Robotics for Environmental Monitoring

Lino Marques, Ryan Smith, Volkan Isler

Multi-Robot Repeated Boundary Coverage Under Uncertainty

Pooyan Fazli, Alireza Davoodi, Alan K. Mackworth
Department of Computer Science
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
Email: {pooyanf,davoodi,mack}@cs.ubc.ca

***Abstract*—This paper describes work in progress addressing the problem of repeated coverage by a team of robots of the boundaries of both a target area and the obstacles inside it. Events are generated randomly on the boundaries and may have different importance weights. In addition, boundaries of the area and the obstacles are heterogeneous, in that events might appear with varying probabilities on different parts of the boundary. The goal is to maximize the reward by detecting the maximum number of events, weighted by their importance, in minimum time. The reward a robot receives for detecting an event depends on how early the event is detected. To this end, a *Markov Decision Process (MDP)* formalism is used to model the coverage problem and capture the uncertainties in the scenario. The performance of the algorithm proposed to solve the *MDP* will be compared with two static algorithms on the basis of the total reward gained during a repeated boundary coverage mission.**

I. INTRODUCTION

Multi-Robot Boundary Coverage is a challenging problem with different applications such as surveillance and monitoring, cleaning, intrusion detection, facility inspection, and so on. In this task, a team of robots cooperatively visits (observes or sweeps) the boundaries of a target area and the obstacles inside it. The goal is to build efficient paths for all the robots which jointly ensure that every point on the boundaries is visited by at least one of the robots. The *Boundary Coverage* is a variant of the *Area Coverage* [3], [8]–[10] problem, in that, the aim is to cover just the boundaries, not the entire area.

There are two classes of boundary coverage problems:

- ***Single Coverage:*** The aim is to cover the boundary until all its accessible points of interests have been visited at least once, while minimizing the time, the distance traversed by the robots, or the number of visits to the points [4], [19].
- ***Repeated Coverage:*** The goal is to cover all the accessible points of interest on the boundary repeatedly over time, while maximizing the frequency of visiting points on the boundary, minimizing the weighted average event detection time, minimizing the sum/maximum length of the paths/tours generated for the robots, or balancing the workload distribution among the robots. Visiting the points on the boundary can be performed with uniform or non-uniform frequency, depending on the priorities of different parts of the boundary[1] [1], [2].

II. PROBLEM DEFINITION AND PRELIMINARIES

In this paper, we address the *Multi-Robot Repeated Boundary Coverage* problem with the following specifications:

- The environment is a simple polygon including rectilinear or non-rectilinear polygonal obstacles.
- The 2D map of the environment is known *a priori*.
- An arbitrary number of homogeneous robots is involved in the coverage mission. The robots are assumed to move at the same unit speed.
- The robots are equipped with a panoramic visual sensor with limited visual range. The sensors are ideal and without noise, that is, they guarantee the detection of an event occurring within the visual range of the robot.
- The events are generated randomly and might occur on any part of the boundaries.
- The events can have different types, and each event type has its own importance weight.

Definition 1. *Event Type:* m types of events might happen in the environment which are of interest to the robots. The set of all event types is represented by $E = \{E_1, E_2, ..., E_m\}$. Similarly, an event of type E_i is denoted as e_i.

Definition 2. *Event Importance:* A degree of importance is defined for each event type E. The importance of an event type E is given by $weight(E)$. It is assumed that $weight(E) \in (0,1]$ such that 1 is the highest degree of importance. The importance can also be referred to as the *priority*, meaning that an event of higher

[1] In this paper, we use the terms 'Coverage' and 'Repeated Coverage' interchangeably.

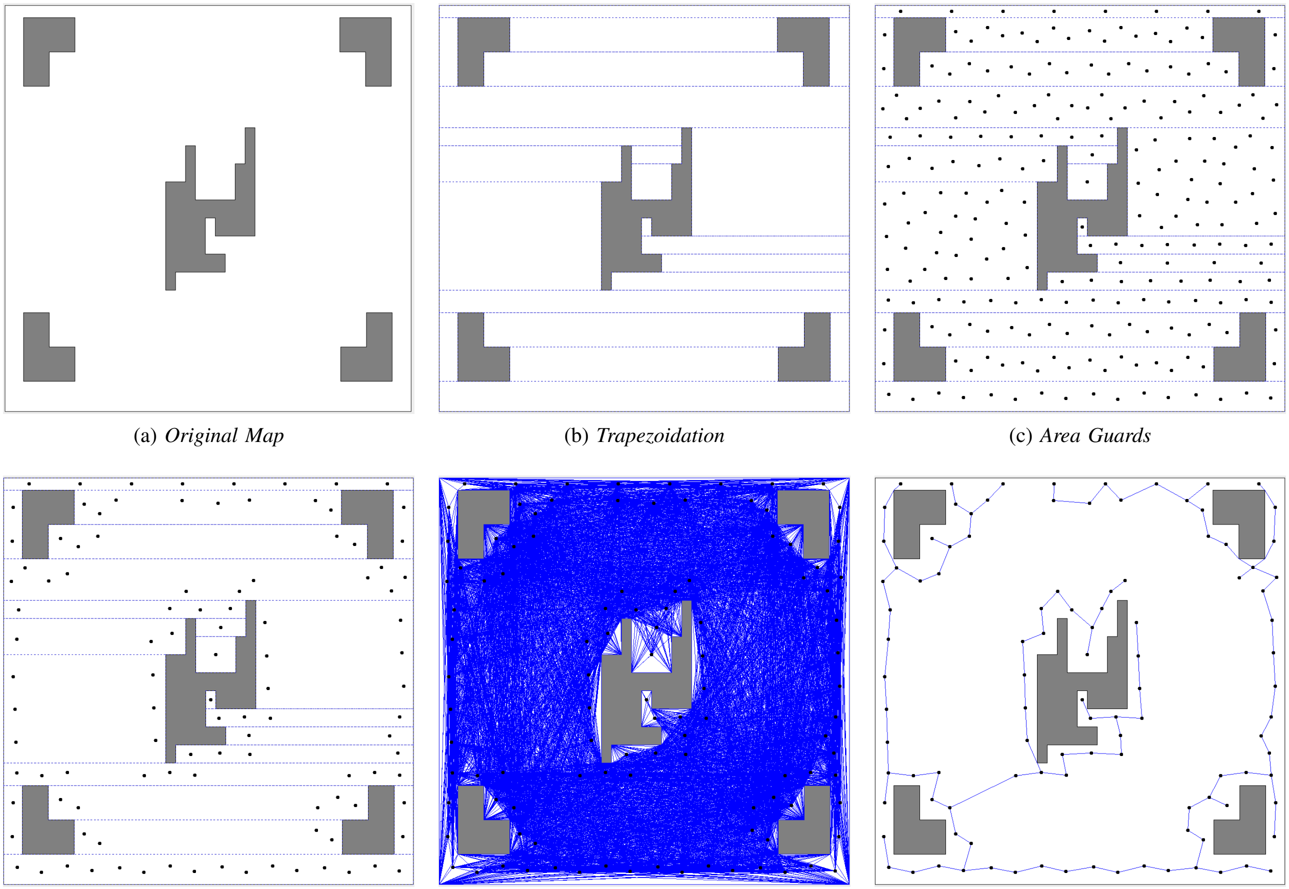


Fig. 1: *Sequential Stages of Building the Boundary Graph*

importance should have higher priority of being detected.

- The reward a robot receives for detecting an event depends on how early the event is detected. At each time step after the event occurrence, the detection reward of the event is decreased by a multiplicative discount factor.
- The boundaries are heterogeneous, in that, events of one type might appear with varying probabilities on different parts of the boundary.

Two types of approaches are proposed to handle the problem: (1) *Uninformed Boundary Coverage* and (2) *Informed Boundary Coverage* algorithms.

Uninformed Boundary Coverage, consisting of two static algorithms, ignores the uncertainties in the coverage mission. In this approach, the robots presume that the events are all of the same importance value and the boundaries are homogeneous. On the other hand, *Informed Boundary Coverage* is primarily based on the *Markov Decision Process (MDP)* formalism in which the robots try to cooperatively maximize the reward function by detecting the maximum number of events, weighted by their importance values, in minimum time. To this end, the robots *learn* the expected reward of visiting a state in the target area at each time step, and based on that, they *plan* to select the best possible path to visit the most promising state at the time in the target area.

The performance of the proposed approaches will be evaluated on the basis of the total reward gained during a finite repeated boundary coverage mission.

III. Environment Modeling

The *Uninformed Boundary Coverage* and *Informed Boundary Coverage* algorithms both require that a roadmap is built within the target area, capturing the connectivity of the free space close to the boundaries while taking into account the limited visual range of the robots. To this end, a graph-based representation called the *Boundary Graph* is constructed on the target area. The *Boundary Graph* provides a roadmap for the robots, enabling them to move throughout the environment to monitor the boundaries of the area and the obstacles. In order to construct the roadmap, a sufficient number of control points, called the *boundary guards*, are placed within the environment, considering the limited visual range of the robots.

A. Locating Guards with Limited Visual Range

In our problem definition, we presume the robots are equipped with panoramic cameras with a 360° field of view. However, the cameras' visual range is limited. The proposed approach initially locates a set of *area guards* required to visually cover an entire area. The term *guard* is taken from the *Art Gallery Problem* [17]. These static *area guards* are control points that can jointly cover the whole environment while satisfying the limited visual range constraint of the robots. In other words, if we had as many robots as the number of *guards*, and each robot was stationed on a *guard*, the entire area would be covered visually by the robots.

To locate the *guards*, the algorithm decomposes the initial target area, a 2D simple polygon with static obstacles, into a collection of convex polygons using a *Trapezoidal Decomposition* method [20], and then applies a post-processing approach to eliminate as many trapezoids as possible (Figure 1b). The post-processing step is more effective in cluttered areas, and since the number of *guards* located by the algorithm is directly correlated to the number of trapezoids, fewer trapezoids will result in fewer guards.

At the next step, a divide-and-conquer method similar to that in [12] is used to successively subdivide each of the resulting convex polygons (trapezoids) into smaller convex sub-polygons until each of them can be covered visually by one *guard* (Figure 1c).

Since, in the current problem, we are interested in monitoring just the boundaries, not all the computed *area guards* are necessary. So, all the *guards* whose visual area does not intersect the boundaries are removed from the set of *area guards*. Figure 1d illustrates the *boundary guards* computed on the sample environment.

B. Building the Graph

Once the boundary guards are located in the target area, a graph called the *Visibility Graph (VG)* [13] is constructed on the guards and the corners of the obstacles (Figure 1e). In order to build the *Visibility Graph*, any two points of interest (a boundary guard or an obstacle corner) which are mutually visible are connected by an edge. Two points are mutually visible if the edge connecting them does not intersect any obstacles in the environment.

C. Boundary Graph

Algorithm 1 describes the steps of the construction of the *Boundary Graph* on a given environment. The input of the algorithm is the *Visibility Graph* made on the map of the area.

The method starts by using the *Floyd-Warshall* algorithm to find the set $MD = \left\{(c_{ij}, v_i, v_j) | v_i, v_j \in V_{vis}\right\}$ of minimum distances, c_{ij}, and the set $SP = \left\{(r_{ij}, v_i, v_j) | v_i, v_j \in V_{vis}\right\}$ of shortest paths, r_{ij}, between any pair of vertices v_i and v_j of the input graph *(line 4)*.

The minimum value of all the minimum distances in *MD* is then selected provided that both the endpoints of the corresponding shortest path in *SP* belong to the set of boundary guards, *SG*, computed in section III-A *(line 5)*. The chosen path *(line 6)*, including all its nodes and edges, forms the initial component of the *Boundary Graph (line 7)*.

Algorithm 1: Boundary Graph

Input:
Graph $G_{vis}(V_{vis}, E_{vis})$, where $V_{vis} = SG \bigcup P$ `/* VG */`
$SG = \{g_1, g_2, ..., g_m\}$ `/* Boundary Guards */`
$P = \{p_1, p_2, ..., p_n\}$ `/* Endpoints of Obstacles */`

Output:
$G_{boundary}(V_{boundary}, E_{boundary})$ where $V_{boundary} = SG \bigcup \widetilde{P}$, $\widetilde{P} \subset P$ `/* Boundary Graph */`

1 **begin**
2 $\quad V_{boundary} \longleftarrow \phi$
3 $\quad E_{boundary} \longleftarrow \phi$
4 $\quad (MD, SP) \longleftarrow FloydWarshall(G_{vis})$
5 $\quad (i,j) \longleftarrow \underset{(i,j)}{\arg\min}\{c_{ij} | (c_{ij}, v_i, v_j) \in MD \;\&\; v_i, v_j \in SG\}$
6 $\quad r_{ij} \longleftarrow GetCorrespondingShortestPath(i,j)$
7 $\quad G_{boundary}(V_{boundary}, E_{boundary}) \longleftarrow InitialBoundaryGraph(r_{ij})$
8 $\quad$ **while** $\neg$ *all the guards added* **do**
9 $\quad\quad g \longleftarrow FindClosestGuardTo(G_{boundary})$
10 $\quad\quad Expand(G_{boundary}, g)$
11 $\quad$ **end**
12 $\quad$ **return** $G_{boundary}(V_{boundary}, E_{boundary})$
13 **end**

Next, among all the guards that have not yet been added to the graph, the algorithm finds the closest guard to the current component *(line 9)*, merging the corresponding shortest path with it *(line 10)*. Following the same process, the algorithm keeps expanding the *Boundary Graph* until there are no more boundary guards to be added to the graph *(lines 8-11)*. The resultant graph is the final *Boundary Graph (line 12)*. The nodes of the *Boundary Graph* includes all the boundary guards *(SG)* and the subset of the obstacles' nodes ($\widetilde{P} \subset P$), collectively referred to as *Points of Interests* ($PoI = SG \bigcup \widetilde{P}$). Traversing the *Boundary Graph* guarantees complete coverage of the boundaries given the limited visual range of the robots.

Figure 1f illustrates the *Boundary Graph* built on the *Visibility Graph* of figure 1e.

D. Boundary Segmentation

The boundaries of the area and the obstacles are divided into identical length *segments*, each of which is small enough to be completely visible by a guard, and such that the probability of event occurrence is uniform along the *segment*.

Definition 3. *Visual Area of a Guard (VA_g):* The *visual area* of a guard, VA_g, is the set of all the *segments* which are visible to the guard g, *i.e.* $VA_g = \{seg_g^1, seg_g^2, ..., .seg_g^p\}$.

Definition 4. *Shared Segment:* A *shared segment* is common to the *visual area* of two or more guards.

Assumption 1. *The events occurring within the visual area of a guard are detected only when the robot is located on the guard.*

IV. Uninformed Boundary Coverage

Uninformed Boundary Coverage ignores the presence of uncertainties in the coverage mission. In this approach, the robots assume that the events are all of the same importance value and the boundaries are homogeneous. We suggest two algorithms for *Uninformed Boundary Coverage*: (1) the *Cyclic Boundary Coverage* and (2) the *Cluster-based Boundary Coverage* algorithms.

A. Cyclic Boundary Coverage

In *Cyclic Boundary Coverage (Algorithm* 2*)*, a tour is constructed on the *Boundary Graph* using the *Chained Lin-Kernighan* algorithm.

Chained Lin-Kernighan (CLK), a modification of the *Lin-Kernighan* algorithm [14], is generally considered to be one of the best heuristic methods for generating *optimal* or *near-optimal* solutions for the *Euclidean Traveling Salesman Problem* [6]. Given the distance between each pair of a finite number of nodes in a *complete* graph, the *Travelling Salesman Problem (TSP)* is to find the shortest tour passing through all the nodes exactly once and returning to the starting node [5].

This *Lin-Kernighan* algorithm, a *local search* algorithm [11], is a generalization of the *k-opt* algorithm [7]. A *k-opt* algorithm explores all the *TSP* tours which can be obtained by removing k edges from the original tour and adding k different edges, such that the resulting tour is feasible. In order to improve the efficiency, Lin and Kernighan introduce a variable *k-opt* algorithm, which adaptively decides at each iteration what value of k to use [14]. Given the computation time limit, the process is repeated by generating new initial tours and applying the *Lin-Kernighan* algorithm to possibly find a tour shorter than the best one thus far. Martin *et. al* [15], [16] suggest that instead of repeatedly starting from new tours, which is inefficient, the alternative is to perturb the *Lin-Kernighan* tour, and then reapply the algorithm. If this leads to a shorter tour, then discard the old tour, and start with the new one. Otherwise, continue with the old tour and perturb it again.

The input of the *Chained Lin-Kernighan* algorithm needs to be a complete graph. To this end, the *Boundary Graph* is made complete *(line 2)* by adding edges from the original *VG* graph, when there does not exist an edge between two nodes in the *Boundary Graph*. If there is not an edge between the two nodes in the original graph either, a *virtual edge* is added to the *Boundary Graph* to connect the two nodes. The weights of these edges are set to the length of the shortest path

---

Algorithm 2: Cyclic Boundary Coverage

Input:
$G_{vis}(V_{vis}, E_{vis})$, where $V_{vis} = SG \bigcup P$ `/* VG */`
$SG = \{g_1, g_2, ..., g_m\}$ `/* Boundary Guards */`
$P = \{p_1, p_2, ..., p_n\}$ `/* Endpoints of Obstacles */`
$G_{boundary}(V_{boundary}, E_{boundary})$: the *Boundary Graph*
$|R|$: Number of Robots

Output:
A tour, *dTour*, distributed among the robots, passing through all the nodes *(Points of Interests)* of the *Boundary Graph*

1 **begin**
2 $CG_{Boundary} \longleftarrow CompleteGraph(G_{boundary}, G_{vis})$
3 $tour \longleftarrow BuildTour(CG_{Boundary}, CLK)$
4 $dTour \longleftarrow DistributeRobots(tour, |R|)$
5 **return** *dTour*
6 **end**

---

between the two nodes in the original *VG* graph. The *Chained Lin-Kernighan* algorithm then finds the shortest tour passing through all the nodes of the *Boundary graph*, returning to the start node *(line 3)*. The robots are then distributed equidistantly along the tour *(line 4)* and move repeatedly around it in the same direction.

B. Cluster-based Boundary Coverage

The *Cluster-based Boundary Coverage* algorithm (*Algorithm* 3*)*, uses the *k-Means* clustering algorithm to divide the guards into $|R|$ disjoint clusters. The initial centroids are found as follows: the endpoints of the longest path in the original *VG* graph are selected as the starting points of the first two centroids, such that the endpoints belong to the set of guards, *SG*. For the next centroid, a guard in *SG* is selected such that it maximizes the minimum distance from the starting points of the first two centroids. Similarly, for the next centroid, a guard is selected that maximizes the minimum distance from the starting points of the other three centroids. This continues until $|R|$ initial centroids are found for the $|R|$ clusters of the guards. In the next iterations, since the computed centroids may not lie on the nodes of the *Boundary Graph*, they are matched to the closest guard in the environment *(line 2)*.

Distance from the centroids is determined based on the distance in the original *VG* graph rather than the Euclidean distance. Having built the $|R|$ clusters on the guards *(line 3)*, we connect each pair of guards in each cluster if they have a corresponding edge in the *Boundary Graph (line 5)*. Thereafter, we do a connectivity test on all the clusters, meaning that each pair of guards in each cluster should be connected through a path. For this purpose, we first find the disconnected components within the cluster *(line 6)* and then

Algorithm 3: Cluster-based Boundary Coverage

Input:
$G_{vis}(V_{vis}, E_{vis})$, where $V_{vis} = SG \bigcup P$ `/* VG */`
$SG = \{g_1, g_2, ..., g_m\}$ `/* Boundary Guards */`
$P = \{p_1, p_2, ..., p_n\}$ `/* Endpoints of Obstacles */`
$G_{boundary}(V_{boundary}, E_{boundary})$: the *Boundary Graph*
$|R|$: Number of Robots

Output:
A set of $|R|$ tours, $Tours = \{T_1, T_2, \ldots, T_{|R|}\}$ where $\bigcup_{i=1}^{|R|} V_{T_i} = SG$, SG is the set of guards of the *Boundary Graph* and V_{T_i} is the set of guards of the tour T_i

1 **begin**
2 $\quad initialCentroids \longleftarrow FindInitialCentroids(G_{vis}, |R|)$
3 $\quad Tours \longleftarrow kMeans(G_{vis}, |R|, initialCentroids)$
4 $\quad$ **foreach** $T_i \in Tours$ **do**
5 $\quad\quad ConnectGuards(T_i, G_{boundary})$
6 $\quad\quad disconnectedComponents \longleftarrow FindDisconnectedComponents(T_i)$
7 $\quad\quad MST \longleftarrow BuildMST(G_{vis}, disconnectedComponents)$
8 $\quad\quad T_i \longleftarrow T_i + MST$
9 $\quad\quad T_i \longleftarrow BuildTour(T_i, CLK)$
10 $\quad$ **end**
11 $\quad$ **return** *Tours*
12 **end**

compute a *Minimum Spanning Tree* on them based on the edges of the original *VG* graph *(line 7)*. Finally, we add the *Minimum Spanning Tree*'s corresponding edges and nodes to the cluster *(line 8)*, and the *Chained Lin-Kernighan* algorithm is used to build a tour on it *(line 9)*. The tour is then assigned to a robot, and the robot repeatedly traverses the tour.

V. Informed Boundary Coverage

Informed Boundary Coverage is primarily based on the *Markov Decision Process (MDP)* formalism in which the robots try to cooperatively maximize the reward by detecting the maximum number of events in the minimum time, considering the importance value of the events. To this end, the robots *learn* the expected reward of visiting a state in the target area at each time step, and based on that, they *plan* to select the best possible path to visit the most promising state at the time in the area.

The algorithm starts by decomposing the *Boundary Graph* into as many clusters as there are robots in the environment using the *k-Means* algorithm, similar to the process discussed before in Section IV-B. Each robot then traverses the cluster assigned to it according to the policy being learned. In this approach, the robots do not need to communicate about every one of the guards they visit. They only update each other about the guards with one or more *shared segments*, and those *shared segments* belong to more than one robot. When a robot, R_i, visits a guard having a *segment* in common with another guard assigned to robot R_j, it notifies R_j about the visit and the events detected in the *shared segment*.

Definition 5. *Time of Last Visit (TLV):* Each robot keeps track of the time of the last visit to its guards, and to the guards of the other robots with some *segments* shared with one of the robot's guards. If $\{g_1, g_2, ..., g_p\}$ is the set of guards monitored by a robot, then for each g_i, TLV_{g_i} represents the last time the guard g_i was visited.

Assumption 2. *The robots are aware of the types of the events occurring on the boundaries and their importance weights.*

The *Multi-Robot Repeated Boundary Coverage* problem is formulated as a tuple (S, A, ST, STR) where:

- $S = SG \times TLV$ is the set of states, where SG is the set of guards and TLV is the set of last visits to the guards.
- A is the set of actions available for a robot in each state. An action is defined as moving from one guard to another. So in each state, the robot might have one or more actions available.
- ST is the state transition function which is deterministic, that is, it guarantees reaching the target state chosen by the robot when the action is performed.
- STR is the state reward, which is equal to the sum of the discounted importance of the detected events at the state. If $t(e)$ is the time interval between starting event e and the detection time, the STR is formulated as:

  $$STR(g) = \sum_{seg_g \in VA_g} \sum_{E_i \in E} \sum_{e_i \in E_i} weight(e_i) \times \gamma^{t(e_i)}$$

  Once a robot arrives at a guard g, it can detect all the events occurring within the VA_g, the *visual area* of the guard g. If a robot were aware of the starting time of the events, it would receive the *STR*. It is assumed that the reward a robot receives for an event depends on how early the event is detected. At each time step after the event occurrence, the detection reward of the event is multiplied by a discount factor of $\gamma = 0.95$.

Definition 6. *Policy:* A policy, $\pi : S \rightarrow A$ at each state determines what action should be performed next by the robot.

A. Learning

If the robots had knowledge of the probability of occurrence of the different events in each state as well as the starting time of the events, they would be able to utilize the *STR* to find a policy which maximizes the total reward of the boundary coverage mission. But since this information is not available to the robots, the *STR* is estimated by the sum of the *Expected*

Segment Reward (ESR) of the *segments* comprising the state:

$$STR(g) \simeq \sum_{seg_g \in VA_g}(ESR(seg_g,t))$$

Expected Segment Reward (ESR) is defined to represent the expected reward of seg_g at time t. The *ESR* can be calculated using the sum of the discounted importance of the events occurred between the last visit, TLV_g, and the current visit time, t, to the *segment's* corresponding guard, g:

$$ESR(seg_g,t) = \sum_{E_i \in E}\sum_{e_i \in E_i}(1+\gamma^1+\gamma^2+...+\gamma^{t-TLV_g}) \times PSE(e_i,seg_g) \times weight(E_i).$$

where γ is the reward discount factor. We assume that for every time step after an event occurs without being detected, the event detection reward is discounted by γ. Furthermore, the *Probability of Segment Event (PSE)* is defined for each event type $E_i \in E$ and each *segment* seg_g, $g \in SG$, to indicate the probability of events of type E_i occurring within the seg_g at each time step.

The *Expected Segment Reward* can be reformulated as below:

$$ESR(seg_g,t) = (1+\gamma^1+\gamma^2+...+\gamma^{t-TLV_g}) \times \sum_{E_i \in E}\sum_{e_i \in E_i} PSE(e_i,seg_g) \times weight(E_i)$$

In the above formula, $\sum_{E_i \in E}\sum_{e_i \in E_i} PSE(e_i,seg_g) \times weight(E_i)$ is called the *Potential Segment Reward (PSR)*, indicating the potential reward of all the events at seg_g, per time step, and is represented by $PSR(seg_g)$. If a robot knows the *PSR* of the events at each *segment* of the *visual area* of the guards, it can calculate the *ESR* for any arbitrary time t.

To this end, a learning procedure for estimating the *PSR* gradually updates its initial value. In the initialization step, we assume that all the events have the same probability of occurrence at each *segment*. Therefore, all the *PSR*s are initialized to 1. When a robot arrives at a guard g, it can detect whether an event has occurred at any of the *segments* belonging to VA_g. If there is at least one event occurring in the seg_g, the $PSR(seg_g)$ is updated using the following formula:

$$PSR(seg_g) = (1-\alpha) \times PSR(seg_g) + \alpha \times \frac{\sum_{E_i \in E}\sum_{e_i \in E_i}(weight(e_i))}{t-TLV_g}$$

where α is the learning rate set to 0.9 and t is the time of the visit to g. This formula gives more weight to the new information than that given to the past information. The robot performs the updating process for all the event types and all the *segments* of the guards.

On the contrary, if no event occurs during the time period between the last visit, TLV_g, and the current visit time, t, the *PSR* is updated to reflect the new fact. To this end, the current value of the *PSR* is multiplied by the discount factor of $\beta = 0.9$:

$$PSR(seg_g) = PSR(seg_g) \times \beta$$

This helps to gradually discard the effects of the *segments* in which events occur rarely.

In summary, the *PSR* is updated once a robot visits a guard. As already mentioned, the *PSR* represents the potential reward of each *segment* per time step. Now, we can use the *PSR* to calculate the *ESR* using the following procedure:

At the beginning, the *ESR*s of all the *segments* are initialized to zero. Then, at each time step, if the robot has yet to arrive at a guard, the value of *ESR* is updated using the following equation:

$$ESR(seg_g,t) = \gamma \times ESR(seg_g,t-1) + PSR(seg_g),\ seg_g \in VA_g \ and \ g \in SG$$

If the robot arrives at g, it detects all the events and consequently:

$$\forall seg_g \in VA_g,\ ESR(seg_g,t) = 0.$$

This updating process continues during the boundary coverage operation.

B. Planning

Once a robot arrives at a guard and detects all the events which might have occurred at the *segments* of the guard, it selects the next action to perform. As already mentioned, the action in the boundary coverage operation is defined as moving from one guard to another. At each state, the robot considers all the *shortest paths* to all the other guards it can move to. Note that, since we divide the *Boundary Graph* among the robots, each robot can only move to the guards assigned to it. For each path, $path(g_i,g_j)$, where g_i is the current guard and g_j is the target guard, *Path Reward (PR)* is defined as the reward the robot receives when moving from the guard g_i to the guard g_j. The path from g_i to g_j includes zero or more intermediate guards and can be represented as:

$$path(g_i,g_j) = [g_i,g_{i+1},g_{i+2},...,g_{j-1},g_j].$$

Given the speed of the robot, the arrival time at each of the guards on the path can be estimated. Hence, the robot can have an estimate of the $ESR(seg_g,t_g)$ for each *segment* of the guard g, in which t_g is the arrival time to the guard g. For such a path, the *PR* is calculated as below:

$PR(path(g_i,g_j)) = \sum_{g \in path(g_i,g_j)} \sum_{seg_g \in VA_g} ESR(seg_g,t_g)$.

When calculating the *PR*, the robot should take into account the visits to the *shared segment* by the other robots. Moreover, as long as a robot does not receive a message from another robot regarding a visit to a *shared segment*, the robot assumes that the *shared segment* has not already been visited and will not be visited by any other robot.

Next, for each path, the *Average Path Reward (APR)* is calculated using the following formula:

$$APR(path(g_i,g_j)) = \frac{PR(path(g_i,g_j))}{t_{g_j} - TLV_{g_i}}$$

where g_j is the target guard on the path, and t_{g_j} is the arrival time to the guard g_j. The robot will select a path with the maximum *Average Path Reward* to traverse next.

VI. Proposed Experiments

We have developed a simulator to test the algorithms in different scenarios. The simulator can support different numbers of robots in the workspace, different visual ranges for the robots, and varying degrees of clutter in the environment. A random map generator was also developed as a part of the simulator which extends a library [18] to build rectilinear or non-rectilinear polygons with free form polygonal obstacles within the space. Maps can have different numbers of nodes and percentages of clutter.

We aim to compare the *Informed Boundary Coverage* with the two variants of the *Uninformed Boundary Coverage* algorithm in terms of the total reward received by the team for detecting the events. To this end, we consider three types of environments in the experiments: sparse ($0-25\%$ cluttered), semi-cluttered ($25-50\%$ cluttered), cluttered ($50-75\%$ cluttered). Ten different maps are used in the experiments for each of the three environment types (30 in total). The clutter percentage of an environment is the ratio of the area of the obstacles to the whole target area (*i.e.* obstacles + free space). The experiments are conducted using $1,2,3,\ldots,15$ robots. 5 different event types are also used in the experiments. Finally, the effect of change in the robots' visual range on the performance of the boundary coverage algorithms is investigated.

VII. Future Work

In the *Informed Boundary Coverage* approach, we considered a static decomposition of the *Boundary Graph* among the robots. However, this is not always a reasonable approach, as it is possible that all the events are generated in just one cluster, and so just one robot will be in charge of handling all the events. In other words, for the other robots, there are no events occurring in their clusters. To cope with this issue, we are investigating a dynamic approach to decompose the *Boundary Graph* among the robots, considering the location and the weight of the events occurring on the boundary.

References

[1] N. Agmon, S. Kraus, and G. Kaminka, "Multi-robot perimeter patrol in adversarial settings," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, May 2008, pp. 2339–2345.

[2] N. Agmon, S. Kraus, G. A. Kaminka, and V. Sadov, "Adversarial uncertainty in multi-robot patrol," in *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI*, 2009, pp. 1811–1817.

[3] A. Almeida, G. Ramalho, H. Santana, P. A. Tedesco, T. Menezes, V. Corruble, and Y. Chevaleyre, "Recent advances on multi-agent patrolling," in *Proceedings of the Brazilian Symposium on Artificial Intelligence, SBIA*, 2004, pp. 474–483.

[4] P. Amstutz, N. Correll, and A. Martinoli, "Distributed boundary coverage with a team of networked miniature robots using a robust market-based algorithm," *Annals Mathematics Artificial Intelligence*, vol. 52, no. 2-4, pp. 307–333, 2008.

[5] D. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*. Princeton, NJ, USA: Princeton University Press, 2007.

[6] D. Applegate, W. Cook, and A. Rohe, "Chained Lin-Kernighan for large traveling salesman problems," *INFORMS Journal on Computing*, vol. 15, pp. 82–92, January 2003.

[7] B. Chandra, H. Karloff, and C. Tovey, "New results on the old k-opt algorithm for the traveling salesman problem," *SIAM Journal on Computing*, vol. 28, pp. 1998–2029, August 1999.

[8] H. Choset, "Coverage for robotics – a survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1-4, pp. 113–126, 2001.

[9] P. Fazli, A. Davoodi, P. Pasquier, and A. K. Mackworth, "Multi-robot area coverage with limited visibility," in *Proceedings of The International Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 2010, pp. 1501–1502.

[10] ——, "Complete and robust cooperative robot area coverage with limited range," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2010, pp. 5577–5582.

[11] H. Hoos and T. Sttzle, *Stochastic Local Search: Foundations & Applications*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2004.

[12] G. D. Kazazakis and A. A. Argyros, "Fast positioning of limited visibility guards for inspection of 2D workspaces," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2002, pp. 2843–2848.

[13] J.-C. Latombe, *Robot Motion Planning*. Norwell, MA, USA: Kluwer Academic Publishers, 1991.

[14] S. Lin and B. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, vol. 21, no. 2, pp. 498–516, 1973.

[15] O. Martin, S. Otto, and E. Felten, "Large-step markov chains for the traveling salesman problem," *Complex Systems*, vol. 5, no. 3, pp. 299–326, 1991.

[16] O. Martin, S. W. Otto, and E. W. Felten, "Large-step markov chains for the TSP incorporating local search heuristics," *Operations Research Letters*, vol. 11, pp. 219–224, 1992.

[17] J. O'Rourke, *Art gallery theorems and algorithms*. New York, NY, USA: Oxford University Press, Inc., 1987.

[18] A. P. Tomás and A. L. Bajuelos, "Quadratic-time linear-space algorithms for generating orthogonal polygons with a given number of vertices," in *Proceedings of the International Conference on Computational Science and Its Applications, ICCSA*, 2004, pp. 117–126.

[19] K. Williams and J. Burdick, "Multi-robot boundary coverage with plan revision," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, 2006, pp. 1716–1723.

[20] B. Zalik and G. J. Clapworthy, "A universal trapezoidation algorithm for planar polygons," *Computers & Graphics*, vol. 23, no. 3, pp. 353 – 363, 1999.

A Mobile Reconnaissance Robot for Investigation of Dangerous Sites

M. Hericks, U. Krebs, O. Kuzmicheva, H. Kampe, A. Gräser

***Abstract*—In this paper a novel approach to semi-autonomous sample drawing in unstructured environments is proposed. Therefore several key issues for reconnaissance robots have been addressed. A fully equipped robotic system with an extendable, highly flexible and platform independent software framework was developed, aiming at autonomous sample handling and adaptive task execution. Several sensory devices allow a detailed inspection including stereovision and thermal imaging. The complex task of sample handling is realized with an innovative container concept which avoids cross-contamination without expensive or complex additional devices. For user interaction, intuitive control concepts are implemented, leading to a system that is suitable for non-experts and performs dexterous manipulation tasks in highly unstructured environments.**

***Index Terms*—decontamination, dexterous manipulation, reconnaissance robot, sample handling, software architecture, outdoor robotics**

I. Introduction

IN many situations, the evaluation of risks is a compulsory need in order to protect action forces and civilians alike from being exposed to dangerous situations e.g. caused by hazardous materials. Therefore risky situations like natural catastrophes, structural fires, traffic accidents with unknown chemical materials, terrorist attacks and even industrial applications require a way of estimating the risks for potential helpers. Before anyone can help in case of such incidents, the personal risks of the rescue team must be evaluated in order not to endanger their health as well.

For detecting hazardous substances in an unstructured environment, highly elaborate measuring methods have to be applied, which can indicate the presence of a dangerous substance by its reaction with other chemical products. Some substances might be detectable using massive electronic devices, which are not easily manageable for a human due to their weight and the need of an electric power supply.

Currently the problem of sample drawing is accomplished by specialized forces (e.g. firemen), which sent specialists in protective suits in order to determine the dangerousness of a certain situation through analysis of the gathered samples. This is a very time consuming process, as it requires a lot of different steps that have to be executed in a certain manner, as can be seen in Figure 1:

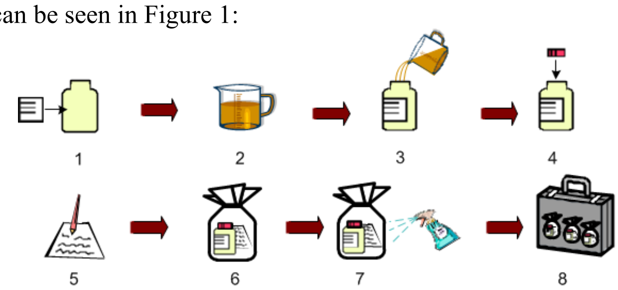


Figure 1, Sample drawing process – (1)label container, (2)fill measuring cup with liquid, (3)pour in prescribed amount into container, (4)close container, (5)edit accompanying protocol, (6)bundle container and protocol in carriage bag, (7)decontaminate carriage bag, (8)gather carriage bags in transport box

Obviously this procedure is an elaborate and exhausting task, since it is extremely important that every single step is performed very thoughtfully. Moreover, a lot of different kinds of samples, like fluid, solid and gaseous samples, may be required in certain situations. This may result in fast fatigue of the workers, possibly influencing the quality of the samples.

Manual sample drawing is a time-consuming process, considering that the experts might have to enter the site multiple times, running through the whole process of decontamination, refilling compressed-air cylinders, and rest periods, in order to get enough samples. All this leads to a situation where humans tend to make errors, which is fatal for sample drawing. So the focus of this work is the research and development of a robot-based concept for the problem of autonomous sample drawing in unstructured environments.

In this paper, a mobile outdoor robotic system, which is designed for handling samples in unstructured environments, is presented. Section II describes the State-of-the-Art in this research field and the subsequent section III deals with the details of the considered system regarding soft- and hardware development. In Section IV current and upcoming results are given and finally section V demonstrates the outcome and future directions.

Manuscript received June 3, 2011. This work was supported by the German Federation of Industrial Research Associations (german: Arbeitsgemeinschaft industrieller Forschungsvereinigungen "Otto von Guericke" e.V. - AiF) with Projectnumber 16336N and the Federal Ministry of Economics and Technology.

M. Hericks, Friedrich-Wilhelm-Bessel-Institute F.m.b.H. (FWBI), Otto-Hahn-Allee 1, 28359 Bremen, Germany (e-mail: hericks@fwbi-bremen.de).

U. Krebs, Friedrich-Wilhelm-Bessel-Institute F.m.b.H. (FWBI), Otto-Hahn-Allee 1, 28359 Bremen, Germany (e-mail: krebs@fwbi-bremen.de).

O. Kuzmicheva, Friedrich-Wilhelm-Bessel-Institute F.m.b.H. (FWBI, Otto-Hahn-Allee 1, 28359 Bremen, Germany (e-mail: olena@fwbi-bremen.de).

H. Kampe, Institute of Automation (IAT) at the Department of Physics/Electrical Engineering, University of Bremen, Otto-Hahn-Allee 1, 28359 Bremen, Germany (e-mail: kampe@iat.uni-bremen.de).

A. Gräser, Institute of Automation (IAT) at the Department of Physics/Electrical Engineering, University of Bremen, Otto-Hahn-Allee 1, 28359 Bremen, Germany (e-mail: ag@iat.uni-bremen.de).

II. STATE-OF-THE-ART

Due to the great significance of reconnaissance robots, a lot of research has been performed in recent years. A multitude of different systems exist, that intend to act more or less autonomously in different kinds of environments. The main fields of research are exploration-, security-, service- and EOD[1]- robots.

One of the most prominent being the HAZBOT system from the JPL Fire Department[1], which serves to localize, characterize, identify and mitigate hazardous materials. Also the Hunter teleoperation system[2], designed by the Yangzhou University serves to inspect and dispose nuclear radiation or harmful chemical materials. The University of Bielefeld designed a robotic systems, which handles samples in a laboratory environment[3]. Other remarkable systems are the Packbot developed by iRobot Corporation[4], Telerob's Safety Guard and tEODor[5] as well as the NAT-II and T.S.R. EOD-robots from Elektroland[6].

However, most of those systems focus on teleoperation where the robot arm serves as an extension of the human arm, offering only a rather low level of automation.

The Safety Guard system from Telerob is the only system capable of taking samples, including the whole procedure as mentioned in section I. All other systems cannot handle fluid and gaseous samples, but only graspable objects. Those systems concentrate on remote controlled inspection, observation and object handling. Hence they must be considered as security- or exploration-robots rather than sample-robots. A common drawback of all systems is that they need experts to be operated, since teleoperation is rather difficult. One of the main objectives, to prevent the user from exhausting and error-prone task execution, cannot be accomplished by these systems. Besides this, all systems suffer from a restricted workspace due to a restricted degree of freedom of the manipulator.

The mobile robotic system RecoRob presented here includes a robot arm with 7 degrees of freedom for dexterous manipulation in unstructured environments, where the user can focus on navigation and decision making. The developed software allows autonomous task execution in order to disburden the user from the complex process of sample drawing by performing the sophisticated manipulation tasks autonomously. This will lead to a system suitable for a wide range of users that neither need any expertise in robotics nor in manipulation.

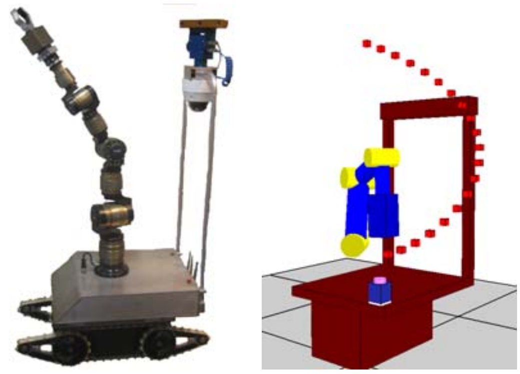
III. SYSTEM CONCEPT

As outlined in the previous section, a multitude of unsolved problems exist in reconnaissance robotics so far. Therefore the RecoRob robotic system is presented.

A. System Setup

The RecoRob system is equipped with various hardware devices as can be seen in Figure 2-a.

[1] EOD – Explosive Ordnance Disposal



(a) (b)

Figure 2, RecoRob reconnaissance robotic system - (a)hardware setup, (b)Mapped Virtual Reality (MVR) used as simulation environment in the RecoRob system

It is based on a mobile *ASENDRO platform* from Robowatch, which consists of a variable drive system that can be equipped with either chains or wheels, depending on the desired application. In addition the chains are supported by swing arms which are capable of continuous rotation, enabling the basis to climb stairs and overcome obstacles. For actuation, there is a *Schunk 7DOF lightweight robot arm* for the object manipulation and a *2DOF Pan-Tilt-Head* for steering the vision system. The sensor system consists of an *ATI Technologies Force-Torque sensor* in the manipulator's wrist, a *PointGrey Bumblebee* stereovision camera for 3D reconstruction, a *Samsung SNC Dome Camera* for workspace observation and an *NEC ThermoTracer IR-Camera* for thermal inspection of the working area. For the computational power, there are two *SPECTRA NISE 3140P2E* embedded computers and for communication a *D-Link DAP-2590 Access Point* is installed. The user interaction is realized with a *Getac M230N-5* ruggedized notebook, including Wireless LAN and a touchscreen.

In order to ensure the systems power supply, two additional accumulators are mounted to support the computers in case the main accumulator inside the mobile basis runs out of energy.

A major problem in sample handling is the avoidance of cross-contamination. In our approach, the samples are stored in disposable one-way containers where the actual sampling tool is an integral part of the container itself, as can be seen in Figure 3.

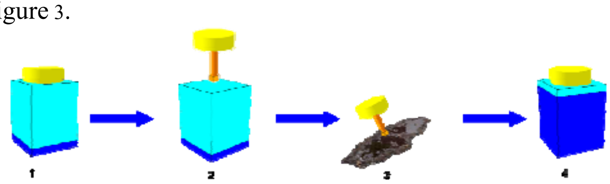


Figure 3, Simulated wipe sample sequence – (1)unused sample container, (2)extracted tool with sponge, (3)taking wipe sample, (4)closed and filled container

Cross contamination is almost impossible, since the robot touches the containers only from the outside and the tool is encapsulated inside the container. In order to avoid a tool changing system for the robot, the containers have an identical outer size and shape and can be handled by a single gripper.

Four different kinds of samples are considered at first: wipe sample, fluid sample, soil sample and pipette sample.

B. Software Architecture

The software architecture of the RecoRob system is constructed to be easily adjustable and extendable to different system configurations. This is achieved by a hierarchical structure with defined interfaces for all its modules. Figure 4 shows a schema of the architecture.

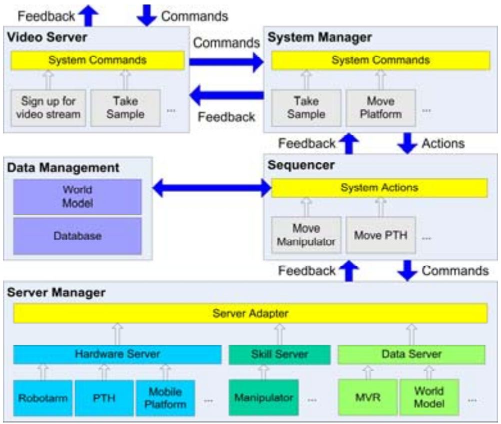


Figure 4, RecoRob software architecture

The system is equipped with two computers. One uses the operating system (OS) Windows and the other uses a Unix OS. The different OSs are presently necessary due to the available drivers for the hardware. Windows is required for the camera modules and therefore also performs the compression of the video streams. The Unix PC controls the platform, the manipulator and the pan-tilt-head of the camera system as well as the sample containers. On the long term it is planned to use one OS only.

Communication with the system is performed using a proprietary command format via a TCP/IP connection. For system control, the user connects and sends the commands to a server on the Windows PC where they are either processed directly or forwarded to the Unix computer, if necessary.

Hardware control is handled by separate server software modules for each hardware component. They are connected to the main system by the CORBA[7] framework which allows the connection of software elements that may be executed on different computers. This approach enables a maximum of flexibility and scalability concerning the hardware. The hardware servers stop continuous movements once the connection to the user is interrupted for too long in order to avoid uncontrolled movements. Next to the hardware servers there are several skill servers which permit hardware actions on a higher level of abstraction. At last, data servers provide necessary information to the other servers, required to accomplish the requested task, e.g. the dimensions of the RecoRob system itself. The servers are controlled by a server manager, which can start, stop and automatically restart them.

Movement of the manipulator is calculated with the help of a mapped virtual reality (Figure 2-b) which represents relevant parts of the system and the environment by geometric primitives (cuboid, cylinder, sphere).

One of the core functions of the RecoRob software architecture is sample handling. Therefore a sample manager exists, which controls the inventory of sample containers and their current state. In case of sample drawing, the available samples are checked and an appropriate one is chosen. For each sample type its unique accompanying manipulation sequence is defined and saved in the system. The only missing information is the target location, which is determined by the user during execution.

C. Communication

As the system setup involves three computers, the communication between them is an important point to consider. Communication must be robust, fast, independent from specialized hardware or operating system and extensible with respect to the number of participants and the bandwidth.

The simple network communication via TCP/IP seems to be a suitable solution to match these prerequisites. Internally the both computers are connected to a gigabit switch, allowing fast transfer of real-time video. The remote control PC is connected via 802.11n Wireless LAN, which is still fast enough to transfer up to two simultaneous video streams with acceptable frame rate.

The currently chosen hardware components can easily be exchanged if the requirements change. If long range communication between remote controller and main system is required, the WLAN part can be replaced with an UMTS/HSDPA capable device without touching the software.

Besides the hardware and low level protocol considerations, the high level protocol is important for an easy interfacing of the system. While the internal communication of the Unix PC utilizes CORBA, which completely hides the communication details from the programmer, the Windows PC uses a command interface, where the commands are simple byte sequences, giving some basic information about the command, followed by the actual data payload, e.g. a video frame. This command interface can easily be implemented on any device that supports TCP/IP communication. By using this interface, restrictions concerning the availability of CORBA for a certain device / operating system are avoided.

D. User interface

Easy and effective guidance of the investigation unit demands a clear overview of the current situation on-site and information about current state of the task execution as well.

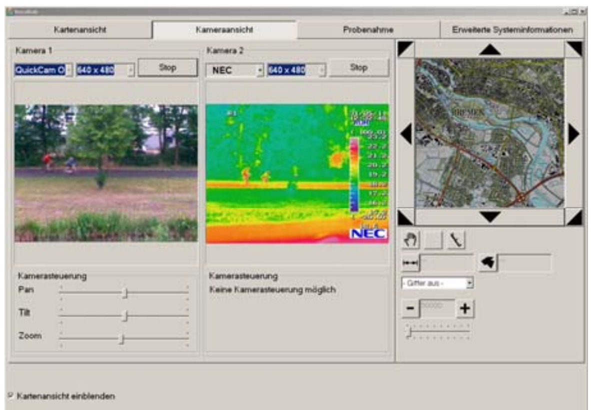


Figure 5, First setup of the RecoRob user Interface - module ‘camera view’: normal camera view (left), thermal image (middle), optional map view (right)

In order to display all available information without overload of the user the GUI of the RecoRob system is divided in four modules:

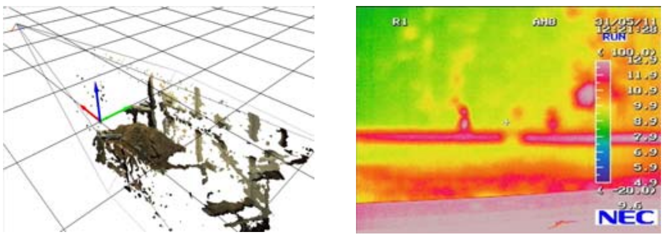
- *Mission overview* integrates map based view of the mission. Allows optional single camera view as well as navigation of the mobile platform and simplified manual control of the robot arm
- *Camera view* provides up to two simultaneous live camera streams and control of available camera features, e.g. pan-tilt-zoom. Allows optional map view as well as mobile navigation and simplified control of the manipulator
- *Sampling control module* aims to provide simple and intuitive mission control: choice of sampling mode (teleoperated / autonomous) like selection of sampling place and type for autonomous sampling and display of the execution state. Dependent on selected sampling mode, extended or simplified manipulator control interface is available
- *Extended system info* integrates additional information about system state like connection status and battery state. Offers a possibility of remote control of the computers integrated in mobile investigation unit

IV. Results

Since the project is in an early stage, only preliminary results can be presented by now. The focus of this project, as outlined in section III, is the autonomous sample handling. In order to verify our expectations to the system and to test all functionalities, we set up a simulation environment using a mapped virtual reality. The environment can be seen in Figure 2-b.

For the reconstruction of three dimensional data, the OpenCV library from willow garage is used to compute a point cloud from the stereo images and gather information for the autonomous task execution. An exemplary point cloud can be seen in Figure 6-a.

Ongoing research is also in progress for the analysis of the thermal images (Figure 6-b), which will be used for localization of heat sources, that might indicate a point of interest for specific sampling acquisition.



(a) (b)

Figure 6, Machine Vision approaches - (a)pointcloud generated with test images using OpenCV and Stereo Blockmatching, (b)thermal imaging with NEC IR-Camera

In order to realize our ideas for sample handling, we are currently investigating possible solutions for the containers with integrated tools.

V. Conclusion / Outlook

This paper focuses on design and development of a new approach in reconnaissance robotics for semi-autonomous sample drawing in unstructured environments. We proposed the robotic system RecoRob which can be operated by users without special expertise in robotics and manipulation. Thus it allows focusing on the main objectives analysis, characterization and identification of hazardous materials by taking samples.

An extendable, hybrid multilayer software architecture was presented, using a virtual environment for simulation and wireless communication in combination with an adequate user interface.

Simulated and experimental results verify that the system concept achieves the targeted goals of object manipulation, sample handling, wireless communication and user interaction.

Upcoming research goals include further machine vision approaches, integration of multiple sample scenarios using developed sample containers and practical tests of the whole system with appropriate users in outside applications.

References

[1] H. W. Stone, G. Edmonds, “HAZBOT: A hazardous materials emergency response mobile robot,” Proceedings of the 1992 IEEE International Conference on Robotics and Automation, Nice, France, May 1992, pp. 67-73.

[2] H. Tang, X. Cao, A. Song, Y. Guo, J. Bao, “Human-Robot Collaborative Teleoperation System for Semi-Autonomous Reconnaissance Robot,” Proceedings of the 2009 IEEE International Conference on Mechatronics and Automation, Changchun, China, August 2009, pp. 1934-1939.

[3] I. Poggendorf, ”Einsatz eines Serviceroboters zur Automatisierung der Probenentnahem und des Probenmanagements während Kultivierung tierischer Zellen in einer Technikumsumgebung,” Universität Bielefeld, 2004

[4] B. M. Yamauchi, “PackBot: A Versatile Platform for Military Robotics,” Proceedings of 2004 SPIE, Orlando, USA, pp. 228-237

[5] A. Saffiotti, “Platforms for Rescue Operations,” AASS Mobile Robotics Laboratory, Örebro University, Örebro, Sweden, 2004

[6] Elektroland Company, Ankara, Turkey, http://www.elektroland.com.tr/

[7] Henning, Michi ; Vinoski, Steve: Advanced CORBA Programming with C++. Addison Wesly Professional, 1999. – 1120 S. – ISBN 0201379279

[8] Kampe, Henning:[9] Integral Modelling of Objects for Service Robotic Systems. 41st International Symposium on Robotics, ISR 2010, 7 – 9 June 2010, Munich

An Adaptive Field Estimation Algorithm for Sensor Networks in Dynamic Environments

Amanda Prorok, William C. Evans and Alcherio Martinoli

Abstract— **The efficiency of distributed sensor networks depends on an optimal trade-off between the usage of resources and data quality. This workshop paper addresses the problem of optimizing this trade-off in self-configured distributed sensor networks. In our case-study example, we investigate a quadtree network topology and describe how we integrate a fully distributed node controller and field estimation algorithm. In a further step, we present a variant control algorithm, which continuously adapts network sampling and node activity to match spatio-temporal field variability. Realistic simulations are performed on the e-puck robot platform, and show that the proposed sampling strategy potentially economizes 20% of resource usage.**

I. Introduction

Since the beginnings of research on sensor networks in the 1970s, the monitoring of environments and habitats has become one of its major application fields [3]. Technological advances in embedded systems, such as the development of reliable wireless communication, and miniaturization and improved efficiency of microcontrollers and sensors have have answered key needs, and encouraged an increasing deployment of wireless sensor networks as a main tool to monitor spaces [8]. Still, one of the challenges presented with the deployment of sensor networks is the accurate estimation of fields with unpredictable environmental phenomena, while simultaneously addressing the critical issues of resource usage such as local memory, communication and processing constraints.

With networks often consisting of a considerable number of sensor nodes, the necessity of limiting energy consumption as well as bandwidth requirements increases. Research in the domain of ad hoc wireless routing has produced a range of algorithms which propose solutions for these problems. Improved routing algorithms have been developed which aim to accomplish in-network load balancing and an increased system lifetime, employing techniques that are mostly based on system information such as remaining energy levels and routing capacities.

A. Spatial & Temporal Suppression

There are two main approaches to optimizing the energy consumption of sensor networks. In temporal suppression schemes, each node uses its own history of measurements to determine if a new value can be inferred by the network sink instead of being transmitted, or even to avoid sampling and local processing entirely. A simple example would be transmitting measurements only when they differ from the previous value. Typically these approaches make use of much more complex models, often providing bounded error.

The Probabilistic Adaptable Query (PAQ) system is one notable such scheme based on time series forecasting [23]. It uses autoregressive models maintained locally per sensor node in order to keep from sending data directly to the sink. Instead, nodes communicate model parameters as necessary in order to keep the sink's predictions within some defined error bound. Tulone and Madden extend this work with their Similarity-based Adaptive Framework (SAF) [24], adding robustness to quick changes in data trends as well as a location-independent clustering technique that allows the detection of redundant nodes.

On the other hand, spatial suppression exploits spatial correlations between nearby sensor nodes in order to reduce communication load. Many spatial suppression algorithms attempt to detect and deactivate sets of redundant nodes. Arici and Altunbasak propose using a first-order model to determine the predictability of particular nodes [1]. They define some of the nodes in the network as *macronodes* which attempt to fit a plane over their neighbors' positions and data, commanding easily predictable nodes to stop reporting measurements for some period of time. Similarly, Willett et al. define the idea of a *fusion center* that is responsible for estimating a field based on received sensor measurements and then directly deactivating redundant nodes [26].

Chu et al. propose the use of replicated dynamic probabilistic models between the sink and *disjoint cliques* of data sources [4]. The sink then uses these models to predict future sensor data. If the *root* of a clique observes data inconsistent with the sink's current prediction model, a subset of the clique's recent observations are sent and the sink's model is updated as necessary.

B. Motivation

In our work, we address the problem of designing distributed sensor networks for surveillance and monitoring. It is clear from [14] that self-configuration is a necessary element for effective as well as efficient performance of such networks. The proposed design paradigm suggests hierarchical topologies, following a top-down control and bottom-up reconfiguration principle. Here, we build upon this design rule, implementing a distributed, multi-layer tree-based routing algorithm and combining it with a threshold-

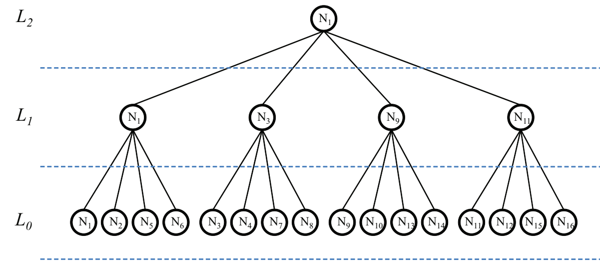
Amanda Prorok, William C. Evans and Alcherio Martinoli are with the Distributed Intelligent Systems and Algorithms Laboratory of the Ecole Polytechnique Fédérale de Lausanne. This work was partially sponsored by the Swiss National Science Foundation (Contract Nr. PP002-116913). William C. Evans is currently sponsored by the NCCR Stabilization Measures Initiative (project "Tamperproof Monitoring Solution for Weather Risk Management"). `firstname.lastname@epfl.ch`

based clustering strategy which is adaptive to the state of the field being estimated. Our algorithm leans on established field estimation methods described in [18] and [26]. The approach is similar to the one described by Arici et al. in [1], which describes an adaptive sensing method also based on a tree-like, hierarchical network structure. Their method exploits the fact that a manual deployment of sensors may offer more information than necessary (over time and space) to reconstruct an accurate field estimate. They propose a self-configuration algorithm which will put nodes into passive mode when their measurements become 'predictable'. Here, also motivated by previous research in the domain of distributed sensor node controllers as presented in [7], we develop a fully distributed node controller that is easily implemented on resource constrained and noisy hardware, which aims to optimize system performance by finding a trade-off between use of resources and data quality. In contrast to the methods described in [2, 13, 27], we base our clustering strategy on field data, rather than on system information. Also, our resulting data aggregation method follows a multi-layer bottom-up principle, which enables global abstraction of the target field, different from the local collaborative processing methods of [15, 28]. Lastly, in contrast to [18] and [26] we focus on the whole system rather than only on communication and routing activities, and our work in [19] demonstrates the approach on real hardware by comparing the performance to theoretical predictions.

The method in this work especially targets heterogeneous sensor-networks, given its non-homogeneous communication constraints. This allows for the deployment of large numbers of cheap sensor nodes to increase granularity, while more expensive, robust sensor nodes are placed at strategically important positions. Nevertheless, in order to guarantee the scalability and robustness of the system, redundancy must be foreseen by implementing efficient role selection strategies. Finally, although our current algorithm does not explicitly take into account node mobility, its design easily accommodates extensions such as node redeployment or network reconfiguration. This capability may equally be deployed non-homogeneously throughout the sensor network.

II. Spatial Suppression using Hierarchical Network Topologies

In accordance with our above-mentioned motivation to port our algorithms onto mobile platforms, we base the following elaborations on robotic sensor networks. As suggested in the theoretical work of [26], we superpose a quadtree (Fig. 1) on the robotic sensor network. Especially when computing spatial problems typical in computer aided design and geo-data applications [12], the quadtree data structure has proven an efficient and powerful tool [11, 20]. An early work in [10] shows how an active quadtree network facilitates image representation and analysis. Also, a recent study in [9] shows how a quadtree can be utilized for in-network data querying in a fully distributed wireless network.



(a) Quadtree hierarchy

Fig. 1. A 16-node quadtree structure. The quadtree hierarchy is decomposed into 3 hierarchy levels. A node will participate in either of the 3 subsets: $\{L_0\}, \{L_0, L_1\}$ or $\{L_0, L_1, L_2\}$.

A. Distributed Network Organization

Here, although our controllers and models are general to any hierarchical topology, we showcase our study on a quadtree based network with each robotic node within our sensor field representing a leaf node in the tree structure. The robots are distributed on a regular grid in a square arena. In a network of a total n nodes, assuming that the robots are aware of their location, each one allocates itself to one of n sensing cells in the decomposed space. We thus obtain a robotic sensor network ordered by the intrinsic hierarchy of the quadtree. Adapted and implemented in a fully distributed sensor network, this hierarchy can be explored in terms of i) communication channels and ii) fine-tuning the spatial resolution of the sensor network. Whereas exploring i) is relatively straightforward as we can directly exploit the quadtree hierarchy, there are many approaches to ii)—our chosen approach will be discussed later in Section II-B.2.

On a global level, the quadtree structure depends only on the number of nodes (implicitly a power of 4), and can be constructed in a distributed manner, assuming that all nodes know their location. As is evident in Fig. 1, a single node may have multiple roles within the network, depending on the status of the network. Thus, we create the notion of layers L_i. In a network of 4^K nodes, we have $K+1$ layers $(L_0, ..., L_K)$, and a node's current role in the network is defined by its current processing layer $L_{current}$. Every node N_i has a maximum layer $L_{k_{max}}$ with $N_i \in L_{k_{max}}$ such that there is no $k > k_{max}$ with $N_i \in L_k$. Also, any node N_i in $L_k, k > 0$ is a clusterhead, with four descending nodes in L_{k-1} as its cluster children (including itself). Lastly, for the sake of clarity, we don't go into the details of an eventual clusterhead rotation or election strategy.

The group of robotic nodes uses wireless communication as a means of inter-node organization. There are two classes of messages being used within the network: control messages and data messages (measurements). The messages typically contain the following elements: *control* or *measurement data*, i and k, with i the id of the sender node N_i and L_k its current processing layer. Control messages are sent top-down through the network structure, and measurement messages bottom-up. Nodes throughout the network or within the communication range of the transmitting node may receive messages at all times and asynchronously from various senders.

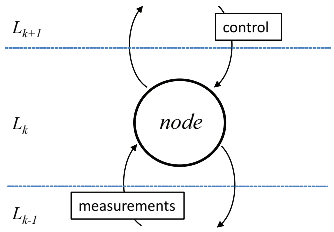


Fig. 2. The node is currently processing data in layer L_k. Measurement messages are sent bottom-up and control messages are sent top-down the quadtree structure.

A clusterhead will only accept measurement data from nodes belonging to its cluster, and following the top-down control principle, a node will only accept control messages from its clusterhead. Fig. 2 illustrates the communication protocol.

B. Control of the Robotic Node

We elaborate two control variants: first, a *naive sensing* strategy (NS), and second, an improved *threshold-based sensing* strategy (TBS). With NS, the nodes are in one of three possible states, whereas with TBS, the nodes are in one of four possible states. The controller is simple and distributed, homogeneous on all nodes.

1) State Machine: The controller can be represented by a simple state-machine, and is depicted in Fig. 3. Initially, a node is in the *sample* state. Each time a node takes a measurement, it will transition to the *process* state. If the node is a leaf node (its processing layer is $L_{current} = L_{k_{max}}$ at all times) it will transition directly to the *broadcast* state, send its measurement and then return to the *sample* state. If the node is a clusterhead, it will increment its processing layer $L_{current}$ once it has received (and aggregated) the data from all the nodes in its cluster, and will enter the *broadcast* state if it has reached its maximal layer $L_{k_{max}}$. Otherwise, it will re-enter the *sample* state. Finally, upon sending the (collected) measurement data in the *broadcast* state, the clusterhead will return to the *sample* state.

In a further step, we develop the controller for TBS, with the goal of optimizing the use of resources by reducing the number of messages sent and measurements taken. The aim is to *prune* certain node-clusters off the quadtree by putting the nodes in those clusters to sleep. A clusterhead will then replace measurement values of all its descendant nodes with its own. A fourth state is added to the NS controller, and is illustrated by dashed line on the right-hand side in Fig. 3. If a node has received a relevant pruning control message, it will be absorbed by the *idle* state.

2) Threshold-Based Pruning Algorithm: In TBS, a clusterhead makes the decision to prune or not prune its child nodes. Thus, we implemented a threshold-based pruning algorithm, which builds on the theoretical formula proposed in [18]. Assuming that the field is anisotropic, the chosen approach is to prune sensor-node clusters which are sampling values in isotropic subparts of the field. The resulting field estimator will display a higher sensing resolution along the boundaries of the anisotropic field and lower resolution in the isotropic subparts. This principle is illustrated by the

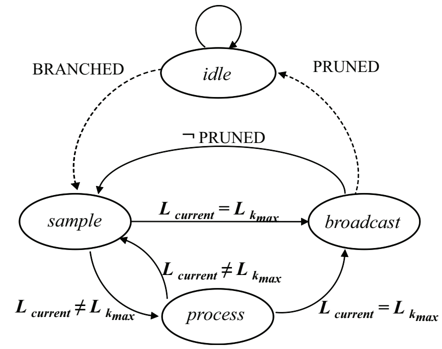


Fig. 3. Schematic illustration of two variant state-machines implemented for the quadtree structure. (a) NS (without dashed line): A node samples environmental events. Measurement data from cluster nodes is received and processed. When the cluster data is complete, a node will broadcast the collected data. (b) TBS (with dashed lines): A node which is shut down is absorbed by the idle state. If change is perceived an idle node may re-enter the sampling state.

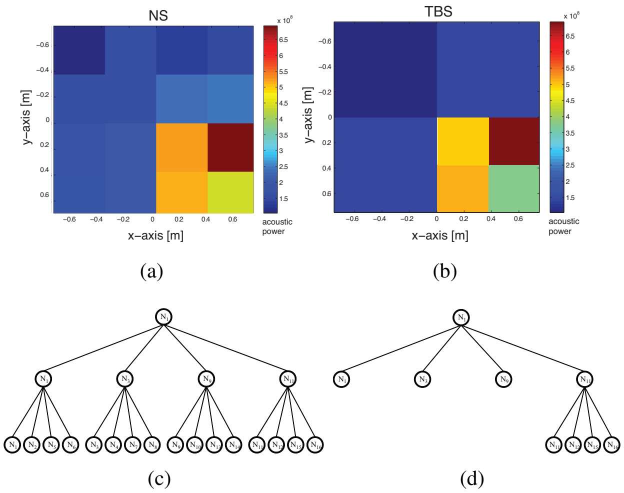


Fig. 4. The graphs show the calculated power of an acoustic event at a given moment. Each of the 16 cells is occupied by one robotic sensor node. An acoustic source is located in the bottom left corner of the arena. (a) A snapshot of the true field values (b) The data sent out of the network by the top-level node after completion of the pruning algorithm

example in Fig. 4. Fig. 4 (a) and (c) show a fully active (un-pruned) quadtree and the values transmitted by the full network, whereas Fig. 4 (b) and (d) show a pruned quadtree and the values transmitted by the remaining active nodes.

The following formal details are as previously elaborated in [19]. From [18] we have

$$\hat{f}_n = \underset{f(\theta),\theta\in\Theta_n}{\operatorname{argmin}} \; R(f(\theta), x) + 2s^2 p(n)|\theta| \qquad (1)$$

where s^2 is the signal noise variance and $p(n)$ a monotonically increasing function of the total number of nodes. The finite set Θ_n includes all possible pruning variations (partitions) of a quadtree with n nodes, and θ is one particular partition. Then, for the set of partitions Θ_n, the algorithm will seek the optimal partition θ which minimizes the cost of the resulting field estimator, $\hat{f}_n$. This cost is comprised of two terms. The first term $R(f(\theta), x)$ is the approximation error resulting from the pruned clusters in the partitions. The error is calculated as in

$$R(f(\theta), x) \quad = \quad \sum_{i=1}^{n} (f_i(\theta) - x_i)^2$$

where $f_i(\theta)$ is the estimated value for a node N_i in a particular partition θ and x_i is the true field value. The aim of the second term in (1), $2s^2p(n)|\theta|$, is to penalize increasing complexity, where the factor $|\theta|$ is the number of not pruned nodes in the partition. In [17], $p(n) = 2/3 \log n$ and s^2 is homogeneous on all sensor nodes.

We can solve equation (1) in a distributed manner by using the bottom-up messaging protocol mentioned in Section II-A. The work in [17] confirms that both terms of the estimator are additive functions, thus the error and the penalty cost of a subsquare can be calculated by each corresponding clusterhead independently. Then, following our messaging protocol, a clusterhead in the quadtree hierarchy will receive from its 4 child nodes (three child nodes and itself) the field estimate which minimizes the estimation cost as given by the formula.

In order to implement the field estimation technique in our distributed network, we propose a threshold-based pruning algorithm. We are interested in studying the performance of a fixed-size sensor network in function of a threshold T_k. At layer L_0, there is no propagated error from lower levels, the cost $\hat{f}_i(\theta_{L_1})$ at a clusterhead N_i is thus equal to

$$\hat{f}_i(\theta_{L_1}) = \begin{cases} 8s^2p & \text{if not pruning} \\ R(f_i(\theta_{L_1}), x) + 2s^2p & \text{if pruning} \end{cases}$$

The algorithm will seek the minimal cost $min\{\hat{f}_i(\theta_{L_1})\}$, therefore the threshold on the approximation error $R(f_i(\theta_{L_1}), x)$ for layer L_1 is

$$T_{1,i}(s, p) \quad = \quad 6s^2p$$

In other words, if the approximation error $R(f_i(\theta_{L_1}), x) < T_{1,i}(s, p)$, the cluster will be pruned. For layers L_k with $k > 1$, the estimator takes into account the propagated errors and complexity penalizers from lower level layers, with

$$\hat{f}_i(\theta_{L_k}) = \begin{cases} \sum_{j \in C_{k,i}} \hat{f}_j(\theta_{L_{k-1}}) & \text{if not pruning} \\ R(f_i(\theta_{L_k}), x) + 2s^2p & \text{if pruning} \end{cases}$$

where $C_{k,i}$ is the set of all children nodes of clusterhead N_i at layer k. Since the network size is fixed, p is constant and the threshold $T_k(s)$ for level L_k, $k > 1$ is then

$$T_{k,i}(s) \quad = \quad 6s^2p + \sum_{j \in C_{k,i}} R(f_j(\theta_{L_{k-1}}), x) \qquad (2)$$

3) Branching Algorithm: Sensor networks often deal with non-static environments. In order to take into account these changes in the environment, we extend the pruning algorithm elaborated above in order to enable an adaptive pruning behavior. We develop a branching mechanism, which enables initially pruned nodes to resume their full activities (sampling, data processing and message sending). This behavior is illustrated by the left dashed arrow in the state-machine depicted in Fig. 3. In contrast to the controller described in Section II-B.2, where pruned clusters remain pruned, nodes can now potentially receive reactivation signals enabling entire clusters to *branch*.

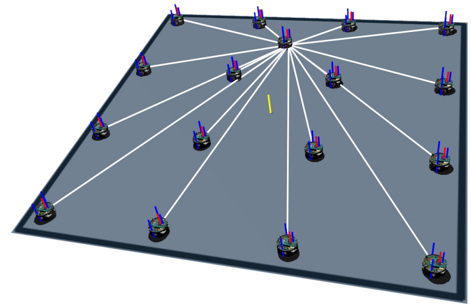


Fig. 5. The figure shows a screenshot from the Webots simulation environment. 16 robotic nodes (e-pucks) are evenly spaces out in a $1.5 \times 1.5\ m^2$ large space. The links show the detection of the acoustic source, a 17^{th} robot, placed in the top half of the arena.

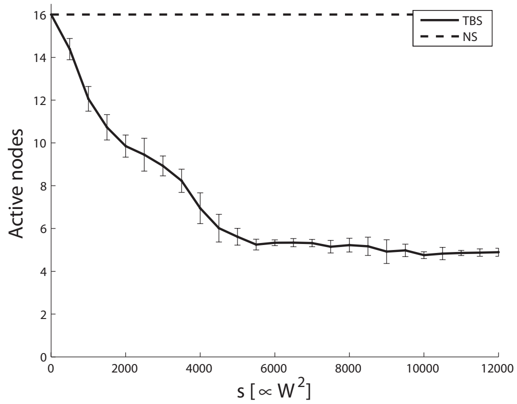
Intuitively, we might implement a simple branching algorithm by defining a constant time interval, at which a branching control message is sent to all nodes within the network. Yet, defining an optimal constant branching interval a-priori may be difficult or even impossible, due to the unknown and unpredictable characteristics of environmental phenomena. Thus, we developed a simple distributed strategy which will branch pruned clusters as a function of change perceived in the environment by the active nodes. This strategy exploits the fact that in a dynamic environment, the boundaries of an anisotropic field are moving. Thus, according to our threshold-based pruning algorithm, in a dynamic environment, active nodes may eventually be pruned as at they no longer cover anisotropic parts of the field. Each time an active node is pruned, it signals the need for a reevaluation of the current quadtree partition. Hence, the quadtree will branch if for a node i

$$R(f_i(\theta_{L_k}), x) \leq T_{k,i}.$$

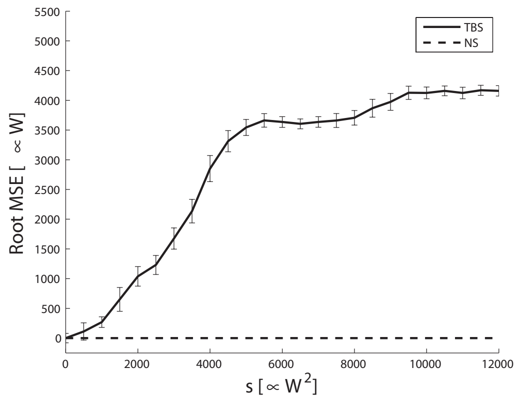
Following this additional threshold-based rule, active nodes in isotropic parts of the field will send branching control messages to pruned nodes in the quadtree.

III. Results

We designed an experimental setup using the robotic simulation software Webots [16]. Our robotic nodes are modeled by simulated e-puck robots [5] (which run on a microcontroller of the dsPIC30 family). The robots have a trinaural microphone array, enabling them to detect acoustic events, and are equiped with radio modules enabling short range communication [5]. An additional robot plays the role of a sound source, which will, depending on the experiment, remain stationary, or move randomly about the arena, avoiding the other robots and boundaries (Braitenberg vehicle with a speed of one robot-size per second). As elaborated in previous work by Cianci et. al. [6], the dynamics of the sound source are accurately modeled, taking into account reflection,

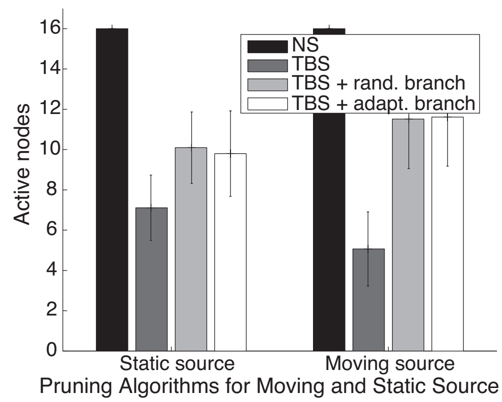


(a)

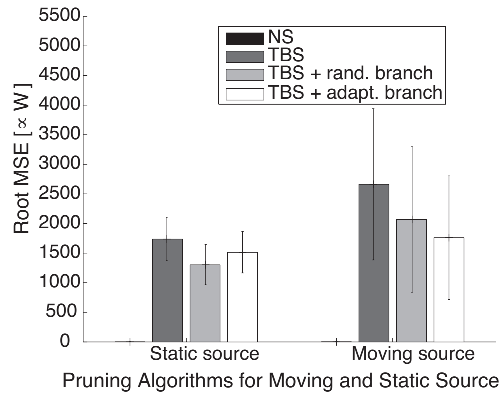


(b)

Fig. 6. Performance with i) NS and ii) TBS. 500 runs were performed per threshold, for 24 different thresholds with s in [0..12000]. (a) Total active nodes (b) MSE. The errorbars show a 95% confidence interval.



(a)



(b)

Fig. 7. The graphs show the MSE and average number of active nodes, for the quadtree structure implemented with three different controlling algorithms. The error-bars show the standard deviation.

fading and mixing. Also in [6], the radio communication is realistically modeled within the simulation software using a plugin based on *OMNeT++* [25], which accurately simulates the physical layer (i.e., with channel fading) and data link layer (i.e. modulation properties, channel coding, MAC protocol).

Fig. 5 shows the experimental setup with 16 robotic nodes spaced out evenly in a $1.5 \times 1.5\,m^2$ arena. The sound source in this setup generates a continuous, local acoustic field. The robotic nodes in the network sample at a frequency of approximately 288 kHz, take measurements at regular intervals of 256 ms, and calculate the power of this acoustic event. Figured 6 (a) and (b) summarize the behavior of the two control variants NS and TBS as elaborated above, with respect to (a) the number of active nodes and (b) the MSE. We performed 500 runs per threshold, for 24 different thresholds with s in [0..12000]. For NS, the total number of active nodes as well as the resulting MSE will remain constant. As expected for TBS, we observe a decreasing number of active nodes and an increasing MSE as the threshold increases.

Figures 7 (a) and (b) show the performance of the sensor network with four variant control algorithms: NS, TBS, TBS with random branching and TBS with adaptive branching. We see that in comparison with the pruning control TBS, adaptive branching reduces the resulting MSE for a moving sound source. Also, the number of active nodes is reduced by over 20% with respect to a fully active network as in NS. Post-evaluation of the data gathered by the adaptive pruning algorithm shows that in 42% of the time, the quadtree was branched. Thus, in order to better evaluate the adaptive pruning controller, we implemented a random branching mechanism with an equivalent branching probability instead of the threshold-based branching rule. We see that for both branching mechanisms the MSE is nearly identical, but that in the case of a dynamic environment, the adaptive algorithm outperforms the random one.

IV. Conclusion & Outlook

In this work we first developed a layer-based fully asynchronous distributed node controller, specific to hierarchical network topologies, and we implemented a self-configuration method based on an estimation technique. Whereas the theory for the estimation technique optimizes communication costs, we decoupled our performance metric by considering a sensor-node as either fully active or shut-down. In our previous work [19], we additionally verified the system's performance on hardware, and developed a probabilistic model that accurately captured the behavior of a real sensor network. Also, we developed a framework which ultimately allows for a specific, user-defined trade-off between the cost and accuracy of a sensor network. Beyond our previous work, this paper explores the feasibility of an augmented node control that envisions the reactivation of nodes absorbed by the idle state through the branching of pruned quadtree nodes. With our simulation results, we showed how the proposed quadtree branching algorithm may lead to significantly reduced resource usage without compromising the quality of the data obtained.

There are a number of possible extensions to this work, but most importantly, the introduction of clusterhead rotation cycles and distributed node responsibilities lead to increased robustness, which is a key factor for large-scale networks. Building upon the current baseline method, we will explore how the controlled movement of robotic sensor nodes affects the spatial resolution of the sensor network as a whole, thus also affecting its performance. Simultaneously, we will explore how to optimally allocate nodes in heterogeneous networks. To this purpose, we will employ SensorScope stations [21] as well as flying robotic vehicles [22] for outdoor operation, offering a promising set of tools for validating our future approaches on mobile systems as well as in outdoor scenarios.

References

[1] Tarik Arici and Yucel Altunbasak. Adaptive sensing for environment monitoring using wireless sensor networks. In *IEEE Wireless Communications and Neworking Conference (WCNC)*, pages 2347–2352, Atlanta, GA, USA, March 2004.

[2] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 85–96, 2001.

[3] Chee Yee Chong and S. P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, August 2003.

[4] D. Chu, A. Deshpande, JM Hellerstein, and W. Hong. Approximate data collection in sensor networks using probabilistic models. In *IEEE ICDE*, pages 48–48, 2006.

[5] Christopher M. Cianci, Xavier Raemy, Jim Pugh, and Alcherio Martinoli. Communication in a swarm of miniature robots: The e-puck as an educational tool for swarm robotics. In *Simulation of Adaptive Behavior (SAB-2006), Swarm Robotics Workshop*, pages 103–115, Rome, Italy, October 2006, Lecture Notes in Computer Science (2007), vol. 4433.

[6] Christopher M. Cianci, Jim Pugh, and Alcherio Martinoli. Exploration of an incremental suite of microscopic models for acoustic event monitoring using a robotic sensor network. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3290–3295, Pasadena, CA, USA, May 2008. IEEE Press.

[7] Christopher Michael Cianci. *Distributed intelligent algorithms for robotic sensor networks monitoring discontinuous anisotropic environmental fields*. PhD thesis, Lausanne, 2009. URL `http://library.epfl.ch/theses/?nr=4247`.

[8] David Culler, Deborah Estrin, and Mani Srivastava. Overview of sensor networks. *IEEE Computer, Special Issue in Sensor Networks*, 37(8):41–49, August 2004.

[9] M. Demirbas and Xuming Lu. Distributed quad-tree for spatial querying in wireless sensor networks. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 3325–3332, June 2007. doi: 10.1109/ICC.2007.551.

[10] T. Dubitzki, A. Y. Wu, and A. Rosenfeld. Parallel region property computation by active quadtree networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(6):626–633, 1981.

[11] R. A. Finkel and J. L. Bentley. Quad trees: A data structure for retrieval on composite keys. *JournalActa Informatica*, (1):1–9, 1974. ISSN 0001-5903.

[12] Antonin Guttman. R-trees: a dynamic index structure for spatial searching. *SIGMOD Rec.*, 14(2):47–57, 1984. ISSN 0163-5808. doi: http://doi.acm.org/10.1145/971697.602266.

[13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, pages 10 pp. vol.2–, Jan. 2000.

[14] S. Sitharama Iyengar and Richard R. Brooks. *Distributed Sensor Networks*. Chapman & Hall/CRC, USA, 2005.

[15] Juan Liu, James Reich, and Feng Zhao. Collaborative in-network processing for target tracking. *EURASIP Journal on Applied Signal Processing*, (4):378–391, 2003. doi: doi:10.1155/S111086570321204X.

[16] Olivier Michel. Webots: Professional mobile robot simulation. *Journal of Advanced Robotic Systems*, 1(1):39–42, 2004.

[17] Robert Nowak and Urbashi Mitra. Boundary estimation in sensor networks: Theory and methods. In *Information Processing in Sensor Networks (IPSN)*, pages 85–90, Palo Alto, CA, USA, April 2003. Springer-Verlag.

[18] Robert Nowak, Urbashi Mitra, and Rebecca Willett. Estimating inhomogeneous fields using wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):999–1006, August 2004.

[19] Amanda Prorok, C.M. Cianci, and Alcherio Martinoli. Towards optimally efficient field estimation with threshold-based pruning in real robotic sensor networks. *IEEE ICRA*, pages 5453–5459, 2010.

[20] Hanan Samet. The quadtree and related hierarchical data structures. *ACM Comput. Surv.*, 16(2):187–260, 1984. ISSN 0360-0300. doi: http://doi.acm.org/10.1145/356924.356930.

[21] Sensorscope Sàrl. `http://www.sensorscope.ch`.

[22] Ascending Technologies. `http://www.asctec.de`.

[23] Daniela Tulone and Samuel Madden. PAQ: Time series forecasting for approximate query answering in sensor networks. In *EWSN*, pages 21–37, 2006.

[24] Daniela Tulone and Samuel Madden. An energy-efficient querying framework in sensor networks for detecting node similarities. In *ACM MSWiM*, pages 191–300, 2006.

[25] Andras Varga. Software tools for networking: "OMNeT++". *IEEE Network Interactive (2002)*, 16(4), July 2002.

[26] Rebecca Willett, Aline Martin, and Robert Nowak. Backcasting: Adaptive sampling for sensor networks. In *Information Processing in Sensor Networks (IPSN)*, pages 124–133, Berkeley, CA, USA, April 2004. ACM Press.

[27] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 70–84, New York, NY, USA, 2001. ACM. ISBN 1-58113-422-3. doi: http://doi.acm.org/10.1145/381677.381685.

[28] Feng Zhao, Jie Liu, Juan Liu, L. Guibas, and J. Reich. Collaborative signal and information processing: an information-directed approach. *Proceedings of the IEEE*, 91(8):1199–1209, Aug. 2003. ISSN 0018-9219. doi: 10.1109/JPROC.2003.814921.

Environmental Monitoring with the Particle Plume Explorer Algorithm

Gonçalo Cabrita and Lino Marques

Abstract— **This paper presents an environmental monitoring application for monitoring gas concentrations achieved by means of an odor guided exploration and plume tracking algorithm (Particle Plume Explorer) through manipulation of the gas mapping algorithm (Particle Plume). This is achieved by giving an expiration time to the gas data being collected so that the robot is forced to continuously revisit previously explored areas, thus arriving at a monitoring-like behavior. The idea was validated through both simulations and real world experiments on two different floors of an indoor environment using a robot equipped with a gas sensor. The results show that the robot is able to visit most of the environment with a good distribution giving however more emphasis to areas where chemicals are present in the air.**

I. INTRODUCTION

Environmental health hazards are slowly slipping into the homes of developed countries. Among the most common hazards is indoor air pollution. This problem is finding its way to the top of the agenda due to the significant burden of disease it imposes. Environmental monitoring can be described as the process of characterizing and monitoring the quality of any given environment. Developments in the field of mobile robotics during recent years have turned the mobile robot into a viable tool for environmental monitoring. Research on the field of odor related mobile robotics has however been focusing on the problem of odor source localization.

The existing solutions to the odor source localization problem can be divided in three families of algorithms, (1) hill climbing or gradient ascent techniques; (2) biologically inspired algorithms; and (3) probabilistic methods.

Gradient based techniques require comparisons between two or more spatially separated chemical measurements, meaning that there must be more than one gas sensor on the same agent or, in the case of a unique sensor, the agent must perform multiple measurements separated in time and space in order to estimate the gradient and make a decision for the next move. This behavior has been widely observed in several insects (e.g. Drosophila) and in some bacteria (e.g. E. coli). However, this type of behavior is satisfactory only if the agent is either close to or inside the plume, otherwise it will perform random movements while trying to re-acquire the plume.

Biologically inspired algorithms have been broadly used in robotics as an attempt to mimic the successful behaviors of animals, in this case while performing odor tracking or odor source declaration. The bacterium E. coli, which is also referred in smooth gradient ascend techniques, presents a chemotaxis behavior that consists of a series of movements that probabilistically lead towards highest concentrations. However, the moth is the being that inspired the most algorithms in this area. The moth performs a set of movements to reach the odor source [1], [2] which consist of moving straight upwind while inside the odor plume - the surge; then, the moth performs counter-turning patterns (zig-zag movements) along the plume and according to the wind direction, aiming to acquire odor cues - the casting. Some silkworm moths [3] also describe some kind of irregular spiral movements when the casting strategy bears no fruits. Along with this case, other bio-inspired algorithms have been implemented and tested in mobile robots to accomplish the task of odor source localization [4], [5], using search strategies like Lévy-taxi [6], [7] or Biased Random Walk [8]. Different approaches of bio-inspired algorithms which rely on large groups of individuals carrying out a task as a whole (swarm behavior) can be found in [9], [10].

The odor source localization problem was also researched employing probabilistic methods such as Hidden Markov Methods [11] or Bayesian methods [12], [13]. These works proposed to localize the odor source by building a probability map of the source localization and, whenever new information was available, updating each cell in the map.

Environmental monitoring is usually achieved by means of a sensor network. The network nodes can be static, mobile or a combination of both. Static wireless sensor networks (WSN) consist of small nodes equipped with sensors capable of measuring the desired phenomena. The available solutions are usually cheap and easy to deploy, even over large areas, both indoors and outdoors. Static WSN have found their way into many monitoring applications, from museums [14] to large glaciers [15]. Mobile robots equipped with multiple sensors can create a mobile WSN. Mobile robot platforms come in many shapes, from small ground robots to unmanned aerial vehicles (UAVs) or even underwater unmanned vehicles (UUVs), allowing for their deployment in almost any scenario. A mobile WSN will ultimately perform the same task a static WSN would, however a small number of mobile sensors is able to achieve a similar spatial resolution to that of a static WSN installed over a larger area. Furthermore a team of robots can be deployed virtually anywhere in a short amount of time, hence being a far more flexible solution [16]. Finally some applications can benefit from the use of both static and mobile WSNs. Mobile robots deployed within an environment equipped with a static WSN can tap into the existing network to access the environmental data

G. Cabrita and L. Marques are with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, 3030-290 Coimbra, Portugal.
{goncabrita, lino}@isr.uc.pt

of the covered area. This allows the robot to make decisions based on this data and get more detailed readings, thus improving the coverage and spatial resolution of the complete system [16].

Traditionally mobile robot environmental monitoring is achieved by means of patrolling or area coverage, or sweeping algorithms. Sensors placed on the robot take readings, thus characterizing the environment. The most common types of chemical variables monitored are: volatile organic compounds (VOCs), air contaminants, and other type of toxic or hazardous gases [17], [18], [19]. So how is environmental monitoring related to odor source localization? Algorithms from the families of patrolling, area coverage or sweeping will produce a monitoring behavior oriented to the surrounding environment, the robot moves in order to optimize the frequency at which each map node is visited or to optimize the covered area. In this paper a solution that produces an odor guided monitoring behavior is proposed, where the chemical data being gathered is not merely being stored, but also being used to guide the robot. To accomplish this the odor guided exploration and plume tracking algorithm Particle Plume Explorer and the Particle Plume gas mapping algorithm were be used.

The remaining of this paper is organized as follows, section II presents the proposed approach for solving the presented problem. In section III one can find the experimental setup used to validate the proposed method. Section IV holds the results and section V the discussion. The conclusions can be found in section VI.

II. ODOR GUIDED ENVIRONMENTAL MONITORING

The goal of this work is to monitor a known indoor environment using an odor guided exploration and plume tracking algorithm. The proposed method focuses on Particle Plume (PP), a plume mapping algorithm and Particle Plume Explorer (PPE), an odor guided exploration and plume tracking algorithm. Although these were presented in [20] an overview of both algorithms follows.

A. Particle Plume

The chemical readings from the gas sensor mounted on the robot are converted into a point cloud around the sensor's location on a global frame, following a gaussian distribution. The number of points generated for each reading is proportional to the chemical reading in parts per million (ppm). The point cloud is confined to a predefined radius around the point of origin. The newly created point cloud is finally added to the plume. If the volume occupied by the new points contains older points, these older points are removed from the plume. This process is represented in Figure 1.

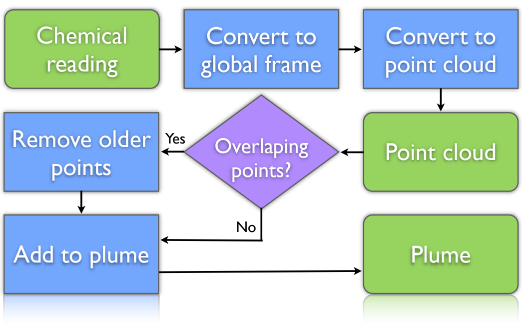


Fig. 1. Flowchart for the Particle Plume plume mapping algorithm.

The result is a low-pass-filter-like behavior, producing a smooth representation of the plume. Point clouds with bigger radius produce smoother plumes while smaller radius result in bigger variations inside the plume. The goal is to tackle the intermittency problem found in gas distribution mapping [21].

Along with the particle cloud representation of the plume PP is in charge of generating a grid of visited cells. This means that it is possible to differentiate between explored areas with no odor detected and unexplored areas.

Due to the fact that a plume is not static PP provides the possibility to give an expiration time to the particles on the plume. This feature is the key in obtaining the monitoring behavior presented in this work.

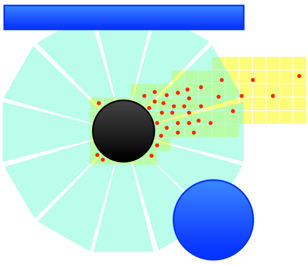
B. Particle Plume Explorer

The PPE algorithm is in charge of odor oriented exploration and plume tracking. This is accomplished in four steps. (1) Initially a circle of radius r is drawn around the robot and divided into n slices. (2) Next the slices which overlap with obstacles are removed. (3) The slices which contain a percentage of explored area over a certain threshold are then also removed. (4) Finally the cost of each remaining slice is calculated using Equation 1, where W_o, W_v and W_d represent the weight of the odor, visited cells and direction, respectively; the *mad()* is the minimum distance angle function; α, β and θ are the angles of the odor, the slice and the robot, respectively; finally, the A_v is the visited area. The slice with the lowest cost is chosen and the goal pose is set. The four stages of the PPE algorithm are illustrated in Figure 2.

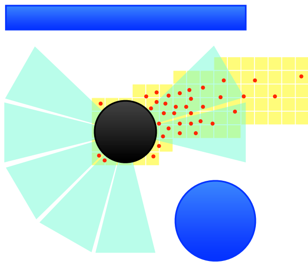
$$S_{cost} = W_o \cdot |mad(\alpha, \theta)| + W_v \cdot A_v + W_d \cdot |mad(\beta, \theta)| \quad (1)$$

There might be situations where all slices are removed before the final stage. When this happens PPE enters a recovery behavior where the area surrounding the robot is scanned for the closest viable spot to continue the exploration. This is done respecting the obstacles present in the environment. Determining if a certain location is suitable for further exploration is accomplished using the four-stage algorithm discusses earlier. If while performing the recovery behavior PPE is unable to locate a suitable pose to continue exploration the algorithm comes to an end.

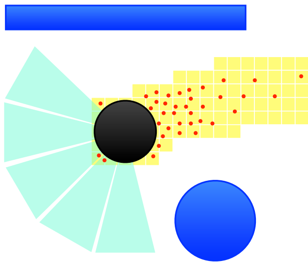
Since the purpose of PPE is odor exploration and plume tracking, areas which appear to have odor cues are given more importance and are thus more thoroughly explored. This is achieved by means of a dynamic threshold in the third stage (Figure 2(c)) of the algorithm. The presence of nearby odor cues results in higher thresholds for the percentage of



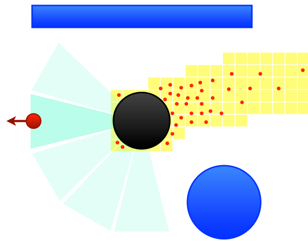
(a) Step 1.



(b) Step 2.



(c) Step 3.



(d) Step 4.

Fig. 2. The four steps of the Particle Plume Explorer algorithm: **a)** Draw a circle of radius r around the robot and divide it in n slices; **b)** Remove the slices which overlap with obstacles; **c)** Remove the slices which contain a percentage of explored area over a certain threshold; **d)** Calculate the goal using the cost function in Equation 1. The robot is represented in black, obstacles in blue, the visited cells in yellow and the particle plume in red.

visited area for each slice while the absence of odor cues results in little tolerance for visited areas.

C. *Achieving the Monitoring Behavior*

As stated previously the monitoring behavior is achieved by using the expiration time on the particles generated by PP. As the robot moves it leaves a trail of visited cells behind which will begin to fade as time goes by. As a result the robot will re-visit previously explored areas. In order for the robot not to explore the environment completely and exit the algorithm the expiration time on the plume should be lower than the time that takes for the robot to complete a full exploration. On the other hand if the expiration time is too low the robot will not leave the same area. The choice for the expiration time will depend on the speed of the robot and the size of the environment being monitored.

D. *Implementation*

Both PP and PPE were implemented in C++ under the ROS framework [22]. All the source code developed for this project is open source and can be downloaded for free at **http://www.ros.org/wiki/pp_explorer**.

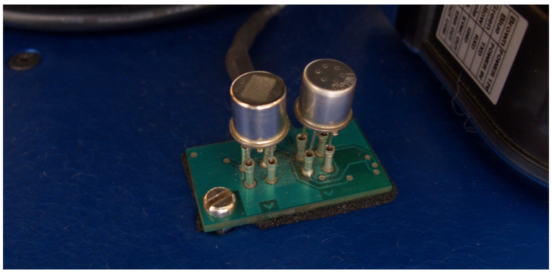
III. EXPERIMENTAL SETUP

For the real world experiments an Erratic robot was used. It was equipped with a Hokuyo LASER range finder and a Microsoft Kinect for obstacle avoidance and a Figaro TGS2620 MOX gas sensor for reading chemical concentrations. The robot can be viewed in Figure 3. The robot was running the ROS navigation stack for autonomous navigation and localization. The metric map of the environment was provided to the robot for each experiment. The Figaro TGS2620 gas sensor is considerably slow, with about 2.0 seconds of rising time and about 15.0 seconds of decay time. In spite for this the maximum robot speed allowed during the experiments was $0.50m/s$, although this produced a drag effect on the plume the desired monitoring effect was still achieved without having to decrease the speed of the robot.

The Figaro TGS2620 MOX sensor was calibrated to provide readings in ppm. The sensor was placed inside a sealed chamber of known volume containing clean air. Next a known quantity of CH_3OH (the chemical used throughout the experiments) was released into the chamber. After the sensor stabilized a sensor reading was recorded along with the current quantity of chemical inside the chamber in ppm. MOX sensors produce changes in resistance as the chemical concentration changes. The procedure was repeated for several calibration points until the sensor was close to saturation. The obtained data set was then fitted by least square minimization to a logarithmic model.

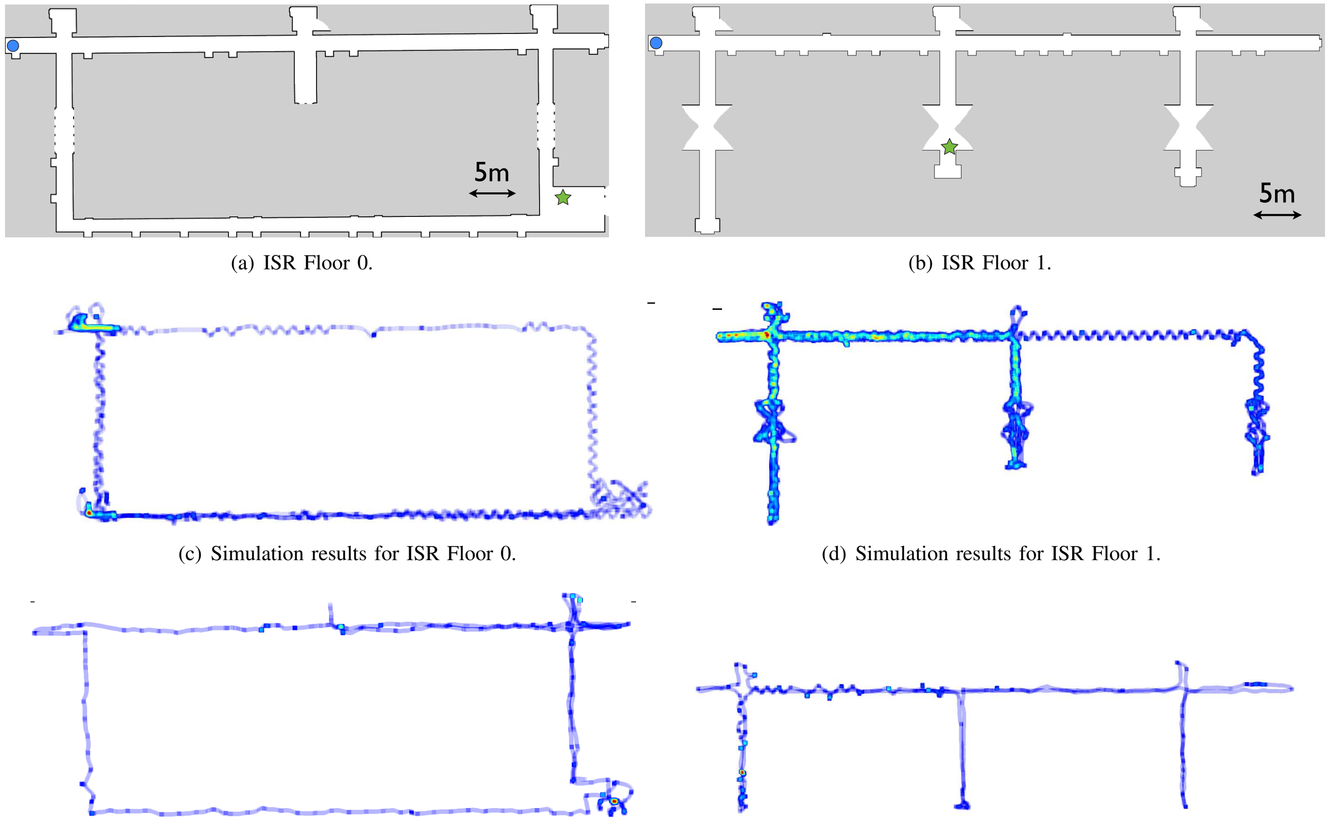
The following set of parameters was used for all the experiments: For PP the sphere radius used was $0.50m$ and the particle life time was $10min$. For PPE the pie radius r used was $2.0m$; the number of slices n used was 32; the visited area, current heading and odor weights used were respectively 1.00, 0.01 and 0.10; the maximum allowed percentage of discovered area per slice on the absence of odor was 20% and for the presence of odor 50%.

(a) The Erratic used for the experiments.



(b) Figaro TGS2620 MOX sensor on the left.

Fig. 3. The Erratic equipped with a Hokuyo LASER range finder, a Microsoft Kinect and a gas sensor.



(a) ISR Floor 0.

(b) ISR Floor 1.

(c) Simulation results for ISR Floor 0.

(d) Simulation results for ISR Floor 1.

(e) Real world results for ISR Floor 0.

(f) Real world results for ISR Floor 1.

Fig. 4. Results of the experiments.

Two real world experiments were performed in floors 0 and 1 of the Institute of Systems and Robotics of the University of Coimbra. For each experiment the robot was allowed to monitor the environment during 1 hour. An experiment was performed on each floor. Each time the robot visited a cell of the map the corresponding cell count was increased allowing to determine how homogeneous the monitoring was during the experiment, ideally the robot should visit each cell map the same number of times. The maps being used have a resolution of $0.05m$ per cell. The maps are shown in Figures 4(a) and 4(b) where the green stars mark the location of the odor source used in the experiments. The chemical substance released during the real world experiments was methanol vapor (CH_3OH), appropriate for the gas sensor being used. It was always released at a point about 0.3m above the ground using the bubbling method.

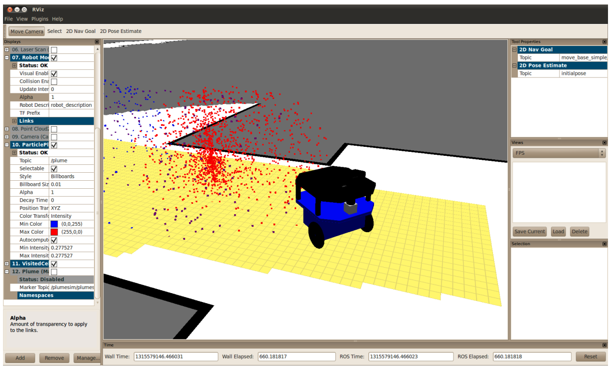
A. Simulated Experiments

Initially a set of simulations was conducted using the robot simulator Stage and the plume simulator PlumeSim [23] on a ROS environment. The robot used for the real world experiments was simulated in the same maps used for the real world experiments to guarantee the simulations were as close as possible to the real world. A PlumeSim gaussian plume model was used having the following parameters, diffusion in x of 0.2, diffusion in y of 0.005, 20 max points per cell, a radius of $0.1m$ and a plume length of $7m$. The location of the odor sources used during the simulated experiments are shown in Figures 4(a) and 4(b) as blue circles. During a first stage the simulations were used to extract the best parameters for PP and PPE. Finally simulations of the real world experiments were produced for longer times than those allowed by the battery of the real robot. Two simulation of 2 hours each were performed, one for floor 0 and another for floor 1 of the Institute of Systems and Robotics of the University of Coimbra. Once more a count was recorded for each time a map cell was visited by the robot.

IV. RESULTS

The results of the experiments can be found in Figure 4. A map of each floor is presented followed by the map cell visit count for each experiment. The less visited cells show up in blue whereas the most visited cells are displayed in red. The graphics in Figures 4(c), 4(d), 4(e) and 4(f) were obtained by applying a sliding window filter of size 10x10 cells to the raw cell count data in order to allow for a better visualization of the data.

It is also possible to see a screenshot of the robot performing a real world experiment in Figure 5. The robot is near the odor source in floor 1 of the ISR University of Coimbra. Figure 5(a) shows a screenshot taken from rviz, a 3D visualization software where it is possible to see the robot, the visited cells in yellow and the particle plume.



(a) Rviz screenshot.

(b) Erratic robot during an experiment.

Fig. 5. The Erratic performing environmental monitoring near the odor source on floor 1 of the ISR University of Coimbra.

Notice that recent particles appear in red. The particle's color slowly turns into blue as they approach the expiration time.

V. DISCUSSION

Analyzing Figures 4(c) and 4(d) it is possible to see that during the simulations the robot visited almost all of the environment having skipped the upper rooms of floor 0 (Figure 4(a)) and the upper right room of floor 1 (Figure 4(b)) which is acceptable. However in floor 1 the robot did not visit the right most corridor during the 2 hours period it was running. In both graphics it is noticeable the tendency of the robot to visit the left part of the maps more often, more precisely the upper left part of the maps, where the odor sources were placed. This is the the result that was intended.

In Figures 4(e) and 4(f) it is possible to see that once more the robot did not visit all the map and that each area was visited less often when in comparison to the simulation results. This is obviously natural as the robot was running for only 1 hour against the 2 hours simulations. Furthermore in the real world the robot has to deal with dynamic obstacles such as people passing by which further delays the advancing of the robot. In floor 1 (Figure 4(f)) the robot did not visit the center upper room which is acceptable. It is also possible to see that the cells near the odor source do not present an increase in activity in comparison to neighbor cells. This is probably due to the fact that the odor source was located in a narrow passage. The middle of each vertical corridor of floor 1 is in fact a glass bridge with metal bars, although in the simulations the robot is allowed to fully explore the hourglass shaped areas in the real world experiments the obstacle avoidance layer does not allow it. Figure 4(e) it is possible to see that the robot failed to visit the left upper room and the small branch at the center. The area near the odor source does however present an increase in activity.

VI. CONCLUSIONS

An environmental monitoring behavior for monitoring gas concentrations was implemented using an odor guided and plume tracking algorithm. The proposed method was validated through simulations and real world experiments on floors 0 and 1 of the Institute of Systems and Robotics of the University of Coimbra. The results show that the robot performed well, having visited most of the designated areas. Furthermore the robot succeeded in exploring the areas where odor was present more extensively, thus achieving the desired effect.

In order for the algorithm to work properly, i.e. for the robot to visit all areas of interest, the expire time on the gas mapping algorithm must be adjusted depending on the environment which is a limitation. Furthermore the robot tens not to visit small branches off the main path. Future work will focus on creating a hybrid solution where the robot will be aware of the need to visit all areas yet optimize its path with regard to the chemical concentrations present in the environment.

The possibility of integrating a fixed sensor network into the algorithm will also be studied, as with little cost it is possible to greatly increase the robustness and effectiveness of the system.

REFERENCES

[1] J. Belanger and E. Arbas, "Behavioral strategies underlying pheromone-modulated flight in moths: lessons from simulation studies," *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, vol. 183, no. 3, pp. 345 – 360, 1998.

[2] T. Lochmatter and A. Martinoli, "Tracking odor plumes in a laminar wind field with bio-inspired algorithms," in *Proc. of the 11th Int. Symposium on Experimental Robotics 2008 (ISER 2008)*. Springer, 2009, pp. 473–482.

[3] R. A. Russell, *Odour Detection by Mobile Robots*, ser. Robotics and Intelligent Systems. World Scientific, 1999, vol. 22.

[4] L. Marques, U. Nunes, and A. de Almeida, "Olfaction-based mobile robot navigation," *Thin Solid Films*, vol. 418, no. 1, pp. 51–58, 2002, selected from 1st Int. School on Gas Sensors.

[5] R. Russell, A. Bab-Hadiashar, R. Shepherd, and G. Wallace, "A comparison of reactive robot chemotaxis algorithms," *Robotics and Autonomous Systems*, vol. 45, no. 2, pp. 83–97, 2003.

[6] Z. Pasternak, F. Bartumeus, and F. W. Grasso, "Lvy-taxis: a novel search strategy for finding odor plumes in turbulent flow-dominated environments," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 43, 2009. [Online]. Available: http://stacks.iop.org/1751-8121/42/i=43/a=434010

[7] S. Nurzaman, Y. Matsumoto, Y. Nakamura, S. Koizumi, and H. Ishiguro, "Biologically inspired adaptive mobile robot search with and without gradient sensing," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (IROS 2009)*, 2009, pp. 142 – 147.

[8] A. Dhariwal, G. Sukhatme, and A. Requicha, "Bacterium-inspired robots for environmental monitoring," in *Proc. of the IEEE Int. Conf.on Robotics and Automation, (ICRA 2004)*, vol. 2, 2004, pp. 1436 – 1443.

[9] L. Marques, U. Nunes, and A. de Almeida, "Particle swarm-based olfactory guided search," *Autonomous Robots*, vol. 20, no. 3, pp. 277–287, 2006, special Issue on Mobile Robot Olfaction.

[10] V. Gazi and K. Passino, *Swarm Stability and Optimization*. Springer, 2011.

[11] J. A. Farrell, S. Pang, and W. Li, "Plume mapping via hidden markov methods," *IEEE Trans. on Systems, Man, and Cybernetics - Part B*, vol. 33, no. 6, pp. 850–863, 2003.

[12] S. Pang and J. Farrell, "Chemical plume source localization," *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 36, no. 5, pp. 1068 – 1080, 2006.

[13] S. Pang, "Plume source localization for AUV based autonomous hydrothermal vent discovery," in *OCEANS '10*, 2010, pp. 1 – 8.

[14] D. Abruzzese, M. Angelaccio, B. Buttarazzi, R. Giuliano, L. Miccoli, and A. Vari, "Long life monitoring of historical monuments via wireless sensors network," in *Proc. of the 6th Int. Conf. on Symposium on Wireless Communication Systems*. IEEE Press, 2009, pp. 570–574.

[15] K. Rose, J. Hart, and K. Martinez, "Seasonal changes in basal conditions at Briksdalsbreen, Norway: the winter–spring transition," *Boreas*, vol. 38, no. 3, pp. 579–590, 2009.

[16] R. Stranders, "Coordinating Teams of Mobile Sensors for Monitoring Environmental Phenomena," School of Electronics and Computer Science, University of Southampton, Tech. Rep., May 2009. [Online]. Available: http://eprints.ecs.soton.ac.uk/17535/

[17] R. Shepherd, S. Beirne, K. Lau, B. Corcoran, and D. Diamond, "Monitoring chemical plumes in an environmental sensing chamber with a wireless chemical sensor network," *Sensors and Actuators B: chemical*, vol. 121, no. 1, pp. 142–149, 2007.

[18] M. A. Pillai, S. Veerasingam, and Y. S. D., "Implementation of Sensor Network for Indoor Air Quality Monitoring Using CAN Interface," in *IEEE Int. Conf. on Advances in Computer Engineering*, Los Alamitos, CA, USA, 2010, pp. 366–370.

[19] J. Hayes, S. Beirne, K. Lau, and D. Diamond, "Evaluation of a low cost wireless chemical sensor network for environmental monitoring," *IEEE Sensors*, pp. 530–533, 2008.

[20] G. Cabrita, P. Sousa, and L. Marques, "Odor guided exploration and plume tracking - particle plume explorer," in *European Conference on Mobile Robots (ECMR 2011)*, 2011.

[21] A. Loutfi, S. Coradeschi, A. Lilienthal, and J. Gonzalez, "Gas distribution mapping of multiple odour sources using a mobile robot," *Robotica*, vol. 27, no. 02, pp. 311–319, 2008.

[22] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[23] G. Cabrita, P. Sousa, and L. Marques, "Player/stage simulation of olfactory experiments," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2010)*, 2010, pp. 1120 – 1125.

Decentralized Multi-Robot Active Exploration for Probabilistic Classification of Hotspots

Kian Hsiang Low, Jie Chen, John M. Dolan, Steve Chien, and David R. Thompson

***Abstract*— A central problem in environmental sensing and monitoring is to classify/label the hotspots in a large-scale environmental field. This paper presents a novel decentralized multi-robot active exploration (DEC-MAX) strategy for probabilistic classification/labeling of hotspots in a Gaussian process-based field. In contrast to existing state-of-the-art exploration strategies for learning environmental field maps, the time needed to solve the DEC-MAX strategy is independent of the map resolution and the number of robots, thus making it practical for *in situ*, real-time active sampling. Its exploration behavior exhibits an interesting formal trade-off between that of boundary tracking until the hotspot region boundary can be accurately predicted and wide-area coverage to find new boundaries in sparsely sampled areas to be tracked. We provide a theoretical guarantee on the active exploration performance of the DEC-MAX strategy: under conditional independence assumption, we prove that it can optimally achieve two formal cost-minimizing exploration objectives based on the misclassification and entropy criteria. Importantly, this result implies that the uncertainty of labeling the hotspots in a GP-based field is greatest at or close to the hotspot region boundaries. Empirical evaluation on real-world plankton density and temperature field data shows that, subject to limited observations, the DEC-MAX strategy can achieve better classification of the hotspots than state-of-the-art active exploration strategies.**

I. INTRODUCTION

A fundamental problem in environmental sensing and monitoring is to identify and delineate the hotspot regions in a large-scale environmental field [1], [2]. It involves partitioning the area spanned by the field into one class of regions called the hotspot regions in which the field measurements exceed a predefined threshold, and the other class of regions where they do not. Such a problem arises in many real-world applications such as precision agriculture, monitoring of ocean and freshwater phenomena (e.g., plankton bloom), forest ecosystems, rare species, pollution (e.g., oil spill), or contamination (e.g., radiation leak). In these applications, it is necessary to assess the spatial extent and shape of the hotspot regions accurately due to severe economic, environmental, and health implications, as discussed in [2]. In practice, this aim is non-trivial to achieve because the constraints on the sampling assets' resources (e.g., energy consumption, mission time, sensing range) limit the number and coverage of *in situ* observations over the large field that can be used to infer the hotspot regions. Subject to limited observations, the most informative ones should therefore be selected in order to minimize the uncertainty of estimating the hotspot regions (or, equivalently, classifying/labeling the hotspots) in the large field, which motivates our adaptive sampling work in this paper.

Mobile robot teams are particularly desirable for performing the above environmental sensing task because they can actively explore to map the hotspot regions at high resolution. On the other hand, static sensors lack mobility and are therefore not capable of doing this well unless a large quantity is deployed. While research in multi-robot exploration and mapping have largely focused on the conventional task of building occupancy grids [3], some recent efforts are put into the more complex, general task of sampling spatially distributed environmental fields [4], [5]. In contrast to occupancy grids that assume *discrete, independent* cell occupancies, environmental fields are characterized by *continuous-valued, spatially correlated* measurements, properties of which cannot be exploited by occupancy grid mapping strategies to select the most informative observation paths. To exploit such properties, exploration strategies for learning environmental field maps have recently been developed and can be classified into two regimes: (a) *wide-area coverage* strategies [4], [5], [6] consider sparsely sampled (i.e., largely unexplored) areas to be of high uncertainty and consequently spread observations evenly across the field; (b) *hotspot sampling* strategies [7] assume areas of high uncertainty and interest to contain extreme, highly-varying measurements and hence produce clustered observations. Formal, principled approaches of exploration [4], [5] have also been devised to simultaneously perform hotspot sampling when a hotspot region is found as well as wide-area coverage to search for new hotspot regions in sparsely sampled areas. These strategies optimize their observation paths to minimize the uncertainty (e.g., in terms of mean-squared error or entropy) of mapping the entire continuous-valued field. They are, however, suboptimal for classifying/labeling the hotspots in the field, which we will discuss and demonstrate theoretically and empirically in this paper.

This paper proposes a novel decentralized multi-robot active exploration (DEC-MAX) strategy for probabilistic classification/labeling of hotspots in a large-scale environmental field (Section V). The environmental field is assumed to be realized from a rich class of probabilistic spatial models called Gaussian process (GP) (Section II) that can

K. H. Low and J. Chen are with the Department of Computer Science, National University of Singapore, Republic of Singapore. {lowkh, chenjie}@comp.nus.edu.sg

J. M. Dolan is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. jmd@cs.cmu.edu

S. Chien and D. R. Thompson are with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA. {steve.chien, david.r.thompson}@jpl.nasa.gov

A portion of this work was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

formally characterize its spatial correlation structure. More importantly, it can provide formal measures of classification/labeling uncertainty (i.e., in the form of cost functions) such as the misclassification and entropy criteria (Section III) for directing the robots to explore highly uncertain areas of the field. The chief impediment to using these formal criteria is that they result in cost-minimizing exploration strategies (Section IV), which cannot be solved in closed form. To resolve this, they are reformulated as reward-maximizing dual strategies, from which we can then derive the approximate DEC-MAX strategy to be solved in closed form efficiently. The specific contributions of our work include:

- analyzing the time complexity of solving the DEC-MAX strategy: we prove that its incurred time is independent of the map resolution and the number of robots, thus making it practical for *in situ*, real-time active sampling. In contrast, existing state-of-the-art exploration strategies [4], [5], [6] for learning environmental field maps scale poorly with increasing map resolution and/or number of robots;
- analyzing the exploration behavior of the DEC-MAX strategy through its formulation: it exhibits an interesting formal trade-off between that of boundary tracking until the hotspot region boundary can be accurately predicted and wide-area coverage to find new boundaries in sparsely sampled areas to be tracked. In contrast, ad hoc, reactive boundary tracking strategies [8], [9] typically require a hotspot region boundary to be located manually or via random exploration and are not driven by the need to maximize the fidelity of estimating multiple hotspot regions given limited observations;
- providing theoretical guarantee on the active exploration performance of the DEC-MAX strategy: we prove that, under conditional independence assumption, it produces the same optimal observation paths as that of the centralized cost-minimizing strategies, the latter of which otherwise cannot be solved in closed form. This result has a simple but important implication: the uncertainty of labeling the hotspots in a GP-based field is greatest at or close to the hotspot region boundaries;
- empirically evaluating the active exploration performance of the DEC-MAX strategy on real-world plankton density and temperature field data: subject to limited observations, the DEC-MAX strategy can achieve better classification of the hotspots than state-of-the-art active exploration strategies [5], [10].

II. Gaussian Process-Based Environmental Field

The Gaussian process (GP) can be used to model an environmental field as follows: the environmental field is defined to vary as a realization of a GP. Let $\mathcal{X}$ be a set of sampling locations representing the domain of the environmental field such that each location $x \in \mathcal{X}$ is associated with a realized (random) measurement y_x (Y_x) if x is sampled/observed (unobserved). Let $\{Y_x\}_{x\in\mathcal{X}}$ denote a GP, that is, every finite subset of $\{Y_x\}_{x\in\mathcal{X}}$ has a multivariate Gaussian distribution [11]. The GP is fully specified by its prior mean $\mu_x \triangleq \mathbb{E}[Y_x]$ and covariance $\sigma_{xs} \triangleq \text{cov}[Y_x, Y_s]$ for all $x, s \in \mathcal{X}$. In the experiments (Section VI), we assume that the GP is second-order stationary, i.e., it has a constant *prior* mean and a stationary *prior* covariance structure (i.e., σ_{xs} is a function of $x - s$ for all $x, s \in \mathcal{X}$). The prior mean and covariance structure of the GP are assumed to be known. Let $\mathcal{S}$ denote a subset of locations of $\mathcal{X}$ sampled a priori (either by the robot team or other sampling assets) and $y_\mathcal{S}$ be a row vector of corresponding measurements. Given the set $\mathcal{S}$ of sampled locations and corresponding measurements $y_\mathcal{S}$, the distribution of Y_x at any unobserved location $x \in \mathcal{X} \setminus \mathcal{S}$ remains Gaussian with the following *posterior* mean and variance

$$\mu_{x|\mathcal{S}} = \mu_x + \Sigma_{x\mathcal{S}}\Sigma_{\mathcal{S}\mathcal{S}}^{-1}\{y_\mathcal{S} - \mu_\mathcal{S}\}^\top \quad (1)$$

$$\sigma^2_{x|\mathcal{S}} = \sigma^2_x - \Sigma_{x\mathcal{S}}\Sigma_{\mathcal{S}\mathcal{S}}^{-1}\Sigma_{\mathcal{S}x} \quad (2)$$

where $\mu_\mathcal{S}$ is a row vector with mean components μ_s for every location $s \in \mathcal{S}$, $\Sigma_{x\mathcal{S}}$ is a row vector with covariance components σ_{xs} for every location $s \in \mathcal{S}$, $\Sigma_{\mathcal{S}x}$ is the transpose of $\Sigma_{x\mathcal{S}}$, and $\Sigma_{\mathcal{S}\mathcal{S}}$ is a covariance matrix with components $\sigma_{ss'}$ for every pair of locations $s, s' \in \mathcal{S}$. To map the entire field, the measurements at its unobserved areas can be predicted using the posterior mean (1) and the uncertainty of each of these point-based predictions is represented by the posterior variance (2). An important property of GP is that the posterior variance $\sigma^2_{x|\mathcal{S}}$ (2) is independent of the observed measurements $y_\mathcal{S}$.

If the environmental field evolves over time, then its domain is extended to include the temporal dimension: let $\mathcal{X}$ instead denote a set of *spatiotemporal* inputs such that each input $x \in \mathcal{X}$ comprises both the spatial location and time. The rest of the GP model formulation remains unchanged.

III. Cost Functions

Recall that the exploration objective is to select observation paths that minimize the uncertainty of estimating the hotspot regions in the field. To achieve this, formal measures of uncertainty (specifically, in the form of cost functions) have to be defined. Let us first consider the feasibility of using cost functions that quantify the uncertainty of mapping the entire continuous-valued field, such as (a) sum of posterior variances (2) over the unobserved locations in $\mathcal{X} \setminus \mathcal{S}$ [4]

$$\sum_{x\in\mathcal{X}\setminus\mathcal{S}} \sigma^2_{x|\mathcal{S}}$$

and (b) posterior joint entropy of the measurements $Y_{\mathcal{X}\setminus\mathcal{S}}$ at the unobserved locations in $\mathcal{X} \setminus \mathcal{S}$ [5]

$$\mathbb{H}[Y_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}] \triangleq -\int P(y_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}) \log P(y_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}) \; \mathrm{d}y_{\mathcal{X}\setminus\mathcal{S}} \, .$$

These cost functions have been utilized in [4], [5] to guide exploration: the resulting active exploration strategies for learning GP-based field maps are non-adaptive and perform wide-area coverage, that is, observation paths are distributed evenly across the field. Do these wide-area coverage strategies also optimize our exploration objective or should observation paths be directed to sample specific features of the field instead? In the rest of this paper, we will show that, by defining cost functions to measure the uncertainty of classifying the hotspots in the field, our objective can be better achieved by performing the latter.

Let us begin by framing the problem of estimating the hotspot regions in a field formally as one of classifying/labeling the hotspots in the field: A location x is defined as a hotspot if its corresponding field measurement Y_x is greater than or equal to a predefined threshold, denoted by γ. Let $\{Z_x\}_{x\in\mathcal{X}}$ denote a binary random process such that Z_x is an indicator variable of label 1 if $Y_x \geq \gamma$ (i.e., location x is a hotspot), and label 0 otherwise. Then, our problem of estimating the hotspot regions is equivalent to one of labeling the hotspots in the field, specifically, by predicting the label of Z_x for every location $x \in \mathcal{X}$. As a result, our exploration objective can be achieved through the use of cost functions that measure the uncertainty of labeling the hotspots in the field. Two such cost functions will be defined next.

Let $\widehat{Z}_x$ be the predicted label of Z_x for every location $x \in \mathcal{X}$ and the cost of predicting (or, more precisely, misclassifying) the label of Z_x with $\widehat{Z}_x$ be denoted by the following $0-1$ loss function

$$L(Z_x, \widehat{Z}_x) = \left|Z_x - \widehat{Z}_x\right| = \begin{cases} 1 & \text{if } Z_x \neq \widehat{Z}_x \ , \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

That is, (3) counts a false positive (i.e., the location x is labeled as a hotspot but it is not) or false negative (i.e., x is not labeled as a hotspot but it is) as a misclassification. If Z_x is unlabeled (i.e., location x is unobserved), then we calculate the *expected* cost (or risk) of predicting the label of Z_x with $\widehat{Z}_x$ instead, which is denoted by

$$\begin{aligned} R_{\widehat{Z}_x|\mathcal{S}} &= \textstyle\sum_{i=0}^{1} L(Z_x = i, \widehat{Z}_x)\ P(Z_x = i|y_\mathcal{S}) \\ &= \widehat{Z}_x\ (1 - P(Z_x = 1|y_\mathcal{S})) + (1 - \widehat{Z}_x)\ P(Z_x = 1|y_\mathcal{S}) \\ &= P(\widehat{Z}_x \neq Z_x|y_\mathcal{S}) \end{aligned} \tag{4}$$

where $P(Z_x = 1|y_\mathcal{S}) = P(Y_x \geq \gamma|y_\mathcal{S})$, the second equality results from $P(Z_x = 0|y_\mathcal{S}) = 1 - P(Z_x = 1|y_\mathcal{S})$, and the last equality states that the risk (4) is equal to the probability of misclassification.

The risk (4) is minimized by the Bayes decision/classification rule

$$\begin{aligned} \widehat{Z}^*_x &= \begin{cases} 1 & \text{if } P(Z_x = 1|y_\mathcal{S}) \geq 0.5 \ , \\ 0 & \text{otherwise.} \end{cases} \\ &= \underset{i\in\{0,1\}}{\arg\max}\, P(Z_x = i|y_\mathcal{S}) \ . \end{aligned}$$

Using $\widehat{Z}^*_x$ as the predicted label of Z_x, the risk (4) reduces to

$$R_{\widehat{Z}^*_x|\mathcal{S}} = \min\left(P(Z_x = 1|y_\mathcal{S}), 1 - P(Z_x = 1|y_\mathcal{S})\right) \ . \tag{5}$$

Consequently, the sum of risks (or expected number of misclassifications) over the unobserved locations in $\mathcal{X} \setminus \mathcal{S}$ is

$$\sum_{x\in\mathcal{X}\setminus\mathcal{S}} R_{\widehat{Z}^*_x|\mathcal{S}} \ , \tag{6}$$

which defines our first cost function. We call this (6) the *misclassification criterion*.

The second cost function, which we call the *entropy criterion*, is defined as the posterior joint entropy of the labels of $Z_{\mathcal{X}\setminus\mathcal{S}}$ at the unobserved locations in $\mathcal{X} \setminus \mathcal{S}$

$$\mathbb{H}[Z_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}] \ . \tag{7}$$

IV. Centralized Active Exploration

In this section, we will formulate greedy cost-minimizing exploration strategies based on the misclassification (6) and entropy (7) criteria defined in Section III. Unfortunately, these centralized strategies cannot be evaluated in closed form, as explained in this section. To resolve this, these cost-minimizing strategies must first be reformulated as reward-maximizing dual strategies, from which we can then derive the approximate DEC-MAX strategy (Section V) to be solved in closed form efficiently.

Supposing the misclassification criterion (6) is used and a set $\mathcal{S}$ of locations are previously sampled, the exploration strategy for directing a team of k robots has to select the next set $\mathcal{O} \subseteq \mathcal{X} \setminus \mathcal{S}$ of k locations to be observed that minimize the sum of *expected* risks:

$$\min_{\mathcal{O}} \sum_{x\in\mathcal{X}\setminus\mathcal{S}} \mathbb{E}_{Y_\mathcal{O}|y_\mathcal{S}} \left\{R_{\widehat{Z}^*_x|\mathcal{S}\bigcup\mathcal{O}}\right\} \ . \tag{8}$$

This cost-minimizing strategy (8) can be reformulated as the following reward-maximizing dual strategy, which selects the next set $\mathcal{O}$ of locations to be observed that maximize the sum of expected risk reductions:

$$\max_{\mathcal{O}} \sum_{x\in\mathcal{X}\setminus\mathcal{S}} R_{\widehat{Z}^*_x|\mathcal{S}} - \mathbb{E}_{Y_\mathcal{O}|y_\mathcal{S}} \left\{R_{\widehat{Z}^*_x|\mathcal{S}\bigcup\mathcal{O}}\right\} \ . \tag{9}$$

The equivalence between these two strategies follows immediately from observing that the first term $\sum_{x\in\mathcal{X}\setminus\mathcal{S}} R_{\widehat{Z}^*_x|\mathcal{S}}$ in (9) remains constant with any choice of $\mathcal{O}$. Both strategies cannot be solved exactly due to the expectation term, which cannot be evaluated in closed form.

If the entropy criterion (7) is used instead, then the exploration strategy has to select the next set $\mathcal{O}$ of locations to be observed that minimize the *expected* posterior joint entropy of the labels of $Z_{\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})}$:

$$\min_{\mathcal{O}} \ \mathbb{E}_{Z_\mathcal{O}|y_\mathcal{S}} \left\{\mathbb{H}[Z_{\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})}|y_\mathcal{S}, Z_\mathcal{O}]\right\} \ . \tag{10}$$

This cost-minimizing strategy (10) can be reformulated as the following reward-maximizing dual strategy, which selects the next set $\mathcal{O}$ of locations with maximum label entropy to be observed:

$$\max_{\mathcal{O}} \ \mathbb{H}[Z_\mathcal{O}|y_\mathcal{S}] \ . \tag{11}$$

To show their equivalence, $\mathbb{H}[Z_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}]$ (7) is first expanded using chain rule of entropy:

$$\mathbb{H}[Z_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}] = \mathbb{H}[Z_\mathcal{O}|y_\mathcal{S}] + \mathbb{E}_{Z_\mathcal{O}|y_\mathcal{S}} \left\{\mathbb{H}[Z_{\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})}|y_\mathcal{S}, Z_\mathcal{O}]\right\} \ . \tag{12}$$

From (12), since $\mathbb{H}[Z_{\mathcal{X}\setminus\mathcal{S}}|y_\mathcal{S}]$ is a constant, the choice of $\mathcal{O}$ that maximizes $\mathbb{H}[Z_\mathcal{O}|y_\mathcal{S}]$ (i.e., (11)) minimizes $\mathbb{E}_{Z_\mathcal{O}|y_\mathcal{S}} \left\{\mathbb{H}[Z_{\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})}|y_\mathcal{S}, Z_\mathcal{O}]\right\}$ (i.e., (10)). When $|\mathcal{O}| = k \geq 2$, both strategies cannot be solved exactly due to the entropy terms, which contain multivariate Gaussian cumulative distribution functions that cannot be evaluated in closed form.

V. Decentralized Active Exploration

This section presents a novel decentralized multi-robot active exploration (DEC-MAX) strategy that can approximately achieve both cost-minimizing exploration objectives (8) and (10) (Section IV) based on the misclassification and entropy criteria, respectively. Unlike the centralized cost-minimizing and reward-maximizing exploration strategies (Section IV), the DEC-MAX strategy can be solved in closed form efficiently.

The DEC-MAX strategy for directing each of the k robots has to select the next location $x \in \mathcal{X} \setminus \mathcal{S}$ to be observed that trades off between (a) minimizing the difference between its predicted measurement $\mu_{x|\mathcal{S}}$ and the boundary threshold γ, and (b) maximizing the square root of its posterior variance $\sigma^2_{x|\mathcal{S}}$:

$$\min_x \ |\gamma - \mu_{x|\mathcal{S}}| / \sigma_{x|\mathcal{S}} \ . \tag{13}$$

Intuitively, the behavior of the DEC-MAX strategy exhibits an interesting trade-off between that of (a) boundary tracking and (b) wide-area coverage: it simultaneously tracks a hotspot region boundary that is found until it can be accurately predicted as well as searches for new hotspot region boundaries in sparsely sampled areas to be tracked.

In this paper, the domain $\mathcal{X}$ of the field is assumed to be a grid of sampling locations. The next location x to be observed by each robot is then constrained to be selected from the 4-connected neighborhood $\mathcal{N}$ of the robot's current location instead of from $\mathcal{X} \setminus \mathcal{S}$.

Theorem 1 (Time Complexity): Solving the DEC-MAX strategy (13) requires $\mathcal{O}(|\mathcal{S}|^2(|\mathcal{S}| + |\mathcal{N}|))$ time.
The above result reveals that the time needed to compute the DEC-MAX strategy is independent of the map resolution (i.e., domain size $|\mathcal{X}|$) and the number k of robots, thus making it practical for *in situ*, real-time active sampling.

Theorem 2 (Communication Overhead): Let the communication overhead be the number of broadcast messages sent by each robot over the network. Then, the communication overhead of DEC-MAX strategy (13) is $\mathcal{O}(1)$.
In terms of data sharing, each robot broadcasts a message to the other robots sharing its sampled observations since its last broadcast. Coordination between robots is needed only if their neighborhoods intersect: in this case, they may select the same next location to be observed. To avoid this, each robot can broadcast on the same or another message sharing its selected location to be observed next.

Under conditional independence assumption, the DEC-MAX strategy (13) produces the same observation paths as that of the centralized cost-minimizing strategies (8) and (10) (Section IV), as established in the result below:

Theorem 3 (Performance Guarantee): If the unobserved measurements $Y_{\mathcal{X}\setminus\mathcal{S}}$ are conditionally independent given the sampled measurements $y_{\mathcal{S}}$, then the DEC-MAX strategy (13) is equivalent to both cost-minimizing strategies (8) and (10) based on the misclassification and entropy criteria.
The proof of the above result can be found in Appendix A. The proof construction in fact describes how the DEC-MAX strategy (13) can be derived from either reward-maximizing dual strategy (i.e., (9) or (11)) (Section IV). A simple but important implication of this result is that the uncertainty of estimating the hotspot regions in a GP-based field (i.e., in terms of misclassification or entropy criterion) is greatest at or close to the hotspot region boundaries.

In practice, how restrictive is the conditional independence assumption? We conjecture that the assumption becomes less restrictive (i.e., Theorem 3 becomes more reliable) when the number $|\mathcal{S}|$ of sampled locations increases to potentially reduce the degree of violation of conditional independence, the spatial correlation between field measurements decreases, and the robots are sufficiently far apart (this last case applies only to the entropy criterion).

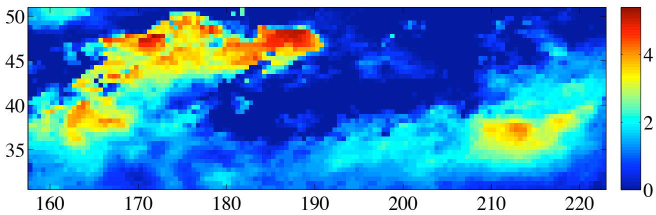


Fig. 1. Temperature field bounded within lat. $30.75 - 50.75$N and lon. $157.75 - 222.25$E: γ is set to $3\,^\circ$C, which results in a hotspot region in the top left and another one in the bottom right.

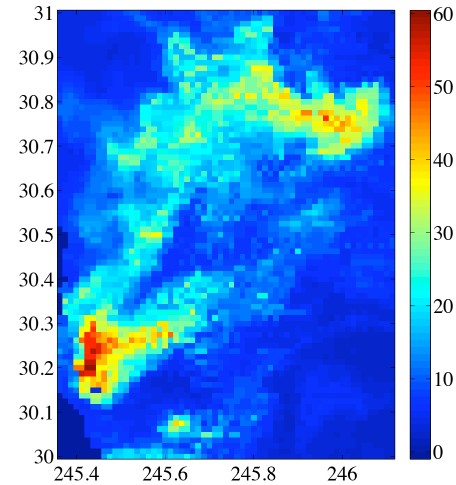


Fig. 2. Plankton density field bounded within lat. $30 - 31$N and lon. $245.3625 - 246.1125$E: γ is set to 30 mg/m^3, which results in a hotspot region in the top right and another one in the bottom left.

VI. Experiments and Discussion

This section evaluates the active exploration performance of the DEC-MAX strategy (13) empirically on 2 real-world spatial datasets off the west coast of USA: (a) August 2009 AVHRR temperature data (Fig. 1), and (b) March 2009 MODIS plankton density data (Fig. 2). These regions are discretized, respectively, into (a) 130×41 (i.e., $|\mathcal{X}| = 5330$) and (b) 61×81 (i.e., $|\mathcal{X}| = 4941$) grids of sampling locations. Each location x is, respectively, associated with (a) temperature measurement y_x in $^\circ$C, and (b) chlorophyll-a (chl-a) measurement y_x in mg/m^3. Using a team of $k = 2, 4, 8$ robots, each robot is tasked to, respectively, explore $1250, 625, 312$ locations in its path to sample a total of about 2500 observations. The robot team is given 120 randomly selected observations as prior data before exploration. We

TABLE I

COMPARISON OF ACTIVE EXPLORATION STRATEGIES (WC: WIDE-AREA COVERAGE, HS: HOTSPOT SAMPLING, BT: BOUNDARY TRACKING).

Exploration strategy	Behavior	Coordination type	Time complexity	Map resolution $\lvert\mathcal{X}\rvert$	Number k of robots
Maximize mutual information [6]	WC	Centralized	$\mathcal{O}(\lvert\mathcal{N}\rvert^k\lvert\mathcal{X}\rvert^2(\lvert\mathcal{X}\rvert+k^2))$	Cubic	Exponential
Minimize sum of variances [4]	WC	Centralized	$\mathcal{O}(\lvert\mathcal{N}\rvert^k\lvert\mathcal{S}\rvert^2\lvert\mathcal{X}\rvert)$	Linear	Exponential
Maximum entropy sampling (MES) [5]	WC	Centralized	$\mathcal{O}(\lvert\mathcal{N}\rvert^k\lvert\mathcal{S}\rvert^2(\lvert\mathcal{S}\rvert+k^2))$	Independent	Exponential
MES coupled with HS [5]	WC+HS	Centralized	$\mathcal{O}(\lvert\mathcal{N}\rvert^k\lvert\mathcal{S}\rvert^2(\lvert\mathcal{S}\rvert+k^2))$	Independent	Exponential
Straddle [10]	WC+BT	Decentralized[1]	$\mathcal{O}(\lvert\mathcal{S}\rvert^2(\lvert\mathcal{S}\rvert+\lvert\mathcal{N}\rvert))$	Independent	Independent
DEC-MAX	WC+BT	Decentralized	$\mathcal{O}(\lvert\mathcal{S}\rvert^2(\lvert\mathcal{S}\rvert+\lvert\mathcal{N}\rvert))$	Independent	Independent

use 2000 randomly selected observations to learn the hyperparameters (i.e., mean and covariance structure) of GP through maximum likelihood estimation [11].

Since the domains $\mathcal{X}$ of both fields are considerably large, it is prohibitively expensive to compare meaningfully with the wide-area coverage strategies [4], [6] that scale poorly with increasing map resolution and are thus not practical for *in situ*, real-time active sampling. For example, it was reported in [12] that the greedy mutual information-based strategy of [6] incurred more than 62 hours to generate paths for 3 robots to sample a total of 267 observations in a grid of only $|\mathcal{X}| = 1424$ locations. The performance of the DEC-MAX strategy is therefore compared to that of three state-of-the-art exploration strategies whose incurred times are independent of the map resolution: (a) The decentralized[1] *straddle* strategy [10] for directing each robot selects the next location x to be observed using $\max_x 1.96\sigma_{x|\mathcal{S}} - |\gamma - \mu_{x|\mathcal{S}}|$. Its exploration behavior is expected to be similar to that of DEC-MAX, but emphasizes boundary tracking more than wide-area coverage (due to space constraint, this analysis is not provided here). As a result, it tends to persist in tracking boundaries that are already well-predicted before deciding to search for new ones. Subject to limited observations, it may consequently not perform as well as DEC-MAX in a field with multiple hotspot regions. Also, it cannot be formally related to achieving the cost-minimizing exploration objectives (8) and (10); (b) The centralized *maximum entropy sampling* (MES) strategy [5] for directing the robot team performs only wide-area coverage by selecting the next set $\mathcal{O}$ of locations with maximum entropy to be observed using $\max_{\mathcal{O}} \mathbb{H}[Y_{\mathcal{O}}|y_{\mathcal{S}}]$; (c) It can be coupled with hotspot sampling (HS) by modifying the exploration objective to $\max_{\mathcal{O}} \mathbb{H}[Y_{\mathcal{O}}|y_{\mathcal{S}}] + \sum_{x\in\mathcal{O}} \mu_{x|\mathcal{S}}$. We call this the MES+HS strategy [5]. For these centralized strategies, the joint action space is exponential in the number of robots. So, they scale poorly with increasing number of robots. Table I summarizes and compares the characteristics of the above-mentioned active exploration strategies; it does not include the communication overhead, which is $\mathcal{O}(1)$ for all strategies.

A. Performance Metric

The performance metric used to evaluate the tested strategies is the number of misclassifications

$$M(\mathcal{A}) \triangleq \sum_{x\in\mathcal{A}} L(z_x, \widehat{Z}^*_x)$$

over all locations in a given set $\mathcal{A}$ where the function L is previously defined in (3). Three cases are considered:

(a) $\mathcal{A} = \mathcal{X}$ (i.e., all locations in the domain of the field),

(b) $\mathcal{A} = \mathcal{X}'$ where

$$\mathcal{X}' = \{x \in \mathcal{X} \mid |\gamma - y_x| \leq 0.2(\max_{x'\in\mathcal{X}} y_{x'} - \min_{x'\in\mathcal{X}} y_{x'})\}$$

(i.e., all locations with measurements that are close to the boundary threshold of 30 $\mathrm{mg/m^3}$ for the plankton density field and $3\,^\circ\mathrm{C}$ for the temperature field), and

(c) $\mathcal{A} = \mathcal{X} \setminus \mathcal{X}'$.

We observe that $|\mathcal{X}'|$ is only about 22% of $|\mathcal{X}|$ for both fields.

B. Temperature Field Data

Fig. 3 shows the results of the performance of tested strategies averaged over 5 randomly generated starting robot locations for the temperature field. In terms of the $M(\mathcal{X})$ performance, Figs. 3a−3c show that the DEC-MAX strategy quickly outperforms the MES and MES+HS strategies as the number of observations increases: their performance differences have been verified using t-tests ($\alpha = 0.1$) to be statistically significant after a total of 500, 750, and 800 observations sampled by teams of 2, 4, and 8 robots, respectively. Hence, the boundary-tracking DEC-MAX strategy reduces a greater number of misclassifications over the entire field than wide-area coverage and hotspot sampling. With more observations, the DEC-MAX strategy can also perform better than the straddle strategy: their performance differences have been verified using t-tests ($\alpha = 0.1$) to be statistically significant after a total of 500, 1000, and 1600 observations sampled by teams of 2, 4, and 8 robots, respectively. To explain this, we examine the observation paths of a team of 2 robots in one of the 5 test runs, as shown in Fig. 4. The initial performance of the DEC-MAX and straddle strategies are similar because they are both searching for hotspot region boundaries (Fig. 4a). As the number of observations increases further, DEC-MAX's performance improves over that of the straddle strategy because we observe that it directs the robots to search for new boundaries when the ones that are currently being tracked are well-predicted. In contrast, the straddle strategy tends to persist in tracking boundaries that are already well-predicted before deciding to search for new ones (Figs. 4b−4e). In terms of the $M(\mathcal{X}')$ and $M(\mathcal{X} \setminus \mathcal{X}')$ performance, Figs. 3d−3i reveal that, with increasing observations, the DEC-MAX strategy also reduces a greater number of misclassifications than the other evaluated strategies whether they are over locations close to the boundaries (i.e., in $\mathcal{X}'$) or away from

[1]The original straddle strategy proposed by [10] is developed for a single robot. To transform it into a decentralized multi-robot strategy, we simply execute the single-robot straddle strategy on every robot in the team.

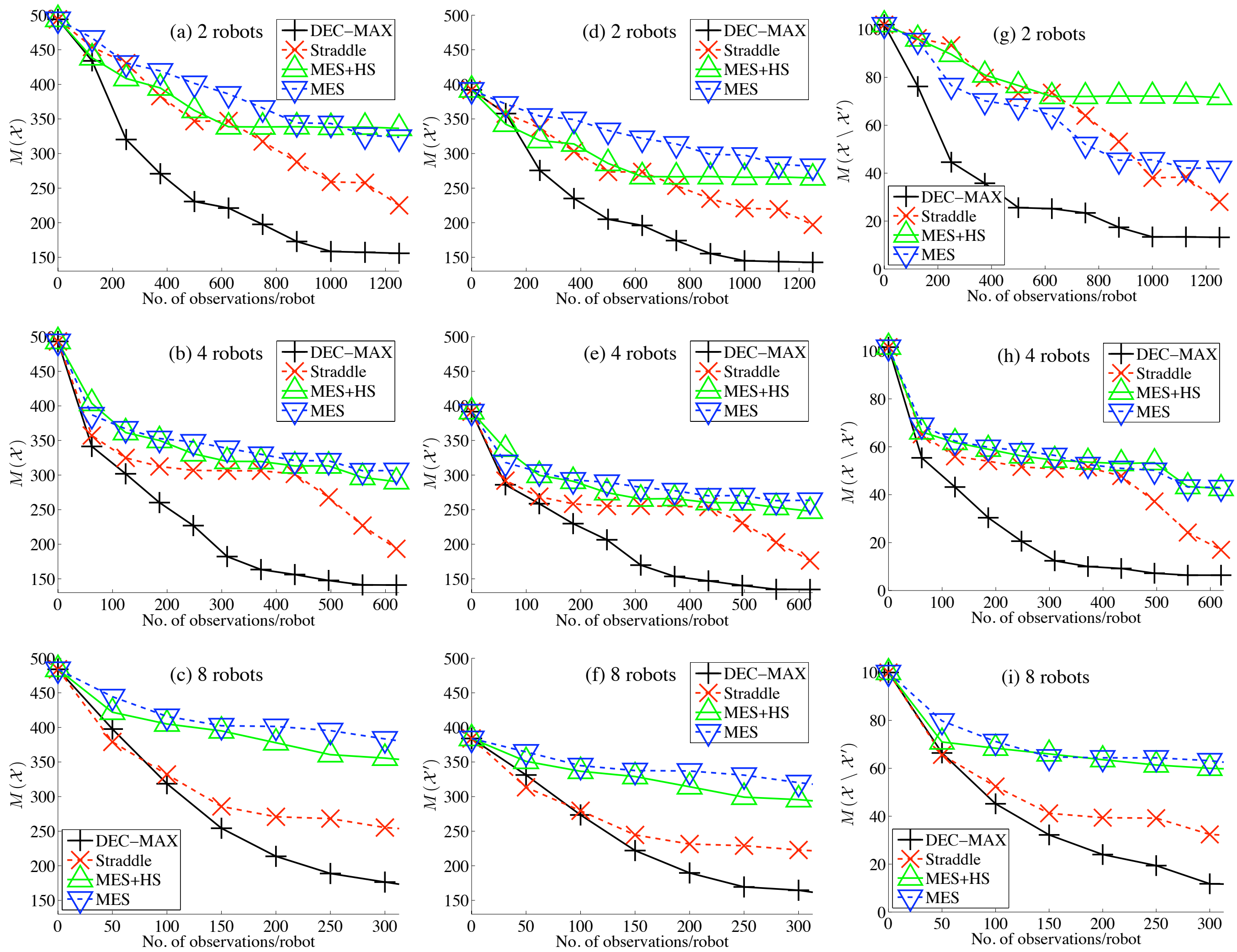


Fig. 3. Graphs of (a-c) $M(\mathcal{X})$, (d-f) $M(\mathcal{X}')$, and (g-i) $M(\mathcal{X} \setminus \mathcal{X}')$ vs. no. of observations/robot for varying number of robots actively exploring the temperature field.

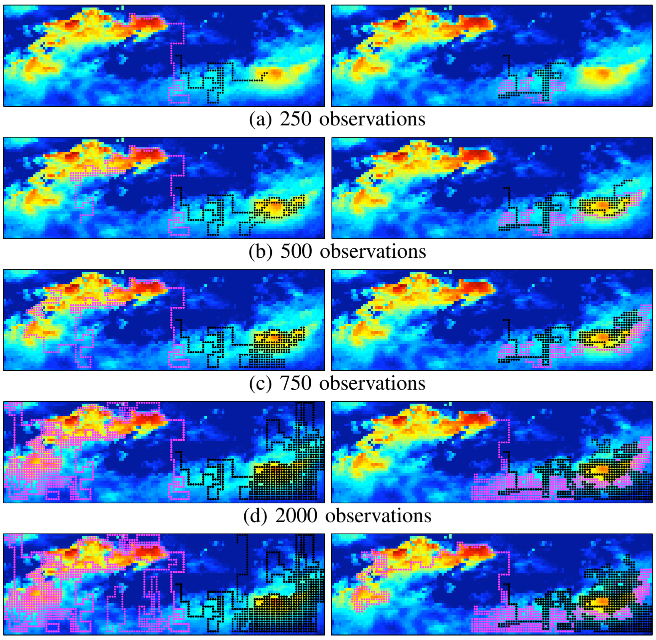
the boundaries (i.e., in $\mathcal{X} \setminus \mathcal{X}'$). It is interesting to note that locations close to the boundaries incur the majority of the misclassifications as compared to those away from the boundaries, which further corroborates the implication of Theorem 3 that there is higher uncertainty in labeling the locations close to the hotspot region boundaries.

C. Plankton Density Field Data

Fig. 5 shows the results of the performance of tested strategies averaged over 5 randomly generated starting robot locations for the plankton density field. The results are very similar to that of the temperature field (Section VI-B) except that the performance of the straddle strategy approaches that of the DEC-MAX strategy with excessive observations: their performance differences have been verified using t-tests ($\alpha = 0.1$) not to be statistically significant after a total of 2000 and 2240 observations sampled by teams of 2 and 4 robots, respectively. This is expected because the straddle strategy can track and predict the boundaries as well as the DEC-MAX strategy given a long enough exploration. However, subject to limited observations (which is more practical, as explained in Section I), the performance of the DEC-MAX strategy is clearly superior to that of the straddle strategy.

VII. Conclusion

This paper describes a decentralized multi-robot active exploration (DEC-MAX) strategy for probabilistic classification of hotspots in a large-scale Gaussian process-based environmental field. It has the practical advantage of being significantly more time-efficient over existing state-of-the-art active exploration strategies [4], [5], [6] because its incurred time is independent of the map resolution and the number of robots. In terms of active exploration performance, we have theoretically guaranteed that, under conditional independence assumption, the DEC-MAX strategy can optimally achieve the formal cost-minimizing exploration objectives based on the misclassification and entropy criteria, both of which otherwise cannot be optimized exactly to yield closed-form solutions. We have demonstrated theoretically and empirically that the uncertainty of labeling the hotspots in a GP-based field is greatest at or close to the hotspot region



(a) 250 observations (b) 500 observations (c) 750 observations (d) 2000 observations (e) 2500 observations

Fig. 4. Evolution of 2-robot observation paths produced by DEC-MAX (left column) and straddle (right column) strategies sampling a total of (a) 250, (b) 500, (c) 750, (d) 2000, and (e) 2500 observations. The robots start at locations of different lat. 37.75N and 40.75N and same lon. 192.75E.

boundaries. The DEC-MAX strategy is capable of exploiting this to produce an exploration behavior that formally trades off between that of boundary tracking until the hotspot region boundary can be accurately predicted and wide-area coverage to find new boundaries in sparsely sampled areas to be tracked. Empirical evaluation on real-world plankton density and temperature field data shows that, given limited observations, the DEC-MAX strategy can reduce a greater number of misclassifications than state-of-the-art active exploration strategies.

Appendix

A. *Proof of Theorem 3*

In Section IV, we have already shown the equivalence between the cost-minimizing and reward-maximizing strategies based on the misclassification and entropy criteria. Therefore, it suffices to prove that the DEC-MAX strategy (13) is equivalent to the reward-maximizing strategies.

Let us first prove that the reward-maximizing strategy (9) for the misclassification criterion is equivalent to the DEC-MAX strategy (13). From (9),

$$
\begin{aligned}
& \max_{\mathcal{O}} \sum_{x\in\mathcal{X}\setminus\mathcal{S}} R_{\widehat{Z}^*_x|\mathcal{S}} - \mathbb{E}_{Y_{\mathcal{O}}|y_{\mathcal{S}}}\left\{R_{\widehat{Z}^*_x|\mathcal{S}\bigcup\mathcal{O}}\right\} \\
= \;& \max_{\mathcal{O}} \sum_{x\in\mathcal{O}} R_{\widehat{Z}^*_x|\mathcal{S}} + \\
& \sum_{x\in\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})} \left(R_{\widehat{Z}^*_x|\mathcal{S}} - \mathbb{E}_{Y_{\mathcal{O}}|y_{\mathcal{S}}}\left\{R_{\widehat{Z}^*_x|\mathcal{S}\bigcup\mathcal{O}}\right\}\right) \\
= \;& \max_{\mathcal{O}} \sum_{x\in\mathcal{O}} R_{\widehat{Z}^*_x|\mathcal{S}} + \\
& \sum_{x\in\mathcal{X}\setminus(\mathcal{S}\bigcup\mathcal{O})} \left(R_{\widehat{Z}^*_x|\mathcal{S}} - \mathbb{E}_{Y_{\mathcal{O}}|y_{\mathcal{S}}}\left\{R_{\widehat{Z}^*_x|\mathcal{S}}\right\}\right) \\
= \;& \max_{\mathcal{O}} \sum_{x\in\mathcal{O}} R_{\widehat{Z}^*_x|\mathcal{S}} \\
= \;& \sum_{i=1}^{k} \max_{x_i} R_{\widehat{Z}^*_{x_i}|\mathcal{S}} \; .
\end{aligned}
$$

The first equality follows from $R_{\widehat{Z}^*_x|\mathcal{S}\bigcup\mathcal{O}} = 0$ for $x \in \mathcal{O}$. The second equality is due to the conditional independence assumption that is provided as a sufficient condition in the theorem. The third equality is due to the second summation term evaluating to zero. The last equality follows from the observation that each risk term in the summation depends only on the choice of the next location x to be observed by a single different robot. Hence, we can maximize each risk term in the summation independently and in a decentralized manner to achieve the same result as that in the third equality.

$$
\begin{aligned}
& \max_{x} R_{\widehat{Z}^*_x|\mathcal{S}} \\
= \;& \max_{x} \{\min\left(P(Z_x = 1|y_{\mathcal{S}}), 1 - P(Z_x = 1|y_{\mathcal{S}})\right)\} \\
= \;& \max_{x} \{\min\left(P(Y_x \geq \gamma|y_{\mathcal{S}}), 1 - P(Y_x \geq \gamma|y_{\mathcal{S}})\right)\} \\
\equiv \;& \max_{x} \left\{\min\left[-\mathrm{erf}\left(\frac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}}\sqrt{2}}\right), \mathrm{erf}\left(\frac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}}\sqrt{2}}\right)\right]\right\} \\
= \;& \max_{x} - \left| \mathrm{erf}\left(\frac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}}\sqrt{2}}\right) \right| \\
\equiv \;& \min_{x} \left| \mathrm{erf}\left(\frac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}}\sqrt{2}}\right) \right| \\
\equiv \;& \min_{x} \frac{|\gamma - \mu_{x|\mathcal{S}}|}{\sigma_{x|\mathcal{S}}\sqrt{2}} \\
\equiv \;& \min_{x} \frac{|\gamma - \mu_{x|\mathcal{S}}|}{\sigma_{x|\mathcal{S}}} \; .
\end{aligned}
$$

The first equality follows from (5). The first equivalence is due to $P(Y_x \geq \gamma|y_{\mathcal{S}}) = \dfrac{1}{2}\left[1 - \mathrm{erf}\left(\dfrac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}}\sqrt{2}}\right)\right]$.

Now, let us prove that the reward-maximizing strategy (11) for the entropy criterion is equivalent to to the DEC-MAX strategy (13). From (11),

$$
\begin{aligned}
& \max_{\mathcal{O}} \; \mathbb{H}[Z_{\mathcal{O}}|y_{\mathcal{S}}] \\
= \;& \max_{\mathcal{O}} \sum_{x\in\mathcal{O}} \mathbb{H}[Z_x|y_{\mathcal{S}}] \\
= \;& \sum_{i=1}^{k} \max_{x_i} \; \mathbb{H}[Z_{x_i}|y_{\mathcal{S}}] \; .
\end{aligned}
$$

The first equality follows from chain rule of entropy and conditional independence assumption. The second equality follows from observing that each entropy term in the summation depends only on the choice of the next location x to be observed by a single different robot. Hence, we can maximize each entropy term in the summation independently and in a decentralized manner to achieve the same result as that in the first equality.

$$
\begin{aligned}
& \max_{x} \; \mathbb{H}[Z_x|y_{\mathcal{S}}] \\
\equiv \;& \min_{x} P(Z_x = 1|y_{\mathcal{S}}) \log P(Z_x = 1|y_{\mathcal{S}}) + \\
& \quad (1 - P(Z_x = 1|y_{\mathcal{S}})) \log(1 - P(Z_x = 1|y_{\mathcal{S}}))
\end{aligned}
$$

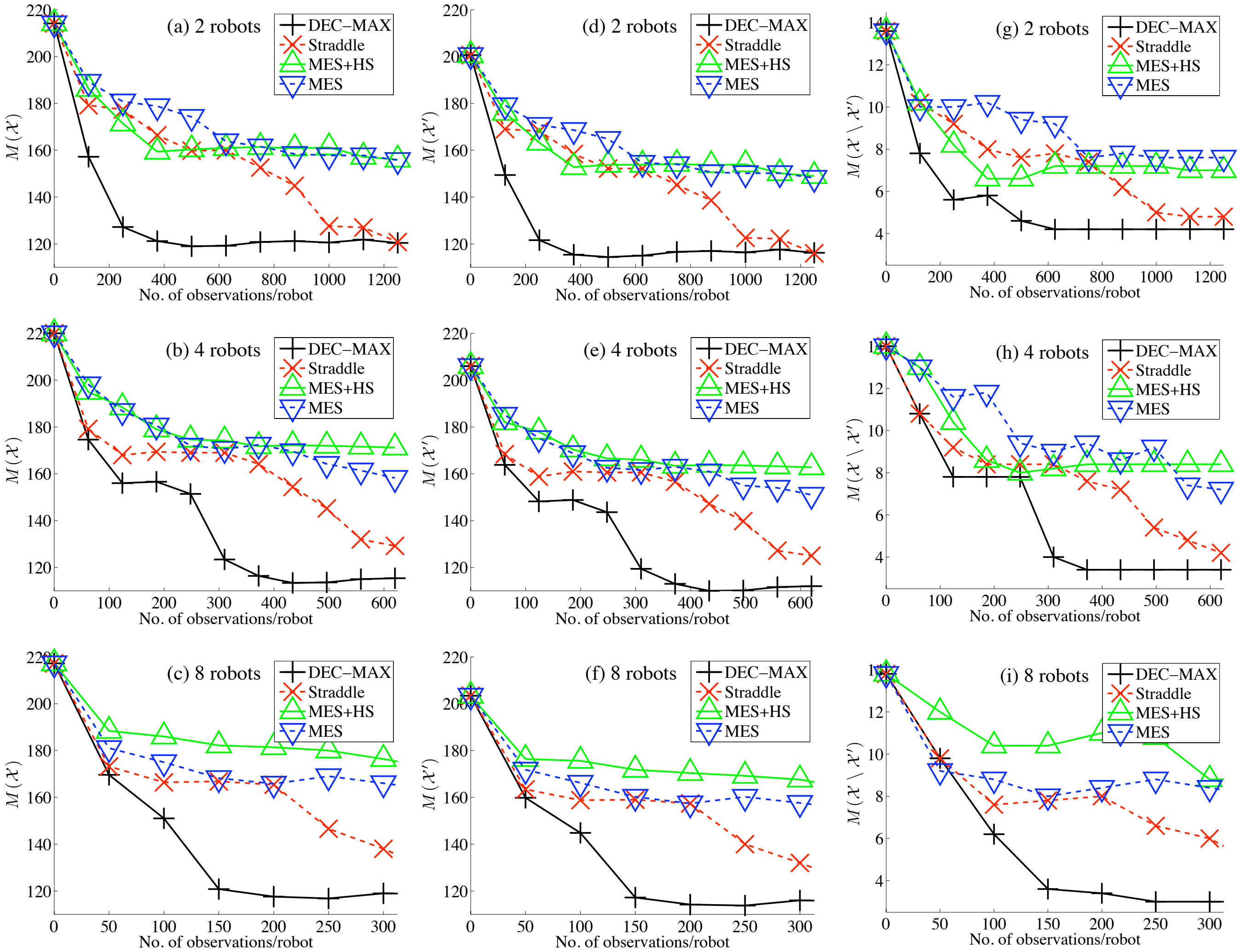


Fig. 5. Graphs of (a-c) $M(\mathcal{X})$, (d-f) $M(\mathcal{X}')$, and (g-i) $M(\mathcal{X} \setminus \mathcal{X}')$ vs. no. of observations/robot for varying number of robots actively exploring the plankton density field.

$$
\begin{aligned}
&\equiv \min_x \left| \frac{1}{2} - P(Z_x = 1 | y_{\mathcal{S}}) \right| \\
&= \min_x \left| \frac{1}{2} - P(Y_x \geq \gamma | y_{\mathcal{S}}) \right| \\
&\equiv \min_x \left| \operatorname{erf}\left(\frac{\gamma - \mu_{x|\mathcal{S}}}{\sigma_{x|\mathcal{S}} \sqrt{2}} \right) \right| \\
&\equiv \min_x \frac{|\gamma - \mu_{x|\mathcal{S}}|}{\sigma_{x|\mathcal{S}} \sqrt{2}} \\
&\equiv \min_x \frac{|\gamma - \mu_{x|\mathcal{S}}|}{\sigma_{x|\mathcal{S}}} \, .
\end{aligned}
$$

REFERENCES

[1] N. A. C. Cressie, *Statistics for Spatial Data*, 2nd ed. NY: Wiley, 1993.

[2] R. Webster and M. Oliver, *Geostatistics for Environmental Scientists*, 2nd ed. NY: John Wiley & Sons, Inc., 2007.

[3] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.

[4] K. H. Low, J. M. Dolan, and P. Khosla, "Adaptive multi-robot wide-area exploration and mapping," in *Proc. AAMAS*, 2008, pp. 23–30.

[5] ——, "Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing," in *Proc. ICAPS*, 2009, pp. 233–240.

[6] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *JMLR*, vol. 9, pp. 235–284, 2008.

[7] K. H. Low, G. J. Gordon, J. M. Dolan, and P. Khosla, "Adaptive sampling for multi-robot wide-area exploration," in *Proc. IEEE ICRA*, 2007, pp. 755–760.

[8] R. N. Smith, A. Pereira, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "Autonomous underwater vehicle trajectory design coupled with predictive ocean models: A case study," in *Proc. IEEE ICRA*, 2010, pp. 4770–4777.

[9] F. Zhang and N. E. Leonard, "Cooperative control and filtering for cooperative exploration," *IEEE Trans. Automat. Contr.*, vol. 55, no. 3, pp. 650–663, 2010.

[10] B. Bryan, J. G. Schneider, R. Nichol, C. Miller, C. Genovese, and L. A. Wasserman, "Active learning for identifying function threshold boundaries," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, pp. 163–170.

[11] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[12] K. H. Low, J. M. Dolan, and P. Khosla, "Active Markov information-theoretic path planning for robotic environmental sensing," in *Proc. AAMAS*, 2011, pp. 753–760.

An Artificial Potential Field based Sampling Strategy for a Gas-Sensitive Micro-Drone

Patrick Neumann, Sahar Asadi, Jochen H. Schiller, Achim Lilienthal and Matthias Bartholmai

Abstract— **This paper presents a sampling strategy for mobile gas sensors. Sampling points are selected using a modified artificial potential field (APF) approach, which balances multiple criteria to direct sensor measurements towards locations of high mean concentration, high concentration variance and areas for which the uncertainty about the gas distribution model is still large. By selecting in each step the *most often suggested close-by* measurement location, the proposed approach introduces a locality constraint that allows planning suitable paths for mobile gas sensors. Initial results in simulation and in real-world experiments with a gas-sensitive micro-drone demonstrate the suitability of the proposed sampling strategy for gas distribution mapping and its use for gas source localization.**

Index Terms—**autonomous UAV, chemical sensing, gas distribution modelling, gas source localization, gas sensors, mobile sensing system, quadrocopter, sensor planning, artificial potential field.**

I. Introduction

Gas distribution modelling and gas source localization play an important role in environmental management applications such as leak detection and landfill monitoring [1], for example. The response of many gas sensors, however, is caused by direct interaction with the chemical compounds and thus represents only a small area around the sensor surface. For practical applications either a large number of stationary sensors or mobile sensors are required. In this paper we consider the case of gas sensors carried by a mobile robot, which offers a number of advantages including rapid deployment, adaptation to changing environmental conditions, and the possibility to move to areas of high concentration, to name but a few. A crucial element for gas-sensitive mobile robots is a sensor planning strategy that selects preferable sampling locations based on the current knowledge about the environment and more specifically the current knowledge about the gas distribution. The purpose of the sensor planning component is to reduce the time that is necessary to converge to the final gas distribution model or to reliably identify important parameters of the distribution such as areas of high concentration, for example. Sensor planning is especially important in the case of a flying gas-sensitive robot such as the one considered in this paper due to its limited battery life time.

In this paper, we adapt a newly developed sensor planning approach by introducing locality constraints to plan the path for a micro-drone. The sensor planning algorithm uses information about the target area and previous sampling locations. In addition, it considers the current statistical gas distribution model to direct sensor measurements towards locations of high mean concentration, high concentration variance and areas for which the uncertainty about the gas distribution model is still large. The different objectives are combined in an Artificial Potential Field (APF) in a way that allows to include additional objectives, e.g. from human operators, in an intuitive and straightforward way. In addition to the introduction of the modified APF-based sensor planning algorithm and the demonstration on a gas-sensitive micro-drone, we also demonstrate that the peak in the predictive variance model can provide an accurate estimation of the location of a stationary gas source.

In the reminder of this paper, we first describe the APF-based approach for sensor planning and its modification to provide meaningful search paths for a mobile gas sensor (Sec. II). Next, we describe the robotic platform used (Sec. III) and the experimental set-up (Sec. IV). Finally we present the results (Sec. V), draw conclusions and identify directions of future work (Sec. VI).

II. Sensor and Path Planning

A. Statistical Gas Distribution Modelling

The first step in the proposed algorithm is to create a statistical gas distribution model using the Kernel DM+V/W algorithm introduced by Reggente and Lilienthal [6]. The input to this algorithm is a set of measurements $D = \{(x_1,r_1,v_1), \ldots, (x_n,r_n,v_n)\}$ with gas sensor measurements r_i and airflow measurements v_i collected at locations x_i. The output is a grid model that computes an estimate of distribution mean and variance for each cell. We use the 2D version of the Kernel DM+V/W algorithm as basis for the APF sensor planning algorithm to avoid the higher computational complexity of the 3D Kernel DM+V/W algorithm [7] and because of the limited battery capacity of

Manuscript received June 03, 2011. This work was supported by BMWi within the MNPQ programme (file number 28/07) and the EC (contract number FP7-224318-DIADEM).

Patrick Neumann and Matthias Bartholmai work for the BAM Federal Institute for Materials Research and Testing, D – 12205 Berlin, Germany (corresponding author to provide phone: +493081043629; fax: +493081041917; e-mail: Patrick.Neumann@bam.de).

Jochen H. Schiller works for the Institute of Computer Science, FU University, D – 14195 Berlin, Germany.

Achim Lilienthal and Sahar Asadi are members of the AASS Research Centre, School of Science and Technology, Örebro University, SE – 70182 Örebro, Sweden.

the micro-drone, which does not permit a full 3D search. In the experiments, the drone was kept in a single 2D plane.

The Kernel DM+V/W algorithm works as follows. In the first step, it computes weights $\omega_i^{(k)}$ that model the information content of measurement i at grid cell k. This is done by evaluating a two-dimensional, multivariate Gaussian kernel N at the distance between the location of the measurement i and the center $x^{(k)}$ of cell k:

$$\omega_i^{(k)} = N(|x^{(k)} - x_i|, v; \sigma, \gamma). \tag{1}$$

The shape and orientation of the kernel depends on the local airflow vector v and on two meta-parameters that determine a spatial scale (σ) and a wind scale (γ). If no wind is measured (or if no wind information is available), the Gaussian kernel has a circular shape. In case of a non-zero wind measurement the kernel takes the shape of an elongated ellipse with the semi-major axis rotated in wind direction and stretched according to the strength of the wind.

Second, weights $\omega_i^{(k)}$, weighted sensor readings $\omega_i^{(k)} \cdot r_i$, and weighted variance contributions $\omega_i^{(k)}(r_i - r^{(k(i))})^2$ are integrated and stored in temporary grid maps.

$$\Omega^{(k)} = \sum_{i=1}^{n} \omega_i^{(k)}, \tag{2}$$

$$R^{(k)} = \sum_{i=1}^{n} \omega_i^{(k)} \cdot r_i, \tag{3}$$

$$V^{(k)} = \sum_{i=1}^{n} \omega_i^{(k)} \cdot \left(r_i - r^{k(i)}\right)^2 \tag{4}$$

The variance contributions are computed using the difference between the actual measurements r_i and the corresponding prediction of the model $r^{(k(i))}$, i.e. the predictive mean for the grid cell $k(i)$ closest to the point at which r_i was measured.

Third, a confidence map $\alpha^{(k)}$ is computed from the integrated weights $\Omega^{(k)}$ using another scaling parameter σ_Ω as a soft threshold:

$$\alpha^{(k)} = 1 - e^{\frac{\Omega^{(k)}}{\sigma_\Omega^2}}. \tag{5}$$

The confidence map expresses an increased confidence at locations for which we have a large number of sensor readings in the close vicinity ("close" is to be understood relative to the kernel width σ).

Finally, the map estimate of the mean $r^{(k)}$ and the corresponding variance estimate $v^{(k)}$ is calculated using (6) and (7) as

$$r^{(k)}(\sigma, \gamma, \sigma_\omega) = \alpha^{(k)} \frac{R^{(k)}}{\Omega^{(k)}} + (1 - \alpha^{(k)}) r_0, \tag{6}$$

$$v^{(k)}(\sigma, \gamma, \sigma_\omega) = \alpha^{(k)} \frac{V^{(k)}}{\Omega^{(k)}} + (1 - \alpha^{(k)}) v_0. \tag{7}$$

The final estimate is obtained by linear interpolation between the map prediction and an *a priori* estimate for cells with low confidence. For the mean, the *a priori* estimate r_0 is computed as the average concentration over all sensor readings. Similarly, the average over all variance contributions v_0 is used to estimate the distribution variance in regions far away from measurement points.

B. Artificial Potential Field (APF) based Sensor Planning

In each step, the sensor planning component suggests a selectable number n_{sp} of locations to place sensors in the area of interest in the next iteration. The algorithm uses information about the target area, previous sampling locations, and the current statistical gas distribution model (described in the previous section). The selection process considers three objectives to direct the sensor towards areas of (1) high predictive mean, (2) high predictive variance and (3) areas in which the model uncertainty is high.

The first two objectives implement exploitation of the information in the gas distribution model. They are realized with an attractive potential generated by charges placed in each grid cell center. The strength of these charges is given by the corresponding predictive mean and variance. Accordingly, two APF contributions are computed for each cell k as

$$APF_M^{(k)} = \sum_{j \neq k} r^{(k)} \cdot e^{-\frac{|x^{(j)} - x^{(k)}|}{\sigma_d}}, \tag{8}$$

$$APF_V^{(k)} = \sum_{j \neq k} v^{(k)} \cdot e^{-\frac{|x^{(j)} - x^{(k)}|}{\sigma_d}}. \tag{9}$$

The third objective that corresponds to exploration is implemented by a repulsive potential generated by placing charges at all $|D| = n$ previous measurement locations:

$$APF_R^{(k)} = \sum_{i=1}^{|D|} q \cdot e^{-\frac{|x^{(i)} - x^{(k)}|}{\sigma_d}}. \tag{10}$$

Associating the sensors to be placed as negative charges, the virtual charge q has to be negative as well. In the current implementation, we assign the same repulsive force to all previous measurements and select q = -1. Finally, the APF contributions are additively combined with importance factors β_M, β_V, and β_R for each objective:

$$APF^{(k)} = \beta_M APF_M^{(k)} + \beta_V APF_V^{(k)} + \beta_R APF_R^{(k)}. \tag{11}$$

Finally, n_{sp} locations are identified iteratively by selecting the location at which the potential takes its minimum as a suggested measurement point and updating the APF by temporarily placing an additional measurement charge at the selected location. Theoretically it could happen that the attractive forces towards an increased mean in one direction and towards and increased variance in the opposite direction cancel themselves out. In practice, we did not observe such an effect. It is unlikely that the attractive forces are completely balanced at the position of the sensor, and if they are not the sensor will be directed towards one of the directions so that in the next step the symmetry is broken.

C. Selection of the Next Measurement Location

The sensor planning approach detailed in the previous section distributes its suggestions over the target area without any spatial order. Moving the mobile gas sensor directly to these locations could create a seesaw movement, which tends to empty the batteries sooner, resulting in fewer measurements. Therefore, we add a locality constraint by selecting out of the n_{sp} suggestions from the sensor planning component the *most often suggested close-by* measurement location. This is implemented by a matrix S that has the same discretization as the gas distribution model. For each grid cell $S^{(k)}$ it counts how often the cell was suggested since it was actually visited the last time. The next measurement point is ultimately selected as the one with the highest ratio $S^{(k)}/d(k)$, where $d(k)$ is the distance between the current position of the sensor and grid cell k. Thus, a location far away from the current position will only be selected if it is frequently suggested.

In the current implementation, we increase not only the counter for a suggested cell but also the counter of neighboring cells within a radius of 0.5 m by one, which corresponds to the scale of the drainage area below the drone (that has a diameter of 1m).

D. Path Planning Algorithm for the Micro-Drone

The initial measurement location is chosen randomly in the target area. Then the following steps are iteratively performed:

- collect gas sensor and wind measurements while keeping the drone at a fixed position for a prolonged time (here: 20 s);
- average the wind measurements over the measurement time (20s);
- compute the predictive gas distribution model using the Kernel DM+V/W algorithm (Sec. II.A), the input to the algorithm are the current positions and gas sensor readings and the averaged wind measurement;
- derive an estimate of the source location from the predictive gas distribution map (detailed below);
- determine the n_{sp} suggested sampling locations with the APF based sensor planning component (Sec. II.B);
- update the matrix S and select a sampling location that maximizes the ratio $S^{(k)}/d(k)$ as described in Sec. II.C;
- fly the drone autonomously to the chosen sampling location and repeat with the first step. (Measurements in between two sampling locations are not used to decrease the influence of a memory effect in the sensor response due to the slow sensor recovery.)

Fig. 1. Pollution source and micro-drone during one of the experiments.

The algorithm terminates either if the battery runs out or the confidence map $\alpha^{(k)}$ is above a defined threshold for each cell k.

III. Robotic Platform

Federal Institute for Materials Research and Testing (BAM), in cooperation with Airrobot GmbH & Co. KG, has developed a mobile and flexible measurement system as part of an R&D project funded by the Federal Ministry of Economics and Technology [2], [3], [4]. The result of the project is a gas-sensitive sensor module (approx. 200 g) for the Airrobot drone AR100-B (Fig. 1). The drone can be flown by line of sight, via onboard video camera and video goggles as well as by autonomous waypoint tracking.

The Inertial Measurement Unit (IMU) is an important part of the drone. It provides the basis for flight control and wind vector estimation and can be read out during operation. The IMU consists of three orthogonally arranged accelerometers, which detect linear accelerations along the x-, y- and z-axis, and three orthogonally arranged rotation rate sensors, which measure angular accelerations along the x-, y- and z-axis. Magnetic field sensors (compass) and GPS are used to improve the accuracy of the IMU and to compensate for sensor drift. The IMU of the drone also contains a barometric pressure sensor to control the drone's altitude.

A commercially available gas detector (Dräger X-am 5600), which was originally designed as a handheld device for personal safety, is the base unit of the gas sensitive payload. Depending on the scenario, it can measure many combustible gases and vapors with the catalytic sensor as well as different (toxic) gases, e.g. O_2, CO, H_2S, NH_3, CO_2, SO_2, PH_3, HCN, NO_2, and Cl_2 with electrochemical and infrared sensors.

An additional electronic circuit controls the communication between the gas detector and the drone via appropriate device interfaces. A temperature and humidity sensor was also integrated as both factors may affect the measurement data (however, no compensation for varying temperature or humidity was applied in the experiments presented in this paper). The casing of the gas detector is protected against water and dust according to IP 67 (see [5] for further information) and therefore capable of working outdoors.

IV. Experimental Setup

All experiments were carried out inside an 8 x 12 m^2 area in an outdoor environment with a micro-drone equipped with electrochemical CO sensors. Gas concentration and wind measurements were recorded with 1 Hz. Measurements were taken at each measurement position for about 20 s, which is of the same order as the T_{90} response times of the used sensors. The wind vector was then averaged over all the measurements collected at the measurement point. Each gas sensor measurement was then included into the gas distribution model as if it was acquired together with a measurement of the average wind vector, i.e. the average wind vector was used for all individual gas sensor measurements acquired at the measurement position.

The parameters of the Kernel DM+V/W algorithm were heuristically set to $c = 0.15$ m (grid cell size), $\sigma = 0.40$ m (kernel width), $\sigma_\Omega = N(0, \sigma = 0.4) \approx 1.0$, and $\gamma = 0.2\ s^{-1}$. Equal importance factors β_M, β_V, and β_R were chosen for the APF contributions. The flight speed of the drone between the measurement positions was set to 1 ms^{-1}. Because of the low flight height of about 1 m, the height of the drone was controlled manually during the experiments. Each run took around 14 – 19 minutes to complete. A barbecue filled with burning coal and fresh, damp wood was used as a pollution source (Fig. 1) and was placed approximately in the middle of the experimental area (at approx. (6.3, 3.8) m from the bottom left corner). The drone was set to autonomous waypoint mode directly after take-off, which started the experiment.

V. Results

The results presented in Fig. 2 and Table I demonstrate the suitability of the proposed algorithm for gas distribution mapping and its use for localization of a stationary gas source. Table I shows for all five runs the distance between the true gas source location and three different estimates after the last measurement point. The first estimate is derived by selecting grid cells in which the predictive mean is larger than 90% of the maximum. The center of this area is taken as the source location estimate and the maximum extension in x- or y-direction is used to specify a confidence interval. In the same way the other two estimates are computed using the variance (second result) or the product of mean and variance (third result). The true source location was within the mean estimation area only in one trial and within the variance estimation area in two trials. This is in line with previous

TABLE I
Results of the Experiments

run	measurement points	distance to true source location, estimate using mean / variance / mean·variance
1	27	(1.66±0.75) m / (1.52±0.71) m / (1.57±0.50) m
2	24	(2.64±0.65) m / (2.84±0.68) m / (2.75±0.46) m
3	31	**(0.68±0.77) m / (0.25±0.80) m / (0.46±0.57) m**
4	20	(3.01±2.43) m / (2.05±0.89) m / (1.89±0.64) m*
	34	(1.84±1.32) m / (2.51±0.50) m / (2.28±0.49) m
5	32	(1.72±0.80) m **/ (0.74±0.86) m /** (1.41±0.55) m

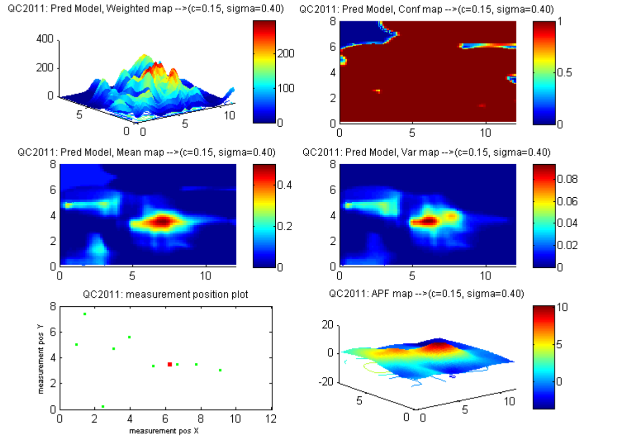


Fig. 2. The top row shows the weight map $\Omega^{(k)}$ (left) and the confidence map $\alpha^{(k)}$ computed with the Kernel DM+V/W algorithm (right). The middle row shows the corresponding mean distribution map $r^{(k)}$ (left) and variance distribution map $v^{(k)}$ (right). Bottom row, left: area with the suggested next measurement points (green dots) and the source location estimate (red dot). Bottom row, right: visualization of the APF. All plots were created after the last time step of the SP algorithm (run 3, measurement 31).

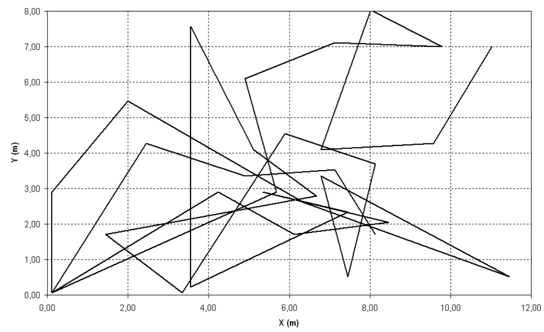


Fig. 3. Calculated sample trajectory of the SP algorithm (run 3) with starting position $(x, y) = (11.02, 7.00)$ m.

observations that the concentration variance often provides a better indication of the gas source location [8] than the mean.

Fig. 2 shows the final snapshot of run number 3 after 31 measurement points. The first four diagrams are related to the Kernel DM+V/W algorithm and show the weight map $\Omega^{(k)}$, the confidence map $\alpha^{(k)}$, the mean distribution map $r^{(k)}$, and the variance distribution map $v^{(k)}$. The last two diagrams show the measurement positions suggested by the sensor planning component (green dots) together with the predicted source location (red dot) and the APF map. Fig. 3 shows exemplarily the trajectory produced by the SP algorithm in run 3 with starting position $(x, y) = (11.02, 7.00)$ m (compare with Fig. 2).

Keeping the gas emission rate constant over time with the chosen gas source was difficult. A re-ignition of the almost extinguished source in run #4 for example (after the 20th measurement) created an intense emission that likely caused very high concentrations also far away from the source. The 21st measurement taken at position $(x, y) = (8.85, 4.95)$ m was affected by this outburst, which caused a strong change in the gas source location estimate and is therefore. Results are given in Table I for the 20 measurement points up to this event, marked with an asterisk (*), and for the full duration of the experiment. Another difficulty, which should be mentioned here, is the slow sensor decay. Flying from one measurement position to another directly over the source can also lead to wrong source estimates when the sensor still responds to the high concentrations close to the source.

VI. Conclusions and Future Work

Statistical gas distribution modelling indicates areas of high mean and variance and the respective maxima suggest areas that are good candidates for further inspection. The proposed APF-based approach balances objectives related to exploration and exploitation. The repulsive part prevents repeated measurements at the same locations and thus promotes exploration. The attractive part directs the attention to areas for which higher gas accumulation or higher variance in the predictive gas dispersion is predicted (exploitation). Through the introduction of a locality constraint, implemented by selecting in each step the most often suggested close-by measurement location, the results of the sensor planning component could be used to plan suitable paths for a mobile gas sensor. The proposed algorithm was tested in real-world experiments with a gas sensitive micro-drone. The initial results presented in this paper show the potential of this approach for gas distribution mapping and highlight that the produced maps can provide good estimates of the gas source location.

The method presented in this paper leaves ample room for future work. With respect to real-world applications, it should be investigated how robust the proposed approach is with respect to changing wind directions and to different levels of turbulence.

The current implementation does not take into account the time when the measurements were made. We will study therefore an extension of the approach proposed in this paper that introduces time-dependency at two points. First, a time dependent statistical gas distribution modeling algorithm will be used, for example the Time-Dependent (TD) Kernel DM+V/W algorithm introduced in [9]. Second, the charge q that scales the strength of the repulsive potential exerted from previous measurement points should also be time-dependent, namely it should be lower for earlier measurements. We also plan to study an extension to a 3D approach.

Finally, we will investigate methods to select optimal relative weights (β parameters) for the different objectives in Eq. (11), and include more real-world experiments and simulations to test the algorithm.

Acknowledgment

The authors thank the participating colleagues from BAM and Örebro University. The authors also would like to express their gratitude to BMWi (MNPQ Program; file number 28/07) and to the EC (contract number FP7 – 224318 - DIADEM: Distributed Information Acquisition and Decision-Making for Environmental Management) for funding the research.

References

[1] Kowadlo, G.; Russell, R. A.: Robot Odor Localization: A Taxonomy and Survey. *In: Int. J. Rob. Res. 27 (2008)*, Nr. 8, S. 869 – 894.

[2] Neumann, Patrick P.; Bartholmai, Matthias; Schiller, Jochen H.; Manolov, Manol; Wiggerich, Burkhard: Self Optimizing Search and Characterization of Gaseous Hazardous Substance Sources using a Micro-Drone: A new Approach to determine Wind Speed and Direction. *IEEE International Workshop on Robotic and Sensors Environments (ROSE 2010)*, Phoenix, AZ, USA.

[3] Neumann, Patrick; Bartholmai Matthias: Self Optimizing Search and Characterization of Gaseous Hazardous Substance Sources using a Micro-Drone. *Proceedings of the International Micro Air Vehicle Conference and Flight Competition (IMAV), 2010.*

[4] Bartholmai, Matthias; Neumann, Patrick: Micro-Drone for Gas Measurement in Hazardous Scenarios via Remote Sensing. *Proceedings of* the 6th International Conference on REMOTE SENSING (REMOTE'10), Iwate, Japan.

[5] ANSI/IEC 60529-2004: *Degrees of Protection Provided by Enclosures (IP Code).* National Electrical Manufacturers Association, Virginia.

[6] Reggente, Matteo; Lilienthal, Achim J.: Using Local Wind Information for Gas Distribution Mapping in Outdoor Environments with a Mobile Robot. *Proceedings of IEEE Sensors*, *2009*, 1715 – 1720.

[7] Reggente, Matteo; Lilienthal, Achim J.: The 3D-Kernel DM+V/W Algorithm: Using Wind Information in Three Dimensional Gas Distribution Modelling with a Mobile Robot. *Proc. IEEE Sensors, 2010, pp. 999-1004.*

[8] Lilienthal, Achim J.; Duckett, Tom; Werner, Felix and Ishida, Hiroshi: Indicators of Gas Source Proximity using Metal Oxide Sensors in a Turbulent Environment. *Proc. IEEE / RAS-EMBS Int. Conf. on Biomedical Robotics and Biomechatronics (Biorob), 2006.*

[9] Asadi, Sahar; Pashami, Sepideh; Loutfi, Amy and Lilienthal, Achim J.: TD Kernel DM+V: Time-Dependent Statistical Gas Distribution Modelling on Simulated Measurements. Olfaction and Electronic Nose - *Proceedings of the 14th International Symposium on Olfaction and Electronic Nose (ISOEN)*, 2011, to appear.

A Networked Telerobotic Observatory for Collaborative Remote Observation of Avian Activity and Range Change

Siamak Faridani, Bryce Lee, Selma Glasscock, John Rappole, Dezhen Song, and Ken Goldberg

***Abstract*—The scientific field study of wildlife often requires vigilant observation of detailed animal behavior over extended periods. In remote and inhospitable locations, observation can be an arduous, expensive, and dangerous experience for field scientists. We are developing a new class of networked teleoperated robotic "observatories" that allows "citizen scientists" and professional scientists to remotely observe, record, and index animal activity and behaviors via the internet. This paper describes CONE-Welder, installed at the Rob & Bessie Welder Wildlife Foundation in Texas to gather photographic and quantitative data for a biological study of avian activity and hypothesized range change for selected subtropical bird species. Since the system was deployed on 12 May 2008, over 600 users ("players") have participated online. Players have requested over 2.2 million camera frames and captured over 29,000 photographs. Within these photos, citizen scientists have classified 74 unique species, including eight avian species previously unknown to have breeding populations within the region. The collected dataset quantifies seasonal presence of birds of particular interest, e.g., the Green Jay (*Cyanocorax yncas*). This paper describes the system architecture, the game interface that provides incentives for player participation, and initial data collected. CONE-Welder is available online at: http://cone.berkeley.edu/**

Fig. 1. Sample screenshot of CONE-Welder from an Internet browser. The interface (using Adobe Flash) allows users to share control of a networked telerobotic video camera to capture photos, and classify the photos taken by other players. This screenshot includes a Great Kiskadee (*Pitangus sulfuratus*) and a color-marked Green Jay (*Cyanocorax yncas*), two of the species of interest in this project.

I. Introduction

To assist field biologists, we are developing Collaborative Observatories for Natural Environments (CONEs), a new class of networked teleoperated robotic "observatories" that allows "citizen scientists" and professional scientists to remotely observe, record, and index wildlife activity via the internet. Our broader goal is to advance understanding of automated and collaborative systems that combine sensors, actuators, and human input to observe and record natural behavior in remote settings.

This paper presents the latest in a series of field installations based on a joint effort among UC Berkeley, Texas A&M, the Smithsonian Institution, and the Rob & Bessie Welder Wildlife Foundation. CONE-Welder is deployed at the Rob & Bessie Welder Wildlife Refuge, 12 km NE of Sinton, Texas (28E6'51.1" N, 97E25'2.2" W). The region in which the refuge is located has the highest diversity of bird species in North America outside of the tropics. Welder has detailed records of its avifauna, as well as many other aspects of its ecological communities, dating back to its establishment in 1954 [1], [2].

The field research objectives of CONE-Welder are as follows: 1) To collect data documenting daily and seasonal presence of subtropical bird species not previously known to breed as far north as the Welder Refuge (Fig 1); and 2) To record the daily and seasonal presence of individuals of some of these species that have been banded and color-marked allowing individual recognition by photo [3]. This study is relevant to larger questions regarding the proximate and ultimate causes for such shifts, which may include global effects such as climate change.

At Welder, scientists including co-authors Glasscock and Rappole designed, constructed, and maintain an avian feeding station. To document the presence of the new species, they capture and band individuals of these species, and to allow individual recognition, they color-band birds from these species. Rappole and colleagues have also undertaken two years of nest searches to locate nests and document breeding. The Welder Foundation and resident scientists provide additional field and laboratory support for the project and collect information on range change in these bird species.

The CONE-Welder networked robotic camera system engages citizens around the world, including students from local and non-local schools, to systematically photograph and collect data on the daily and seasonal occurrence of subtropical

S. Faridani, B. Lee, and K. Goldberg are with University of California, Berkeley, CA. S. Glasscock is with Welder Wildlife Foundation, Sinton, TX. J. Rappole is with Smithsonian National Zoological Park and Roger Tory Peterson Institute, Jamestown, NY. D. Song is with Texas A&M University, College Station, TX.

birds at the site.

The project builds on our past installations that have developed new models for collaborative observation drawing on computational geometry, stochastic modeling, and optimization [4].

CONE-Welder introduces several new features:

- Remote environment with extreme bird diversity
- Professionally designed feeding stations
- Lights for observation at night
- Flash interface for cross-browser compatibility
- Zone based image classification
- Multi-dimensional image and classification scoring metrics

A. Related Work and Previous Experience

Shortly after the introduction of the World Wide Web, Goldberg and his colleagues developed the Mercury (1994) and the Telegarden (1995-2004) networked telerobot systems. In the Telegarden, users could plant and water seeds remotely over a nine-year period [5].

These and many subsequent Internet-based telerobotic systems are surveyed in [6]. Goldberg and colleagues later explored multi-user control of telerobots and human "Tele-Actors' [7]. For recent examples see [8]–[10]. In other work, Kimber, Liu, Foote et al developed a multi-user robotic camera for video conferencing [11], [12].

The "frame selection problem" for a shared networked telerobotic camera was defined in [13]. They study the problem of controlling a single, online, robotic camera based on simultaneous frame requests from n users. The initial algorithm based on grouping and sorting of virtual corners had time complexity $O(n^2m)$ for n users and m zoom levels. Har-Peled et al. improved this to $O(mn^{3/2}log^3n)$ and proposed a near linear ϵ-approximation algorithm. Song et al. describe the approximate and distributed algorithms for solving the frame selection problem [14]. They show that with approximation bound ϵ, their algorithms runs in $O\left(n/\epsilon^3\right)$ time. They also show that their algorithm can be distributed to run with time complexity $O\left(n/\epsilon^3\right)$ at each client and in $O\left(n+1/\epsilon^3\right)$ at the server. See [15] for recent results.

In the context of avian observation, Chen and colleagues developed the Bird-Watching Learning system (BWL) [16]. BWL participants must be physically present at the observation site where they use camera-equipped PDAs. BWL explores how the network can be used to classify the resulting images.

CONE-Welder's zoning and classification game was inspired by von Ahn's PeekaBoom [17] game where remote users collaborate to identify and label photos. In CONE-Welder, users are restricted to a finite set of image classification labels.

B. System Architecture

CONE-Welder has three major components: a robotic camera, a multi-purpose network server, and a cross-platform client interface. As shown in Fig. 2, channeling communication through an intermediate server improves performance and ease of maintenance. This layout also allows us to modify content and functionality at a single location, promoting thin-client development.

C. Collaborative Camera Control

Since the camera can only stream video from one viewpoint at a time, we developed an algorithm to satisfy multiple, simultaneous user requests. Given a set of n simultaneous frame requests $R = \{r_1...r_n\}$, we must select an optimal single frame f (Fig. 3). We formalize and then minimize user dissatisfaction.

Specifically, we are interested in reducing the time taken to fulfill a request r_i or "time-dissatisfaction". Requests that have been waiting longer get weighted more. This metric prevents starvation, where frame requests are left unfulfilled indefinitely.

The CONE frame-selection algorithm minimizes both the mean and the variance of time-dissatisfaction across all requests r_i. Mean is a measure of general satisfaction and variance is a measure of fairness. The memory-less algorithm [18] only minimizes user dissatisfaction for a single frame. It does not take into account satisfaction over time. To solve this problem the time- dissatisfaction model is used in CONE [4]. In this model users expects to view their request for some amount of time T_{SAT}. Ideally, in a single user camera, a user's request is satisfied immediately. In a shared camera, the resulting frame may only cover a portion of a requested frame. In the time-dissatisfaction model, the "coverage-time" is also taken into account and a frame that is partially covered receives some satisfaction.

We define request fulfillment as how well the current response satisfies the original request and use intersection over maximum (IOM) metric:

$$S(f, r_i) = \frac{P_i}{max\left(A_f, A_{r_i}\right)} \tag{1}$$

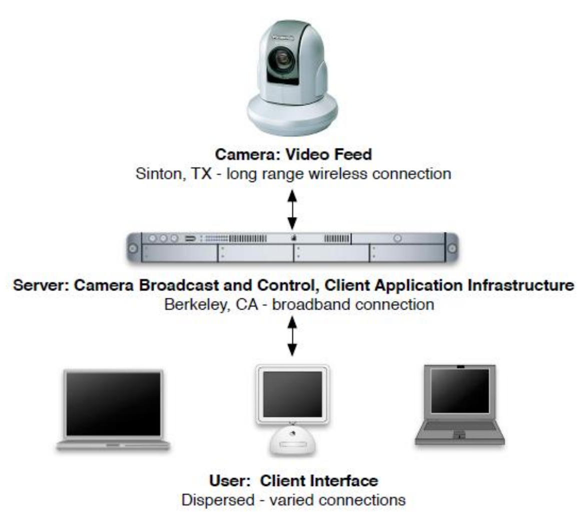


Fig. 2. Block diagram of the CONE system. The camera is communicated with using a relay server, and users receive video stream from the server. The server processes the user requests and runs the frame selection algorithm.

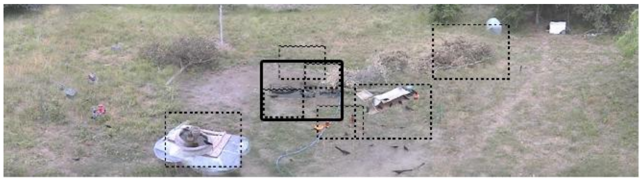


Fig. 3. Camera frame selection among multiple requests, adapted from [4]. Dotted rectangles represent user frame requests r_i. The solid bordered rectangle indicates the frame f chosen by the camera.

where f is a candidate frame, r_i is user i's frame selection, A_{r_i} is the area enclosed by r_i, A_f is the area enclosed by f, and P_i is the intersection of A_f and A_{r_i}. The maximum value is achieved when the requested frame and selected frame are identical, resulting in a complete intersection. This maximum considers both frame location and zoom level, since each requester is expecting a level of detail with their coverage. Fulfillment also depends on coverage time. Ideally, we would provide coverage as long as desired. Since this is often not possible, we define a set duration of time that a request can receive attention before being considered fulfilled.

Schmidt and Dahl implement these metrics in the "d-weighted" frame selector, developed for CONE Sutro Fores [4]. They used the time-satisfaction model as an input to the partial-frame satisfaction [14]. The goal is to maximize the total satisfaction per request weighted by time-dissatisfaction (eq. 2).

$$\sum_{r_i \in R} S(f, r_i) D_{r_i}(t) \tag{2}$$

The dissatisfaction value ($D_{r_i}(t)$) is defined recursively:

$$D_{r_i}(t) = \begin{cases} 1 & t = 0 \\ D_{r_i}(t-1)\left(1 - \frac{S(f,r_i)}{T_{SAT} - E(r_i,t)}\right) & S(f,r_i) > 0, t \neq 0 \\ D_{r_i}(t-1) + 1 & S(f,r_i) = 0, t \neq 0 \end{cases} \tag{3}$$

$E(r_i, t)$ is the cumulative coverage function. $E(r_i, t)$ is also calculated recursively:

$$E(r_i, t) = \begin{cases} 0 & t = 0 \\ E_{r_i, t-1} + S(f, r_i) & t > 0 \end{cases} \tag{4}$$

When $E(r_i, t) = T_{SAT}$ the request is satisfied. A requests dissatisfaction will decrease during fulfillment, but incrementally increase for each frame selection where the requested frame is neglected.

II. Client Interface

The CONE-Welder interface shown in (Fig. 1) is designed to look like a "bird lookout". Figure 4 is an example of zone classification where users can define a bounding box (zone) around the bird in the picture, and propose the name of the species. Submitted classifications help establish zone identifications through a voting scheme. When the total number of classifications exceeds a predefined threshold (currently 3), the zone is classified based on the majority vote (if there is one). If these conditions are not met, the zone remains unclassified. Figure 5 is a snapshot of the online classification report. It uses the timestamps of classified photographs to plot the number of daily sightings for each species. A general sighting plot is also available to help identify trends.

III. Game Model for Users

As an incentive for online users to gather data and generate useful information for field biologists, we designed a game where users gain points for taking and classifying photos. Leading players are recognized on leader boards. The system is designed to be self-organizing, in that scores are assigned automatically based on activity of all users. For example, there is a *Daily Parrot Award* for the most useful commentary of the day; *Cumulative Photographer Award* for the user who generates interest in another user's photo; and *Cumulative Primary Classification Award* given to the first person to classify a new zone correctly. There are currently seven different types of awards in the system. As of April 2009 there were over 97,000 awards with total value of over 125,000 points. A diagram of the total daily values of awards is shown in (Fig. 6).

A. Rating-based Awards

User interest of a photograph can be related to the presence of novel events that are valuable to researchers. We attempt to capture this interest through the "star" rating system (Fig. 4). Aggregate calculations over the data can help identify these interesting pictures. We reward the photographers of these pictures with the *daily "Eagle Eye" award*. To determine the winner, we consider all photographs taken on the target day. Valid candidates within this set must also have at least one rating from the target day. Each candidate photograph p_i is assigned a value $V(p_i)$ according to the following formula:

$$V(p_i) = \sum_{r \in R_i} \left(r - \frac{S_{max} + S_{min}}{2} \right), S_{min} \leq r \leq S_{max} \tag{5}$$

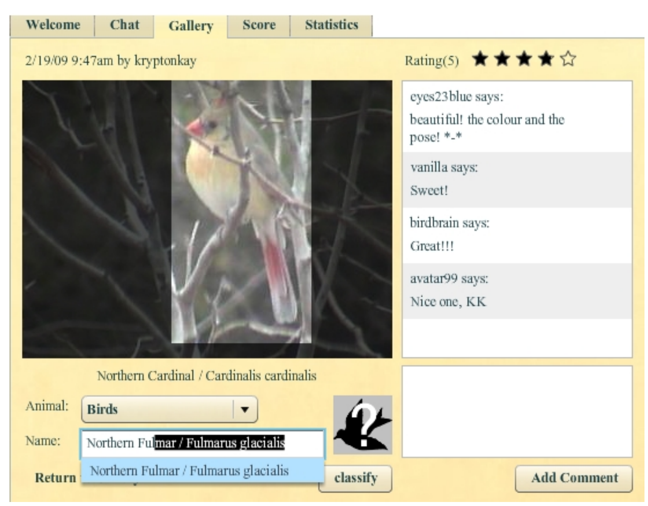


Fig. 4. The box drawn around the bird is an example of Zone Classification. On this image, the photographer and at least three other participants have classified the bird in the Zone as a Northern Cardinal. Users can also rate each photo by assigning stars to each picture (top right)

where R_i is the set of submitted ratings for p_i, S_{min} is the least possible stars given by a rating (In the current

version of CONE $S_{min} = 0$), and S_{max} is the greatest possible stars given by a rating (currently $S_{max} = 5$). This formula is based on the idea that not every submitted rating is positive. It adjusts the rating system so that a mid-range rating is considered neutral. The photograph with the highest, positive, calculated value receives the award. In the event of a tie, all winners receive the award, but points are equally distributed among them. Low-rated photographs could potentially win this award when the candidate set is small. We account for differing levels of competition by rewarding points as a function of the candidate size. Each award a has a point value $P(a)$, calculated as follows:

$$P(a) = B_A \left(\frac{C_a}{D_A} \right), a \in A \qquad (6)$$

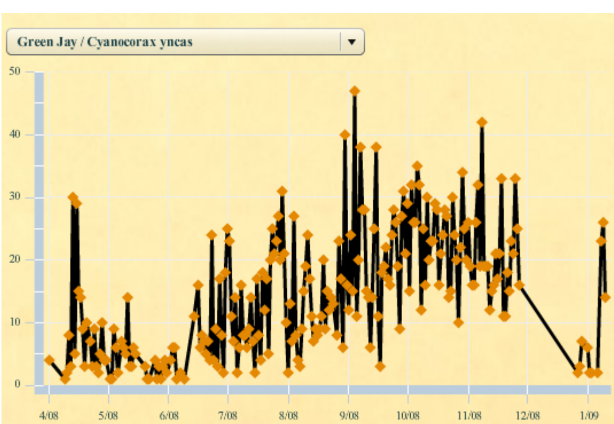
where A is the award type of a, B_A is the base point value for A, C_a is size of the candidate photograph set for a, and D_A is a scaling factor for A. A maximum and minimum value is also imposed to restrict the possible point range.

B. Time-based Awards

To encourage users to use the system throughout a 24-hour period, we initiated the "Early Bird" and "Night Owl" awards. We consider these conditions in our scoring model by comparing the winning photograph to the runner up. The awarded point value is determined by an exponential decay function. The function awards points based on how close the winning photograph is time-wise to the runner up photograph.We designed this function to have a maximum value of 10 points and a half-life of one hour. It helps prevent users from targeting set times by awarding a negligible amount of points for solitary camera operation.

IV. Data

Table I summarizes system usage. These data are summarized in Figure 7 where the number of log-ins to the system is modeled as a non-homogenous Poisson process.



Graph of Green Jay sightings over time.

Fig. 5. Avian data visualizations on the CONE-Welder site.

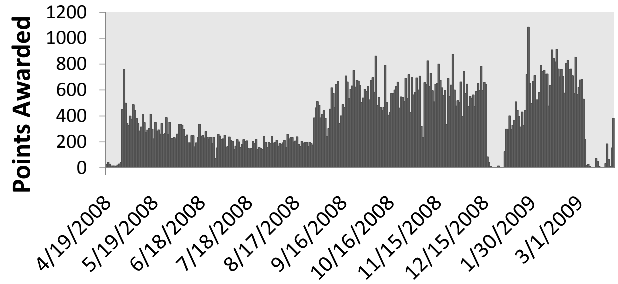


Fig. 6. Daily value of game points as of 6 April 2009. There are two maintenance periods in the diagram during which no points were allocated

Over 460 users have logged in to the system between April 18, 2008 to April 6, 2009. Of these, 256 users have contributed to classification and zoning. CONE has a highly dedicated community of active users. The 30 most active users account for 120,838 score points, 96.4% of the total. A histogram of the number of snapshots is shown in Fig. 8.

A. Image Classification

Image classifications are useful to researchers to help document new species and track previously seen species. Participants defined zones, each a classification opportunity, on 93 percent of all photographs. Among these zoned photographs, 73 percent had at least one zone with a consensus classification. Furthermore, consensus classifications were established with an average of 4.5 votes. Users have identified a total of 74 unique species to date, shown in Fig 9. These results confirm that Welder Refuge is extraordinarily diverse, and also confirm the the presence of eight species whose range did not extend to the Welder refuge 30 years ago [3].

The project produced a collection of bird images with more than 29,000 individual photos. This collection can serve as a training set for bird detection algorithms. The data set is available at (http://cone.berkeley.edu/frontdesk/gallery/).

B. Avian Range Change Data

There are fifteen Subtropical or Balconian (central Texas) species that now occur at Welder during the breeding period that were not there as breeders 30 years ago [1], [19], [20]. We have documented the presence of eight of these species by photos taken at the CONE-Welder site (Table II). Photographs of a newly-fledged Green Jay and a juvenile Bronzed Cowbird

TABLE I
SUMMARY OF STATISTICS AS OF APRIL 6,2008

Case	Amount
Frames requested by users	2,294,535
Frames selected by the system	2,014,196
Species	723
Subselections	33,110
Comments	15,609
Ratings	15,362
Awards distributed	97,326
Total value of awards	125,375

TABLE II
SUBTROPICAL OR BALCONIAN (CENTRAL TEXAS) SPECIES THAT NOW OCCUR AT WELDER DURING THE BREEDING PERIOD THAT WERE NOT THERE AS BREEDERS 30 YEARS AGO

Species	Photos
Green Jay (Cyanocorax yncas)	3659
Bronzed Cowbird (Molothrus aeneus)	1710
Buff-bellied Hummingbird (Amazilia yucatanensis)	1671
Black-chinned Hummingbird (Archilochus alexandri)	768
Great Kiskadee (Pitangus sulphuratus)	516
Eastern Bluebird (Sialia sialis)	144
Audubon's Oriole (Icterus graduacauda)	28
Couch's Kingbird (Tyrannus couchii)	12

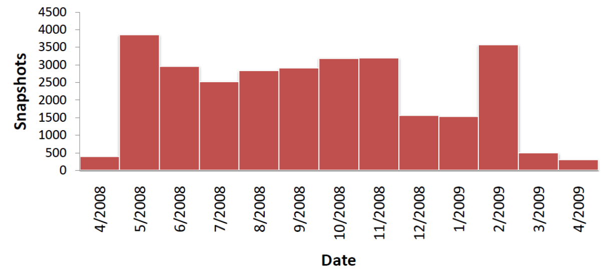


Fig. 8. Histogram of the number of snapshots taken by users.

being fed by a Northern Cardinal confirm breeding by those species. A juvenile Eastern Bluebird was photographed in July 2008. In addition, we obtained photos of color-banded Green Jays from every month of the year, demonstrating year- round residency for this species at Welder.

V. CONCLUSION AND FUTURE WORK

Our three-year NSF project is drawing to a close; this paper describes our latest Collaborative Observatory for a Natural Environment to date. CONE-Welder is installed at the Welder Wildlife Foundation in Texas to gather photographic and quantitative data for a biological study of avian distribution and activity. We described the system architecture, the game interface that provides incentives for player participation and initial data collected.

In future work, to collect data when no users are online and to provide an ongoing stream of interesting video for passive viewers, we are developing an "autonomous agent" to control the camera. It will be based on statistical learning of patterns and desired frames from the first year of user/frame/timing data that we collected. We are working on software to automatically detect if a bird is visible in the video frame using machine vision. We are interested in determining if it is possible to automate the "search" for birds in the camera's workspace.

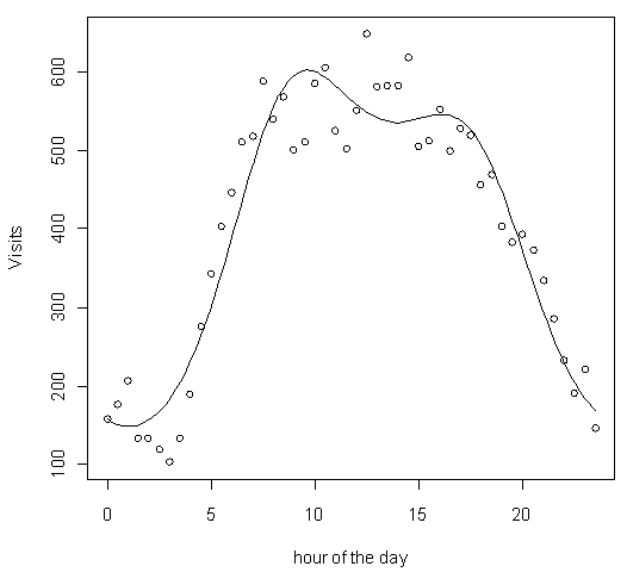


Fig. 7. Poisson regression on the number of visits per hour of the day. Midnight is represented as time $t = 0$.

We hope CONE-Welder can be a useful model for observatories in other locations. The system has been remarkably robust, remaining online 24 hours a day for approximately one year so far, with short periods of downtime due to network interruptions and maintenance. It has attracted over 600 "citizen scientists", with over 30 dedicated regular (daily) players, who have collectively identified over 70 unique species, including eight unexpected bird species whose breeding range previously did not include the Welder Refuge. A large corpus of photographic, taxonomic, and timing data have been collected. In many photos, colored leg bands can be clearly distinguished. This set also provides valuable machine learning information for researchers in the field of computer vision. We look forward to analyzing these data, and will report results in a future publication.

VI. ACKNOWLEDGMENTS

Creating and maintaining CONE-Welder requires the expertise and dedication of many individuals across multiple disciplines. We greatly appreciate NSF grants 0535218/0534848 and support from Microsoft Research HRI program and Panasonic. We wish to thank Jeff Tang, Connie Wang, Yan Zhang, Ephrat Bitton, George Lee, Paul Oh, the Berkeley Center for New Media, Berkeley Automation Sciences Research Lab, Jean-Claude Junqua and Deanna Wilkes-Gibbs from Panasonic Research Labs and Stewart Tansley of Microsoft Research and Tom Kellie. John Callender set up the CONE weblog (http://www.elanus.net/sutro/). We thank Chang Young Kim, Yiliang Xu, and Hongpeng Wang at Texas A&M. Support for field aspects of the project comes from the Smithsonian National Zoological Park, Conservation and Science Directorate, and the Welder Wildlife Foundation. Permits for the bird work are provided by Texas Parks and Wildlife Department, the National Bird Banding Lab, and the Smithsonian National Zoological Park Animal Care and Use Committee. Field and logistical assistance are provided by the late James Cox, B. C. Glasscock, Baldomar Martinez, and Alyssa Taylor, Chris McLean, Marlin Frederick and many Welder Foundation volunteers. Linda Kelley, Kay Loughman, Eleanor Pirry, Kirsten Hunter, Brooke Miller,Rafa Serra, Paul Lynch, Arleen Vieira and Ann Resnick provided feedback on the CONE system. We also wish to thank former Welder Wildlife Foundation Director, Lynn Drawe, and current Director, Terry Blankenship, for permission to work at the Welder site and the hundreds of online users of CONE for their invaluable data.

References

[1] G. Blacklock, "Checklist of Birds of the Welder Wildlife Refuge," in *Welder Wildlife Foundation, Sinton, Texas.*, 1984.

[2] S. Glasscock and T. Blankenship, "Checklist of the Birds of the Welder Wildlife Refuge," in *San Patricio County, Texas. Rob & Bessie Welder Wildlife Foundation, Sinton, Texas*, 2007.

[3] J. Rappole, G. Blacklock, and J. Norwine, "Apparent rapid range change in south texas birds: response to climate change," *The changing climate of south Texas 1900*, vol. 2100, pp. 133–146, 2007.

[4] A. Dahl, "Implementation of a Collaborative Observatory for Natural Environments," 2007.

[5] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and J. Wiegley, "Desktop teleoperation via the world wide web," in *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, vol. 1. IEEE, 1995, pp. 654–659.

[6] K. Goldberg and R. Siegwart, *Beyond webcams: an introduction to online robots*. The MIT Press, 2002.

[7] K. Goldberg, D. Song, and A. Levandowski, "Collaborative teleoperation using networked spatial dynamic voting," *Proceedings of the IEEE*, vol. 91, no. 3, pp. 430–439, 2003.

[8] J. Kim, B. Choi, S. Park, K. Kim, and S. Ko, "Remote control system using real-time mpeg-4 streaming technology for mobile robot," in *Consumer Electronics, 2002. ICCE. 2002 Digest of Technical Papers. International Conference on*. IEEE, 2002, pp. 200–201.

[9] J. Schiff, M. Meingast, D. Mulligan, S. Sastry, and K. Goldberg, "Respectful cameras: Detecting visual markers in real-time to address privacy concerns," *Protecting Privacy in Video Surveillance*, pp. 65–89, 2009.

[10] D. Song, N. Qin, Y. Xu, C. Kim, D. Luneau, and K. Goldberg, "System and algorithms for an autonomous observatory assisting the search for the ivory-billed woodpecker," in *Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on*. IEEE, 2008, pp. 200–205.

[11] D. Kimber, Q. Liu, J. Foote, and L. Wilcox, "Capturing and presenting shared multi-resolution video," in *SPIE ITCOM 2002. Proceeding of SPIE, Boston*, vol. 4862. Citeseer, 2002, pp. 261–271.

[12] Q. Liu, D. Kimber, L. Wilcox, M. Cooper, J. Foote, and J. Boreczky, "Managing a camera system to serve different video requests," in *Multimedia and Expo, 2002. ICME'02. Proceedings. 2002 IEEE International Conference on*, vol. 2. IEEE, 2002, pp. 13–16.

[13] D. Song, A. van der Stappen, and K. Goldberg, "Exact algorithms for single frame selection on multiaxis satellites," *Automation Science and Engineering, IEEE Transactions on*, vol. 3, no. 1, pp. 16–28, 2006.

[14] D. Song, A. Pashkevich, and K. Goldberg, "Sharecam part II: Approximate and distributed algorithms for a collaboratively controlled robotic webcam," in *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings*, vol. 2, 2003.

[15] D. Song, *Sharing a Vision: Systems and Algorithms for Collaboratively-Teleoperated Robotic Cameras (Springer Tracts in Advanced Robotics)*. Springer, 2009.

[16] Y. Chen, T. Kao, and J. Sheu, "A mobile learning system for scaffolding bird watching learning," *Journal of Computer Assisted Learning*, vol. 19, pp. 347–359, 2003.

[17] L. Von Ahn, R. Liu, and M. Blum, "Peekaboom: a game for locating objects in images," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM New York, NY, USA, 2006, pp. 55–64.

[18] D. Song, N. Qin, and K. Goldberg, "Systems, control models, and codec for collaborative observation of remote environments with an autonomous networked robotic camera," *Autonomous Robots*, vol. 24, no. 4, pp. 435–449, 2008.

[19] H. Oberholser, "The bird life of texas," *Univ. Tex. Press, Austin*, 1974.

[20] J. Rappole and G. Blacklock, *Birds of the Texas Coastal Bend: abundance and distribution*. Texas A&M University Press, College Station, 1985, vol. 84.

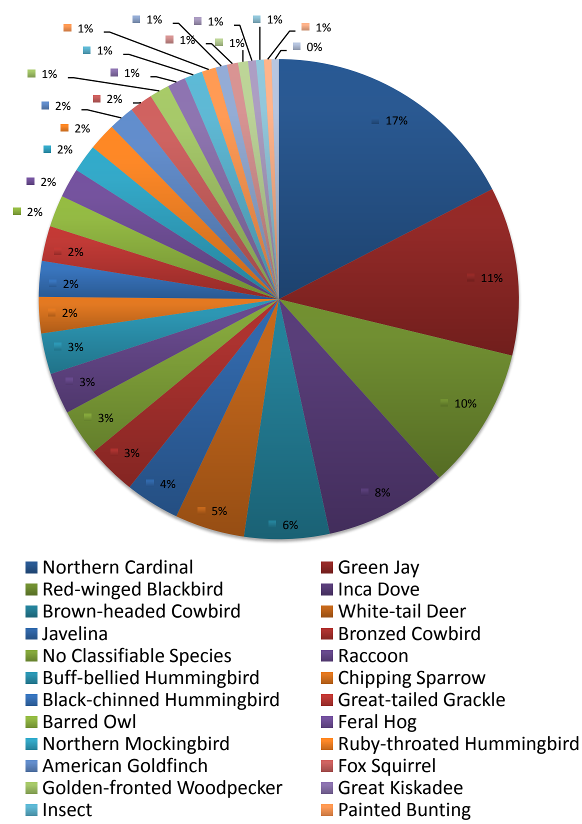


Fig. 9. Species Classification Totals: April 28, 2008 - April 6, 2009

Robotic Aircraft for Remote Sensing of the Environment

Mitch Bryson, Salah Sukkarieh, Ali Göktoğan, Jeremy Randle, Muhammad Esa Attia, Steve Keep, Alistair Reid, Calvin Hung, Tariq Abuhashim, Zhe Xu, Nick Lawrance, Todd Lupton, Abdallah Kassir, Jen Jen Chung, Graham Brooker and Nazifa Tahir

***Abstract*— Robotic aircraft provide a unique and valuable means for remote sensing of the environment. They can traverse large distances with minimal energy requirements and hence are more environmentally friendly than manned systems; can carry significant sensor payload capacity compared to other robotic systems; can operate at lower altitudes than manned aircraft and hence can attain higher spatial and temporal resolution; and can be configured for intelligent and adaptive sampling. Over the last five years the aerial robotics group at the Australian Centre for Field Robotics (ACFR) has conducted a number of significant experiments and field trials using robotic aircraft in a number of remote sensing applications including terrain mapping, vegetation classification, and animal tracking. Our research has focussed principally on novel data fusion and classification algorithms suited for such tasks, as well as control strategies for adaptive mapping, search and track. This workshop paper provides an overview and examples of this research and a discussion of the future potentials for aerial robotics in environmental monitoring.**

I. Introduction

Robotic aircraft or Unmanned Aerial Vehicles (UAVs) provide a unique and valuable means for remote sensing of the environment. They can traverse large distances with minimal energy requirements and hence are more environmentally friendly than manned systems; can carry significant sensor payload capacity compared to other robotic systems; can operate at lower altitudes than manned aircraft and hence can attain higher spatial and temporal resolution; and can be configured for intelligent and adaptive sampling.

There has thus been much recent interest in the use of UAVs in environmental monitoring and ecology studies; recent applications of interest include weed monitoring [1], [4] crop health assessment [6], rangeland monitoring [7], antarctic ecology [5] and wildlife surveys [8], [11]. These applications take advantage of the low-cost and high-resolution imagery available from UAV systems. Remote sensing data available from traditional means such as from satellites or manned aerial photography often suffer from fundamental limits to the spatial and temporal resolution available; spatial resolution is often limited by the high operating altitude of these platforms and temporal resolution is often limited by the high-cost of data collection (particularly from manned aircraft), which is particularly important in small-scale environmental science and ecology studies.

This work was based at the Australian Centre for Field Robotics, University of Sydney, NSW 2006, Australia `m.bryson@acfr.usyd.edu.au`, `salah@acfr.usyd.edu.au`

Fig. 1. Aerial robotic platforms used in environmental monitoring at the ACFR: Left, 1/3 scale J3 Cub fixed-wing UAV with sensor payload consisting of GPS, IMU and downwards-looking colour vision camera. Right, autonomous helicopter with attached spray booms used in precision herbicide delivery, and sensor payload consisting of GPS and downwards-looking colour vision camera.

This paper summarises research efforts at the Australian Centre for Field Robotics (ACFR) in using UAVs for environmental monitoring applications. Results from three example applications are presented, the first two focussing on weed detection and mapping using both hovering and fixed-wing UAV systems, and the last application focusing on the use of UAVs for studying insect band movements.

II. Aquatic Weed Surveillance using a Robotic Helicopter

This work focused on the development of a hovering UAV system (robotic helicopter) (see Figure 1, right), for the monitoring and spraying of aquatic weed species that grow in river systems, wetlands and irrigation channels. Aquatic weeds such as Salvinia (Salvinia molesta) and Alligator weed (Alternanthera philoxeroides) have been named as weeds of national significance in Australia due to their invasive spread into waterways, creating thick blankets of vegetation across water surfaces that kill bottom-growing native vegetation, deoxygenate water and kill fish populations. The developed

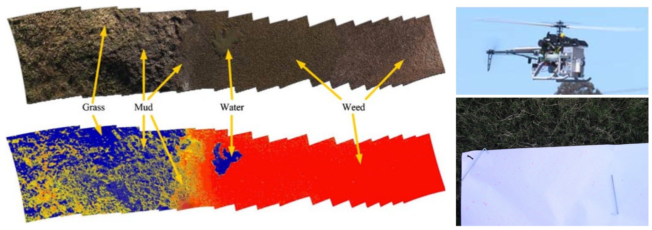


Fig. 2. Monitoring and spraying of aquatic weeds using a hovering UAV system: Left top, stitched mosaic of images of a weed-infested area taken by the hovering UAV. Left bottom, The same stiched mosaic colour-labelled based on weed probability from an offline weed classification process (high-probability in red, medium probability in yellow and low probability in blue). Right top, autonomous spraying system in action for delivering herbicide to invasive weeds. Right bottom, ground-based photo of a white paper target sprayed with colour dye by the hovering UAV during an autonomous spray test.

system carries a sensor payload consisting of a downwards-looking colour camera, magnetometer and GPS receiver to collect geo-referenced imagery while hovering at a low-altitude over river banks and wetlands. An onboard autopilot was used to guide the helicopter over a target area to collect imagery; after landing this data was then processed on the ground to detect weed species in the imagery. The detected weed locations were then mapped based on GPS recorded positions and the UAV was used in a second flight to hover over weed points and deliver herbicide to the effected area.

A. *Results*

Figure 2, left, illustrates a stitched mosaic of images collected from the helicopter and the associated classification colour associated with the probability of areas in the image corresponding to a weed or non-weed class. A Support Vector Machine (SVM) classifier was used to distinguish each patch of collected imagery as either weed or non-weed based on human-expert training examples provided on collected data. Figure 2, right, illustrates the spraying system installed on the helicopter in action. Due to the hazards involved in handling herbicide, the spray system was demonstrated using coloured dye. White paper targets were placed into the environment as spray targets for the helicopter, and the amount of dye transferred onto the paper during spraying was used as a measure of the spraying accuracy.

III. Landscape Mapping and Classification using a fixed-wing UAV

This work focused on the development of a fixed-wing UAV system (see Figure 1, left), sensor payload and data processing algorithms for producing high resolution 3D maps of the environment with classified object labels. The use of the fixed-wing platform allowed for traversal over large scale areas (with transects on the order of 2km long) but with a spatial resolution of 3.5cm on the ground, an unprecedented level of resolution when compared to manned aerial surveys. This work was motivated by two applications, the first which involved the mapping of woody weed species in open agricultural grasslands, the second involved the mapping of invasive cacti over inaccessible semi-arid rangelands. Woody weeds such as Prickly Acacia (Acacia nilotica) and Mesquite (Mesquite prosopis) are a huge problem in Australia, particularly on agricultural land where they outcompete native vegetation, cause land access and cattle mustering issues and use water resources in grassland and woodland habitats.

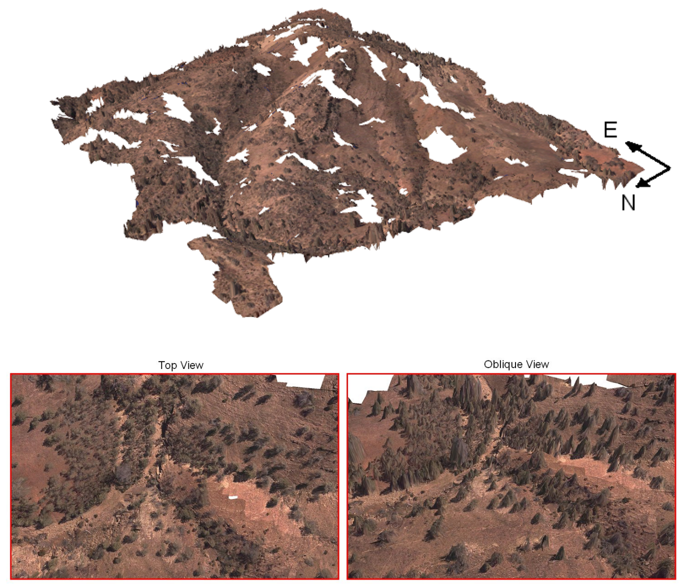


Fig. 3. 3D terrain mapping results using sensor data collected over the Flinders Ranges, South Australia: Vision data is collected from a low-altitude (100m above the ground) compared to manned aircraft, enabling 3D reconstruction of the landscape including the rough 3D shape of tree canopies without the need for airborne radar or lidar.

A. *Overview of Robotic Data Collection*

Sensor data was collected over target areas by flying a trajectory of overlapping swaths from a fixed altitude above the terrain. A flight path as developed such that subsequent images captured by the on-board camera would present points on the ground in at least 3 images while flying in a straight line. Parallel swath trajectories were planned such that sufficient overlap would be achieved across swaths as to observe objects in the terrain for a variety of perspectives. The fixed-wing UAV was fitted with an off-the-shelf autopilot [9]; remote control flight with a pilot on the ground was used during takeoff and landing of the UAV on an unprepared runway (a dirt track) with control handed to the autopilot when the UAV was in the air. The autopilot system allowed for autonomous over-the-horizon operation of the system where telemetry data was sent to a ground station operated by a single person to monitor the progress of the mission and ensure the UAV was operating safely.

B. *Overview of Mapping and Classification Algorithms*

Once the UAV had completed the mission, the collected sensor data was downloaded from the aircraft and processed

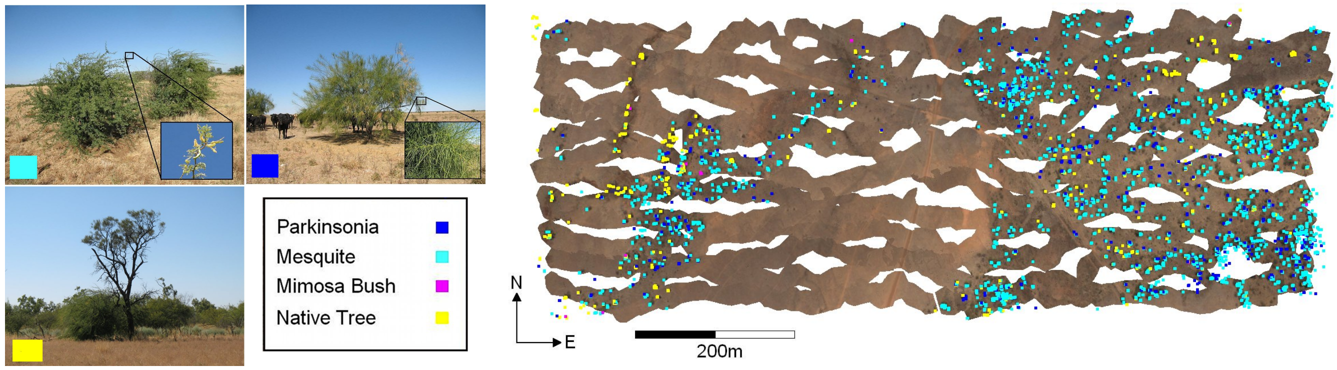


Fig. 4. Classified map results of woody weed and native vegetation using sensor data collected over farmland in West Queensland, Australia: The coloured points in the map represent different classes detected by the supervised classification approach and allow for the distinction of target weed species using low-cost visual-band imagery.

at the site using a collection of laptop computers. The aim of the data processing was to produce accurate, geo-referenced 3D maps of the terrain under the flight area and to detect and classify the different species of vegetation present in the map.

Mapping was performed in a multi-stage process. IMU and GPS information were used to compute an initial estimate of the position and orientation (full six degree of freedom pose) trajectory along the flight using an Extended Kalman Filter (EKF). Scale Invariant Feature Transform (SIFT) feature points were extracted from the images with correspondences found across both subsequent frames and overlapping swaths. Triangulation of 3D feature points was performed using extracted image coordinates and corresponding pose data computed using the IMU/GPS EKF. A non-linear least squares bundle adjustment phase was then used to optimise both the trajectory and 3D feature point estimates using all of the raw IMU, GPS and extracted vision feature data. The 3D feature points and corresponding vision frames were finally used to build a surface model of the terrain on which the collected imagery was used to texture the surface.

In parallel to the map construction process, image-based detection and classification was performed using the collected image data. A tree detection algorithm was used to segment regions in the image corresponding to tree crowns and vegetation [2]. A feature descriptor composed of colour and texture measures was then extracted from every detected tree segment in the image data. A multi-class classifier was then used to label each of the detected tree points and to add this label to the constructed 3D map. The classifier itself was created using a supervised learning approach in which a human expert was provided with several examples of vegetation from the aerial images and asked to label these into one of several classes based on vegetation species. The supervised learning approach allowed for the system to be tailored for different applications involving different environments and target species of vegetation.

C. Results

Figure 3 illustrates results of the 3D terrain mapping algorithms applied over inaccessible semi-arid rangelands in the Flinders Ranges, South Australia. The use of vision data collected from a low-altitude (100m above the ground) enables 3D reconstruction of the landscape including the rough 3D shape of tree canopies without the need for airborne radar or lidar. Figure 4 shows an equivalent terrain map with labelled vegetation points corresponding to different classes of woody weed and native tree species in farmland in West Queensland, Australia. The low-altitude flight of the robotic aircraft allowed for very high spatial resolution in the imagery which reduced the spectral mixing often present in manned aircraft or satellite remote sensing data and allowed for the discrimination of vegetation based on properties such as canopy texture, which would otherwise not be resolvable at a lower spatial resolution.

IV. Autonomous Tracking of Australian Plague Locusts using UAVs

This project focused on using UAVs for tracking and quantifying the movements of migratory bands of Australian plague locusts for the purposes of studying insect behaviour. Locust swarms can have a devastating impact on agriculture as locusts march along the ground, stripping and consuming crops. The aim of the project was to thus use the unique vantage point available to UAVs to track locust band movements and swarm behaviours, and to use this data to build models of locust swarm behaviour, which could later be used for efficient and targeted delivery of aerial pesticides.

The project involved the development of a sensor payload consisting of a high-shutter speed camera and flashing strobe mounted to the UAV and the development of upwards-looking micro-retro-reflectors that were attached to plague locusts on the ground. The retro-reflectors were designed to be small (2.5mm in diameter) and light weight such that when glued to the back of an adult locust they would not affect it's behaviour or motion. Locusts tend to march and migrate only during the day thus several locusts within a swarm were to be captured and tagged the night before flight

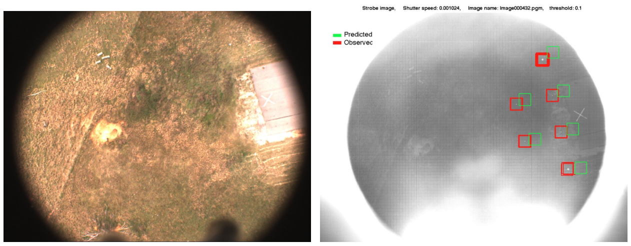


Fig. 5. UAV-mounted Camera Strobe System Test Results: Left, normal exposure colour image captured over the terrain. One end of a runway can be seen in the image; the distance across the image corresponds to approximately 120m on the ground. Right, strobe-synchronised short-exposure image with detected retro-reflector returns and predicted image locations of GPS-surveyed retro-reflectors on the ground. The offset between the predicted and detected locations is due to small errors in sensor timing and navigation system accuracy.

operations would be performed. The shutter speed of the UAV-mounted camera was tuned and synchronised to the on-board strobe such that the camera would return pinpoint reflections of locusts on the ground.

A. Initial Results and Ongoing Work

To date, the system described above has been demonstrated using retro-reflector targets at stationary points on the ground to assess the feasibility of the sensing strategy. Figure 5 illustrates camera imagery collected from the fixed-wing UAV system (see Figure 1, left) while flying over stationary reflector targets. The left subfigure illustrates an ordinary colour image taken with the camera using a shutter speed and exposure suitable to image the terrain. The right subfigure illustrates the high-shutter speed/strobe image over the same area. Image processing was used to extract the high-intensity responses corresponding to retro-reflector positions on the ground. The results indicate that reflectors can be reliably distinguished from background objects.

Future and ongoing work in the project is now focusing on tracking retro-reflectors mounted to live locusts. Since each reflector does not provide a unique return signal (compared to other retro-reflectors), robust data-association and tracking algorithms are being developed to correspond reflector measurements between captured frames and eventually track the position and velocity of individual locusts across the landscape.

V. Conclusions and Future Work

This paper has presented research into UAVs for environmental monitoring in a number applications including mapping, 3D landscape modelling, object detection, classification and tracking. The use of UAVs in environmental monitoring is an exciting new research area with many recent demonstrations, and has huge potential for future work. The following sub-sections summarise important areas of future research.

A. UAV Applications in Environmental Monitoring

Data collection in environmental monitoring has traditionally been performed using remote sensing systems such as satellite imagery, manned aerial photography and human visual observation from aircraft, watercraft or from the ground. UAVs and other robotic platforms offer a new vantage point from which to observe the natural world, and future work should focus on communicating with scientists, ecologists, land managers and decision makers to discover how some of this data collection can be supplemented with or replaced by robotic/autonomous platforms. UAVs provide the advantage of cheap, low-altitude, persistent and high-time resolution data collection and have been been identified in potential applications such as atmospheric science [3] and archaeology [10] amongst the other applications demonstrated in this paper.

B. Autonomous and Intelligent Data Collection

Existing developments in real-time airborne data fusion and mapping are currently limited to small areas, with much recent work focused on real-time mapping from robotic platforms in large 3D environments. Building maps online during data collection will allow future UAV systems to adapt operations from fixed and inflexible data collection plans, to real-time informed data collection strategies based on the information they collect during a mission. This is an exciting future area for robotic monitoring of the environment that will rely on advances in real-time data fusion, path planning and research into relating high-level mission or science goals into feasible data collection strategies such as focusing on specific areas or phenomena of interest.

C. Multi-Platform Robotic Exploration of the Environment

Multi-platform, cooperative robotic systems would further enable efficient data collection over large areas and enable applications involving time-synchronous collection of data over multiple sites or of one site from different spatial scales. This area of future research also includes the fusion of data collected from robotic systems with data collected by other means such as satellite and manned-airborne remote sensing, thus benefiting from the advantages that different platforms offer.

VI. Acknowledgments

This work is supported by Land and Water Australia under project code USY13/USY004628, Meat and Livestock Australia (MLA) under project code B.NBP.0474, the Australian Weeds Research Council (AWRC) under project code AWRC08-04, the ARC Centre of Excellence programme and Linkage Project LP0989291, funded by the Australian Research Council (ARC) and the New South Wales State Government.

References

[1] M. Bryson, A. Reid, F. Ramos, and S. Sukkarieh. Airborne Vision-Based Mapping and Classification of Large Farmland Environments. *Journal of Field Robotics*, 27(5):632–655, 2010.

[2] C.Hung, M. Bryson, and S. Sukkarieh. Vision-based Shadow-aided Tree Crown Detection and Classification Algorithm using Imagery from an Unmanned Airborne Vehicle. In *International Symposium on Remote Sensing of the Environment (ISRSE)*, 2011.

[3] T.H. Cox, C.J. Nagy, M.A. Skoog, and I.A. Somers. Civil UAV capability assessment. In *NASA Capability Assessment Report*, 2004.

[4] A. Goktogan, S. Sukkarieh, M.Bryson, J.Randle, T. Lupton, and C. Hung. A Rotary-wing Unmanned Air Vehicle for Aquatic Weed Surveillance and Managemen. *Journal of Intelligent and Robotic Systems*, 57(1-4):467–484, 2010.

[5] A. Lucieer, S. Robinson, and D. Turner. Using an Unmanned Aerial Vehicle (UAV) for Ultra-high Resolution Mapping of Antarctic Moss Beds. In *Australian Conference on Remote Sensing and Photogrammetry*, 2010.

[6] S. Nebiker, A. Annen, M. Scherrer, and D. Oesch. A light-weight multispectral sensor for Micro UAV - opportunities for very high resolution airborne remote sensing. In *The International Archives of the Photogrammetry Remote Sensing and Spatial Information Sciences*, 2008.

[7] A. Rango, A. Laliberte, C. Steele, J.E. Herrick, B. Bestelmeyer, T. Schmugge, A. Roanhorse, and V. Jenkins. Using Unmanned Aerial Vehicles for Rangelands: Current Applications and Future Potentials. *Environmental Practice*, 8:159–168, 2008.

[8] R. Speck, A.H. Goktogan, and S. Sukkarieh. Autonomous Airborne Wildlife Tracking Using Radio Signal Strength. In *International Conference on Intelligent Robots and Systems*, 2010.

[9] B. Vaglienti, R. Hoag, M. Niculescu, J. Becker, and D. Miley. Piccolo system user's guide. In *Cloud Cap Technology, http://www.cloudcaptech.com/download/Piccolo/User and Integration Guides/Version 2.1.2/Piccolo User's Guide.pdf*, 2009.

[10] G.J.J. Verhoeven. Providing an Archaeological Birds-eye View - an Overall Picture of Ground-based Means to Execute Low-altitude Aerial Photography (LAAP) in Archaeology. *Archaeological Prospection*, 16:233249, 2009.

[11] A.C. Watts, J.H. Perry, S.E. Smith, M.A. Burgess, B.E. Wilkinson Z. Szantoi, P.G. Ifju, and H.F. Percival. Small Unmanned Aircraft Systems for Low-Altitude Aerial Surveys. *Journal of Wildlife Management*, 74(7):1614–1619, 2010.

Efficient autonomous image mosaicing with applications to coral reef monitoring

Armagan Elibol,
Dept. of Mathematical Engineering
Yildiz Technical University, Istanbul Turkey
aelibol@yildiz.edu.tr

Nuno Gracias, Rafael Garcia
Computer Vision and Robotics Group
University of Girona, Girona Spain
{ngracias, rafa}@eia.udg.edu

Art Gleason, Brooke Gintert,
Diego Lirman and R. Pam Reid
University of Miami, Miami USA
{agleason,bgintert,dlirman,preid}@rsmas.miami.edu

Abstract—**Over the past decade, several image mosaicing methods have been proposed in robotic mapping and remote sensing applications. Owing to the rapid developments on obtaining optical data from areas beyond human reach, there is a high demand from different scientists for creating large-area image mosaics often using images as the only source of information. One of the most important steps in the mosaicing process is the motion estimation between overlapping images to obtain the topology, i.e, the spatial relationships between images. In this paper, we propose a generic framework for feature-based image mosaicing capable of obtaining the topology with a reduced number of matching attempts and to get the best possible trajectory estimation. Innovative aspects include the use of a fast image similarity criterion combined with a minimum spanning tree (MST) solution, to obtain a tentative topology. This topology is improved by attempting image matching over the pairs of higher overlap evidence. Unlike previous approaches for large-area mosaicing, our framework is able to naturally deal with the cases where time consecutive images cannot be matched successfully, such as completely unordered sets. We conclude this paper by presenting an environmental application of this mosaicing approach for monitoring coral reefs.**

I. Introduction

Image mosaicing methods have been widely used for panoramic imaging [1] and mapping [2]. Aerial and satellite imagery of the Earth's surface, merged with high-resolution topographic models, have proven a key tool to understand physical processes of our planet (geological, hydrological, biological, etc.), to monitor environmental changes, whether man-induced or natural, resource management, development and infrastructure planning (public works, remediation plans, land use, etc.). Robots are becoming more and more important in gathering optical data from places where human cannot reach. In robot mapping(i.e., aerial and/or underwater), when a robot is surveying a large area using only a down-looking camera, it is of interest to obtain a global view of the area. To have a wide-area visual representation of the scene, it is necessary to create large-area maps (mosaics). Mosaics enable different applications such as geological [3], [4] and archaeological surveys [5], ecology studies [6], [7], [8], environmental damage assesment [9], [10] and detection of temporal changes in [11]. Therefore, there is a high demand from different science communities for creating optical maps of areas where human cannot reach.

When a robot is mapping an area in a scattering media [4], illumination effects, noise, lack of image contrast and blurring are phenomena that make image registration difficult. This leads to inaccuracies in image registration that cause misalignment when images are mapped onto the mosaic (global) frame. To compose images into a mosaic form, several steps are needed and one of the most important steps is image registration. In the absence of any other information, most of the existing methods try the exhaustive strategy of matching all images against all. However, this approach is only feasible for a small number of images. Since large-area surveys might comprise of several hundreds to tens of thousands of images [4], [5], the all-against-all strategy becomes impractical.

To overcome this problem, we propose in this paper a generic image mosaicing scheme aiming to get the complete topology with minimum number of image matching attempts while simultaneously obtaining a globally coherent mosaic. The algorithm takes as input a set of images that has been previously acquired. The time order is not taken into account; therefore, the image set can be totally unordered. Our technique first infers image similarity information using a fast method based on the Euclidean distance between feature descriptors. Then, this similarity is used to create a tentative topology with associated uncertainty. The estimated uncertainty, although large at the beginning, is gradually reduced by successive iterations of image matching and bundle adjustment.

II. Related Work

Quality constraints of image mosaics are usually very strict, especially for mapping purposes, as the mosaic might be used for global navigation [12], localization of interest areas [4] and detection of temporal changes [11]. Several image mosaicing approaches for creating underwater mosaics have been proposed over the last decade [13], [14], [15], [16]. Pizarro et al. [13] proposed a mosaicing system that exploited navigation and attitude information for bundle adjustment. Madjidi and Negahdaripour [14], addressed the global alignment problem for a submersible equipped with stereo cameras, using a mixed adjustment model to recursively determine the pose of the vehicle. Rzhanov et al. described in [15] a methodology that exploited navigation data to build geo-referenced photomosaics of the mid-ocean ridges at the East Pacific Rise. Ferrer et al. [16] proposed a global alignment method for creating

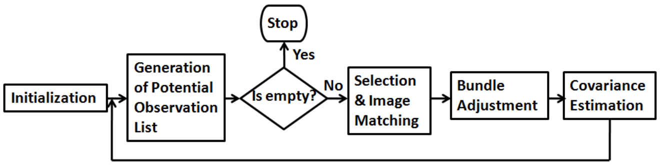


Figure 1. Pipeline of the algorithm.

large-scale underwater photo-mosaics that combines image registration information and 3D position estimates provided by navigation sensors available in deep water surveys. Bulow et al. [17] proposed an online mosaicing (image-to-mosaic registration) method for Unmanned Aerial Vehicles using an image registration method based on Fourier-Mellin transformation. As they have stated, the proposed method fails if there is not enough overlapping area between time consecutive images. None of the methods mentioned above have concentrated on the challenge of finding the non-time consecutive matches when only image information is available.

III. Topology Estimation

In this work, we assume that the optical axis of the camera is kept perpendicular to the scene, which is approximately flat [1]. Each image has an associated planar transformation [18] with 4 degrees of freedom, ${}^{M}\mathbf{H}_i$, detaied in Eq. (1), that relates the image frame i to a common mosaic frame M (i.e., absolute homography).

$$ {}^{M}\mathbf{H}_i = \begin{bmatrix} a_i & -b_i & c_i \\ b_i & a_i & d_i \\ 0 & 0 & 1 \end{bmatrix} \qquad (1) $$

For simplicity, we consider the reference frame to be the coordinate system of the first image, so that ${}^{M}\mathbf{H}_1$ is equal to identity, thus it is not part of the parameter vector to be estimated.

Our scheme is composed of five different steps: (1) Initialization, (2) Generation of the list of potential overlapping image pairs, (3) Image selection and matching, (4) Minimisation of the reprojection error and (5) Covariance propagation. The pipeline of the proposed method is illustrated in Fig. 1

Initialization The initialization step aims at obtaining information on the similarity between images and to establish an initial link between them. This similarity information is intended to be computed in a fast and approximate way. First, Scale Invariant Features (SIFT) [19] are extracted. Then a small subset of feature descriptors (between 100 and 200) are randomly selected from each image, and compared against the subsets of all other images. This comparison is performed using the Euclidean distance between feature descriptors [19].

For a given pair of images, our similarity measure is proportional to the number of descriptors that are associated using the distance criterion. The computational cost of this similarity measure is comparatively low, since it mainly involves computing the angles between a small set of descriptor vectors. Our multi-threaded C implementation allows for computing the measure in 2.5 miliseconds on a standard desktop machine for a pair of images with 200 descriptors each. In order to establish the initial link between images, we use a MST where weights of the edges are the inverted initial similarity values. MST of a weighted graph is a subset of edges that form a tree whose sum of weights of edges is minimum[20]. MST provides a connected tree which is composed of the most similar image pairs according to the similarity information. The initial relative homographies between those image pairs that are in the MST are treated as identity mappings with very large uncertainty. Using these relative homographies, the absolute homographies are computed along with its uncertainty which is propagated using a first order approximation [21].

Finding Potential Overlapping Image Pairs This step aims to find the overlapping image pairs given an estimate of the absolute homographies and its uncertainty. We propose to use an approach which employs two successive different tests. The first test computes the distance between image centers by taking into account their uncertainties. If this distance is smaller than a selected threshold (such as, size of the image diagonal) then the second test is applied. The second test consists of generating several noisy instances of homographies using the propagated covariances and computing the overlapping area between images. If the normalized average overlapping area is above a given threshold (e.g., 30%), then the pair is added to the list of potential overlapping image pairs.

Image Selection and Matching This step starts by selecting a subset of image pairs from the potential overlapping pair list. The main reason for this selection is that it is not feasible to attempt to match the whole list since the list might contain many non-overlapping pairs due to the high uncertainty and drift on the current absolute homographies. We have used the estimated overlapping area between potential overlapping pairs as a ranking criterion. The size of the subset is determined by a simple Computational Time criterion, where the total matching time for the subset is approximately equal to the computational time spent on all other steps in the iteration (generation of list of potentially overlapping pairs, bundle adjustment and covariance propagation). For image matching, features are detected and matched using SIFT [19], followed by outlier rejection and motion estimation [18].

Minimizing the Reprojection Error The error terms resulting from image registration are measured in the image reference frames. We have employed a standard Bundle Adjustment (BA) approach [12] which minimizes the weighted reprojection error over homographies. Reprojection error is expressed as follows:

$$ \varepsilon = \sum_{k} \sum_{t} \sum_{j=1}^{n} \parallel {}^{k}\mathbf{p}_j - {}^{M}\mathbf{H}_k^{-1} \cdot {}^{M}\mathbf{H}_t \cdot {}^{t}\mathbf{p}_j \parallel_2 + \parallel {}^{t}\mathbf{p}_j - {}^{M}\mathbf{H}_t^{-1} \cdot {}^{M}\mathbf{H}_k \cdot {}^{k}\mathbf{p}_j \parallel_2 \qquad (2) $$

where k and t are a pair of images that were successfully

[1]In this work, it is assumed that the navigation altitude of the vehicle is large with respect to the 3D relief of the scene

matched, n is the total number of correspondences between the overlapping image pairs, $({}^M\mathbf{H}_k, {}^M\mathbf{H}_t)$ are the absolute homographies for images k and t, respectively. ${}^k\mathbf{p}_j = ({}^k x_j, {}^k y_j, 1)$ encodes the coordinates of the j^{th} feature point in image k, while ${}^t\mathbf{p}_j$ are the coordinates of the same scene point in image t. The weight included cost function is given in Eq. (3), which is the L_2 norm of a stack of weighted residues. f is minimized over θ, which contains the parameters for all image homographies.

$$f = \mathbf{R}^T \cdot \mathbf{W} \cdot \mathbf{R} \tag{3}$$

where $\mathbf{R} = \left[\left| \begin{array}{c} {}^i\mathbf{r}_j^k = {}^i\mathbf{p}_k - {}^i\mathbf{H}_j \cdot {}^j\mathbf{p}_k \\ {}^j\mathbf{r}_i^k = {}^j\mathbf{p}_k - {}^j\mathbf{H}_i \cdot {}^i\mathbf{p}_k \end{array} \right|_{stack} \right]$ is a $4N_{pm} \times 1$ vector and $\mathbf{W}$ is a diagonal $4N_{pm} \times 4N_{pm}$ matrix of weights for each residue. N_{pm} is the total number of correspondences. Finally, ${}^i\mathbf{H}_j = {}^i\mathbf{H}_M \cdot {}^M\mathbf{H}_j$ and ${}^j\mathbf{H}_i = {}^j\mathbf{H}_M \cdot {}^M\mathbf{H}_i$. The minimisation of the cost function in Eq. 3 was carried out using the MATLAB™ *lsqnonlin* function for large-scale methods. The optimization algorithm requires the computation of the Jacobian matrix containing the derivatives of all residuals with respect to all parameters. The Jacobian matrix has a clearly defined block structure, and the sparsity pattern is constant [22], [23]. In our implementation, analytic expressions were derived and used for computing the Jacobian matrix.

Covariance Propagation We apply Haralick's method [21] to propagate the uncertainty of the resulting homography estimations of BA. f in Eq. 3 is a function of parameter vector θ and $\mathbf{x}$ containing all data affected by noise. After optimization, the first order approximation to the uncertainty in the parameters is given by [21]

$$\mathbf{\Sigma}_\theta = \left(\frac{\partial g}{\partial \theta}\right)^{-1} \cdot \frac{\partial g}{\partial \mathbf{x}} \cdot \mathbf{\Sigma}_\mathbf{x} \cdot \left(\frac{\partial g}{\partial \mathbf{x}}\right)^T \cdot \left(\frac{\partial g}{\partial \theta}\right)^{-1} \tag{4}$$

where $\mathbf{\Sigma}_\mathbf{x}$ is the covariance matrix of $\mathbf{x}$ and g is the jacobian of f with respect to θ.

IV. Experimental Results

The generic scheme described in the previous section was tested on a general setup for image surveys using an underwater robot equipped with a down-looking camera. We have tested our scheme on a real data set in which some time-consecutive images do not have overlapping areas. The dataset was extracted from an underwater image sequence acquired by a Phantom 500 ROV during a survey in Andros, the Bahamas [7]. This data set is composed of two horizontal and three vertical transects. The total number of images is 112. In addition, we have changed the order of the images to have more non-overlapping consecutive pairs between ordered images. The initial similarity matrix is depicted in Fig. 2. The resulting final trajectory and uncertainty can be seen in Fig. 3 while the resulting mosaic is depicted in Fig. 4 and Table I summarises the results. The first column of the table corresponds to the tested method. The second column shows the total number of successfully matched image pairs that have least 20 inliers. The third column contains the total number of image pairs that were not successfully matched (*unsuccessful observations*). The percentage of the total number of image matching attempts with respect to all-against-all image matching attempts is given in the fourth column. The last column corresponds to the average reprojection error calculated using all the correspondences with the resulting set of homographies for each tested strategy.

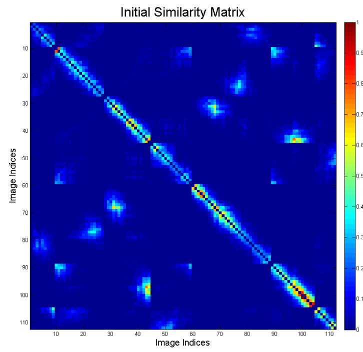


Figure 2. Initial Similarity Matrix of the dataset. This matrix was computed using a maximum of 200 feature points. Values are scaled to $[0, 1]$.

As there are some broken links between the time-consecutive images, the traditional iterative topology estimation method proposed in [12] cannot be applied. It can be observed in Table I that our scheme was able to get 94% of the total overlapping pairs with a considerably smaller number of image matching attempts. The second line shows the results for matching all the pairs for which the similarity matrix provides at least 20 descriptor associations to attempt RANSAC [18]. The third line is for the all-against-all strategy. Initial similarity matrix almost suggests all-against-all matching.

Table I
SUMMARY OF RESULTS.

Strategy	Successful Obs.	Unsuccessful Obs.	% of attempts wrt all-against-all	Avg. Error in pixels	Std. Dev. in pixels
1. Proposed Scheme	278	1,201	23.79	5.12	3.67
2. Similarity Matrix	294	5,900	99.65	4.86	3.61
3. All Against All	294	5,922	100.00	4.86	3.61

In order to show that the proposed scheme is not dependent on the image order in the dataset. We have also tested our scheme on a small dataset that is composed of approximately two transects having a few overlapping pairs between them. We have changed the image order fully in random manner only keeping the first image same. This is mainly to represent the topology in common global frame. Then, we reorganize the initial similarity matrix by taking into account this randomly generated new image order. The initial similarity matrices are depicted in Fig. 5. We have run our scheme on both original captured order and the randomly generated order of images. Results are summarized in Table II.

From the results, it can be seen that our scheme can work with fully unordered datasets as it uses similarity matrix obtained from images. Final trajectory and uncertainties on image centers are given in Fig.6 Final trajectory for the randomly ordered images is given in Fig.7.

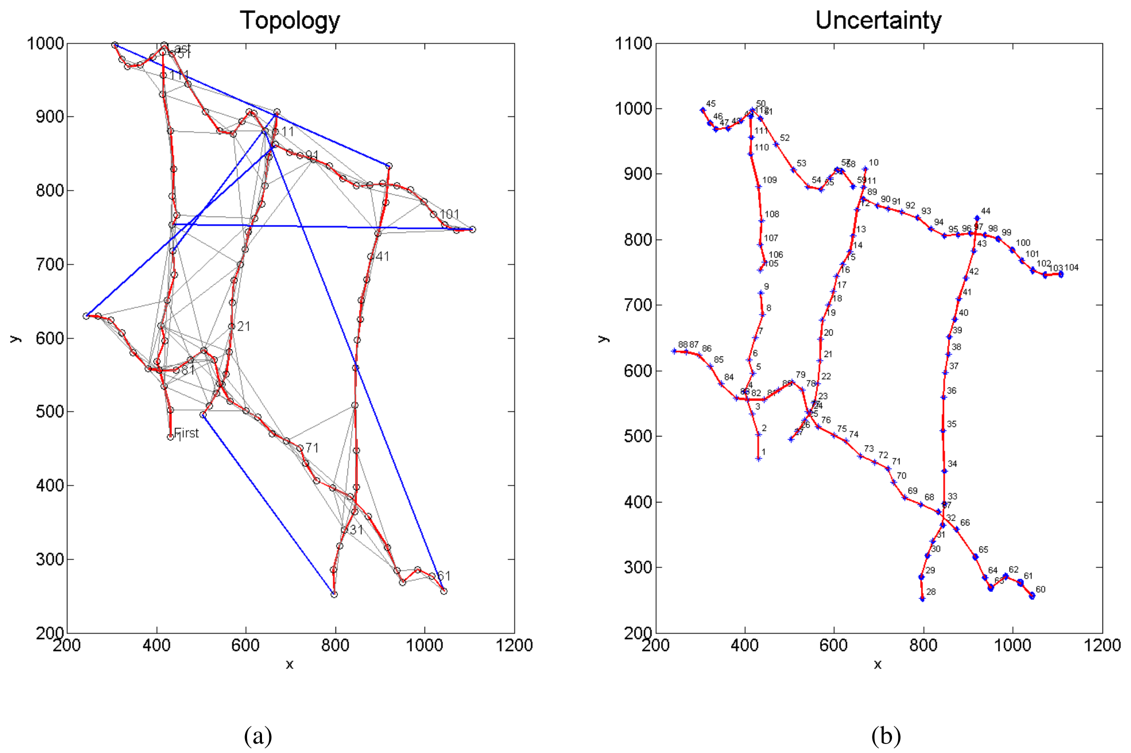


Figure 3. Axes are in pixels and approximately 200 pixels per meter. (a) Final trajectory obtained by the proposed method. Red lines are links between time consecutive overlapping images while the black ones are between the non-time consecutive. Blue lines show the non-overlapping time consecutive images. (b) Uncertainty on the final trajectory. Uncertainty ellipses are drawn with 95% confidence level.

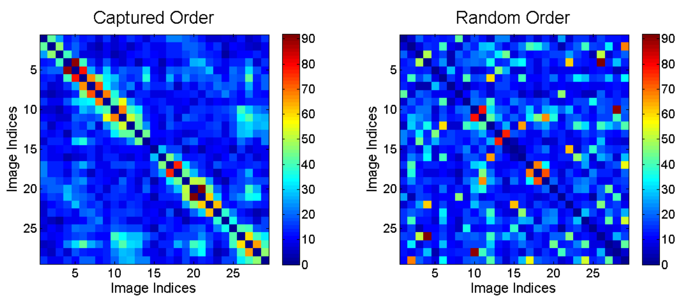


Figure 5. Initial Similarity Matrices of the second dataset. These matrices were computed using a maximum of 100 feature points.

Table II
SUMMARY OF RESULTS.

Strategy	Successful Obs.	Unsuccessful Obs.	% of attempts wrt all-against-all	Avg. Error in pixels	Std. Dev. in pixels
1. Captured Order of images	62	30	22.66	7.05	4.20
1. Random Order of images	62	31	22.90	7.03	4.20
2. Similarity Matrix	64	342	100.00	6.63	4.17
3. All Against All	64	342	100.00	6.63	4.17

V. ENVIRONMENTAL APPLICATION OF MOSAICS ON CORAL REEFS

Recent declines in coral reefs across the globe underscore the need for new scientific tools to better understand ecological patterns and rates of change. Given that multiple factors are typically responsible for changes within reef ecosystems, the monitoring of reef health must be carried out at multiple spatial and temporal scales, rather than relying on measuring only a few parameters. Comprehensive assessment of coral reef resources demands a hierarchical mapping strategy involving micro-scale to macroscale measurements. Image-based mosaics of the seabed enable observations on a mesoscale of 10's to 100's of m, with mm-scale resolution.

Underwater image-based mosaics address several limitations of traditional, diver-based, coral reef monitoring techniques. First, mosaics provide a landscape view of coral reefs that has previously been unobtainable [7]. Second, mosaics are efficient tools for tracking patterns of change over time [25]. Third, mosaics have high spatial accuracy at both the scale of an individual coral colony [7] and at the scale of the entire mosaic [10].

The potential use of mesoscale, or "landscape," mosaics has been investigated for several coral reef-related applications, including: documenting hurricane damage at both the colony and reef-framework scale [9], mapping mesophotic [26], [27] and deep-water [28] coral ecosystems, quantifying the area damaged by a ship that had run aground [10], and tracking individual colonies through time [9], [25]. Of these, the ship grounding and individual monitoring take particular advantage of the new scale of observation enabled by landscape mosaics.

Accurately documenting patterns of physical damage (and subsequent recovery patterns) to benthic habitats can be especially challenging when the spatial extent of injuries exceeds tens of square meters. Such injuries are often too large and

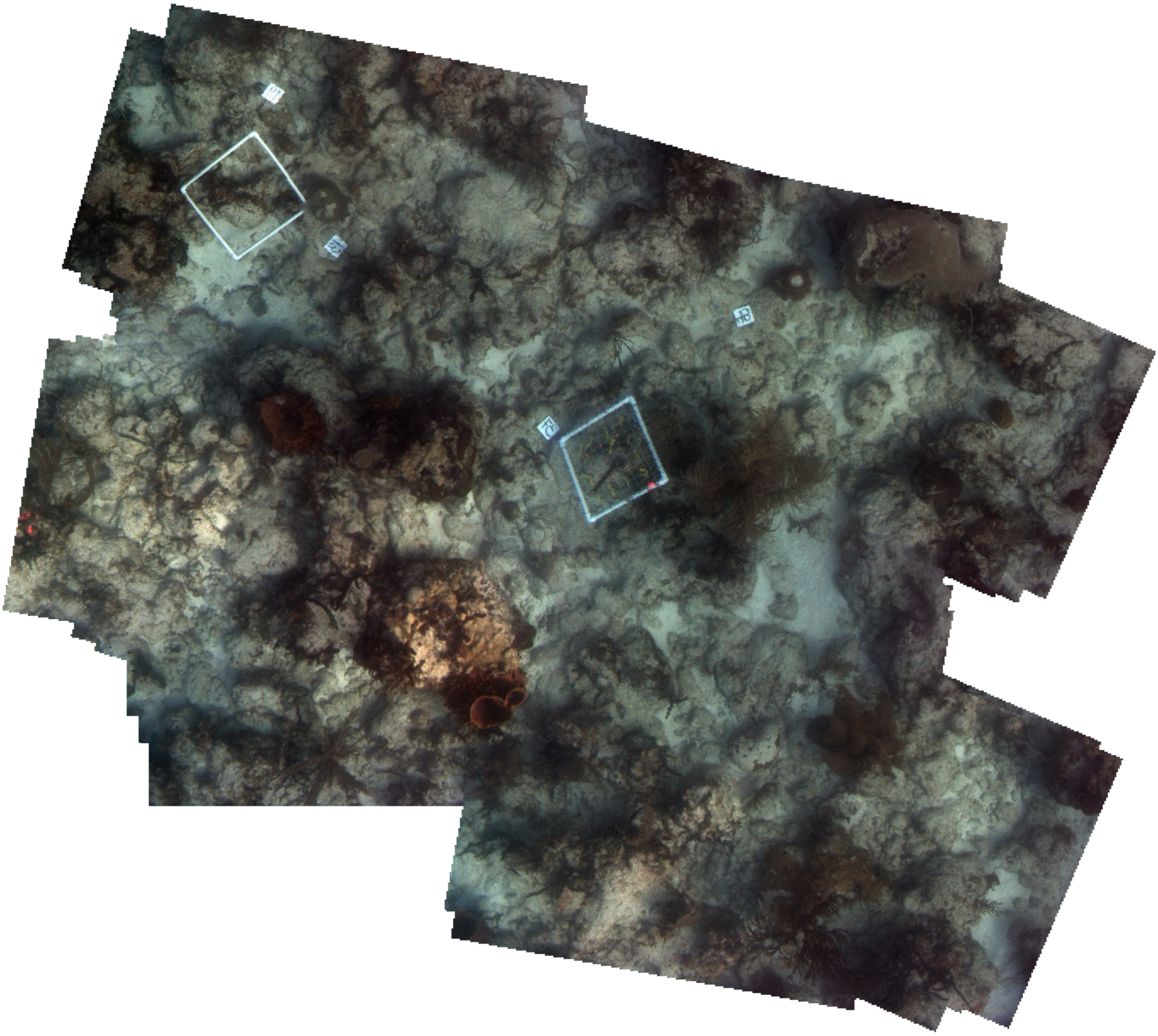
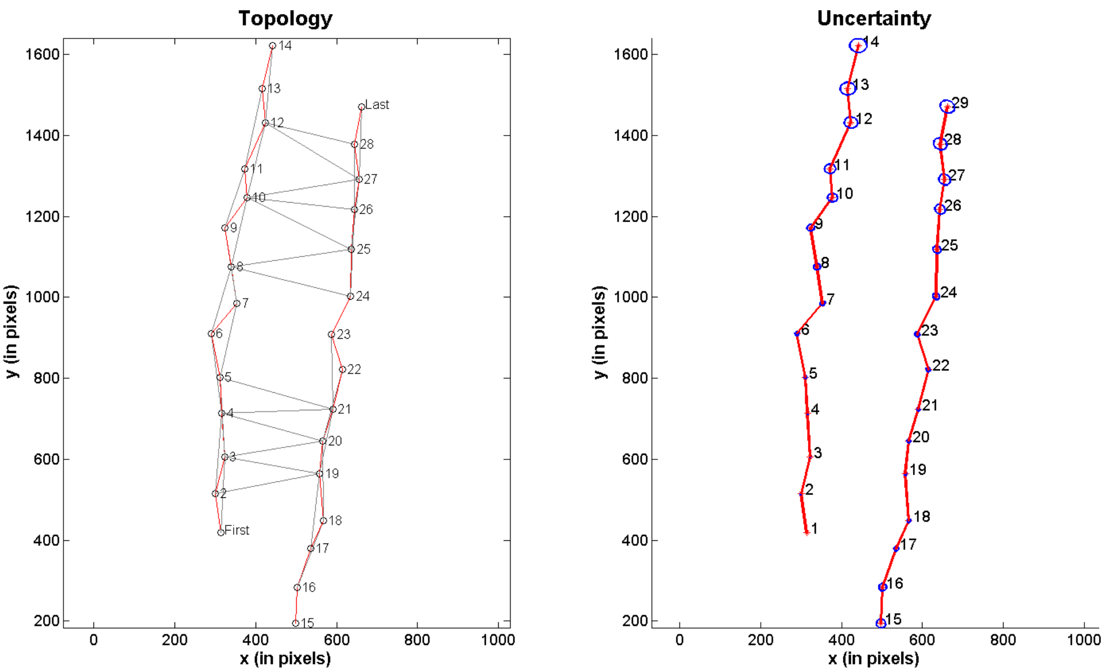


Figure 4. Resulting final mosaic image. After global alignment, the final mosaic was blended using graph cut algorithms [24]

difficult to measure in situ by divers and too small or costly to be quantified effectively using aerial and satellite remote sensing tools. Documenting the extent of damage caused by physical disturbance is one of the main challenges of post-damage surveys in coral reef habitats. In the case of vessel groundings, the effective and accurate assessment of the extent of the damage caused is a crucial first step in the Habitat Equivalency Analysis (HEA) required to determine the amount of compensatory restoration required [29], [30]. Grounding scars are commonly measured in situ by divers using flexible tapes following the "fishbone" method described by [31]. In addition, the boundaries of the damaged areas or the positions of objects of interest (e.g., injured corals) are surveyed using surface-deployed GPS units positioned over specific locations, and the extent of the damage is later calculated from the polygon delineated by the GPS locations.

Landscape mosaics are advantageous for assessment of damage and recovery operations because they permit simultaneous mapping of both the scale of the entire injury as well as the fine scale appropriate to assess individual colony damage. Lirman et al. [10] showed that an estimate of the damaged area derived from a landscape mosaic agreed within 2% with an estimate produced by a diver using differential GPS. Gleason et al. [32] mapped a large scar ($>3,000m^2$) in Puerto Rico by assembling multiple individual landscape mosaics acquired by divers. Despite the huge area, the Puerto Rico mosaic was rendered at 1 cm spatial resolution, allowing the assessment of individual coral colonies (Fig. 8).



(a) (b)

Figure 6. Axes are in pixels and approximately 200 pixels per meter. (a) Final trajectory obtained by the proposed method. Red lines are links between time consecutive overlapping images while the black ones are between the non-time consecutive. Blue lines show the non-overlapping time consecutive images. (b) Uncertainty on the final trajectory. Uncertainty ellipses are drawn with 95% confidence level.

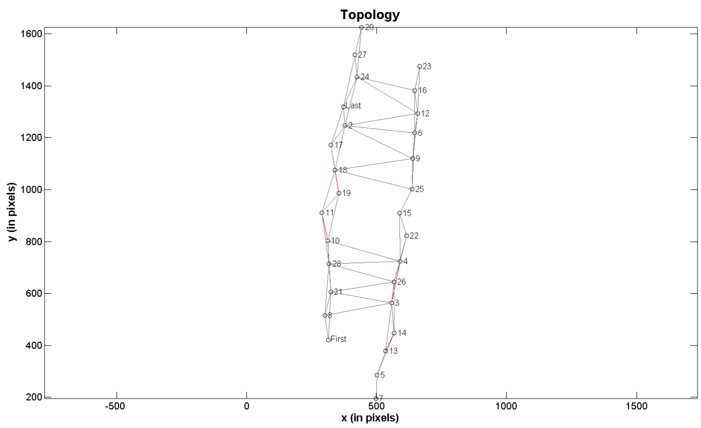


Figure 7. Axes are in pixels and approximately 200 pixels per meter. Final trajectory obtained by the proposed scheme using randomly ordered images in the dataset.

Monitoring individual coral colonies requires establishing a permanent site and periodically measuring the size and condition of each colony within that site. Currently, state-of-the-art assessment techniques rely on divers to measure colony sizes using tape measures or meter sticks, and to estimate colony condition visually. The drawbacks of this technique are, first, that the divers must tag each colony so the specific coral can be identified in the future, and, second, that the divers must have the relevant biological/ecological training to identify corals and assess their condition in the field. This diver-based tagging method is the state-of-the-art method used to establish new permanent plots today.

Landscape mosaics have two key advantages relative to the diver-based method that improve colony-based monitoring. First, tags are not necessary because repeat mosaics taken over the same area can be registered to one another. Removing the reliance on tags eliminates the need for physical contact with corals, thereby greatly reducing the potential for inadvertent damage and the amount of gear that must be permanently attached to the seafloor (e.g., nails, tags, markers). Tagging is

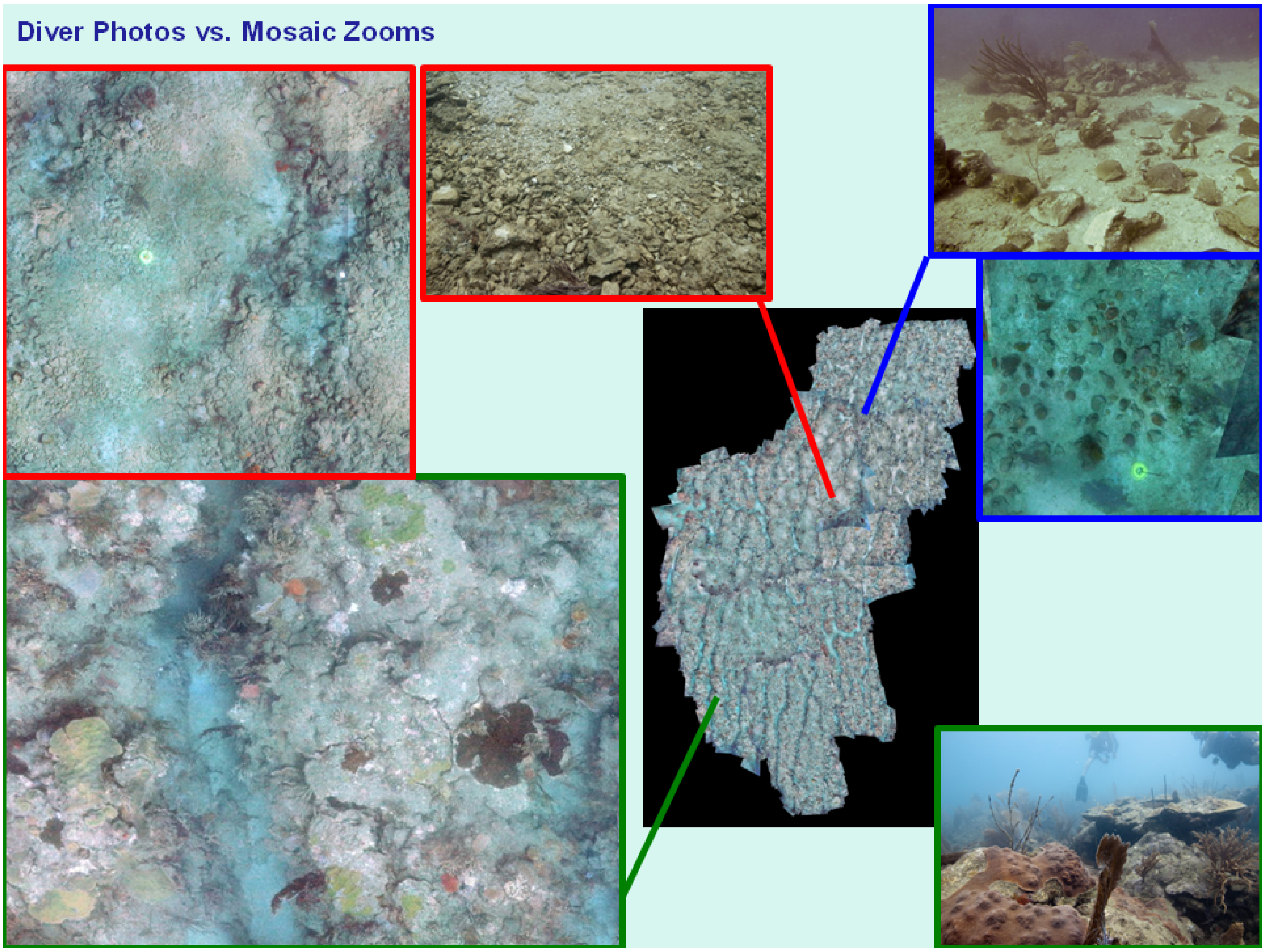


Figure 8. Landscape mosaic of a ship-grounding scar in Puerto Rico. The dimensions of the mosaiced area are 117×67 m, covering over 4,700 m^2 with 1×1 cm pixels. The entire mosaic is presented here at $<5\%$ of its full resolution, but the insets show portions of the mosaic at full resolution to give an idea of the level of detail in comparison with oblique images acquired by divers. Note the pulverized rock within the area of maximum damage (red insets), the condition of the unaffected area surrounding the scar (green insets), and the coral fragments ready to be adhered to the substrate as part of the remediation process (blue insets).

labor intensive both during the initial establishment of the plot and during re-location of colonies through time. Furthermore, tags can get lost due to burial, failure of the attachment mechanism, biofouling, or simply diver error, and lost tags represent lost data as colonies can no-longer be identified. Second, divers who collect the data do not necessarily have to have extensive training in coral reef biology.

VI. Conclusions

We have presented a generic scheme for creating large-area mosaics with application to environmental mapping over areas where only image information is available. Our scheme aims at obtaining the topology with minimum number of image matching attempts as well as obtaining the best possible trajectory estimation. The proposed approach is able to deal with the cases where time consecutive images do not have overlapping areas.

Acknowledgments.

This work was partially funded through the Spanish Ministry of Education and Science (MCINN) under grant CTM2010-15216, EU Project FP7-ICT-2009-248497 and US SERDP program project CS-1333. Nuno Gracias was supported by MCINN under the Ramon y Cajal program.

References

[1] H. Sawhney, S. Hsu, and R. Kumar, "Robust video mosaicing through topology inference and local to global alignment," in *European Conference on Computer Vision*, vol. II, Freiburg, Germany, June 1998, pp. 103–119.

[2] R. Kumar, H. Sawhney, S. Samarasekera, S. Hsu, H. Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hirvonen, M. Hansen, and P. Burt, "Aerial video surveillance and exploitation," *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1518–1539, Oct 2001.

[3] Z. Zhu, E. Riseman, A. Hanson, and H. Schultz, "An efficient method for geo-referenced video mosaicing for environmental monitoring," *Machine Vision and Applications*, vol. 16, no. 4, pp. 203–216, 2005.

[4] J. Escartin, R. Garcia, O. Delaunoy, J. Ferrer, N. Gracias, A. Elibol, X. Cufi, L. Neumann, D. J. Fornari, S. E. Humpris, and J. Renard, "Globally aligned photomosaic of the Lucky Strike hydrothermal vent field (Mid-Atlantic Ridge, 3718.5'N): Release of georeferenced data, mosaic construction, and viewing software," *Geochemistry Geophysics Geosystems*, vol. 9, no. 12, p. Q12009, 2008.

[5] R. M. Eustice, H. Singh, J. J. Leonard, and M. R. Walter, "Visually mapping the RMS titanic: Conservative covariance estimates for SLAM information filters," *International Journal of Robotics Research*, vol. 25, no. 12, pp. 1223–1242, 2006.

[6] K. Jerosch, A. Ldtke, M. Schlter, and G. Ioannidis, "Automatic content-based analysis of georeferenced image data: Detection of beggiatoa mats in seafloor video mosaics from the hakon mosby mud volcano," *Computers & Geosciences*, vol. 33, no. 2, pp. 202 – 218, 2007.

[7] D. Lirman, N. Gracias, B. Gintert, A. Gleason, R. P. Reid, S. Negahdaripour, and P. Kramer, "Development and application of a video–mosaic survey technology to document the status of coral reef communities," *Environmental Monitoring and Assessment*, vol. 159, pp. 59–73, 2007.

[8] O. Pizarro and H. Singh, "Toward large-area mosaicing for underwater scientific applications," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 4, pp. 651–672, October 2003.

[9] A. Gleason, D. Lirman, D. Williams, N. Gracias, B. Gintert, H. Madjidi, R. Reid, G. Boynton, S. Negahdaripour, M. Miller, and P. Kramer, "Documenting hurricane impacts on coral reefs using two–dimensional video–mosaic technology," *Marine Ecology*, vol. 28, no. 2, pp. 254–258, June 2007.

[10] D. Lirman, N. Gracias, B. Gintert, A. Gleason, G. Deangelo, M. Dick, E. Martinez, and R. P. Reid, "Damage and recovery assessment of vessel grounding injuries on coral reef habitats using georeferenced landscape video mosaics," *Limnology and Oceanography: Methods*, vol. 8, pp. 88–97, 2010.

[11] O. Delaunoy, N. Gracias, and R. Garcia, "Towards detecting changes in underwater image sequences," in *OCEANS 2008-MTS/IEEE Techno-Ocean*, Kobe, Japan, 2008, pp. 1–8.

[12] N. Gracias, S. Zwaan, A. Bernardino, and J. Santos-Victor, "Mosaic based navigation for autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 4, pp. 609–624, Oct. 2003.

[13] O. Pizarro, R. Eustice, and H. Singh, "Relative pose estimation for instrumented, calibrated imaging platforms," in *Digital Image Computing Techniques and Applications*, Sydney, Australia, December 2003, pp. 601–612.

[14] H. Madjidi and S. Negahdaripour, "Global alignment of sensor positions with noisy motion measurements," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1092–1104, 2005.

[15] Y. Rzhanov, L. Mayer, S. Beaulieu, T. Shank, S. Soule, and D. Fornari, "Deep-sea geo-referenced video mosaics," in *MTS/IEEE OCEANS Conference*, Boston, USA, sep. 2006, pp. 2319–2324.

[16] J. Ferrer, A. Elibol, O. Delaunoy, N. Gracias, and R. Garcia, "Large-area photo-mosaics using global alignment and navigation data," in *MTS/IEEE OCEANS Conference*, Vancouver, Canada, November 2007, pp. 1–9.

[17] H. Bulow and A. Birk, "Fast and robust photomapping with an unmanned aerial vehicle UAV," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 10-15 2009, pp. 3368 –3373.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Harlow, UK: Cambridge University Press, 2004.

[19] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[20] S. Skiena, *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Reading, MA: Addison-Wesley, 1990.

[21] R. M. Haralick, "Propagating covariance in computer vision," in *9. Theoretical Foundations of Computer Vision*, March 1998, pp. 95–114.

[22] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – A modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. LNCS. Springer Verlag, 2000, pp. 298–375.

[23] D. Capel, *Image Mosaicing and Super-resolution*. London: Springer-Verlag, 2004.

[24] N. Gracias, M. Mahoor, S. Negahdaripour, and A. Gleason, "Fast image blending using watersheds and graph cuts," *Image and Vision Computing*, vol. 27, no. 5, pp. 597–607, 2009.

[25] B. Gintert, N. Gracias, D. Lirman, A. Gleason, P. Kramer, and R. Reid, "Second-generation landscape mosaics of coral reefs," in *Proceedings of the 11th International Coral Reef Symposium*, 2008, pp. 7–11.

[26] R. Armstrong, "Deep zooxanthellate coral reefs of the puerto rico: Us virgin islands insular platform," *Coral Reefs*, vol. 26, no. 4, pp. 945–945, 2007.

[27] A. Gleason, N. Gracias, D. Lirman, B. Gintert, T. Smith, M. Dick, and R. Reid, "Landscape video mosaic from a mesophotic coral reef," *Coral Reefs*, vol. 29, no. 2, pp. 253–253, 2010.

[28] M. Ludvigsen, B. Sortland, G. Johnsen, and H. Singh, "Applications of geo-referenced underwater photo mosaics in marine biology and archaeology," *Oceanography*, vol. 20, no. 4, pp. 140–149, 2007.

[29] J. Milon and R. Dodge, "Applying habitat equivalency analysis for coral reef damage assessment and restoration," *Bulletin of marine science*, vol. 69, no. 2, pp. 975–988, 2001.

[30] S. Shutler, S. Gittings, T. Penn, and J. Schittone, *Compensatory restoration: how much is enough? Legal, economic, and ecological considerations*. Taylor & Francis, 2006, pp. 77–93.

[31] J. Hudson and W. Goodwin, "Assessment of vessel grounding injury to coral reef and seagrass habitats in the florida keys national marine sanctuary, florida: protocol and methods," *Bulletin of marine science*, vol. 69, no. 2, pp. 509–516, 2001.

[32] A. C. R. Gleason, D. Lirman, N. Gracias, T. Moore, S. Griffin, M. Gonzalez, B. Gintert, and R. P. Reid, "Damage assessment of vessel grounding injuries on coral reef habitats using underwater landscape mosaics," in *Proceedings of the 63rd Gulf and Caribbean Fisheries Institute*, November 2010.

High-Resolution 3D Reconstruction of the Seafloor for Environmental Monitoring and Modelling

Rafael Garcia, Ricard Campos
Computer Vision and Robotics Group
Underwater Vision Lab
University of Girona, Spain
rafael.garcia@udg.edu, rcampos@eia.udg.edu

Javier Escartin
Équipe de Géosciences Marines
CNRS UMR 7154
IPGP, Univ. Paris Diderot, France
escartin@ipgp.fr

***Abstract*—Underwater robots are appearing in the last few years as necessary exploration tools which provide scientists with valuable data and samples, allowing the implementation of experiments to monitor natural processes occurring on the seafloor. In this paper a video processing pipeline is proposed for high-resolution 3D mapping of an underwater area of interest at 1,700m depth along the Mid-Atlantic Ridge axis. First, the robot trajectory is estimated using Structure from Motion (SFM). Then, based on the estimated camera trajectory, a dense 3D point cloud is obtained. Next, this cloud of points is used to create a mesh of triangles defining the shape of the mapped structure. Finally, the estimated camera trajectory and surface mesh are used to automatically obtain the best texture information for each of the triangles forming the model. The texture mapping of the surface is key to enable the monitoring of environmental changes through repeated surveys in time. This surface model is suitable not only for visualization purposes, but also for scientific interpretation. Furthermore, the proposed method is generic, allowing for the mapping of arbitrary surfaces with no a priori constraints about the needed structure, as was the case for 2.5-D acoustic bathymetric maps commonly used in underwater mapping. This allows the imaging of complex structures with concavities, whose characterization is precluded for traditional acoustic imaging systems.**

I. Introduction

The complexity of natural environments presents numerous challenges for surveying with robots. However, with more reliable and intelligent systems as a result of technology advances, robotic platforms are now necessary exploration tools to gain an understanding of the processes that operate and shape the Earth and other planets. In the mostly unexplored deep-sea, underwater robots are providing scientists with valuable data and samples, and allow the implementation of experiments to monitor natural processes occurring on the seafloor. As a result, they have been adopted as a necessary tool in marine and environmental research. A key aspect of these studies is the understanding of active geological, biological, and oceanographic processes, and the links that exist between them. To help unravel these interdependencies, we propose imagery-based tools that allow us to use robots so as to characterize the seafloor at fine scales, allowing scientists to detect changes over time.

Underwater 3D surveying has gained considerable relevance in the later years using acoustic multi-beam systems, often combined with side-scan sonar imagery. However, the inherent limitations of these tools, including the insonification geometry and their vocation to characterize relatively large areas of the seafloor, strongly restrict the resolution of the recovered maps. In particular, these systems are configured –and the data processed– so as to obtain a 2.5-D terrain model having the form $h(x, y) = z$. Thus, complex three-dimensional geometries are not characterized, or are poorly represented. These 3D terrain models also have a limited spatial resolution (tens of cm for near-bottom robot surveys). Hence traditional sonar surveys provide data at a resolution that is at least order of magnitude lower than that of the optical imagery which can be acquired by Remotely-Operated Vehicles (ROVs) or Autonomous Underwater Vehicles (AUVs) near the seafloor.

We present in this paper a processing pipeline to model the terrain of an underwater area of interest using solely a video camera as a mapping sensor. This method bridges the gap that exists between acoustical and optical systems in resolution and capability to obtain accurate 3D models of the seafloor. Furthermore, the implementation of this technique can be widespread and become routine, as cameras are standard sensors equipping underwater robots, and video imagery can be continuously acquired with no additional cost. Furthermore, the proposed method is generic, allowing for the mapping of arbitrary surfaces with no a priori constraints about the needed structure, as was the case for 2.5-D. This allows the imaging of complex structures with concavities, whose characterization is precluded for traditional acoustic imaging systems.

On the other hand, as we are using optical images, the result is a 3D reconstruction of the surveyed site in the form of a textured surface model, thus adding the richness of imagery that cannot be obtained using multibeam systems. The proposed pipeline works as follows. The camera trajectory is first estimated using a Structure from Motion (SFM) method. Then, a dense 3D reconstruction is computed, based on the estimated camera positions from the previous step, thus obtaining a highly sampled model through a 3D point cloud. This point cloud is then used as input for a surface reconstruction method to obtain a mesh of triangles defining the shape of the object. This model is suitable for visualization purposes, and for scientific interpretation. Finally, the known camera positions and surface mesh are used to automatically obtain the best texture information for each of the triangles

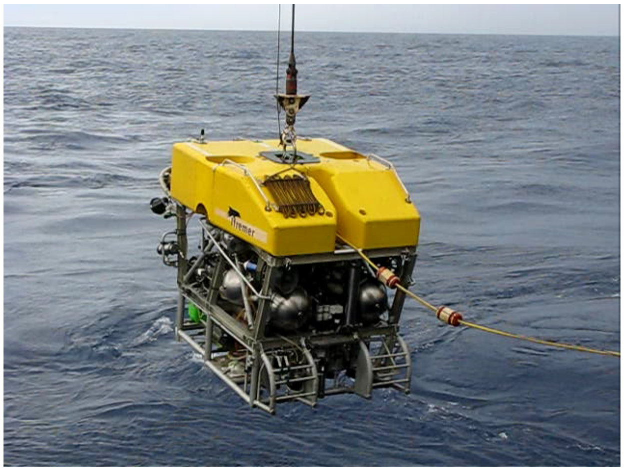


Fig. 1. VICTOR6000 ROV being deployed at the Mid-Atlantic Ridge during one of the deep-sea surveys. Copyright: CNRS/IFREMER.

forming the model. The texture mapping of the surface is key to enable the monitoring of environmental changes through repeated surveys in time.

In this paper we first provide a brief overview of the mosaicing system structure and functionality. Next, the proposed framework is validated and demonstrated on a video survey of a deep-sea hydrothermal vent in section III. Then, some experiments are reported, illustrating the obtained results. Finally, the conclusions are presented, along with further work.

II. Seafloor Modelling

The datasets presented below were collected at 1,700m depth in the Mid-Atlantic Ridge, using the IFREMER robot VICTOR6000 [1] equipped with a forward-looking color camera (see Fig. 1). Beyond its specificity, the methodology described in this section can be applied to any sequence of natural images. The only pre-requisite is the calibration of the intrinsic camera parameters (here performed underwater using a calibration pattern) [2]. The proposed video processing pipeline is based on a series of successive steps detailed in the following subsections, namely: trajectory estimation; dense reconstruction; surface reconstruction; and texture mapping. Since this pipeline relies only in the information of a single camera, the reconstructed model is affected by an unknown scale factor [3]. A discussion on scale recovery of the model is therefore presented below.

A. Estimation of the Camera trajectory

First, the trajectory of the camera (pose at each frame) is estimated by a Structure From Motion (SFM) procedure similar to the one described in [4]. This algorithm performs an incremental reconstruction based on directly registering the cameras to the already reconstructed points, and consists of two main steps: first, an initial model is obtained using a standard motion estimation technique, and then this initial model is used to directly register new cameras.

The model is initialized by fixing an image (normally the first one in the sequence) as the reference frame. Then, another image is selected so as to maximize the baseline. Normally, SFM algorithms use the fundamental matrix for the camera motion estimation, and this is the correct approach when the scene has high 3D relief or the sequence has noticeable parallax. However, when the scene is nearly planar or there is low parallax in the sequence the fundamental matrix can be ill-conditioned, as pointed out in [3]. In this latter case, a more robust approach consists in modelling this motion using a homography. This SFM algorithm uses a dual approach, and selects one model or the other depending on the observed scene.

The fundamental matrix F is computed from correspondences between images using a least-squares procedure inside a RANSAC method, and then decomposed into a rotation and a translation using SVD decomposition. Four solutions are provided when decomposing F (2 rotations and 2 translations), but the correct one is that following the cheirality constraints, *i.e.*, the reconstructed points must lay in front of the camera [5]. In the case of the homography H, it is also obtained from correspondences using a RANSAC method, and decomposed through SVD into a rotation and translation matrix. At this point, two possible solutions for the motion have been computed, and the best one has to be chosen. The transformation that minimizes the reprojection error is the one selected.

Once an initial model has been built, poses of new cameras can be computed by finding correspondences between imaged 2D points and reconstructed 3D points. To perform this matching between 3D and 2D features, each 3D point preserves a description vector, which is the mean of the different descriptors of each of the 2D points that have generated it (*e.g.*, using SIFT [6] this descriptor vector has 128 elements). By having the correspondences between 2D and 3D points, the procedure to compute the pose follows also a dual approach to use a projection matrix or a homography based on the planarity of the scene. In this case, the RANSAC algorithm tries to find a consistent model that minimizes reprojection error by selecting at each iteration a random subset of the matched 3D points, fitting a plane to them, and computing a homography or a projection matrix by looking at the distances between the computed plane and the rest of 3D points. If all the points lie close (according to a threshold) to the plane, the homography model is used, otherwise a projection matrix is computed using the DLT algorithm [3]. Once the RANSAC method has estimated the corresponding homography/projection matrix, it is decomposed into a rotation and a translation, and this absolute pose is further refined through *Bundle Adjustment* [7] to further minimize the reprojection error.

B. Dense reconstruction

The processing step described in the previous section generates a set of sparse 3D points, providing an up-to-scale 3D representation of the mapped object/structure. Since this is a sparse point cloud, it does not allow for a detailed representation of the appearance of the object. For this reason, and once we have solved the full geometry of the system (spatial relations from world to camera pose and from camera pose to image plane), we can focus on a more relaxed solution of the correspondence problem. In this step we do not focus on finding matches across the most distinctive features in the images, like in the SFM procedure. Instead, the goal is to obtain as many correspondences as possible by relaxing this distinctiveness measure on the features.

For this purpose, we use the method described in [8], which aims to find a dense reconstruction of the object. In this approach small planar patches –instead of single points– are reconstructed. A planar patch is represented by a small planar square, a center, and a normal vector pointed towards the cameras that observe it. Furthermore, each patch keeps track of two sets of images, one containing those where it is visible, and the other containing the ones where it *should* be visible. A patch *should* be visible in an image if it is visible according to geometric constraints, but it is not photometrically consistent across images. Starting with a set of sparse reconstructed patches, the method tries to expand the number of patches by looking at those patches nearby the ones that have already been reconstructed. At the same time, it filters out outliers by using photoconsistency and visibility constraints.

The algorithm consists of three main steps: matching, expansion and filtering. The matching step tries to find a set of initial sparse matches using Harris and Difference of Gaussians feature detectors, and matches them across the images using epipolar constraints to restrict the correspondences search. The photoconsistency metric used in matching is the Normalized Cross Correlation score. Once the set of correspondences has been set, the computation of the patch is performed by fixing its position to be along a line of sight of a reference image, and then finding the position and orientation parameters of the patch by minimizing the photometric consistency of its projection across images. Once the matching step has been performed, the expansion and filtering steps are iterated. The expansion step aims to extend the matches near patches that have been already reconstructed, while the filtering step accounts for visibility consistency of the recovered patches, *i.e.*, it removes outliers taking into account self-occlusions among patches in 3D.

C. Surface reconstruction

Since, a priori, the 3D points in natural environments do not follow a regular structure, the points are said to be "unorganized". Therefore, so as to obtain a 3D reconstruction, our processing requires a method capable of dealing with an unorganized set of points. The most widely used method for surface reconstruction from a set of unorganized points is the Poisson method of Kazhdan *et al.* [9], which by nature is expected to yield the best and most robust results for our problem.

The method starts by discretizing in a regular voxel grid the working space where the points are located. The surface will be then represented in this space implicitly, being $f(v_i) = 0$ if the voxel $v_i = (x_i, y_i, z_i)$ is located inside the object, and $f(v_i) = 1$ if it is located outside. From this indicator function, the triangle mesh representation is obtained by means of an isosurface extractor method such as the *marching cubes* algorithm by Lorensen and Cline [10].

The points obtained from section II-B, along with their normal vectors, are used as samples of the gradient of the indicator function described above. The problem is then to find the inverse of this gradient, *i.e.*, to find the indicator function whose gradient best approximates the vector field $\vec{V}$ defined by the samples: $min \left\| \nabla X - \vec{V} \right\|$.

To solve this problem, we perform Gaussian splatting and propagation of the points and their normals into the voxel grid, obtaining a discretized field $\vec{V}$ defined throughout our space. Then, applying the divergence operator, this problem is transformed into a Poisson problem to find the scalar function X whose Laplacian (divergence of gradients) equals the divergence of the vector field $\vec{V}$: $\Delta X \equiv \nabla\nabla X = \Delta\vec{V}$.

However, this approach searches for a closed surface, without boundaries. Despite having a scene with boundaries, the surface recovered by this method will remain closed. To solve this problem, we modify [9] by adopting a simple filtering procedure based on eliminating triangles having an edge length greater than a threshold. This post-processing is motivated by the fact that *false* closing parts are formed by large triangles. Furthermore, this reconstruction results in a smooth surface, which is desirable so as to compensate for the possible measurement errors of the input 3D points.

D. Texture mapping

Using optical images instead of multi-beam sonar provides the means not only to obtain higher resolution maps of interest sites (see Fig. 6), but also to add texture information to the obtained mesh surface. Thus, as images are the source of texture information, we take profit of the derived geometry of the cameras to automatically obtain a texture map for the constructed model. Given the camera positions and the mesh of triangles representing the object, we back-project the three vertices of each triangle into one of the compatible views to get its texture.

The surface mesh obtained in section II-C is not formed by the input points obtained in section II-B. This means that the notion of visibility for each point (*i.e.*, which images see this point) is lost and, thus, we need to process the visibility of the triangles.

For each triangle, its compatible views are selected by following a set of filtering steps. First the projections of the triangle are checked to verify whether they fall within the image plane of the view. Then, normals of the triangle are checked to be compatible (using the dot product) with the *line of sight*, which is the segment joining the center of the triangle with the camera center. Views passing these tests are checked against the most computationally expensive test, which determines whether the triangle is occluded by other parts of the surface. For a view to pass this test, lines of sight from the view center to each of the vertices forming the triangle must not intersect any other triangle in the surface. Since using a brute force search approach is computationally expensive, we describe the surface using an Axis Aligned Bounding Boxes (AABB) tree structure. This data structure consists of a tree hierarchy of bounding boxes whose faces are aligned with the reference axis. The leaf nodes of the tree correspond to the set of triangles. Intersection tests are then reduced to run down the tree by testing the intersection of the line with each node, *i.e.*, with an AABB. AABB intersection checks are simpler, and reduce computational cost, as the real intersection point with the triangle is computed only when we are in the leaf node of the tree; in our case, a unique intersection test is sufficient.

From the previous filtering steps, we obtain a set of valid views or images from which this triangle can be seen. Given these views, the triangle is projected onto the selected valid images. Then, we finally choose the view that covers the largest area inside that triangle. It is noteworthy that this simple approach implicitly takes the texture from a view that is both close and quite orthogonal to the 3D triangle. Since light is strongly attenuated in water, we favor the selection of texture from images that are closer to the reconstructed object or scene, so as to preserve color information.

E. Scale recovery

It is well-known that reconstructing a scene with a monocular camera is subject to arbitrary scaling and that this scale factor cannot be recovered only from the images themselves. Therefore, additional information needs to be provided to estimate the unknown overall scale factor.

In photogrammetry, scale is typically obtained by exploiting the information about true lengths in the scene, *i.e.*, using a known object for which we can fix the distance between two points belonging to the object. By taking an image where the known points are located, we can manually select the two image positions corresponding to those points. Then, the intersection between the rays formed by the camera position and those image projections can be tested for intersection with the reconstructed model (we can use the previously built AABB structure to speed up the process). Once the distance between the two points in the 3D model has been measured, the scale factor can be extracted. For monitoring purposes, the Tour Eiffel hydrothermal chimney is equipped with cylindrical temperature probes installed at fluid outflows, that have a diameter of 150mm (yellow object in Fig. 4b). However, we discarded this option since the probes are small relative to the size of the chimney (150mm vs. $>$15m), and thus a large uncertainty in the obtained scale is expected.

On the other hand, other sources of information can be exploited to extract this scale by fixing any other single length associated with the motion of the camera. In our case, the acoustic navigation system suite of the robot allows us to associate each image with its corresponding depth below the seafloor. By using this information, it is also possible to retrieve the scale factor. Obviously, the reference frames of the world and of the cameras are not aligned, thus there is an

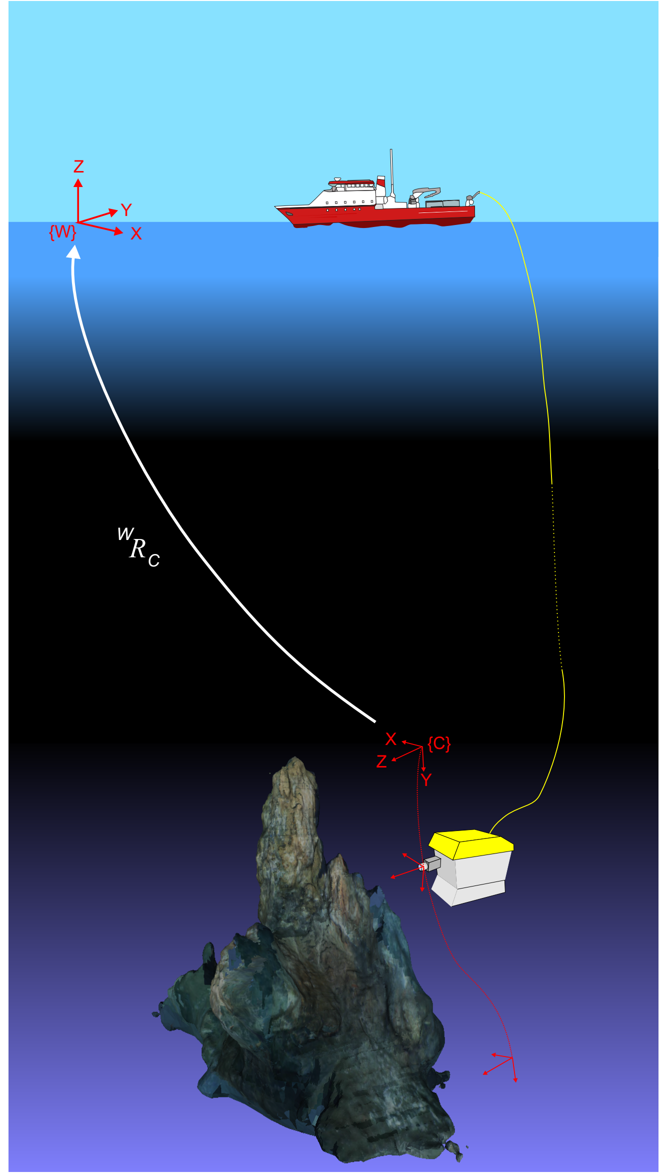
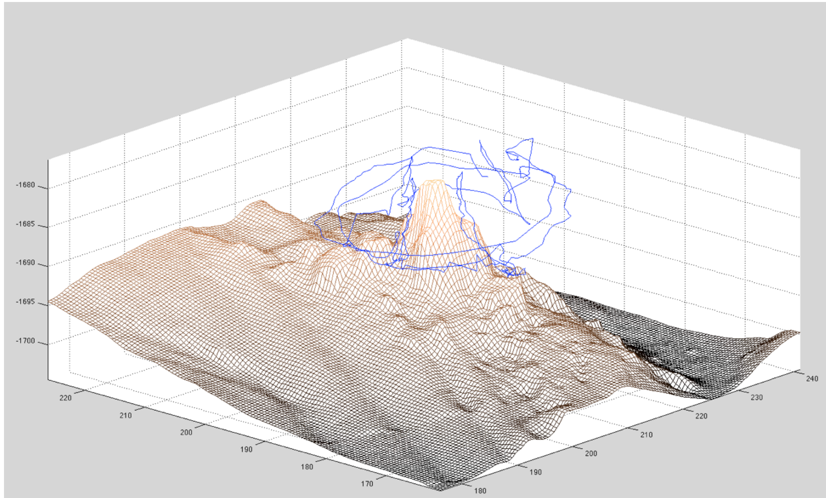


Fig. 2. Relationship between the world coordinate system $\{W\}$, corresponding to the UTM coordinates, and map coordinate system $\{C\}$, corresponding to the reference frame of the first image of the sequence. ${}^W\mathbf{R}_C$ is the rotation between the first image of the sequence and the world.

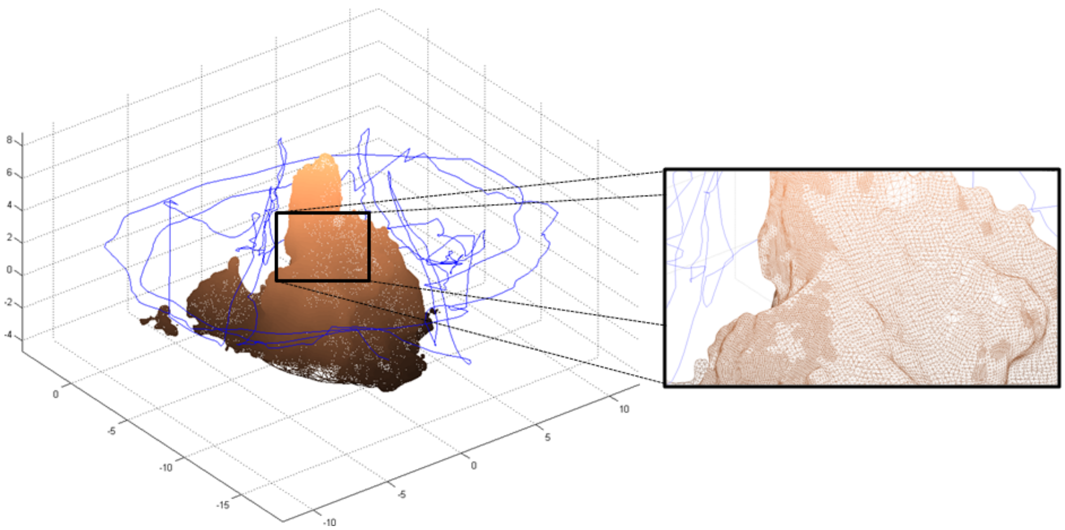
unknown rotation, translation and scaling between them. If we neglect the translation (which will be obtained later from the robot's acoustic navigation system), the transformation that we need is the one that aligns the z axis of the world reference frame $\{W\}$, with the y axis of the coordinate system belonging to the first image of the sequence $\{C\}$:

$$^{W}\begin{pmatrix} x \\ -z \\ y \end{pmatrix} = s \cdot {}^{W}\mathbf{R}_{C} \cdot {}^{C}\begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

where ${}^W\mathbf{R}_C$ is a 3×3 rotation matrix between the coordinate system of the first image $\{C\}$ and the world coordinate system



(a) Multibeam bathymetric map (from Ondréas et al. [11]), and robot trajectory from acoustic navigation (blue line).



(b) 3D high-resolution map obtained by the proposed approach, with scale in meters, and recovered camera trajectory.

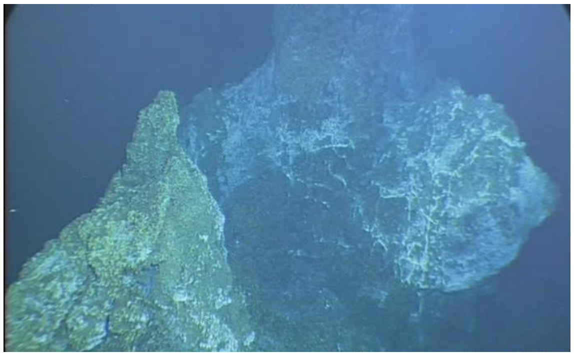
Fig. 3. Comparison of multibeam bathymetry vs. terrain reconstruction of the Tour Eiffel chimney from the proposed approach. Note the higher resolution obtained with the proposed method.

$\{W\}$; s corresponds to the unknown scale factor. It should be noted that the world's z coordinate is negative, since we are below the sea level taking depth measurements. Leading superscript in the equation stands for the frame in which every vector is expressed.

We then just need to find the rotation and scaling (we can neglect translation) that minimize the differences in depth associated to any pair of camera positions of the sequence. Those camera positions that have the largest depth distances provide better constraints to fix the scale, image pairs with small translational motion are avoided. Thus, we finally estimate the scale factor and the 3 rotation angles through a Levenberg-Marquardt optimization [12] in which we consider several pairs of images acquired at different depths along the z axis of the world coordinate system (see Fig. 2).

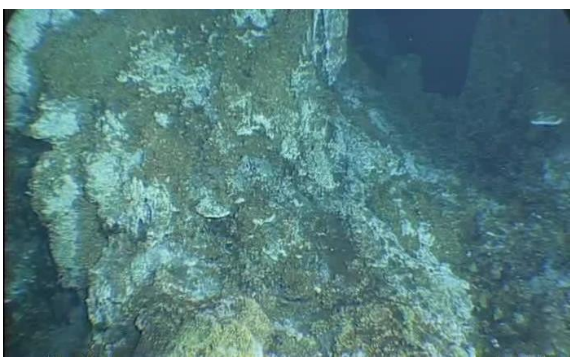
III. 3D MAPPING FOR ENVIRONMENTAL MONITORING

The video survey of the Tour Eiffel hydrothermal vent (Lucky Strike hydrothermal field along the Mid-Atlantic Ridge [13]) was carried out with VICTOR6000 [1] operated from the oceanographic vessel *PourQuoi Pas?* (IFREMER, France) during the Bathyluck09 oceanographic cruise. Hydrothermal activity in the oceanic crust facilitates heat, chemical and mass exchanges between the deep earth and the overlying fluid envelopes (ocean), sustaining rich ecosystems. This hydrothermal activity displays focused fluid discharge zones, massive and complex chimneys develop through mineral precipitation, and whose surfaces sustain macrobial and microbial ecosystems. The Tour Eiffel vent is more than 15 m high, and is the target of geological, chemical and biological studies, and of its monitoring for the last decade. During the survey, the robot was teleoperated from the research vessel,



(a)

(b)



(c)

Fig. 4. Example input data, from a set of 928 images. It should be noted that color information is affected by the differences in distance between the scene and the camera. Temperature probes (yellow) are visible in (b).

and position estimates were obtained by means of a ultra-short baseline system (USBL) and the onboard navigation system, which includes a Doppler Velocity Log (DVL) and a fibre-optic gyrocompass. The ROV circled twice the vent, and did four vertical up and down transects at closer range (see Fig. 3).

IV. Results

We present below the results obtained for the case of study of section III. Fig. 4 shows three sample input images. The obtained reconstruction of the Tour Eiffel hydrothermal vent is illustrated in Figs. 3 and 6. Images acquired closer to the imaged object contain richer color information and details, while images taken from afar are mostly bluish (see Fig. 4). Intermediate results of our processing pipeline are shown on Fig. 5. In Fig. 5(a) the set of points resulting from the SFM method is very sparse and contains outliers. This step is only needed to get the camera positions, from which we apply the dense reconstruction method, yielding results with far better quality and quantity of the 3D points, as in Fig. 5(b). The surface obtained from the Poisson surface reconstruction method is shown in 5(c). Finally, three views of the textured surface are presented in Fig. 6. It should be noted that the recovered geometry is quite complex, and cannot be described using a 2.5-D height map, as obtained from acoustic mapping systems. These 3D textured surfaces allow an accurate location of samples (geological, biological, fluids) and instrumentation (*e.g.*, temperature sensors), while providing a context for accurate interpretation. Imagery also allows the mapping of biological communities, defining their complexity and eventually their temporal evolution. Finally, these data also allow careful planning and installation of instrumentation and observations (*e.g.*, instrumented sites for monitoring). Repeated surveys would also allow scientists to both characterize the geometry of such structures (*e.g.*, volume), as well as its temporal evolution (*e.g.*, growth), providing information on the dynamics of the processes at their origin (submarine hydrothermal activity in this case).

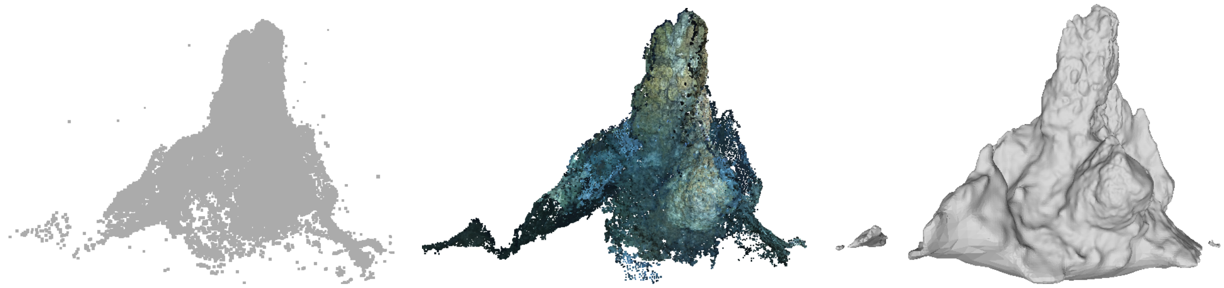
V. Conclusions

This paper described a generic framework for the exploration and mapping of deep-sea complex underwater environments, obtaining highly detailed 3-D models of underwater interest areas. It constitutes the initial step in the development of tools intended to be used by marine scientists in benthic mapping applications. The processing pipeline does not assume any kind of a priori geometry on the mapped structure, allowing reconstruction of structures with complex shapes. This approach relies on the estimation of the robot trajectory using a Structure from Motion algorithm, which is then used to compute a dense cloud of 3D points. Next, a meshing approach allows the procurement of a mesh of triangles from the point cloud. This mesh is later textured-mapped, and the scale of the whole model is finally fixed based on the depth measurements of the underwater robot.

The proposed framework has been tested to reconstruct a deep-sea hydrothermal vent, the Tour Eiffel, at the Mid-Atlantic Ridge. This area in general –and the Tour Eiffel vent in particular– is of special interest, as it has been the object of geological, hydrothermal, chemical and biological studies for more than 15 years. Within this context, our approach is well-suited for conducting temporal studies through repeated surveys of the same area, enabling the detection of changes in these extreme environments.

Acknowledgements

This work has been partially funded through the Spanish Ministry of Science and Innovation under grant CTM2010-15216 (MuMap) and through EU project FP7-ICT-2009-248497 (Trident). Additional support was provided by ANR (France) Mothseim (project NT05-3 42213) and CNRS to J.



(a) Sparse 3D Points (b) Dense 3D Points (c) Poisson Surface

Fig. 5. Results corresponding to the three first stages of the proposed processing pipeline for the Tour Eiffel dataset. (a) shows the sparse set of 3D points obtained by the SFM method; (b) shows the refined topology obtained by the dense 3D reconstruction method; and (c) depicts the mesh of triangles forming the surface as returned by the Poisson method.

Escartin. CNRS and IFREMER (France) financed the Bathyluck'09 cruise where the deep-sea data were acquired. This is IPGP contribution n. 3195.

REFERENCES

[1] V. Rigaud, J. Opderbecke, P. Siméoni, and C. Pitout, "Real time operational mosaics with the deep sea ROV VICTOR," in *International Workshop on Underwater Robotics for Sea Exploitation and Environmental Monitoring*, Rio de Janeiro, Brasil, October 2001.

[2] "Calibration toolbox for matlab." http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.

[3] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[4] T. Nicosevici, N. Gracias, S. Negahdaripour, and R. Garcia, "Efficient three-dimensional scene modeling and mosaicing," *Journal of Field Robotics*, vol. 26, pp. 759–788, October 2009.

[5] L. Robert and O. Faugeras, "Relative 3d positioning and 3d convex hull computation from a weakly calibrated stereo pair," in *Fourth International Conference on Computer Vision*, May 1993, pp. 540–544.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, November 2004.

[7] M. A. Lourakis and A. Argyros, "SBA: A Software Package for Generic Sparse Bundle Adjustment," *ACM Trans. Math. Software*, vol. 36, no. 1, pp. 1–30, 2009.

[8] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 1362–1376, 2010.

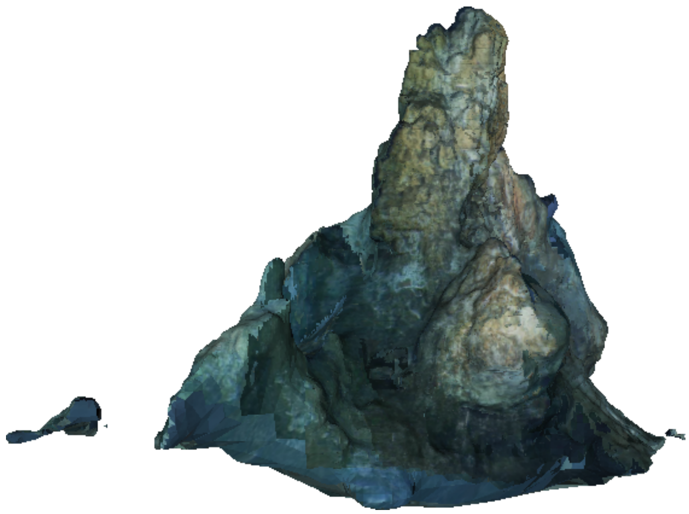
[9] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Eurographics Symposium on Geometry Processing*, 2006, pp. 61–70.

[10] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *SIGGRAPH Comput. Graph.*, vol. 21, pp. 163–169, August 1987.

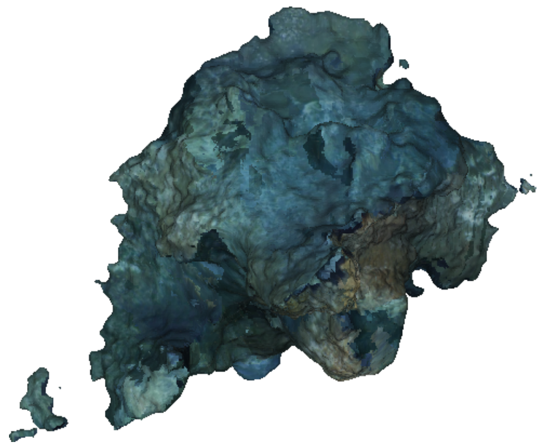
[11] H. Ondréas, M. Cannat, Y. Fouquet, A. Normand, P.-M. Sarradin, and J. Sarrazin, "Recent volcanic events and the distribution of hydrothermal venting at the Lucky Strike hydrothermal field, Mid-Atlantic Ridge," *Geochemistry, Geophysics, Geosystems*, vol. 10, p. Q02006, 2009.

[12] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer-Verlag, 2nd ed., 2006.

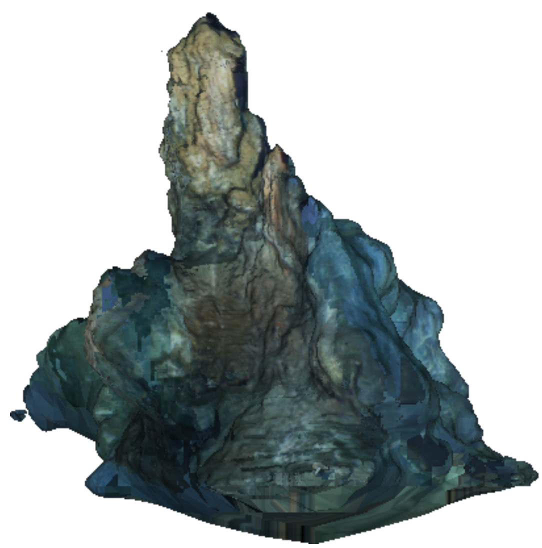
[13] C. Langmuir, S. Humphris, D. Fornari, C. V. Dover, K. V. Damm, M. Tivey, D. Colodner, J. l. Charlou, D. Desonie, C. Wilson, Y. Fouquet, G. Klinkhammer, and H. Bougault, "Hydrothermal vents near a mantle hot spot: the Lucky Strike vent field at 37°N on the Mid-Atlantic Ridge," *Earth and Planetary Science Letters*, vol. 148, pp. 69–91, 1997.



(a) Front view



(b) Top view



(c) Right view

Fig. 6. Three views of the texture-mapped surface of the Tour Eiffel hydrothermal vent, after applying our processing pipeline. The structure is $\sim$15m high.

Coordinated Sampling of Dynamic Oceanographic Features with AUVs and Drifters

Jnaneshwar Das*, Frédéric Py†, Thom Maughan†, Monique Messié†,
Tom O'Reilly†, John Ryan†, Gaurav S. Sukhatme*, Kanna Rajan†
*Robotic Embedded Systems Laboratory, Department of Computer Science, University of Southern California
{*jnaneshd,gaurav*}@*usc.edu*
†Monterey Bay Aquarium Research Institute, Moss Landing, California
{*fpy,monique,oreilly,ryjo,kanna.rajan*}@*mbari.org*

***Abstract*— This paper presents Autonomous Underwater Vehicle (AUV) survey methodologies to track and sample an advecting patch of water. Current AUV-based sampling techniques rely primarily on geographic waypoint track-line surveys that are suitable for static or slowly changing features. We briefly[1] describe extensions to existing oceanographic sampling methodologies to sample within the context of an advecting[2] feature of interest. We use GPS-tracked Lagrangian drifters to tag a patch of interest, and utilize its periodic position updates to make an AUV perform surveys around the drifter as it gets advected by ocean currents. We present results from a five-day offshore experiment carried out in September 2010.**

I. Introduction

Autonomous Underwater Vehicles (AUVs) have allowed oceanographers to collect data over temporal and spatial scales that would be logistically impossible or prohibitively expensive using traditional ship-based measurement techniques. Currently, AUV-based surveys rely primarily on geographic waypoint track-line surveys (Fig. 3b) that are suitable for static or slowly changing features such as bathymetry [3], magnetism, and aquatic environments characterized by weak circulation. When studying dynamic, rapidly evolving oceanographic features (e.g. phytoplankton blooms shown in Fig. 2), such methods at best introduce error through insufficient spatial and temporal resolution, and at worst completely miss the spatial and temporal domain of interest. Our work is situated in the context of a multi-year interdisciplinary field program, the Controlled, Agile and Novel Observing Network (CANON) [3]. This program focuses on understanding rapidly changing coastal ocean processes that have significant impact on local ecosystems. Our objective is to develop methodologies to track and sample bloom patches as they are advected within coastal waterways. We do so by sampling in the Lagrangian frame of reference, that is, the frame of reference moving with the feature of interest [4]. Drifters (Fig. 1) are often used as proxies for advection to study marine transport [5]. This work describes Lagrangian survey experiments where an AUV performs surveys relative to a GPS-tracked drifter used to track a patch of interest. In this *coordinated sampling task*, a survey template is repeatedly undertaken by an AUV in two modes a) by repeating static-plan surveys to stay with the moving patch and b) transformed surveys carried out in the frame of reference of an advecting patch. We present a brief description of our approach, followed by the results from a five day field trial carried out in September 2010.

[1]Preliminary treatment appears in ISER 2010 [1] and a full treatment is in review with IJRR [2].

[2]The horizontal transport of a patch of water.

[3]The study of the floor of water bodies, resulting in a depth contour map.

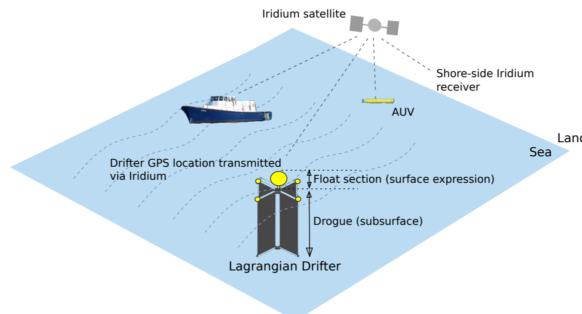


Fig. 1: Illustration of a Lagrangian drifter being tracked on shore and at sea. The drifter has a float section affected mostly by wind and a drogue section which is dragged by the currents. Drifter locations are transmitted via satellite.

II. Technical Approach

Existing AUV sampling methodologies use survey patterns designed in the Earth frame. Hence, by design, these *static-plan surveys* are suitable for features that do not move out of the survey's region of coverage. The scientific goal of this work is to extend existing oceanographic sampling methodologies to sample within the context of an advecting feature of interest. We approach this problem in two parts a) track a patch using GPS-tracked Lagrangian drifters and b) sample within the context of the drifter-tagged advecting patch by extending existing oceanographic survey patterns. Additionally, we define an *enclosure criterion* that ensures that the AUV encapsulates and characterizes the survey volume, by enforcing the constraint that the patch center marked by the drifter always stays within the perimeter of the survey.

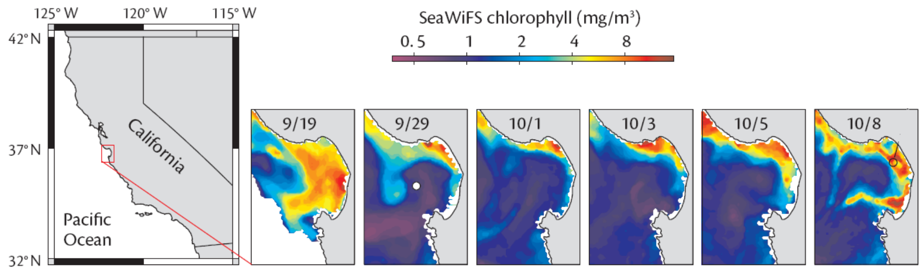
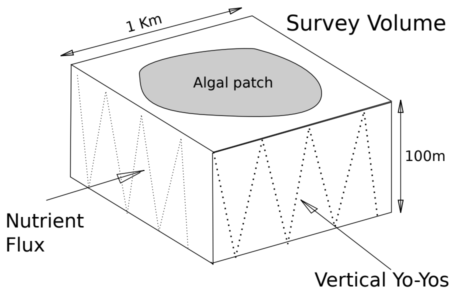
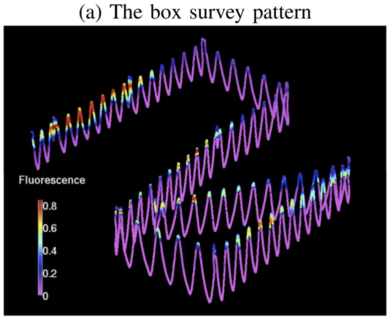


Fig. 2: Dynamics of phytoplankton bloom over a period of 20 days

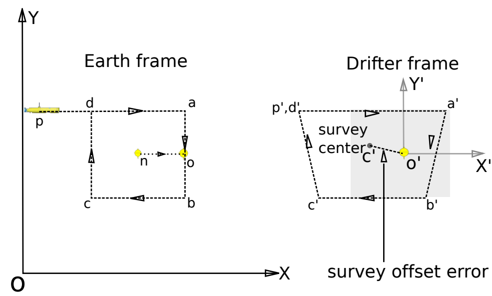


(a) The box survey pattern

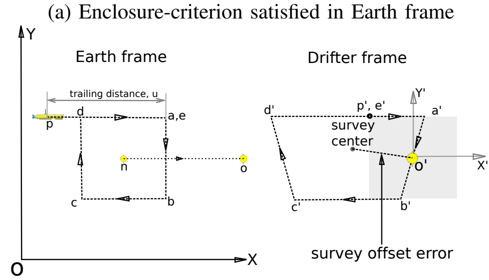


(b) A lawnmower survey pattern showing Chlorophyll fluorescence within vertical yo-yo profiles.

Fig. 3: Box and Lawnmower AUV survey patterns.



(a) Enclosure-criterion satisfied in Earth frame



(b) Enclosure-criterin satisfied in drifter frame

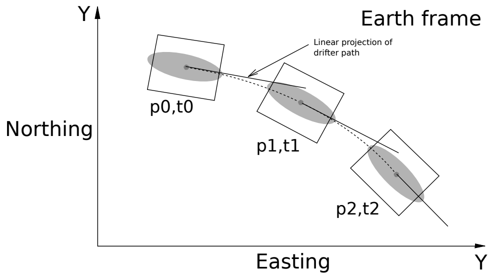
Fig. 4: The goal of our study is to implement surveys such that a drifter that represents a patch of water stays within the boundary of the survey. **n** represents the starting point of the drifter and **o** its termination within a single survey pattern.

Two ways of approaching our goal to perform Lagrangian observation studies are a) repeated static-plan surveys and b) transformed surveys. In repeated static-plan surveys (Fig. 5), we perform existing oceanographic surveys repetitively, repositioning the AUV to the latest location of the drifter once a survey or iteration is complete. In case of transformed surveys, we design the survey pattern to be implemented in the *drifter frame* and transform it back to the Earth frame to obtain the survey plan to command the AUV. For repeated static-plan surveys the enclosure criterion can be satisfied either only in the drifter frame, or both in Earth and drifter frames (Fig. 4). On analysis [2], we find that over multiple iterations, the AUV converges to a fixed trailing distance[4] u^*, dependent solely on the drifter speed, and independent of the trailing distance at the beginning of the experiment. For a nominal AUV speed of $1.5m/s$, AUV pitch angle $\sim 22^o$, the allowable drifter speed to satisfy the enclosure criterion is given by $s_d < 0.146m/s$ for enclosure in Earth frame, and $s_d < 0.364m/s$ enclosure in the drifter frame. From simulation on historical drifter data, we found that repeated static-plan surveys do not satisfy the enclosure criterion for 30% of observed drifter speeds [2]. To address this issue, we design the surveys in the drifter frame of reference. Fig. 6 illustrates this concept by the projection of a box survey template to the Earth frame which defines the goal waypoints for commanding the AUV.

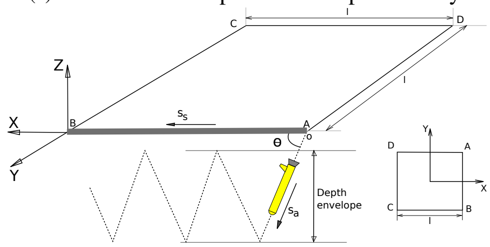
III. Field Trials

A five-day field experiment was carried out in September 2010, 160 Kms off the California coast. MBARI's 4 m long propelled AUV, *Dorado* (Fig. 8) carried out Lagrangian surveys around a specialized advecting drifter that contained an onboard genomic sensor

[4]The distance the AUV lags or trails behind its survey start location for the next iteration.



(a) Illustration of repeated static-plan survey.



(b) AUV and survey parameters

Fig. 5: Repeated static-plan surveys and AUV and survey parameters.

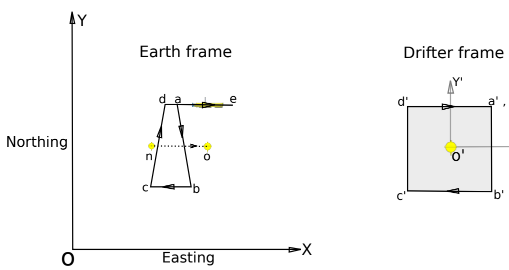


Fig. 6: Box survey pattern in drifter frame and its projection onto an Earth frame.

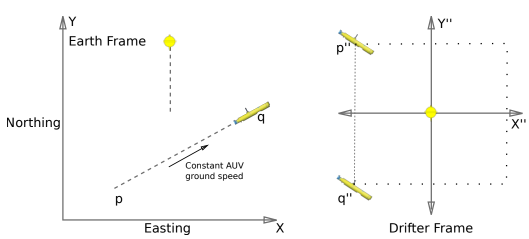


Fig. 7: Illustration of AUV navigation between two waypoints of the box pattern using constant speed in Earth frame.

Fig. 8: The *Dorado* AUV being loaded on the R/V *Zephyr* for the five day drifter tracking experiment in September 2010.

allowing in-situ identification of micro-organisms. The experiment had multiple goals spread across crews on two support vessels, the *R/V Western Flyer* and the *R/V Zephyr*. The Flyer visited the drifter every four hours to carry out a series of ship-based sampling experiments and lab analysis on water samples to ground-truth the genomic sensor data. The *Zephyr* was meanwhile focused entirely on Lagrangian observation studies with the AUV. The goal for this experiment was to monitor the nutrient budget at the perimeter of a 1km X 1km water patch around the advecting drifter while the AUV was used to perform a transformed box pattern (Fig. 3a around the drifter. A number of logistical issues were kept in mind while designing and executing the experiment. Each iteration began with the latest drifter update (position and velocity) received from the drifter through an Iridium satellite link. This was transmitted to the AUV for in-situ adaptation. With this input, the AUV's onboard hybrid plan-execution controller, `T-REX` [6], [7], [8] computed the waypoints for a box pattern using a formulation where the AUV travels at constant velocity in Earth frame [2]. Waypoints were computed once at the beginning of the survey with the AUV surfacing once at every waypoint with each survey lasting ∼1 to 1.5 hrs. In total, 60 iterations were attempted over the course of five days, out of which 45 were completed successfully (some iterations had to be canceled midway or restarted due to operational reasons). Fig. 10 shows the overall track lines of the drifter and the perambulating AUV for all five days. Fig. 9 shows the AUV path in the drifter frame for Day 4 of the experiment.

IV. Analysis

We analyzed the contribution of different sources of error to the quality of surveys during the September field trial (Fig. 11). We defined the survey quality using two metrics a) the mean surfacing error in the drifter-frame (MSE-DF), and b) the survey offset error in the drifter frame [2]. The MSE-DF correlates most with mean surfacing error in Earth-frame (MSE-EF), with a correlation coefficient $R = 0.62$, and with the mean timing error[5] with $R = 0.56$. A detailed discussion on the sources of error is available in Das et al. [2].

V. Acknowledgments

This work was supported in part by the NOAA MERHAB program under grant NA05NOS4781228 and by NSF as part of the Center for Embedded Network Sensing (CENS) under grant CCR-0120778, by NSF grants CNS-0520305 and CNS-0540420, by the ONR MURI program (grants N00014-09-1-1031 and N00014-08-1-0693) by the ONR SoA program and a gift from the Okawa Foundation. We thank the David and Lucile Packard Foundation for supporting our work at MBARI and the crew of the R/V *Zephyr* for help with deployments.

[5]Due to the transformations between the advecting drifter-frame and the Earth-frame, difference between intended and actual time of surfacing in the Earth frame results in surfacing error in the drifter frame.

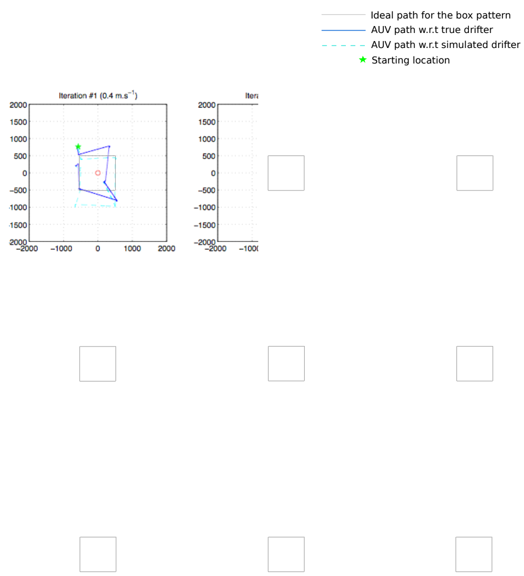


Fig. 9: Iterations from Day 4 of the September trial illustrating the drifter-frame views

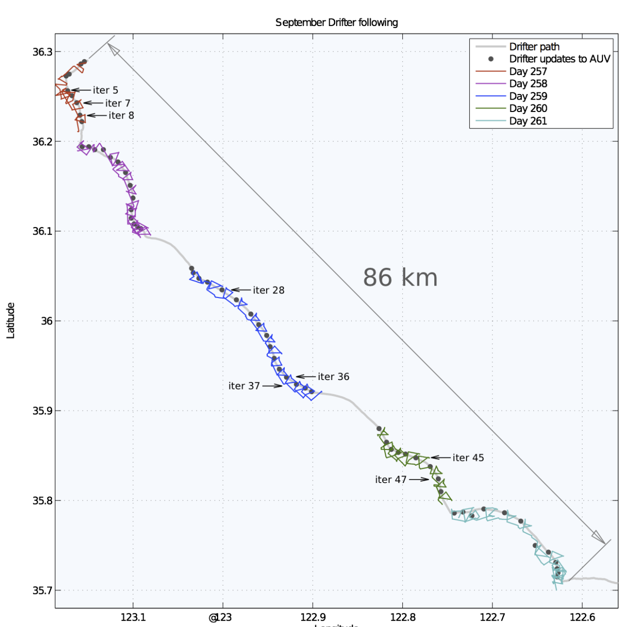


Fig. 10: AUV and drifter paths during the September 2010 five-day field trial

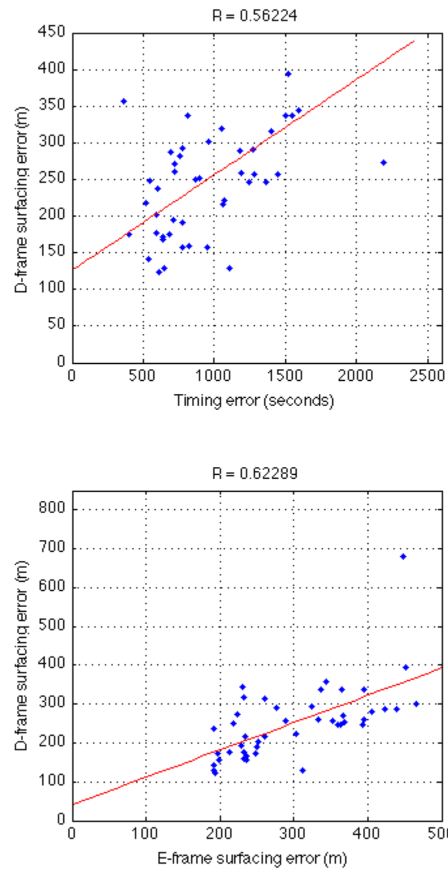


Fig. 11: Scatter plots for error-pairs, along with the correlation coefficient R.

REFERENCES

[1] J. Das, F. Py, T. Maughan, T. O'Reilly, M. Messi, J. Ryan, K. Rajan, and G. S. Sukhatme, "Simultaneous tracking and sampling of dynamic oceanographic features with auvs and drifters," in *12th International Symposium on Experimental Robotics, 2010*, Dec 2010.

[2] J. Das, F. Py, T. Maughan, T. O'Reilly, M. Messi, J. Ryan, G. Sukhatme, and K. Rajan, "Simultaneous tracking and sampling of dynamic oceanographic features with auvs and drifters," *International Journal of Robotics Research*, Submitted.

[3] "CANON: Controlled, Agile and Novel Observing Network," 2010. [Online]. Available: http://www.mbari.org/canon/

[4] R. E. Davis, "Lagrangian Ocean Studies," *Annual Review of Fluid Mechanics*, vol. 23, no. 1, pp. 43–64, 1991.

[5] R. Lumpkin and M. Pazos, "Measuring surface currents with surface velocity program drifters: the instruments, its data and some recent results," *Lagrangian Analysis and Prediction of Coastal and Ocean Dynamics. Cambridge University Press*, 2007.

[6] C. McGann, F. Py, K. Rajan, H. Thomas, R. Henthorn, and R. McEwen, "A Deliberative Architecture for AUV Control," in *IEEE Intnl. Conf. on Robotics and Automation*, Pasadena, CA, May 2008.

[7] C. McGann, F. Py, K. Rajan, J. P. Ryan, and R. Henthorn, "Adaptive Control for Autonomous Underwater Vehicles," in *Proc. Assoc. for the Adv. of AI*, Chicago, IL, 2008.

[8] F. Py, K. Rajan, and C. McGann, "A Systematic Agent Framework for Situated Autonomous Systems," in *9th International Conf. on Autonomous Agents and Multiagent Systems*, Toronto, Canada, May 2010.

Pseudoseeds: Investigating Long-Distance, Ocean Seed Dispersal with Wireless Sensors

Ryan N. Smith, Peter Prentis, Koen Langendoen and Peter Corke

Abstract— **Recent theoretical research has shown that ocean currents and wind interact to disperse seeds over long distances among isolated landmasses. Dispersal of seeds among isolated oceanic islands, by birds, oceans and man, is a well-known phenomenon, and many widespread island plants have traits that facilitate this process. Crucially, however, there have been no mechanistic vector-based models of long-distance dispersal for seeds among isolated oceanic islands based on empirical data. Here, we propose a plan to develop seed analogues, or *pseudoseeds*, fitted with wireless sensor technology that will enable high-fidelity tracking as they disperse across the ocean. The pseudoseeds will be precisely designed to mimic actual seed buoyancy and morphology enabling realistic and accurate, vector-based dispersal models of ocean seed dispersal over vast geographic scales.**

I. INTRODUCTION

The seminal book *The Dispersal of Plants Throughout the World* was published by Ridley [14] over 80 years ago, yet we still know remarkably little about patterns of long-distance seed dispersal among isolated oceanic islands [12]. Dispersal of seeds among isolated oceanic islands, by birds, oceans and man, is a well-known phenomenon [11] and many widespread island plants have traits that facilitate this process. For example, ocean dispersal is responsible for over 78% of plant colonists arriving on the volcanic island of Surtsey [8], highlighting the importance of this vector for island colonization. Although we know that buoyant seeds of many coastal plants have dispersed long distances to colonize isolated islands, remarkably little is known about long-distance seed dispersal in oceanic environments [12].

Recent theoretical research has shown that ocean currents and wind interact to disperse seeds over long distances among isolated landmasses. Crucially, however, there have been no mechanistic vector-based models of long-distance dispersal (LDD) for seeds among isolated oceanic islands based on empirical data. Currently, it is only hypothesised that LDD in the ocean can be represented as a fat tailed dispersal kernel as shown in Fig. 1. Note that the a significant number ($> 80\%$) do not disperse a distance of more than 10 m, implying that only a small percentage of dropped seeds make it off the beach. Further, the ”fat tail” of the dispersal may only comprise 1% of the total dropped seeds, but

R.N. Smith and P. Corke are with the School of Engineering Systems at the Queensland University of Technology, Brisbane, QLD, 4000 Australia. ryan.smith@qut.edu.au, peter.corke@qut.edu.au

P. Prentis is with the School of Biogeosciences in the Chemical, Earth and Life Sciences Portfolio at the Queensland University of Technology, Brisbane, QLD, 4000 Australia. p.prentis@qut.edu.au

K. Langendoen is with the Faculty of Electrical Engineering, Mathematics and Computer Science at Delft University of Technology, Delft, The Netherlands k.g.langendoen@tudelft.nl

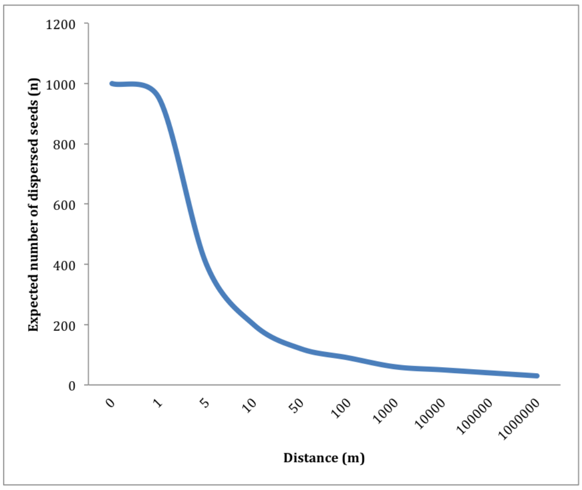


Fig. 1. A stylized representation of a fat tailed dispersal kernel for seeds of coastal plants that remain buoyant for long time periods immersed in seawater. The vectors that displace seeds in the ocean are surface currents and wind.

this percentage remains constant after a threshold distance. Such data is difficult to obtain because 1) LDD is rare and stochastic in nature [3], and 2) reconstructing past patterns of LDD is difficult in most organisms [6]. Drift bottles and drift cards have been used to study ocean currents as analogues for plant dispersal in oceanic environments, but suffer from a number of major limitations [7]. These limitations include a lack of information on which specific dispersal route a seed took, as only the points of release and stranding are known. A second limitation is that drift cards and bottles need to be found to provide data on where they strand, an unlikely situation if they wash up on isolated coastlines. The third major limitation is that drift bottles and cards are very different from seeds, both in terms of morphology and buoyancy as well as how they are released into the ocean. Here, we propose a way to circumvent these limitations and directly estimate a mechanistic vector-based model of seed dispersal in ocean waters.

We plan to develop seed analogues, or *pseudoseeds*, fitted with wireless sensor technology that will enable high-fidelity tracking as they disperse across the ocean. The pseudoseeds will be precisely designed to mimic actual seed buoyancy and morphology enabling realistic and accurate, vector-based dispersal models of ocean seed dispersal over vast geographic scales.

In the remainder of this paper, we will first outline the dispersal experiment in Section II. Then, we will discuss the WSN technology available for building pseudoseeds (Sec-

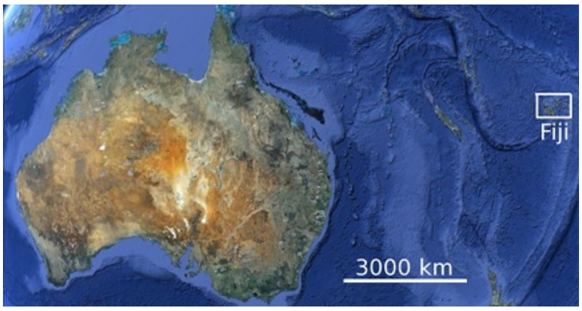


Fig. 2. The geographic location of Fiji relative to Australia, and the scale of the proposed seed tracking experiment. The primary landing location for the seeds is along the north-east coast of Australia.

tion III) and their use in monitoring the dispersal by ocean currents and wind (Section IV). We conclude in Section VI with a brief discussion of the feasibility of the WSN-based approach versus the use of standard satellite-based tracking devices.

II. PROBLEM FORMULATION

Despite recent progress towards understanding patterns of long distance seed dispersal in terrestrial environments, relatively little headway has been made in estimating patterns of long distance seed dispersal in marine environments. Tracking seeds over vast spatial and temporal scales between their source and where they are finally deposited is the major logistical problem that has impaired progress toward developing realistic models of seed dispersal in oceanic environments. In order to overcome this challenge a multidisciplinary approach is required to develop ways to track seeds or biologically realistic pseudoseeds in seawater over long time periods (12 months) and large distances ($>$ 3000 km). The ability to track seeds from initiation to termination of dispersal will enable the estimation of a general mechanistic model of seed dispersal by ocean currents.

A. Geographic Location

This project will focus on seed dispersal in the south west region of the Pacific Ocean. Primarily, we are interested in the transport of seeds between Fiji and Australia. Figure 2 shows the relative locations of Fiji and Australia, and provides the spatial scale of the proposed tracking experiment. The aim of this project is to release the pseudoseeds in Fiji during the appropriate time, and track them, hopefully along with other *real* seeds, across the south west Pacific to the shores of Australia. This will be the first such demonstration of ocean seed dispersal tracking.

B. Seed Specifics

The coconut palm, Cocos nucifera, will be used as a model to develop pseudoseeds. The coconut is a range expanding species commonly found throughout the Pacific and Indian oceans. Seeds of the coconut palm remain positively buoyant and viable in seawater for extended time periods and commonly disperse to the Australian east coast, and Great Barrier Reef islands from the south west Pacific. The coconut palm seed was also chosen as the pseudoseed model to allow for maximum size and payload capacity of a realistic seed to be tracked through the ocean. The details of an average coconut palm seed are presented in Table I. In particular, the large size provides adequate space for the electronics, and the weight allowance allows for a reasonable size battery to extend the deployment life for as long as possible. Depending on the exact route, currents, and wind speed, the pseudoseeds could take from four months to over a year to travel from Fiji to Australia.

TABLE I
MAIN OCEAN SEED PARAMETERS.

weight	1.5 kg
size	
- length	30 cm
- max diameter	12.5 cm
dispersal speed	0.1 - 0.4 m/s

III. PSEUDOSEED CONCEPT

The ongoing trend in miniaturization of digital circuitry has opened up the possibilities for in situ sensing at high temporal and spatial resolution by means of deploying cheap, autonomous sensor nodes configured in a wireless ad-hoc network. Once a fantasy (Smart Dust [9]), now a reality with ever-more successful deployments of Wireless Sensor Networks (WSNs) [2], [5], [13]. We plan to capitalize on this development by utilizing pseudoseeds equipped with WSN technology, such that we can track their whereabouts for up to a year. In particular we will include the following components:

- **Battery** To keep the design of the pseudoseeds as simple as possible, they will be powered by means of a battery; the alternative of using solar power would include additional charging circuitry as well as battery, and introduce uncertainties as the energy harvesting may be compromised by befoulment.
- **GPS** A GPS unit will be used to record the actual location of the pseudoseed during its time out on the ocean. As GPS is a notorious consumer of energy, we will drive it to only take one reading per day.
- **Low-power radio** At some point the logged GPS data must be off loaded, and doing so wirelessly is convenient. It also allows for remote data collection, for example by a UAV, avoiding the need for physical recovery and offers the possibility for online tracking. As for the GPS, the radio must be duty cycled to avoid draining the batteries, a well researched topic with the WSN community [1], [10].
- **Accelerometer** Wireless communication goes best with a direct line of sight between sender and receiver. Since pseudoseeds travel on the ocean surface, it pays off to exchange messages when being on top of a wave, something which can easily be derived from accelerometer data.
- **Satellite modem** Although expensive, the Iridium satellite network provides true global coverage and is the

only option for pseudoseeds to report their trajectory while far out on the ocean. The form factor (20 cm^3) and power consumption (300 mA) make modern satellite modems a viable option, but raises the unit and operation costs considerably (cf. Section IV). Alternatively small low-power Argos tags could be used that are tracked by the Argos satellite network, and can additionally upload short messages. However the cost of these tags is high (c.f. 2000 USD) as is the communications cost (c.f. 1500 USD per year).

- **Processor** A simple microcontroller is required to orchestrate the periodic logging of GPS positions and communication with the outside world. As most time will be spent doing nothing, it is important that the processor can be put in a deep sleep consuming near-zero power.

These components will be fitted in a waterproof enclosure, roughly the shape and size of a coconut. The weight of a pseudoseed needs to be carefully set to get the right buoyancy, and basically determines the size of the battery pack that can be included. Following the parameters provided in Table I we can expect a pseudoseed to carry a payload of about 500 grams, of which about half (250 g) can be reserved for the battery. Using plain alkaline batteries that weight translates into a 30 Ah energy storage capacity, or roughly 80 mAh a day when targeting a lifetime of one year.

GPS units can take anywhere from a second to a minute to acquire a position, depending on the number of satellites in view and the movement of the device since the last reading. As pseudoseeds travel about 10 - 30 km a day, and we only take one reading a day it is highly likely that the GPS will go through a *cold start* every day consuming about 1 mAh (= 60 s $\times$ 50 mA). Note that this cost is insignificant to the complete daily budget, that is, GPS consumes only about 1% of the total available energy budget, indicating that basically all energy can be spent on the communication part of the pseudoseeds, which is the key to successful operation as explained below.

IV. PROPOSED SOLUTION

As an alternative to the readily available satellite tag technologies we propose a solution to monitoring of ocean seed dispersal that uses both wireless sensor network (WSN) and robot technologies. A number of WSN-based pseudoseeds, set to log their position once a day, will be dropped off the coast of Fiji at the start of the experiment. Some of these will be deployed to the ocean, but the vast majority will quickly return to shore. These pseudoseeds must be located and collected as to provide empirical data about the bulk of the dispersal distribution, i.e. the left part of the curve in Figure 1. After physically recovering these short-lived pseudoseeds, they can be redeployed and the experiment repeated. This will provide rich statistical data about the probability of the seeds leaving the coast, based on a sample size that is some multiple of the actual number of pseudoseeds that are built. The devices that leave the coastal region and drift out to sea will eventually reach landfall at a time and position that must be determined.

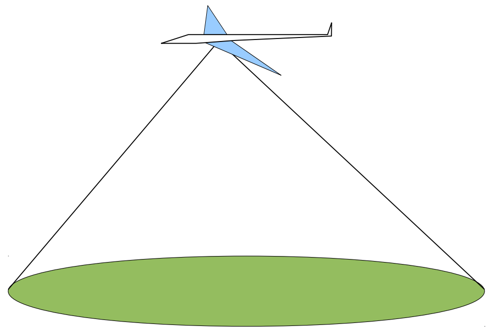


Fig. 3. Aerial reconnaissance of pseudoseeds

A key part of the recovery strategy is that the pseudoseeds (or nodes in WSN speak) broadcast their GPS location, either periodically or in response to a challenge that is detected by a duty-cycled radio receiver. Modern, popular 802.15.4 transceivers such as the CC2430 have identical power consumption for receiving and transmitting (27 mA), so the simplest solution is to periodically broadcast location, rather than periodically enabling the receiver and broadcasting position in response to a challenge. Link level acknowledgement would be used to ensure that the node knows that its message has been received. A GPS measurement comprises 8 bytes of data (timestamp + position), so an 802.15.4 payload of 127 bytes can easily hold 14 GPS readings (from the most recent two weeks) plus device status information.

Communicating the location of the pseudoseed facilitates recovering those that return to shore and also in tracking those that drift out to sea. The shore recovery problem is easiest since it involves searching a bounded strip of coastline either side of the release point. Nodes would detect that they have been beached through the absence of motion detected by their accelerometers, and include this status in their broadcast messages. Beached nodes may increase their broadcast interval since energy considerations do not apply – they could have new batteries fitted before redeployment. A spectrum of beach recovery options are possible. The simplest is to drive along the beach, listening for messages from stranded nodes and using a GPS navigation system to drive toward the node. Alternatively an aircraft (manned or UAV) could fly along the beach and collect location data [16] from beached nodes, and this would be used to plan an efficient and targeted ground recovery mission.

Determining the eventual landfall for the pseudoseeds that drift to sea is a very difficult problem since the potential search area is massive. However they need to be found quickly after landfall since they will slowly become buried and unable to communicate. The most feasible way to solve this problem is to track the pseudoseeds while they drift in the ocean away from the release point. Continuous tracking will provide additional rich information about the paths taken by seeds, rather than just their start and end point. Predictive models of ocean currents do exist, but they are not perfect, and the seeds are also heavily influenced by wind and waves. Over time the uncertainty of the pseudoseeds will grow, so we need to periodically localize the devices and update the drift models. The actual position can be determined by aerial survey using manned aircraft or UAV (see Figure 3). For the purposes of this exercise we consider a UAV in a class similar to the in situ Scan Eagle, which has a cruise speed of 75 knots (40 m/s) and an endurance of 20 hours, which would allow it to cover a total distance of 2800 km in a single mission. A UAV flying at 2000 m should be able to communicate with nodes lying in a circular region of radius 4500 m and an area of 64 sq.km. If the aircraft spent half its mission time flying to and from the search zone, that zone could be 700 km from shore and the searched area would be 12,600 sq.km. If the nodes emitted their GPS data once per minute the aircraft would hear a minimum of two broadcasts while the node was in the reception footprint. The UAV has a much greater power budget than the pseudoseed and would be able to carry a more sensitive receiver (higher weight and power consumption) than a normal node. The radio environment over the ocean is also electromagnetically quieter than on land. Since the accuracy of the drift models in predicting seed movement is currently unknown it would be appropriate to initially track the devices fairly frequently and this is while they are still relatively close to the release point. As the efficacy of the model is determined, a trade-off can be made between the search interval and the area to be searched subject to the operational constraints of the aircraft.

V. Additional Outcomes

A. Ocean Current Modelling

In recent years, many large-scale, regional ocean models have been developed to help deepen our understanding of the complex and dynamic ocean. Model outputs are currently used to study and predict physical and biological phenomena [4], and to guide autonomous underwater vehicles for increased navigation and tracking features of interest [15]. Here, we plan to utilise ocean model predictions for preliminary simulations of the release of the pseudoseeds, and for a general knowledge of the regional currents to assist in tracking the pseudoseeds after they have been released.

A concern with any model is the accuracy of its predictions. Specifically, we are concerned with the spatial structure of the predicted *surface* current velocities. Existing ocean models assimilate surface velocities from and compare predictions to high-frequency radar data measurements. However, these data are generally only available near urbanized coastal regions; not in isolated island chains or the open ocean. To assist in improving the performance, quality and utility of ocean models forecasts, we plan to compare our pseudoseed trajectories with trajectories predicted by an ocean model. This will provide a fine-scale analysis of the surface currents in regions where measurements are difficult to obtain and ground truth.

VI. Conclusions

We have proposed a solution to the problem of monitoring ocean dispersed seeds that uses both wireless sensor network (WSN) and robot technologies. We propose the use of low-cost WSN technology, and by using well known duty-cycling techniques we can easily achieve daily GPS fixes and frequent position broadcast, which facilitates recovery and tracking at sea. The overall cost advantage of this approach over the traditional satellite approach depends critically on the number of nodes and the cost of the aerial monitoring. The cost of the satellite uplink approach is linear in the number of nodes being tracked, while the proposed WSN-approach has an almost constant search cost. The cost of aerial monitoring however has two factors. The first is the operating cost per hour, and while UAVs have theoretically lower operational cost than manned aircraft reliable $/hour figures are very difficult to obtain. The second factor is the number of hours that need to be flown, and this depends on the rate of growth of uncertainty. If this rate is low then flights can be less frequent, but at this stage this is a significant unknown.

References

[1] A. Bachir, M. Dohler, T. Watteyne, and K. Leung. Mac essentials for wireless sensor networks. *Communications Surveys Tutorials, IEEE*, 12(2):222 –248, 2010.

[2] J. Beutel, S. Gruber, A. Hasler, R. Lim, A. Meier, C. Plessl, I. Talzi, L. Thiele, C. Tschudin, M. Woehrle, and M. Yuecel. PermaDAQ: A scientific instrument for precision sensing and data recovery in environmental extremes. In *8th ACM/IEEE Int. Conf. on Information Processing in Sensor Networks (ISPN09) - SPOTS track*, San Francisco, CA, Apr. 2009.

[3] M. L. Cain, B. G. Milligan, and A. E. Strand. Long-distance seed dispersal in plant populations. *American Journal of Botany*, 87(9):1217–1227, September 2000.

[4] Y. Chao, Z. Li, J. Farrara, M. Moline, O. M. E. Schofield, and S. J. Majumdar. Synergistic applications of autonomous underwater vehicles and regional ocean modeling system in coastal ocean forecasting. *Limnol. Oceanogr.*, 53(5, part 2):2251–2263, 2008.

[5] O. Chipara, C. Lu, T. C. Bailey, and G.-C. Roman. Reliable clinical monitoring using wireless sensor networks: experiences in a step-down hospital unit. In *8th ACM Conf. on Embedded Networked Sensor Systems (SenSys 2009)*, pages 155–168, Zurich, Switzerland, Nov. 2010.

[6] M. D. Crisp, S. A. Trewick, and L. G. Cook. Hypothesis testing in biogeography. *Trends In Ecology & Evolution*, 26(2):66–72, February 2011.

[7] P. Heyligers. The role of currents in the dispersal of introduced seashore plants around australia. *Cunninghamia*, 10:167–188, 2007.

[8] S. I. Higgins, R. Nathan, and M. L. Cain. Are long-distance dispersal events in plants usually caused by nonstandard means of dispersal? *Ecology*, 84(8):1945–1956, August 2003.

[9] J. Kahn, R. Katz, and K. Pister. Next Century Challenges: Mobile Networking for ''Smart Dust". In *5th ACM/IEEE Int. Conf. on Mobile Computing and Networks (MobiCom '99)*, pages 271–278, Seatle, WA, Aug. 1999.

[10] K. Langendoen. Medium access control in wireless sensor networks. In H. Wu and Y. Pan, editors, *Medium Access Control in Wireless Networks*, pages 535–560. Nova Science Publishers, Inc., May 2008.

[11] R. Nathan. Long-distance dispersal of plants. *Science*, 313(5788):786–788, August 2006.

[12] R. Nathan, F. M. Schurr, O. Spiegel, O. Steinitz, A. Trakhtenbrot, and A. Tsoar. Mechanisms of long-distance seed dispersal. *Trends In Ecology & Evolution*, 23(11):638–647, November 2008.

[13] S. N. Pakzad, G. L. Fenves, S. Kim, and D. E. Culler. Design and implementation of scalable wireless sensor network for structural monitoring. *Journal of Infrastructure Systems*, 14:89–101, 2008.

[14] H. Ridley. *The dispersal of plants throughout the world*. L. Reeve & Co., Ltd., Lloyds Bank Buildings, 1930.

[15] R. N. Smith, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme. Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model. *International Journal of Robotics Research*, 29(12):1475–1497, October 2010.

[16] S. Teh, L. Mejias, P. Corke, and W. Hu. Experiments in integrating autonomous uninhabited aerial vehicles (UAVs) and wireless sensor networks. In J. Kim and R. Mahony, editors, *Australasian Conference on Robotics and Automation*, Canberra, Australia, 2008.

Mapping and Dilution Estimation of Foz do Arelho Outfall Plume using an Autonomous Underwater Vehicle

Patrícia Ramos
INESC Porto
Campus da FEUP, Rua Dr. Roberto Frias, 378
4200-465 Porto, Portugal
ISCAP, Rua Jaime Lopes Amorim, s/n
4465-004 S. Mamede de Infesta, Portugal
pramos@inescporto.pt

Nuno Abreu
INESC Porto
Campus da FEUP, Rua Dr. Roberto Frias, 378
4200-465 Porto, Portugal
nabreu@inescporto.pt

***Abstract*— In this work geostatistics is used to model and map the spatial distribution of temperature and salinity measurements gathered by an Autonomous Underwater Vehicle in a monitoring campaign to Foz do Arelho outfall, with the aim of distinguishing the effluent plume from the receiving waters, characterizing its spatial variability in the vicinity of the discharge and estimating dilution. The results demonstrate that this methodology provides good estimates of the dispersion of effluent and it is therefore very valuable in assessing the environmental impact and managing sea outfalls.**

I. INTRODUCTION

A. MARES AUV

Autonomous Underwater Vehicles (AUVs) have been used efficiently in a wide range of applications. They were first developed with military applications in mind, for example for mine hunting missions. Later on, scientists realized their true potential and started to use them as mobile sensors, taking measurements in difficult scenarios and at a reasonable cost ([1][2]). MARES (Modular Autonomous Robot for Environment Sampling) AUV has been successfully used to monitor sea outfalls discharges ([3]) (see Fig. 1). MARES is 1.5 m long, has a diameter of 8-inch and weighs about 40 kg in air. It features a plastic hull with a dry mid body (for electronics and batteries) and additional rings to accommodate sensors and actuators. Its modular structure simplifies the system's development (the case of adding sensors, for example). It is propelled by two horizontal thrusters located at the rear and two vertical thrusters, one at the front and the other at the rear. This configuration allows for small operational speeds and high maneuverability, including pure vertical motions. It is equipped with an omnidirectional acoustic transducer and an electronic system that allows for long baseline navigation. The vehicle can be programmed to follow predefined trajectories while collecting relevant data using the onboard sensors. A Sea-Bird Electronics 49 FastCAT CTD had already been installed onboard the MARES AUV to measure conductivity, temperature and depth. MARES' missions for environmental monitoring of wastewater discharges are conducted using a GUI software that fully automates the operational procedures of the campaign ([4]). By providing visual and audio information, this software guides the user through a series of steps which include: (1) real time data acquisition from CTD and ADCP sensors, (2) effluent plume parameter modeling using the CTD and ADCP data collected, (3) automatic path creation using the plume model parameters, (4) acoustic buoys and vehicle deployment, (5) automatic acoustic network setup and (6) real time tracking of the AUV mission.

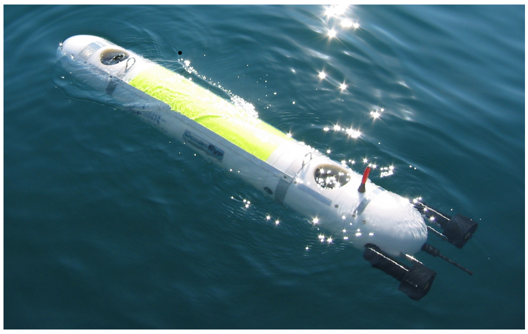


Fig. 1. MARES AUV.

B. Data processing

Data processing is the last step of a sewage outfall discharge monitoring campaign. This processing involves the ability to extrapolate from monitoring samples to unsampled locations. Although very chaotic due to turbulent diffusion, the effluent's dispersion process tends to a natural variability mode when the plume stops rising and the intensity of turbulent fluctuations approaches to zero ([5]). It is likely that after this point the pollutant substances are spatially correlated. In this case, geostatistics appears to be an appropriate technique to model the spatial distribution of the effluent. In fact, geostatistics has been used with success to analyze and characterize the spatial variability of soil properties, to obtain information for assessing water and wind resources, to design sampling strategies for monitoring estuarine sediments, to study the thickness of effluent-affected sediment in the vicinity of wastewater discharges, to obtain information about the spatial distribution of sewage pollution in coastal sediments, among others. As well as giving the estimated values, geostatistics provides a measure of the accuracy of the estimate in the form of the kriging variance. This is one of the advantages of geostatistics over traditional methods of assessing pollution. In this work, universal kriging method [6] is used to model and map the spatial distribution of temperature and salinity measurements gathered by an AUV on a Portuguese sea outfall monitoring campaign. The aim is to distinguish the effluent plume from the receiving waters, characterize its spatial variability in the vicinity of the discharge and estimate dilution.

II. GEOSTATISTICAL ANALYSIS

A. Study site

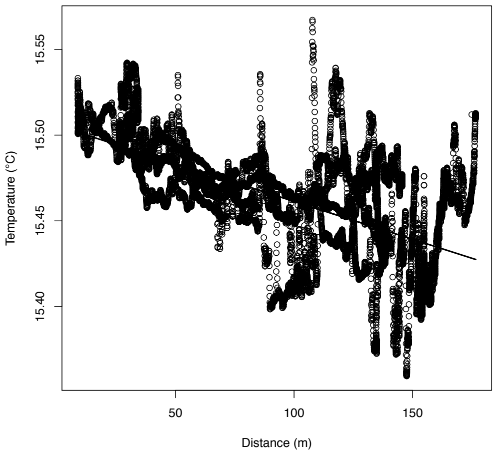
Foz do Arelho outfall is located off the Portuguese west coast near Óbidos lagoon. In operation since June 2005, is presently discharging about 0.11 m^3/s of mainly domestic wastewater from the WWTPs of Óbidos, Carregal, Caldas da Rainha, Gaeiras, Charneca and Foz do Arelho, but it can discharge up to 0.35 m^3/s. The total length of the outfall, including the diffuser, is 2150 m. The outfall pipe, made of HDPE, has a diameter of 710 mm. The diffuser, which consists of 10 ports spaced 8 or 12 meters apart, is 93.5 m long. The ports, nominally 0.175 m in diameter, are discharging upwards at an angle of 90° to the pipe horizontal axis; the port height is about 1 m. The outfall direction is southeast-northwest (315.5° true bearing) and is discharging at a depth of about 31 m. In that area the coastline itself runs at about a 225° angle with respect to true north and the isobaths are oriented parallel to the coastline. A seawater quality monitoring program for the outfall has already started in May 2006. Its main purposes are to evaluate the background seawater quality both in offshore and nearshore locations around the vicinity of the sea outfall and to follow the impacts of wastewater discharge in the area. During the campaign the discharge remained fairly constant with an average flowrate of approximately 0.11 m^3/s. The operation area specification was based on the outputs of a plume prediction model [5] which include mixing zone length, spreading width, maximum rise height and thickness. The model inputs are, besides the diffuser physical characteristics, the water column stratification, the current velocity and direction, and the discharge flowrate. Information on density stratification was obtained from a vertical profile of temperature and salinity acquired in the vicinity of the diffuser two weeks before the campaign. The water column was weakly stratified due to both low-temperature and salinity variations. The total difference in density over the water column was about 0.13 σ-unit. The current direction of 110° was estimated based on predictions of wind speed and direction of the day of the campaign. A current velocity of 0.12 m/s was estimated based on historic data. The effluent flowrate consider for the plume behavior simulation was 0.11 m^3/s. According to the predictions of the model, the plume was spreading 1 m from the surface, detached from the bottom and forming a two-layer flow. The end of the mixing zone length was predicted to be 141 m downstream from the diffuser. The AUV operation area (specified according to the model predictions) was mainly in the northeast direction from the diffuser, covering about 20000 m^2. The vehicle collected CTD data at 1.5 m and 3 m depth, in accordance to the plume minimum dilution height prediction. During the mission transited at a fairly constant velocity of 1 m/s (2 knots) recording data at a rate of 16 Hz. Maximum vertical oscillations of the AUV in performing the horizontal trajectories were less than 0.5 m (up and down).

B. Exploratory analysis

In order to obtain elementary knowledge about the temperature and salinity data sets, conventional statistical analysis was conducted. At the depth of 1.5 m the temperature ranged from 15.359°C to 15.562°C and at the depth of 3 m the temperature ranged from 15.393°C to 15.536°C. The mean value of the data sets was 15.463°C and 15.469°C, respectively at the depths of 1.5 m and 3 m, which was very close to the median value that was respectively 15.466°C and 15.472°C. The coefficient of skewness is relatively low (-0.309) for the 1.5 m data set and not very high (-0.696) for the 3 m data set, indicating that in the first case the distribution is approximately symmetric and in the second case that distribution is only slightly asymmetric. The very low values of the coefficient of variation (0.002 and 0.001) reflect the fact that the distributions do not have a tail of high values. At the depth of 1.5 m the salinity ranged from 35.957 psu to 36.003 psu and at the depth of 3 m the salinity ranged from 35.973 psu to 36.008 psu. The mean value of the data sets was 35.991 psu and 35.996 psu, respectively at the depths of 1.5 m and 3 m, which was very close to the median value that was respectively 35.990 and 35.998 psu. The coefficient of skewness is not to much high in both data sets (-0.63 and -1.1) indicating that distributions are only slightly asymmetric. The very low values of the coefficient of variation (0.0002 and 0.0001) reflect the fact that the distributions do not have a tail of high values. Fig. 2 shows the temperature measurements at depth of 1.5 m (top) and 3 m (bottom) versus distance to the middle point of the diffuser fitted by a linear model. A similar behavior was found for the salinity measurements. These figures show that although some variability there is a certain relation between the measurements and the distance between its location and the middle point of the diffuser. For this reason, universal kriging method was applied.

C. Variogram modeling

For the purpose of this analysis, the temperature and the salinity measurements were divided into a modeling set (comprising 90% of the samples) and a validation set (comprising 10% of the samples). Modeling and validation sets were then compared, using Student's-t test, to check that they provided unbiased sub-sets of the original data. Furthermore, sample variograms for the residuals of the modeling sets were constructed using the Matheron's method-of-moments estimator (MME) and the Cressie and Hawkins estimator (CRE) [6]. The CRE estimator was chosen to deal with outliers and enhance the variogram's spatial continuity. An estimation of semivariance was carried out using a lag distance of 2 m. Table I and Table II show the parameters



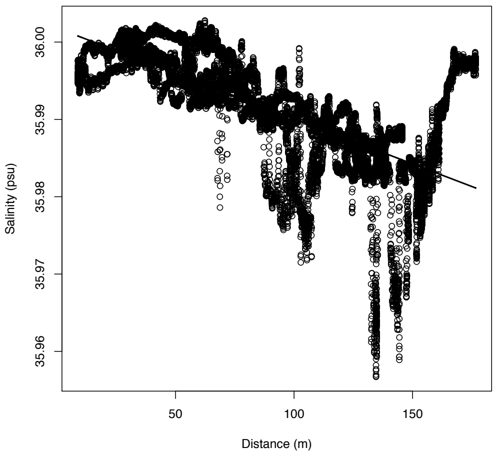


Fig. 2. Temperature measurements at depth of 1.5 m (top) and 3 m (bottom) versus distance to the middle point of the diffuser fitted by a linear model.

of the fitted models to the omnidirectional sample variograms constructed using MME and CRE estimators (for salinity measured at depths of 1.5 m the sample variogram constructed using CRE could not be fitted by any model). All the variograms were best fitted to Matern models. The range value (in meters) is an indicator of extension where autocorrelation exists. The autocorrelation distances are always larger for the CRE estimator (with the exception to temperature at depth of 1.5 m) which may demonstrate the enhancement of the variogram's spatial continuity. All variograms have very low nugget values which indicates that local variations could be captured probably due to the high sampling rate and due to the fact that the variables under study have strong spatial dependence. Anisotropy was investigated by calculating directional variograms. However, no anisotropy effect could be shown.

D. *Cross-Validation*

The block kriging method was preferred since it produced smaller prediction errors and smoother maps than the point

TABLE I

PARAMETERS OF THE FITTED VARIOGRAM MODELS FOR TEMPERATURE MEASURED AT DEPTHS OF 1.5 AND 3.0 M.

Depth	Variogram Estimator	Model	Nugget	Sill	Range
1.5	MME	Matern ($\nu = 0.2$)	0.000	0.001	453.7
	CRE	Matern ($\nu = 0.3$)	0.000	0.001	130.3
3.0	MME	Matern ($\nu = 0.3$)	0.000	0.0001	18.0
	CRE	Matern ($\nu = 0.3$)	0.000	0.00015	83.3

TABLE II

PARAMETERS OF THE FITTED VARIOGRAM MODELS FOR SALINITY MEASURED AT DEPTHS OF 1.5 AND 3 M.

Depth	Variogram Estimator	Model	Nugget	Sill	Range
1.5	MME	Matern ($\nu = 0.2$)	0.000	3.086	95.9
3.0	MME	Matern ($\nu = 0.2$)	0.000	1.522	35.2
	CRE	Matern ($\nu = 0.3$)	0.000	1.459	70.7

kriging. Using the 90% modeling sets of the two depths, a two-dimensional universal block kriging, with blocks of 10×10 m^2, was applied to estimate temperature at the locations of the 10% validation sets. The validation results for both parameters measured at depths of 1.5 m and 3 m depths are shown in Table III and Table IV. At both depths temperature was best estimated by the variogram constructed using CRE. Salinity at the depth of 3 m was also best estimated using CRE. The difference in performance between the two estimators: universal block kriging using the MME estimator (MUBK) or universal block kriging using the CRE estimator (CUBK) is not substantial.

TABLE III

CROSS-VALIDATION RESULTS FOR THE TEMPERATURE MAPS AT DEPTHS OF 1.5 AND 3 M.

Depth	Method	R^2	ME	MSE	RMSE
1.5	MUBK	0.9134	1.1910e-4	8.5402e-5	9.2413e-3
	CUBK[a]	0.9167	1.1348e-4	8.2147e-5	9.0635e-3
3.0	MUBK	0.8753	0.8940e-4	3.6141e-5	6.0117e-3
	CUBK[a]	0.8757	0.8868e-4	3.6045e-5	6.0038e-3

[a] The preferred model.

TABLE IV

CROSS-VALIDATION RESULTS FOR THE SALINITY MAPS AT DEPTHS OF 1.5 AND 3 M.

Depth	Method	R^2	ME	MSE	RMSE
1.5	MUBK[a]	0.9423	4.5058e-5	3.2216e-6	1.7949e-3
3.0	MUBK	0.8931	-6.8442e-5	4.1108e-6	2.0275e-3
	CUBK[a]	0.8973	-6.6000e-5	3.9511e-6	1.9877e-3

[a] The preferred model.

III. RESULTS

A. *Mapping*

Fig. 3 shows the block kriged maps of temperature on a 2×2 m^2 grid using the preferred models. Fig. 4 shows the block kriged maps of salinity on a 2×2 m^2 grid using the preferred models. In the 1.5 m kriged map the temperature ranges between 15.382°C and 15.525°C and the average value is 15.469°C (measured range 15.359°C–15.562°C and average 15.463°C). In the 3 m kriged map the temperature ranges between 15.432°C and 15.502°C and the average value is 15.466°C (measured range 15.393°C–15.536°C and average 15.469°C). We may say that estimated values are in accordance with the measurements since their distributions are similar (identical average values, medians, and quartiles). The difference in the ranges width is due to only 5.0% of the samples in the 1.5 m depth map (2.5% on each side of the distribution) and only 5.3% of the samples in the 3.0 m depth map (3.1% on the left side and 2.2% on the rigth side of the distribution). These samples should then be identified as outliers not representing the behaviour of the plume in the established area. In the 1.5 m kriged map the salinity ranges between 35.965 psu and 36.004 psu and the average value is 35.992 psu, which is in accordance with the measurements (range 35.957psu–36.003psu and average 35.991 psu). In the 3 m kriged map the salinity ranges between 35.984 psu and 36.004 psu and the average value is 35.996 psu, which is in accordance with the measurements (range 35.973psu–36.008psu and average 35.996 psu). As predicted by the plume prediction model, the effluent was found dispersing close to the surface. From the temperature and salinity kriged maps it is possible to distinguish the effluent plume from the background waters. It appears as a region of lower temperature and lower salinity when compared to the surrounding ocean waters at the same depth. At the depth of 1.5 m the major difference in temperature compared to the surrounding waters is about -0.116°C while at the depth of 3 m this difference is about -0.073°C. At the depth of 1.5 m the major difference in salinity compared to the surrounding waters is about -0.044 psu while at the depth of 3 m this difference is about -0.027 psu. It is important to note that these very small differences in temperature and salinity were detected due to the high resolution of the CTD sensor. [7] observed temperature and salinity anomalies in the plume in the order, respectively of -0.3°C and -0.1 psu, when compared with the surrounding waters within the same depth range. The small plume-related anomalies observed in the maps are evidence of the rapid mixing process. Due to the large differences in density between the rising effluent plume and ambient ocean waters, entrainment and mixing processes are vigorous and the properties within the plume change rapidly [7][8]. The effluent plume was found northeast from the diffuser beginning, spreading downstream in the direction of current. Using the navigation data, we could later estimate current velocity and direction and the values found were, respectively, 0.4 m/s and 70°C, which is in accordance with the location of the plume. Fig. 5 shows the variance of the estimation error (kriging variance) for the maps of temperature distribution at depths of 1.5 m and 3 m. The standard deviation of the estimation error is less than 0.03404°C at the depth of 1.5 m and less than 0.00028°C at the depth of 3 m. It's interesting to observe that, as expected, the variance of the estimation error is less the closer is the prediction from the trajectory of the vehicle. The dark blue regions correspond to the trajectory of MARES AUV.

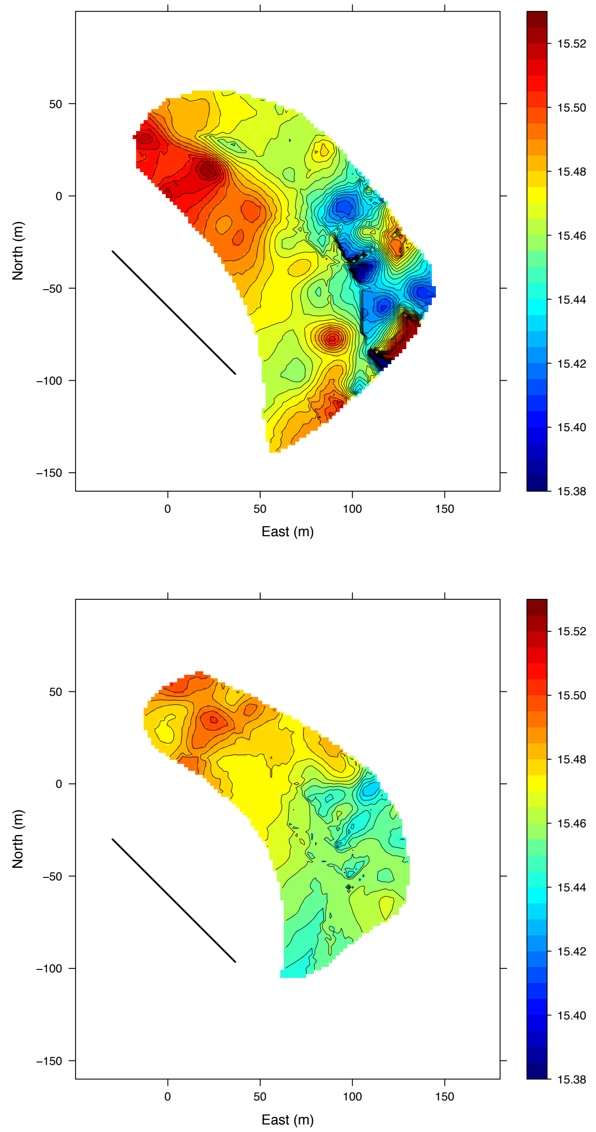


Fig. 3. Prediction map of temperature distribution at depths of 1.5 m (top) and 3 m (bottom).

B. *Dilution estimation*

Using salinity distribution at depths of 1.5 m and 3 m dilution was estimated according to [9] (see the contour maps in Fig. 6). The minimum dilution estimated at the depth of 1.5 m was 778 and at the depth of 3.0 m was 1503 which is in accordance with Portuguese legislation that suggests that

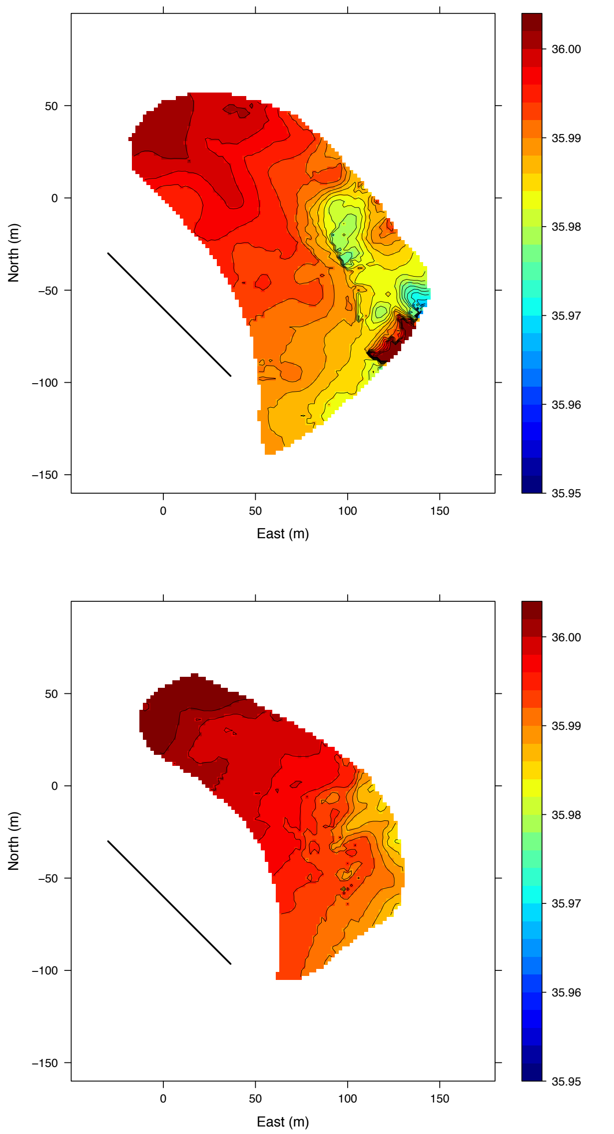


Fig. 4. Prediction map of salinity distribution at depths of 1.5 m (top) and 3 m (bottom).

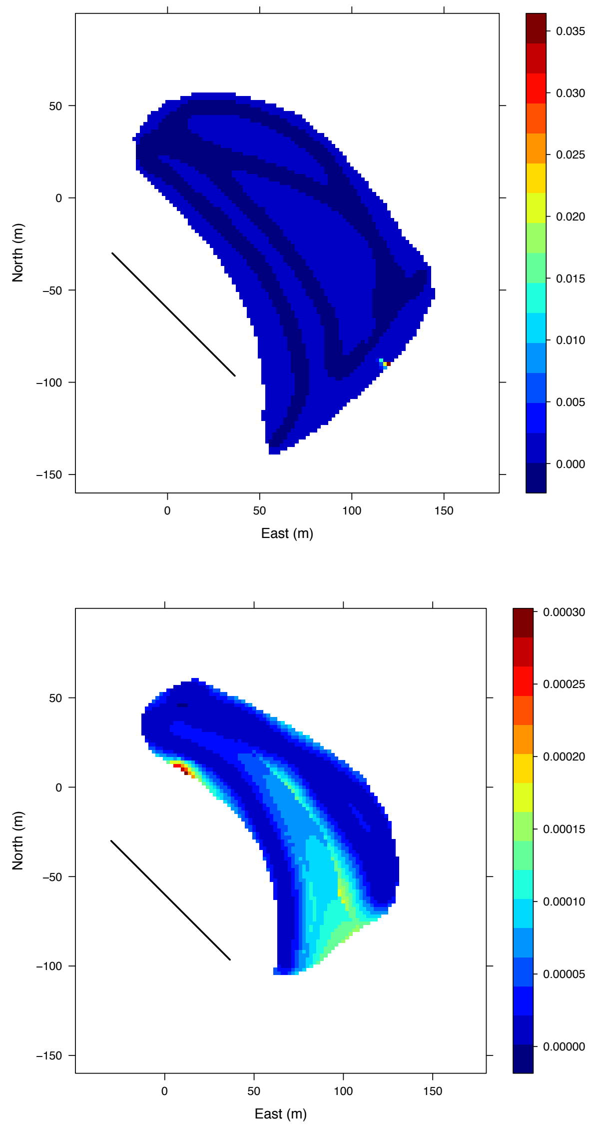


Fig. 5. Variance of the estimation error for the maps of temperature distribution at depths of 1.5 m (top) and 3 m (bottom).

outfalls should be designed to assure a minimum dilution of 50 when the plume reaches surface [10]. (Since dilution increases with the plume rising we should expect that the minimum values would be greater if the plume reached surface [5]).

IV. CONCLUSIONS

Through geostatistical analysis of temperature and salinity obtained by an AUV at depths of 1.5 m and 3 m in an ocean outfall monitoring campaign it was possible to produce kriged maps of the sewage dispersion in the field. The Matheron's classical estimator and Cressie and Hawkins' robust estimator were then used to compute the omnidirectional variograms that were fitted to Matern models. The performance of each competing model was compared using a split-sample approach. In the case of temperature, the validation results, using a two-dimensional universal block kriging, suggested the Matern model ($\nu = 0.3 - 1.5$ m and 3.0 m) with semivariance estimated by CRE. In the case of salinity, the validation results, using a two-dimensional universal block kriging, suggested the Matern model ($\nu = 0.2 - 1.5$ m and $\nu = 0.3 - 3.0$ m) with semivariance estimated by MME, for the depth of 1.5 m, and with semivariance estimated by CRE, for the depth of 3 m. The difference in performance between the two estimators was not substantial. Block kriged maps of temperature and salinity at depths of 1.5 m and 3 m show the spatial variation of these parameters in the area studied and from them it is possible to identify the effluent plume that appears as a region of lower temperature and lower salinity when compared to the surrounding waters, northeast from the diffuser beginning, spreading downstream in the direction of current. Using salinity distribution at depths of 1.5 m and 3 m we estimated dilution at those depths. The values found are in accordance with Portuguese legislation. The results presented demonstrate that geostatistical methodology can provide good estimates of the dispersion of effluent that are very valuable in assessing the environmental impact and managing sea outfalls.

REFERENCES

[1] J. Bellingham, "New Oceanographic Uses of Autonomous Underwater Vehicles," *Marine Technology Society Journal*, vol. 31, no. 3, pp. 34–47, 1997.

[2] X. Yu, T. Dickey, J. Bellingham, D. Manov, and K. Streitlien, "The Application of Autonomous Underwater Vehicles for Interdisciplinary Measurements in Massachusetts and Cape Cod Bayes," *Continental Shelf Research*, vol. 22, no. 15, pp. 2225–2245, 2002.

[3] P. Ramos and N. Abreu, "Using an AUV for Assessing Wastewater Discharges Impact: An Approach Based on Geostatistics," *Marine Technology Society Journal*, vol. 45, no. 2, 2011.

[4] N. Abreu, A. Matos, P. Ramos, and N. Cruz, "Automatic interface for AUV mission planning and supervision," in *MTS/IEEE International Conference Oceans 2010*, Seattle, USA, September 20-23 2010.

[5] C. D. Hunt, A. D. Mansfield, M. J. Mickelson, C. S. Albro, W. R. Geyer, and P. J. W. Roberts, "Plume tracking and dilution of effluent from the Boston sewage outfall," *Marine Environmental Research*, vol. 70, pp. 150–161, 2010.

[6] P. Kitanidis, *Introduction to geostatistics: applications in hydrogeology*, C. U. Press, Ed., New York (USA), 1997.

[7] L. Washburn, B. H. Jones, A. Bratkovich, T. D. Dickey, and M.-S. Chen, "Mixing, Dispersion, and Resuspension in Vicinity of Ocean Wastewater Plume," *Journal of Hydraulic Engineering, ASCE*, vol. 118, no. 1, pp. 38–58, Jan. 1992.

[8] A. A. Petrenko, B. H. Jones, and T. D. Dickey, "Shape and Initial Dilution of Sand Island, Hawaii Sewage Plume," *Journal of Hydraulic Engineering, ASCE*, vol. 124, no. 6, pp. 565–571, June 1998.

[9] H. B. Fischer, J. E. List, R. C. Y. Koh, J. Imberger, and N. H. Brooks, *Mixing in Inland and Coastal Waters*. Academic Press, 1979.

[10] INAG, *Linhas de Orientação Metodológia para a Elaboração de Estudos Técnicos Necessários para Cumprir o Artï¿½ 7ï¿½ do D.L. 152/97. Descargas em Zonas Menos Sensíveis*, Instituto da Água. Ministério do Ambiente, Lisboa, Março 1998.

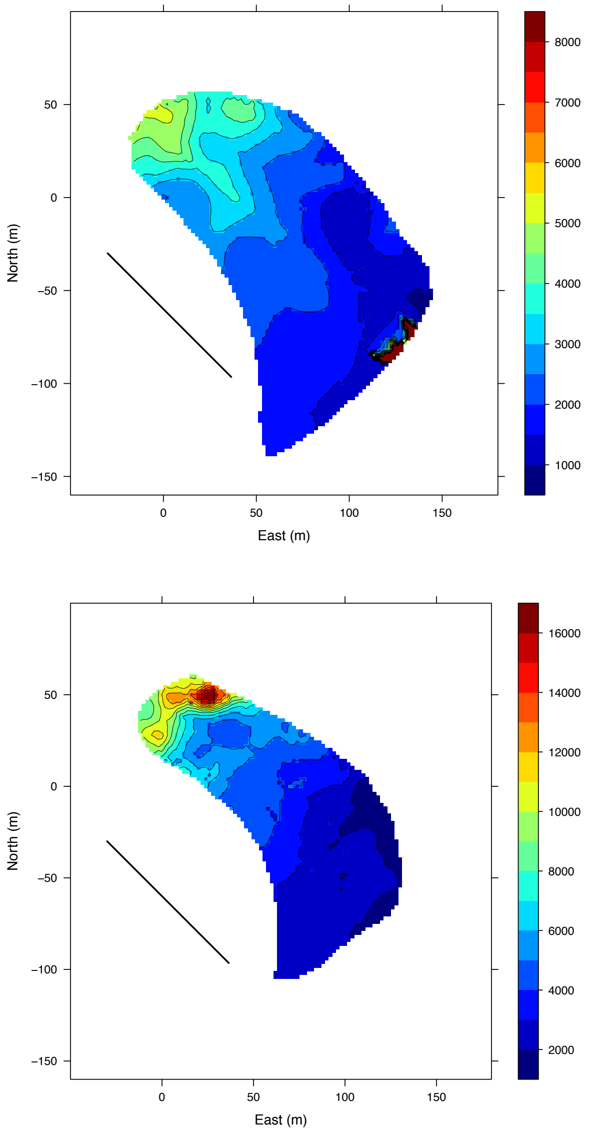


Fig. 6. Dilution maps at depths of 1.5 m (top) and 3 m (bottom).

Technologies for ASV and AUV Cooperation

R. Martins, P.S. Dias, Jose Pinto and J.B. Sousa

***Abstract*— Multiple AUVs can be deployed to quickly accomplish large scale ocean exploration missions. Due to lack of acoustic communication bandwidth and ranges, the AUVs have to depend on wifi to deliver/receive information. Often, the information is collected at the end of the mission which reduces mission situational awareness. In order to enable consistent situational awareness for long missions with frequent information updates, we present a cooperative framework where an autonomous surface vehicle (ASV) visits the AUVs and collects the information which it then transfers to the base station. Due to delay in transferring information from the vehicles to the base station and sending information from the base station to the vehicles, we have included a delay-tolerant network (DTN) capability into the system for smooth information collection and delivery. The proof of concept and initial experiments are presented to show that such a system will provide several benefits for ocean missions.**

I. Introduction

Autonomous underwater vehicles (AUVs) are used for many maritime applications ranging from civilian to military domains. Usually, these missions are of high endurance where, a set of way-points are generated by the human operators. These way-points are uploaded onto the AUVs and they perform the mission. Once, the mission is completed the base station which is a marine vehicle, then travels to the final way-point of the mission, picks up the vehicle, downloads and analyzes the data. In these missions, if the sensors are faulty or biased, or the human operators need to reassess data from a particular region, then the mission has to carried out again. Though, this problem is of low interest for small and low cost exploratory regions, it becomes highly expensive and tedious for missions like counter mine exploratory missions, deep sea exploration missions, etc, that cover tens of kilometers. This problem can be mitigated by intermediate interaction of the AUVs with the base station. However, the AUVs do not have the required acoustic communication bandwidth to interact.

Another approach is to allow AUV to surface periodically and interact with the base station using a wireless network. To communicate, either the AUVs have to travel towards the base station or the marine vessel has to travel close to the AUV location. In the former case, the AUVs spend significant amount of time traveling towards the vessel, while in the later case, the AUVs spend most of the time at the surface waiting for the vessel to arrive.

Department of Electrical and Computer Engineering, University of Porto, Portugal, 4200-465. rasmartins@gmail.com, {pdias,zepinto,jtasso}@fe.up.pt

Also, one can deploy moored bouys that can act as communication gateways. Even in this case, the AUVs need to travel close to the bouys for data transfer. Moreover, these bouys are static by nature and this transfers to one of the above problems. If we use drifting bouys then determining the location of the bouy becomes an issue in itself.

Designing a mechanism using any of the above approaches will not increase mission performance, where performance is measured in terms of the time taken to explore the complete region. To increase the mission performance while allowing the AUV to surface for shorter periods, we deploy a ASV that performs sorties periodically and meets the AUV at a predefined time and place. The ASV collects the data from the AUV, commands a new mission leg and returns to the base station. Thus, a coordination between ASV and AUV emerges from such a system. This system supports human-in-loop behavior to over see the operations with consistent situational awareness obtained through this coordination. Using ASV for coordinated missions is gaining popularity as a static data mule[1], [2]. We are interested in increasing the autonomy of ASV such that it increase mission performance.

The tools and technologies required to perform such cooperative missions need to be advanced at three stages. Primarily, the command and control system must be able to plan for multiple vehicles and provide situational awareness of the mission as and when the information arrives. Secondly, the network connectivity is intermittent, so the communication protocols must take this limitation into account. Thirdly, the cooperative algorithm that tasks the AUVs and the ASV taking communication delays and uncertainty of the mission into account. In this paper, we present the tools and technologies that are necessary to conduct cooperative mission for marine vehicles consisting of ASV and AUVs.

II. Related work

Achieving coordination between various types of unmanned vehicles have been addressed previously in the multi robot literature. However, most of the research is focused on developing theoretical and experimental frameworks for a single class of vehicles. For instance, cooperation between multiple unmanned ground vehicles [3], [4], [5], multiple aerial vehicles [6], [7], [8], and multiple underwater vehicles [9], [10], [11]. Research on cooperation between heterogenous class of vehicles has been limited to ground vehicles and aerial vehicles. Healey et al. [12] develop an algorithm for a stealth mission using a single UAV and ASV. This paper

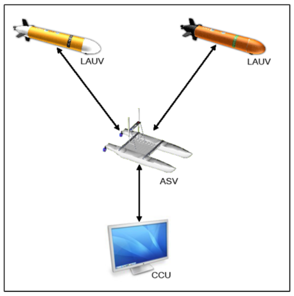


Fig. 1. Network for the operator with ASV and the AUVs

highlights that cooperation between aeria vehicle and marine vehicle is possible.

There is no adequate literature that uses a combination of ASV and AUVs for marine applications [2]. Initial research on using ASV and AUVs with priliminary cooperative mission capability support was developed by Martins et al. [13]. In this paper, the tools and technologies advance the exisiting mode of vehicle operation to achieve more autonomous operations with persistent situational awareness compared to that in [13].

III. System Components

A. Command and control center

One of the main components for autonomous missions using unmanned vehicles is the design of command and control center. We have designed a command and control center called as Neptus, that enables the operators to supervise and control the multi-vehicle network behavior. Neptus supports different phases of unmanned vehicle operation: planning, simulation, execution, revision and dissemination [14]. Neptus provides a plug-in architecture through which vehicle configurations (comprising appearance, networking settings and feasible behaviors), vehicle maneuvers, console widgets, plan generators and post-mission visualizations can be added. A snapshot of the Neptus is shown in Figure 2.

For Neptus, a plan is a graph whose nodes are maneuvers with respective parameters and edges are transitions with respective conditional guards and actions (as shown in Figure 3). This allows an easier access to generate sequential plans (transitions are taken when previous maneuver is complete) or also create more advanced plans with multiple conditional paths and actions that can encode behavior to be carried out by other vehicles, etc. These plans can either be edited manually by operators (with the aid of a planning widget that shows predicted behavior on a map) or can be generated automatically by plan generation plugins.

Plan generation plugins input a selection of vehicles and plugin-specific parameters and in return output one or more plans to be sent to the selected vehicles. We use this facility to automatically divide survey areas between available vehicles, to optimally generate plans to visit a set of points using one or more vehicles and also to generate standardized plans (following a certain pattern). There exists already several different plugins of this type:

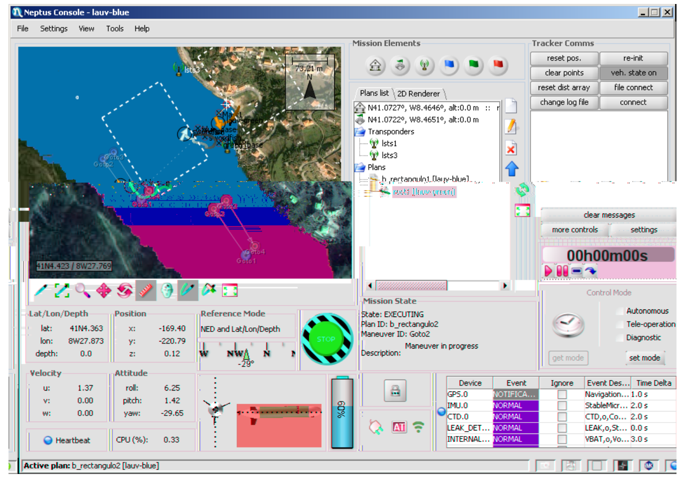


Fig. 2. Neptus - command and control center for multi vehicle deployments

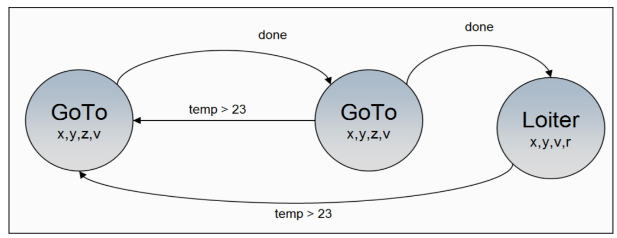


Fig. 3. Plan description

1) Templates – A plugin that will generate a plan from a pattern encoded as a script. This script may specify parameters that will be selected by the user prior to generation and also use any variables reported through the network (as vehicle positions, environmental sensor data, etc).
2) MVTSP – A plugin that, given a set of locations to be visited, will generate a list of plans for visiting all the points in minimal time, taking into consideration varying vehicle speeds.
3) MVCoverage – A plugin that inputs a geo-referenced polygon with an area to be surveyed by a set of vehicles and outputs a set of plans that divide this area by the selected vehicles according to a lawn-mowing algorithm.

Using the plan generation plugins and multi-vehicle control provided by Neptus, it is possible to automatically generate plans that encode cooperative behavior. Neptus allows real-time monitoring and control of multiple vehicles. One or more operator consoles can be connected to the network, visualize gathered data and re-plan vehicle behavior according to operator needs.

B. Intermittent communications

Since multi vehicle systems may be spread across large areas using limited communication means, we cannot rely on continuous communication. The underwater vehicles are disconnected from the base station and other vehicles for most part of their operation and have communication only when at surface. During information transfer, there can be disruption and also delay in information delivery and procurement at different vehicle locations. To facilitate smooth data processing and transfer between vehicles, we have included DTN capabilities. With DTN capabilities, vehicles can be used as data-carrying devices (or data mules) for control and monitoring of other systems even in disconnected networks.

C. Sub-system communications

The DTN is used for information gathering and delivery between vehicles. For communication between different sub-systems inside the vehicle, we developed Inter-Module Communication (IMC) protocol. It defines a modular and layered approach for control and sensing. The IMC protocol comprises of different logical message groups. Figure 4 illustrates the use of IMC within an underwater vehicle system. The message flow corresponds to the several control and sensing layers within IMC:

1) *Mission control messages* define the specification of a mission and it's life-cycle, for the interface between a CCU (Command and Control Unit) such as a Neptus [14] console and a mission supervisor module.
2) *Vehicle control messages* are used to interface the vehicle from an external source, typically to Neptus or a mission supervisor module
3) *Maneuver messages* are used to define maneuvers, associated commands and execution state. Some maneuver are related to waypoint tracking - encoded through a *Goto* message - or loitering patterns - *Loiter*, etc.
4) *Guidance messages* are related to the guidance law used for autonomous maneuvering. A guidance command generates new reference measures for the vehicle heading, depth, and velocity, in the form of a *Desired Guidance* message.
5) *Navigation messages* define the interface for reporting a vehicles navigation state. The *Estimated State* message defines a vehicles navigational state by the SNAME convention [15].
6) *Sensor messages* are used to report sensor readings by the respective hardware controllers. Sensor messages are related to sensor readings which can be from IMU, GPS, LBL (Long Base Line) acoustic positioning system, etc.
7) *Actuator messages* specify the interface with hardware actuator controllers. The actuators that impact in the LAUV guidance are the fins and the thruster, interfaced through the *Set Fin Position* and *Set Thruster Actuation* messages respectively.

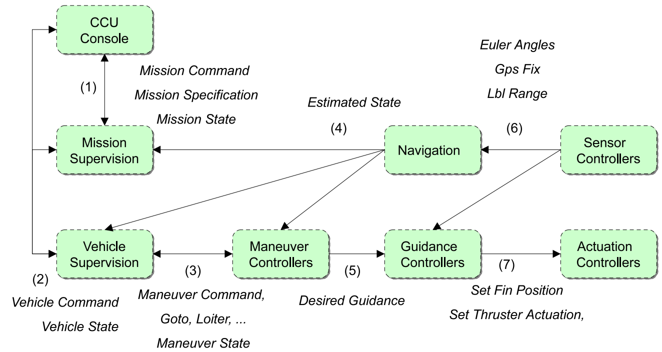


Fig. 4. IMC message flow for AUV.

This layered control and sensing infrastructure is in line with typical control infrastructure for autonomous vehicles, and enables modular development of applications. Software components can run in logical isolation, interfacing with other modules merely through the exchange of IMC messages. The control infrastructure for autonomous vehicles is implemented on-board using the DUNE framework [16], that enables message exchange using a *message bus* abstraction, and provides transport mechanisms for external communications. Vehicles can be monitored and controlled externally using Neptus consoles.

Networking of vehicles and consoles, is enabled through traditional IP-based communication-mechanisms, like raw UDP or TCP sockets, or by other means, such as the Real-Time Publish-Subscribe protocol, or underwater acoustic modems [17]. The drifting and static buoys being used are able to communicate data over long periods of time either by short distance multi-hop networking [18], by using ubiquitous GSM/GPRS communications, or also by communicating large bursts of stored data when a communication link can be established.

IV. Cooperative algorithms

Consider the scenario as shown in Figure 5 where two AUVs are deployed to carry out an ocean exploration mission. The AUVs have limited sensor range, limited communication range and low bandwidth. In order to carry out the mission efficiently a ASV is deployed that acts like a data mule between AUVs and the host. The mission is carried out in the following way: The AUVs are deployed with an initial path and surface time. Based on this information, the ASV will schedule a route to visit the AUVs. The ASV will collect the information from the AUVs and assigns a new path with an associated surface time to the AUV. The AUV will use the new path to explore. The ASV performs this sequence of actions to all the AUVs and then visit the base station to deliver the information within a prescribed time limit called as *Sortie time* (T).

The global objective of the mission is to minimize the total time taken in exploring the region taking human situational awareness into account. The mission completion time depends of two quantities (i) surface time and (ii) exploration time. The objective (i) emphasizes the fact that minimizing the surface time of the AUVs will allow them to explore for longer periods, thus enhancing the search performance. While objective (ii) ensures that the paths generated by the ASV are such that the AUVs spend their search effort on exploring the unknown regions than on explored regions. These two objectives aim at achieving the mission quickly and efficiently. Assume that the surface time of the AUV A_i for the j^{th} sortie is represented as S_j^i and similarly γ_j^i represents the exploration time for the j^{th} sortie. Then the objective and constraints for the mission can be written as:

$$\text{Objective}: \quad \min \sum_{j=1}^{M} \sum_{i=1}^{N} S_j^i \tag{1}$$

$$\max \sum_{j=1}^{M} \sum_{i=1}^{N} \gamma_j^i \tag{2}$$

$$\text{Constraints}: \quad T_j \leq L, \ \forall j \tag{3}$$

$$\gamma_j^i \geq \Delta \tag{4}$$

where T_j is the time taken to perform the j^{th} sortie visiting all the AUVs, L represents the sortie time limit and Δ represents the minimal exploration time that each AUV has to perform. The constraint given in Eq (3) forces the ASV to meet all the AUVs and visit the host within the Sortie time, thus updating the SA at the host vessel. The constraint given in Eq. (4) ensures that all the AUVs perform the exploration for a minimal exploration time and are not idle thus aiding the objective 2.

Sometimes the AUVs may not be able to complete the assigned task, in which case, they will surface. Hence, the ASV has to take this uncertainty into account during route planning stage otherwise, the ASV may not be able to generate a route satisfying the sortie time constraint.

A. Approach

To achieve the objective of minimizing mission time, we need to minimize the surface time and maximize the exploration time of the AUVs. To minimize the AUV surface time, we need to determine a solution to the ASV route planning problem meeting the constraints (3) and (4). While we need to design strategies for the AUVs that maximizes the exploration region to realize the second objective.

We assume that N AUVs are deployed in a region that have limited communication (r_c) and sensor ranges (r_s). We also assume that the AUVs are equipped with autopilots that enable the AUVs to autonomously navigate towards a desired way-point and surface when they reach the assigned way-point. The kinematic equations of the AUVs are:

$$\begin{aligned} \dot{x}_i &= v_i \cos \psi_i \\ \dot{y}_i &= v_i \sin \psi_i \\ \dot{\psi}_i &= k(\psi_i^d - \psi_i) \end{aligned} \tag{5}$$

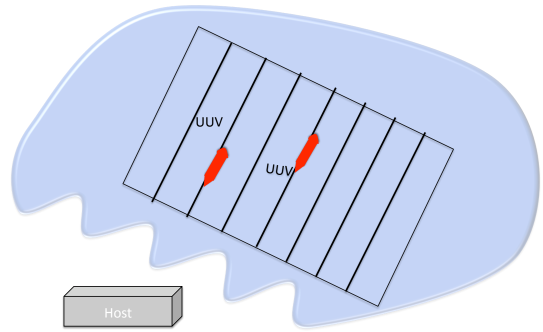


Fig. 5. Search region split into lanes with 2 AUVs performing exploration.

where v_i and ψ_i represent the velocity and the heading of the i^{th} AUV. The change in heading angle is constrained as:

$$-\omega_{\max} \leq \quad \dot{\psi}_i \quad \leq \omega_{\max} \tag{6}$$

The velocity of the AUVs varies between 0 and v_i. When the AUVs are in motion, their velocity is fixed to v_i, while the vehicle velocity is 0 when it surfaces. We assume that the autopilots present in the vehicles can handle the transitions and the depth controller that maintains the desired depth during the mission.

The ASV has kinematics similar to the AUV. However, it can travel at higher velocity than the AUV. The ASV also has limited communication range and it is assumed that the ASV is equipped with autopilot that navigates the ASV to the desired way-points generated by the AUV locations. We assume that the ASV has sufficient fuel to accomplish the mission.

B. Environment model

We model the environment of the exploration mission taking the AUV and ASV constraints into account. The AUVs have to explore the region using their sensor (sonar). The desired pattern for exploration missions using AUVs is the lawn moving pattern. We consider a rectangular search region $\mathcal{B}$ of width W meters and length L meters. We generate lanes on the search region $\mathcal{B}$ for the agents based on their sensor range r_c as shown in Figure 5. These lanes form the line of reference for the AUVs. Since the AUVs have same sensor range r_c, the number of lanes in the search space can be given as: $N_l = \frac{L}{2r_c}$. Each lane is represented as l_n, where $n = \{1, \ldots, N_l\}$.

C. Planning mechanism

The ASV has to visit all the AUVs and return to the base station within the sortie time T. When ASV meets A_i during the j^{th} sortie, then the ASV has to supply a path for the $j+1^{\mathrm{th}}$ sortie. After visiting the AUVs, the ASV returns to the base station. The path assigned to the AUV must be such that the ASV can meet A_i anywhere along the path without violating the sortie time constraint for the $j+1^{\mathrm{th}}$ sortie. Since, the ASV does not know the precise location of the other AUVs, it will estimate the possible locations that the other AUVs may have to travel from their initial and final way-points for at least Δ time units. Taking these constraints, the ASV generates a route that does not violate the sortie constraints given in equations 3 and 4.

The length of the path given to the AUV by the ASV can differ based on the locations of the other AUVs and the sortie time constraints. However, the minimum path length is of 1 time units. When the AUV completes the assigned lanes, the ASV takes the current state of the lanes and determines those lanes that need to be visited and assigns the AUV to the nearest unexplored lane.

V. Initial experiments

Previously, we conducted an experiment in river Douro, Portugal, as a proof of concept for enabling networked marine vehicles to function as team. The experiment consisted of two AUVs and a single ASV as shown in Figure V. The ASV visited the AUVs and delivered the data/commands from the command and control center.

The river width is small and all the vehicles were able to communicate with each other. However, to enable the system to work when communication range limitations are present, the information was routed through the ASV. Although, Neptus could communicate to the AUV directly, due to the restriction, it was communicating through the ASV. The operator tasked the AUV through ASV to perform a lawn moving pattern. Then the operator received continuous updates on the movement of the vehicle through the ASV.

This proof of concept did not have advanced technologies like the DTN for easier transformation of information without losing any data packets and the implementation of cooperative algorithms. Currently, the DTN technology has been implemented into the networked system and the present focus is to develop cooperative algorithm that can efficiently achieve a mission.

Currently, we have demonstrated the developed technologies that are necessary for multi-vehicle systems. Although, we have theoretically developed cooperative algorithms, we have not experimentally demonstrated the performance of the algorithm. We are currently, pursuing to implement the cooperative algorithms on our AUVs and ASV. The plan is to implement the setup on a 4 Sq. Km region at Porto Harbor in Portugal. We will report further develops and results of this experiment.

Fig. 6. Initial field experiment for coordination between heterogeneous vehicles

VI. Conclusion and future research

In this paper, we have presented the a cooperative system design for networked marine vehicles to achieve large scale missions. We have tested an initial system design for networking and with DTN. However, the cooperative algorithms have not been experimentally tested. We are in the process of deploying the algorithms. Also, new cooperative algorithms need to be developed for various other applications.

References

[1] J. Curcio, P. McGillivary, K. Fall, A. Maffei, K. Schwehr, B. Twiggs, C. Kitts, and P. Ballou, "Self-positioning smart buoys, the "un-buoy" solution: Logistic considerations using autonomous surface craft technology and improved communications infrastructure," in *OCEANS 2006*, sept. 2006, pp. 1 –5.

[2] C. C. Sotzing and D. M. Lane, "Improving the coordination efficiency of limited-communication multi-autonomus underwater vehicle operations using a multiagent architecture," *J. Field Robot.*, vol. 27, pp. 412–429, July 2010. [Online]. Available: http://dx.doi.org/10.1002/rob.v27:4

[3] B. Gerkey and M. Mataric, "Sold!: auction methods for multirobot coordination," *IEEE Transactions on Robotics*, vol. 18, no. 5, pp. 758–768, 2002.

[4] D. Vail and M. Veloso, "Dynamic multi-robot coordination," in *In Multi-Robot Systems: From Swarms to Intelligent Automata*, vol. II, 2003, pp. 87–100.

[5] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[6] P. Chandler, M. Pachter, and S. Rasmussen, "Uav cooperative control," in *Proc. of the American Control Conference*, Arlington, VA, June 2001, pp. 50–55.

[7] T. McLain and R. Beard, "Coordination variables, coordination functions, and cooperative timing missions," *AIAA Journal of Guidance Control and Dynamics*, vol. 28, pp. 150–161, 2005.

[8] A. Richards, J. Bellingham, M. Tillerson, and J. P. How, "Coordination and control of multiple uavs," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, Monterey, CA, August 2002, pp. AIAA–2002–4588.

[9] D. Stilwell, A. Gadre, C. Sylvester, and C. Cannell, "Design elements of a small low-cost autonomous underwater vehicle for field experiments in multi-vehicle coordination," in *Proc. of the IEEE/OES Autonomous Underwater Vehicles*, June 2004, pp. 1–6.

[10] R. Kumar and J. Stover, "A behavior-based intelligent control architecture with application to coordination of multiple underwater vehicles," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Cybernetics*, vol. 30, no. 6, pp. 767–784, 2001.

[11] P. Bhatta, E. Fiorelli, F. Lekien, N. Leonard, D. Paley, F. Zhang, R. Bachmayer, R. Davis, D. Fratantoni, and R. Sepulchre, "Coordination of an underwater glider fleet for adaptive sampling," in *International Workshop on Underwater Robotics*, Genova, Italy, 2005, pp. 61–69.

[12] A. Healey, D. Horner, S. Kragelund, B. Wring, and A. Monarrez, "Collaborative unmanned systems for maritime and port security operations," in *Proc. of the IFAC Conference on Control in Advanced Marine Systems*, Bol, Croatia, 2007.

[13] R. Martins, P. Dias, J. Pinto, P. Sujit, and J. Sousa, "Multiple underwater vehicle coordination for ocean exploration," in *Proc. of the International Joint Conference on Artificial Intelligence Exhibition and Workshop: Multi-Robot Teaming Challenge*, Pasadena, CA, July 2009.

[14] P. Dias, S. Fraga, R. Gomes, G. Goncalves, F. Pereira, J. Pinto, and J. Sousa, "Neptus-a framework to support multiple vehicle operation," in *Proc. of the IEEE Oceans*, June 2005, pp. 963 – 968.

[15] *Principles of Naval Architecture - 2nd revision*. Society of Naval Architects and Marine Engineers, 1989.

[16] "Seascout On-Board Software Distribution 0.1," 2007. [Online]. Available: http://whale.fe.up.pt/seascout/

[17] E. Marques, J. Pinto, S. Kragelund, P. Dias, L. Madureira, A. Sousa, M. Correia, H. Ferreira, R. Goncalves, R. Martins, D. Horner, A. Healey, G. Goncalves, and J. Sousa, "Auv control and communication using underwater acoustic networks," in *OCEANS 2007 - Europe*, june 2007, pp. 1 –6.

[18] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

Shallow water Lagrangian floats as versatile sensing and imaging platforms

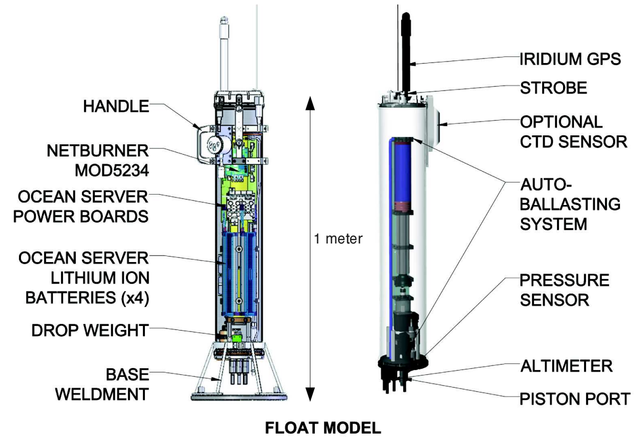
Chris Roman, Gabrielle Inglis, Connor Tennant

Abstract— **This paper presents results from recent work using a shallow water Lagrangian float for sea floor imaging and water column profiling. The float is a flexible platform that is able to carry a suite of sensors and position itself at any point in the water column using ballast control. The low cost of the float and minimal peripheral requirements for personnel and handling gear make it an attractive monitoring platform. Initial results are shown for a sea floor classification study using the float to collect detailed camera images close to the bottom and a routine phytoplankton study demonstrating flexible profiling and adaptive sampling in coastal waters.**

Introduction: The float (Fig. 1) is a low cost environmental monitoring tool designed to work near shore [1], [2]. It has a set of capabilities distinct from the more common open ocean float designs [3], and other small [4] and bottom stationing [5] shallow water platforms. It is optimized to operate on time scales of days rather than years, and depths to 100 meters rather than the 2000 meters typical of open ocean Argo floats. The float's displacement is controlled by a fast acting 450 mL piston type volume changing system. A motor, lead screw and piston change the float's buoyancy to actively move it up and down in the water column. The control system is based on a feedback linearization approach which compensates for the quadratic velocity of the drag force acting on the float when in motion. Shaped inputs are also used to generate depth reference signals the float is capable of following. A 200 kHz Airmar acoustic altimeter is used to measure the distance to the bottom at all times and has a working range of approximately 100 meters. This allows the float to perform constant depth or altitude drifting, and several profiling functions while controlling its vertical position in the water column. In regions of varying bathymetry the altimeter is required to keep the float from hitting the bottom and allows it to drift at a constant altitude off the bottom even as the total water depth changes. At the completion of a mission the float can drop a one kilogram expendable weight to gain more buoyancy on the surface. The weight will also be dropped during a mission by a dead man timer should the microcontoller have a fault or an excessive internal humidity threshold is detected. A GPS and Iridium system is used to record GPS when the float is on the surface and then report that position back via a satellite phone data message.

C. Roman is an Assistant Professor of Oceanography in the Graduate School of Oceanography at the University of Rhode Island, Narragansett, RI 02882 USA cnr@gso.uri.edu

G. Inglis and C. Tennant are students in the Department of Ocean Engineering at the University of Rhode Island, Narragansett, RI 02882 USA gabrielle.inglis@gmail.com and ctennant@my.uri.edu



(a)

(b)

Fig. 1. (**a**) CAD drawing of the shallow water float showing the major systems. (**b**) Photo of the float showing the CTD near the top and the stereo camera system.

The float is able to carry a suite of scientific sensors, including a Neil Brown Ocean Sensors Inc (NBOSI) flow through CTD, Aanderaa optical dissolved oxygen sensor, WET Labs ECO-Puck multi-wavelength fluorometer and custom packaged stereo cameras. The nominal performance of the float consumes between 10 and 40 watts of power, depending on the depth, while actively following the bottom or profiling. The float can run for approximately two days by itself and roughly 8 hours with the camera system taking images. The floats is single person deployable from nearly any boat and requires minimal handle gear. Programming and mission planning are done through a mission script of basic behaviors and parameters similar to most all autonomous underwater vehicle (AUV) mission interfaces.

Seafloor imaging: There are numerous methods for collecting visual images of the sea floor and related biota for

TABLE I

PLATFORM AND SURVEY SUMMARY

	Ship requirements	Personnel requirements	Depth limit	Navigational accuracy	Data Quality	Total cost
SCUBA	Low/dedicated	Moderate/operational	Shallow	Low/tracking	Low	Moderate
Drop cams	Moderate/dedicated	Low/operational	Moderate	Moderate/tracking	Low	Moderate
Tow sleds	Moderate/dedicated	Moderate/technical	Moderate	Moderate/survey	Moderate	Moderate
ROVs	High/dedicated	High/technical	Deep	High/survey	High	High
Manned subs	High/dedicated	High/technical	Deep	High/survey	High	High
AUVs	Moderate/dedicated	High/technical	Deep	High/survey	High	High
Lagrangian Floats	Low/standby	Low/technical	Deep	Moderate/tracking	High	Low

A summary of the typical requirements and costs associated with common survey methods for collecting sea floor images. The Lagrangian float has an attractive set of low cost and high quality attributes that are distinct from the others. The ship requirements categorized as *dedicated* either require special handling gear or full time tending to the operation. A *standby* vessel would be free to complete other tasks during operations and have minimal handling requirements. The personnel are classified as generally lower cost field *operational* or as higher cost *technical* persons with more specialized skill sets. The navigation is considered as *tracking* when it is applied mostly after the fact and *survey* when it is used in real time to offer more precision in the data collection.

benthic habitat studies and fisheries independent stock assessment. SCUBA divers, drop cameras, ship towed camera systems, manned subs, remotely operated vehicles (ROVs) and specialized autonomous underwater vehicles (AUVs) are able to collect near bottom images with varying levels of quality and precision (Table I). Each method also has specific costs in terms of personnel requirements, equipment expense, ship support and overall operational risk. Lagrangian imaging, which uses the float drifting at a constant controlled altitude while taking pictures of the sea floor (Fig. 2), fills a gap in current capabilities. Used in this manner the float is able to collect high resolution images, of comparable quality to those produced by AUVs, at a much lower price and effort point.

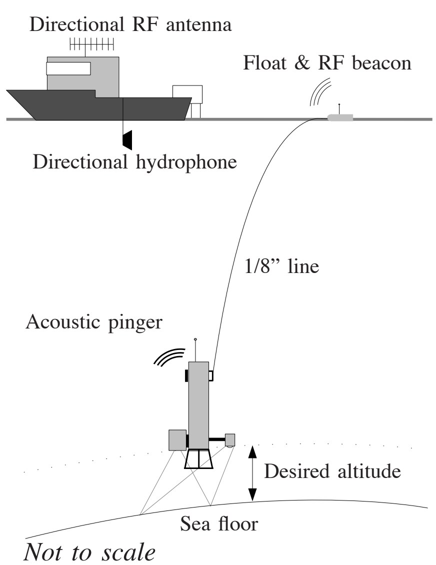
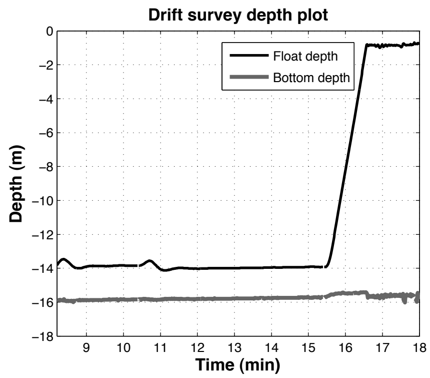


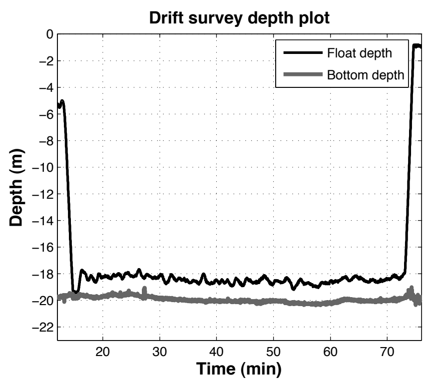
Fig. 2. Diagram of the basic float operation during our Lagrangian imaging tests. The float executes a constant altitude drift at a desired imaging distance from the seafloor using the acoustic altimeter for reference. The float is tracked with a tethered surface buoy, RF tracking and an acoustic locator. This will generally allow the support boat to work within several kilometers of the float during a dive and maintain tracking. When at the surface the float has a second RF beacon and a GPS/Iridium antenna for absolute positioning and communications.

The idea for Lagrangian imaging comes from the observation that data collection with current methods, particularly AUVs, can be broken apart into two essentially independent problems: (1) Getting from location A to location B and (2) knowing where locations A and B are. The first demands a vessel or vehicle capable of locomotion and the ability to travel between way points while maintaining a desirable imaging altitude. The second is the broader problem of underwater navigation, that requires generating an accurate estimate of the platform's location in real time or after the fact using some combination of local and external sensors. The Lagrangian imaging concept essentially removes the need for self propulsion while maintaining the same navigational requirements typical of any underwater vehicle platform. Forsaking propulsion allows the overall vehicle complexity and power requirements to be greatly reduced. This generates a significant cost saving in both the platform and the personnel required to support it. The options for navigation and tracking remain the same for any of the systems described above. The issue is determining the price the end user is willing to pay for the location information and what value it adds to the data. For some applications the value of the images will be completely contingent on the quality of the navigation, but in others it is not a significant factor worthy of the expense. For reconnaissance surveys, broad habitat assessment, and large area coverage there is a minimal need to either visit a specific square meter of the sea floor or identically cover a prior track line. In these cases images taken with the float using semi-structured drift surveys, planned from prior drifts or with the aid of circulation and tidal models, would suffice to cover the survey area of interest for far less cost.

Sample constant altitude drifts from imaging surveys are shown in Fig. 3. In calm seas the float can nominally track a depth or altitude reference to within 5 cm. In rougher seas (Fig. 3(b)), where the pressure oscillations from surface waves are observable by the depth sensor and the surface tether tied to a marker buoy is tugging, the float typically stays within 30 cm of the reference. The reference altitude can be as close as 80 cm from the bottom, enabling detailed photos in moderately turbid conditions. This low altitude capability also lets the float image closer to the sea floor than torpedo style AUVs will comfortably operate due to the risk of bottom collisions. Altitudes less than 80 cm can generate erroneous altimeter data and corrupt the reference trajectory.



(a)

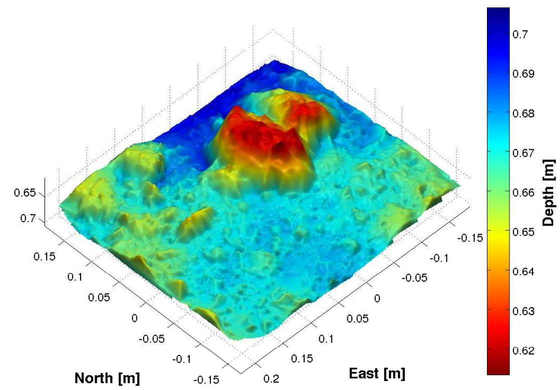


(b)

Fig. 3. Sample constant altitude missions. (**a**) The constant altitude sections of a test mission in perfectly calm seas with no wind. The reference altitude switched from 2.0 meters to 1.8 meters at 10.5 minutes. (**b**) A sample depth plot for an altitude drift at 1.5 meters from the sea floor. The surface conditions, with waves up to one meter and 10-15 knot winds, affected the tethered float and contributing to the depth oscillations.

The stereo camera system consists of one black & white and one color camera made by Prosilica. Each have 12-bit 1360×1024 resolution with fixed focus 9mm lens. The cameras are separated by 12 cm and have a nominally 30×40 degree field of view each. The strobe is a standard Vivitar flash placed in a small glass sphere. The cameras and strobe are controlled by a Linux PC-104 computer in the camera housing. They can be run independently of the float and set to take images at a fixed rate, typically 3 to 5 seconds per image, or in a timed sequence with alternating rates. Due to the varying turbidity and ambient lighting in shallow coastal waters we use the auto exposure feature of the cameras to provide a level of robustness and improve image consistency. The camera housing also has an OceanServer OS5000 compass to orient the images to north and estimate the rotation rate of the float as it is drifting.

The stereo images can be used to create texture mapped bathymetry, allowing us to resolve objects 3 cm and larger in size (Fig. 4). Here we take a simple sparse approach that triangulates pairs of Scale Invariant Feature Transform (SIFT) points [6] automatically extracted and matched from the paired images. The triangulated 3D points, $\sim$5000 per image pair, can then be displayed as a connected mesh (Fig. 4(a)) or as a texture mapped image (Fig. 4(b)). The resulting data products portray shape characteristics of the sea floor and can be used to estimate rugosity and roughness [7].



(a) Bathymetry

(b) Texture mapped image

Fig. 4. Sample image results. (**a**) Bathymetric section constructed from a stereo image pair. (**b**) Texture mapped image over the surface bathymetry.

Overlapping images can also be merged together into single strip photomosaics (Fig. 5). The mosaics can be oriented using the collected compass data and georeferenced using the surface GPS locations or any other additional underwater tracking data. An approximate scale for each image, or the entire photomosaic, can be set using the stereo reconstructions or more simply by the camera field of view and the altimeter data.

Additional estimates of the drift speed and direction are derived using visual odometry. Such near bottom current estimates provide a useful additional piece of information from the drift survey and are otherwise hard to obtain without more expensive instrumentation, such as a bottom tracking Doppler. We have implemented a simple least squares estimate of the drift motion using the movement of corresponding feature points between temporally adjacent image pairs overlapping approximately 30% or more (Fig. 6(a)). To help insure adequate overlap to perform visual

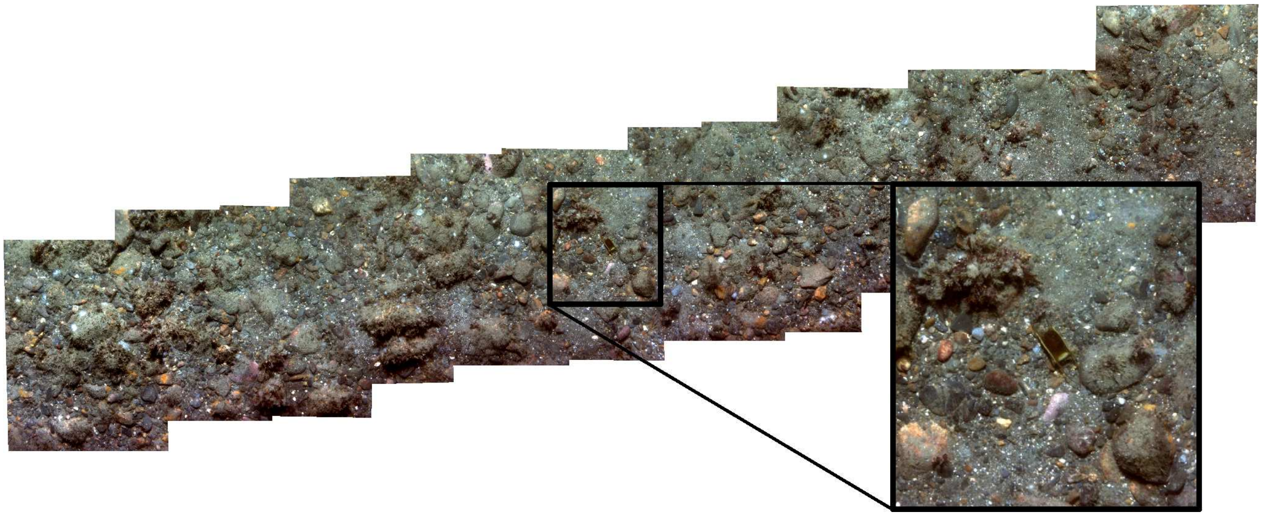


Fig. 5. Sample 12 image mosaic along a drift track. The images were taken at 1.3 meters altitude. The skate egg case, likely *Leucoraja sp.*, in the highlighted section is approximately 8 cm in length.

odometry we use an alternating image timing sequence (Fig. 6(b)). The sequence considers one frame rate for general survey images and periodic bursts of images at a faster rate for odometry. The period between burst sequences can be set by the operator depending how often current estimates are desired. The burst images are taken at the reflesh limit of the strobe, which is about 2 seconds for our current system.

Future work for the Lagrangian imaging concept will utilize in-situ image processing to improve the operational efficiency of the float surveys and produce more consistent results. Real time image processing for visual odometry will provide a measure of the float's actual drift speed during the survey. This could then be used to automatically set the camera frame rate to achieve a desired image coverage or image spacing on the sea floor. The user would then be able to set the image spacing that best fits their monitoring needs rather than setting the frame rate based on initial estimates of the current speed.

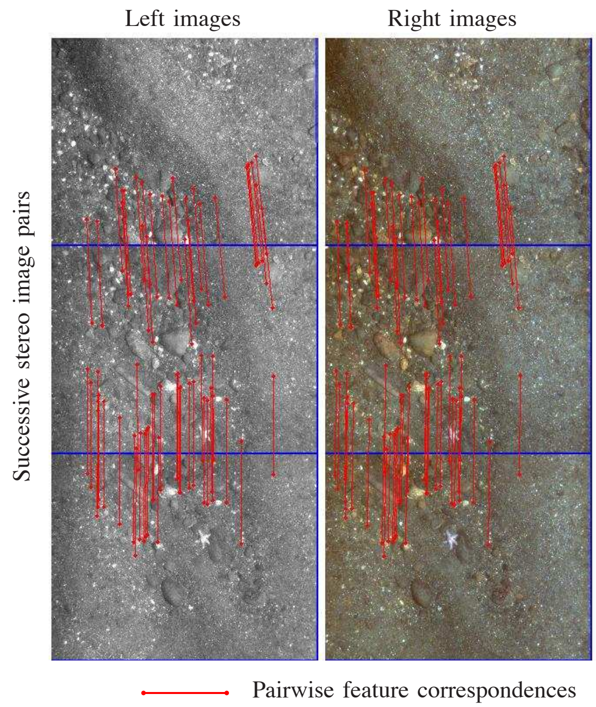
We have also begun work on automated image bottom detection that is able analyze images to determine if they are of sufficient detail in the presence of turbidity. In coastal regions the water quality can vary significant from place to place. This algorithm can verify that the images collected at the specified height above the sea floor are not contrast or clarity limited. If the image quality is poor the float would then be able to lower its altitude, within an acceptable range, to the acquire better images. Such a capability would also contribute greatly to survey efficiency and provide more consistent results.

For greater overall flexibility we would also like to include a higher rate LED strobe. At such low altitudes the current strobe rate, between 3 and 5 seconds, can be a limiting factor in obtaining sufficient image overlap in currents better than a knot. The LED strobe would offer higher rates and more overlap with reduced power consumption.

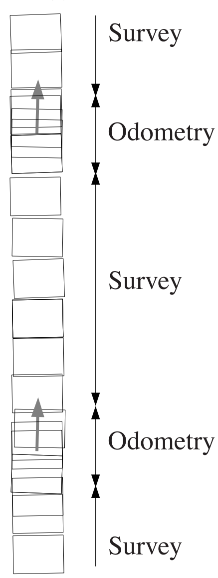
Water column profiling: The float has also been used for water column profiling with its suite of environmental sensors. The float is able to execute controlled vertical profiles at speeds ranging from zero to 10 $\mathrm{m/min}$. The profiles are smooth, decoupled from ship heave that would adversely affect wire hung sensors and reactive to changing bottom contours. A short sample profile mission from Narragansett Bay Rhode Island is shown in Fig. 7. The float is currently being used on a weekly plankton survey at a station in the bay as part of a 50 year long time series [8], [9]. The survey, which involves net tows and water sampling, takes just 20 minutes to complete each week. During the survey the float is quickly deployed and executes several profiles in the shallow water. Figure 7(b) shows the presence of a phytoplankton layer [10], [11] in the fluorometer signal across multiple profiles. Using the surface buoy with a thin spectra tether allows the float to be recovered quickly at the end of the survey. In total the float adds minimal additional time to the survey while providing a complimentary data set of profiles.

The float has also been used on Georges Bank to sample the North Atlantic spring plankton bloom. In this case the float was deployed several days in a row for between six and eight hours each day. During the dives the float was followed by the RV Endeavor. Following the float as it profiled allowed the ship to track a tidally driven water mass moving along the edge of the bank. Shipboard CTDs and water samples were periodically taken and compared to the float profile data. The float data have been compared to both the ship CTD data and collected water samples to calibrate the fluorometer measurements.

Our most recent work with the float has focused on adaptive profiling for features of interest in the water column, such as the thermocline, a halocline or a thin plankton layer. This capability is broken into three basic phases. First, the float completes a series, typically four to eight, vertical profiles through the whole water column or between two specified depths. During the profiles the float records the sensor data and computes running averages for the parameters in depth bins (Fig. 8). Due to the position of the sensors on the top of the float, only the data from upward profiles is used for the averages. Downward profiles tend to be biased by water being pulled down with the wake

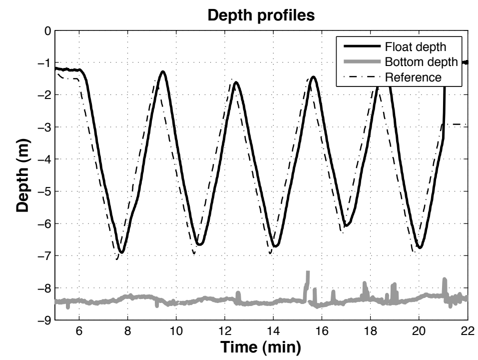


(a)

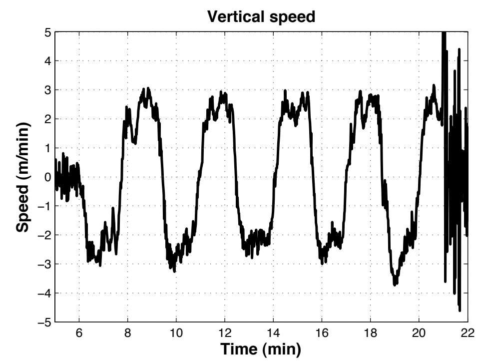


(b)

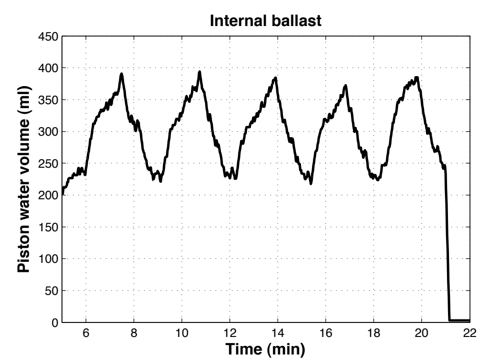
Fig. 6. An example of visual odometry. (**a**) Three left and three right images showing feature correspondences between successive overlapping image pairs. The correspondences can be used to calculate the float velocity and drift track for in-situ current estimates. (**b**) Illustration of the alternating survey-odometry timing sequence. Periodic bursts of high overlap images can be used for current estimation by visual odometry.



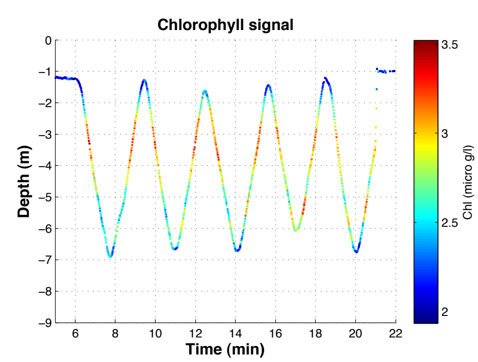
(a)



(b)



(c)



(d)

Fig. 7. A sample float mission. (**a**) Sample depth profile from a weekly plankton survey in Narragansett Bay. (**b**) Vertical profile speed showing no heave reversals. The start and end of the plot show the float on the surface moving in the waves. (**c**) The piston volume change during the profiles, with zero indicating all water expelled from the cylinder (**d**) Chlorophyll signal measured during the profiles showing a layer of plankton.

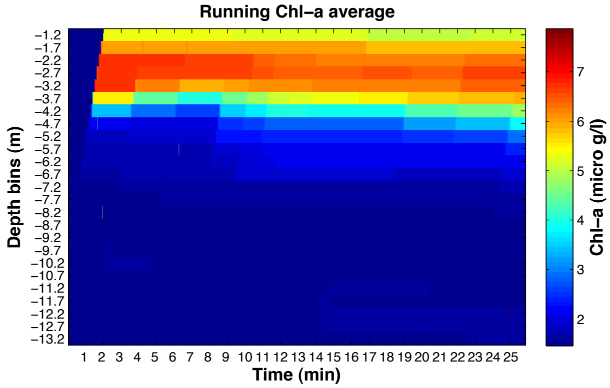


Fig. 8. Running averages of chlorophyll calculated for 50 cm depth bins during a sample mission in Narragansett Bay.

created by the bottom of the float. During upward profiles the sensors are in undisturbed flow. The bins are user specified and typically between 20 and 50 cm thick. The lower limit on the bin thickness is related to the sensor sample rates and the vertical profile speed of the float. Using profile speeds between 3 and 5 m/s the 5 Hz CTD and the 1 Hz fluorometer collect 10's to 100's of readings per bin during the initial full water column profiles. Next, using the bin averages the gradient of any parameters, such as temperature or conductivity, are calculated with simple finite difference. Using this information and conditions set by the user for identifying a feature of interest, such as the peak fluorescence or peak temperature gradient, a new desired depth is determined. Lastly, the float then moves to this depth and completes either a constant depth drift or smaller "narrow banded" profiles in the region of the water column feature. This cycle can then repeat at a prescribed time interval, allowing the float to continually position itself for finer sampling in a dynamic feature of interest. A sample mission to determine a layer of peak chlorophyll fluorescence is shown in Figure 9.

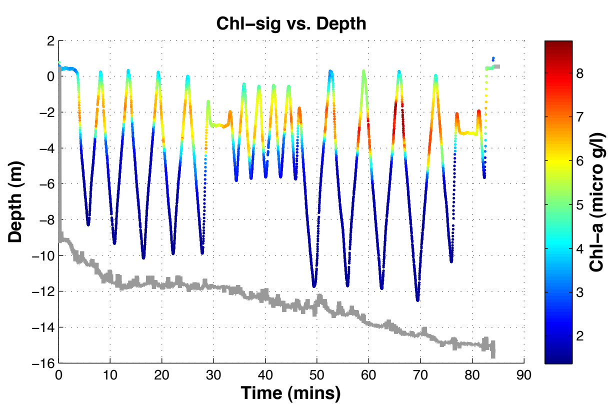


Fig. 9. An adaptive profiling mission looking for a peak in Chl-a fluorescence. The float performs a series of full water profiles, a short drift at the depth showing the peak average Chl-a signal and then smaller profiles about this depth. The cycle then repeats starting a second sequence of full water depth profiles in deeper water. The full profiles were set for the shallower depth of either 15 m or 1.5 m from the bottom.

Conclusion: The presented data confirm the utility of the shallow water float for collecting high quality sea floor images and executing water column profiling. The low cost and ease of deployment makes the float an attractive alternative to other platforms in many applications. Given the demand in application areas such as marine fisheries stock assessment, routine monitoring of marine protected areas and general coastal water quality there is a significant need for such a platform.

For sea floor imaging we envision using the float in conjunction with other survey tools such as towed side scan sonar systems and survey capable AUVs. These systems, which can be used for large scale and structured surveys to produce baseline maps prior to the float surveys. The low cost and flexibility of the float then makes it a cost effective platform for ground truthing areas of interest and performing routine monitoring to assess changes that would motivate additional detailed survey work. In this way the total cost of repeated monitoring with large scale surveys with expensive platforms can be reduced.

For water column profiling and adaptive surveys we are currently improving the capabilities of the system and tuning the parameters for specific survey goals. Settings for the profile rate, depth bins, the duration of profiles and the specific adaptive behaviors can all be related to the dynamics of the processes of interest. We have successfully used the system in Narragansett Bay during the summer months, where the thermocline between 3 and 6 meters depth is generally persistent and related to the vertical structure of phytoplankton. Future work will use the float year round in the Narragansett Bay and again on Georges Bank.

REFERENCES

[1] A. Schwithal and C. Roman, "Development of a new Lagrangian float for studying coastal marine ecosystems," in *MTS/IEEE Oceans*, Bremen, Germany, 2009.

[2] B. McGilvray and C. Roman, "Control System Performance and Efficiency for a Mid-Depth Lagrangian Profiling Float," in *MTS/IEEE Oceans*, Sydney, Australia, 2010.

[3] T. D. Dickey, E. C. Itsweire, M. Moline, and M. J. Perry, "Introduction to the Limnology and Oceanography Special Issue on Autonomous and Lagrangian Platforms and Sensors (ALPS)," *Limnology and Oceanography*, vol. 53, no. 2, pp. 2057–2061, 2008.

[4] J. S. Jaffe, R. Glatts, C. Schurgers, D. Mirza, P. Franks, P. Roberts, and F. Simonet, "AUE: An Autonomous Float for Monitoring the Upper Water Column," in *OCEANS 2007 - Europe*, 2007.

[5] C. Langebrake and E. Lembke, "Design and Initial Results of a Bottom Stationing Ocean Profiler," *MTS/IEEE Oceans*, pp. 98–103, 2002.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, p. 91, Nov. 2004.

[7] C.-C. Wang, B. Hefner, and D. Tang, "Evaluation of Laser Scanning and Stereo Photography Roughness Measurement Systems Using a Realistic Model Seabed Surface," *IEEE Journal of Oceanic Engineering*, vol. 34, no. 4, pp. 466 –475, October 2009.

[8] D. M. Pratt, "The Phytoplankton of Narragansett Bay," *Limnology and Oceanography*, vol. 4, pp. 425–440, 1959.

[9] T. Smayda and D. Borkman, "Nutrient and Plankton Dynamics in Narragansett Bay," in *Science for Ecosystem-based Management*, A. Desbonnet and B. A. Costa-Pierce, Eds. Springer, New York, 2008, pp. 431–484.

[10] M. A. McManus, O. M. Cheriton, P. J. Drake, D. V. Holliday, C. D. Storlazzi, P. L. Donaghay, and C. F. Greenlaw, "Effects of physical processes on structure and transport of thin zooplankton layers in the coastal ocean," *Mar. Ecol. Prog. Ser.*, vol. 301, pp. 199–215, 2005.

[11] T. J. Cowles, "Planktonic layers: physical and biological interactions on the small scale," in *Handbook of Scaling Methods in Aquatic Ecology: Measurements, Analysis, Simulation*. Boca Raton, FL.: CRC Press, 2004.

Adaptive Marine Monitoring via Sensor Web Enablement

Claire D'Este, Andrew Terhorst, Greg Timms, John McCulloch, and Chris Sharman
Tasmanian ICT Centre, CSIRO
Castray Esplanade, Hobart, Australia 7000
claire.deste@csiro.au

***Abstract*—Making use of publicly available real-time sensor data from other organisations allows an autonomous vehicle to significantly augment its situation awareness at low cost. We demonstrate how an autonomous catamaran for monitoring the marine environment can exploit the sensor web to help differentiate between events and sensor faults in a static sensor network, as well as plan more energy efficient paths.**

I. Introduction

Many organisations may be providing real-time sensor data, historical observations and model forecasts within our environmental monitoring region of interest. Tapping into these resources can allow autonomous vehicles to sense phenomena far removed from its current location. This information can be exploited for adaptive sensing, and efficient path planning.

The Tasmanian Marine Analysis Network (TasMAN) project has an autonomous surface vehicle (ASV) and an autonomous underwater vehicle (AUV) for monitoring the waterways of South-East Tasmania (43°S, 147°E). Within this same area, at the time of submission, there were real-time observations from; marine sensors at 9 locations from two different organisations, river flow observations at two locations, 21 weather stations from three different organisations, and hydrodynamic physical model forecasts. All are available in standardised sensor web services implemented to enhance accessibility and interoperability. This paper describes initial work that uses information from the sensor web to help determine if anomalies in marine sensor readings are due to equipment faults or actual environmental events. We also investigate how wind direction and speed from weather stations may help to plan energy efficient paths.

There are many organisations who do not have the budgets to perform large-scale monitoring, but may want to place their own specific sensing needs into a larger context. Cooperative sensing can increase the knowledge of environmental systems for all involved parties. By augmenting the situation awareness of a marine vehicle we are able to look beyond what is possible on board the device itself. This kind of functionality is often achieved by using multiple marine vehicles. However these vehicles can be expensive to purchase and deploy. By making use of observations that are already available we can achieve greater situation awareness at much lower cost. This also allows us to include observations from external parties, which means we are not responsible for the purchase of hardware and maintenance of all the sensor platforms. Static marine sensor nodes can be very expensive to maintain, because of biofouling, so there is great benefit in sharing this load between multiple organisations. We can also make use of data that lays outside of our organisation's skill set, for example the model forecasts.

Although there is not currently adequate indexing to sensor information to perform true sensor discovery, the purpose of this work is to demonstrate what can already be achieved when multiple organisations adhere to sensor web standards. We are also interested in how we can use the augmented sensor capabilities of a sensor web enabled autonomous vehicle.

A. The Sensor Web

Advances in sensor technology and distributed computing, coupled with the development of open standards that facilitate sensor/sensor network interoperability, are contributing to the emergence of a phenomenon known as the 'Sensor Web'[11]. This phenomenon can be described as an advanced Spatial Data Infrastructure (SDI) for [near] real-time situation awareness.

Sensor Web Enablement (SWE) is an Open Geospatial Consortium (OGC) initiative that extends the OGC open web services framework [3] by providing additional services and encodings for integrating web-connected sensors and sensor systems. SWE services are designed to enable *discovery* of sensor assets and capabilities, *access* to these resources through data retrieval and subscription to alerts, and *tasking* of sensors to control observations [3]. SWE enables interoperability between heterogeneous sensors, simulation models and decision support systems.

The SWE initiative has developed specifications for modelling sensors and sensor systems (SensorML), observations from such systems (Observations and Measurements) and processing chains to process observations (SensorML) [2], [5]. The SWE specifications provide semantics for constructing machine-readable descriptions of data, encodings and values, and are designed to improve prospects for plug and play sensors, data fusion, common data processing engines, automated discovery of sensors, and utilisation of sensor data. SWE currently provides four types of web services: Sensor Observation Service (SOS), Sensor Alert Service (SAS), Sensor Planning Service (SPS) and Web Notification Service (WNS) [20], [12], [18], [19].

The SOS provides a standard interface that allows users to retrieve raw or processed observations from different sensors, sensor systems and observation archives. The SAS provides a mechanism for posting raw or processed observations from sensors, process chains or other data providers (including a SOS) based on user-specified alert/filter conditions.

When subscribing to a SAS, users not only define the alert conditions but also the communication protocol for disseminating alerts via the WNS. The WNS provides a standard interface to allow asynchronous communication between users and services and between different services. A WNS is typically used to receive messages from a SAS and to send/receive messages to and from a SPS. The SPS provides a standard interface to sensors and sensor systems and is used to coordinate the collection, processing, archiving and distribution of sensor observations. Users can submit a specification of an observation requirement via the SPS. This specification is encoded using SensorML. Application logic sitting behind the SPS interface translates the specification into a set of instructions mission planning software can understand, assess the feasibility thereof, and then execute.

B. Related Work

Sensor web enabled systems have been used to exchange sensor data and increase autonomy in a distributed set of sensing devices on land, water and air [25]. However, these vehicles used data from within one project rather than attempting to use whatever relevant data could be found. Projects such as GeoCens [10] will soon provide a search engine for sensor observations; crawling the web for sensor web enabled data sources. It will then be even easier to discover relevant information about your monitoring area.

Tenorth *et al.* describe a method for extracting relevant textual information from the web in order to perform household tasks [23]. The system transfers the text into reasoning steps and actions for the robot. Waibel *et al.* use linked data and semantic web standards so that robots can create, share and reuse knowledge [26].

II. Robot Platform

For these initial experiments our ASV is the preferred platform as it gives us continuous communications, as well as the ability to carry more powerful computing, and heavier sensor payloads.

Our ASV 'StrayCat' is a 5.5m Hobiecat catamaran. The rigging has been removed and two independently controllable 80 lb thrust MinnKota electric thrusters have been fitted to the rudders allowing directional thrust to be applied using an electric tiller. Power is supplied through deep cycle lead-acid batteries. An embedded microprocessor implements a closed loop controller for the tiller position, interfaces to a R/C unit, and serves as the interface for the three actuators. The sensors included on the platform include an IMU, Laser scanner, and a suite of Raymarine sensors consisting of wind speed and direction, water depth and temperature, GPS, compass, and speed through the water. The platform also includes multiple I/O connectors that provide power and communications for scientific payloads as required. Previous missions have included sensors for salinity, pH, turbidity, pressure, and fluorescence. These I/O connectors have been standardised and are also provided on the TasMAN project's Starbug AUV 'Searise' [6].

The control system is running on a low power dual core embedded Advantech PC. A 3G modem has been fitted to allow remote monitoring and communication with web services from external parties. A local Wi-Fi network is established for real-time visualisation and near by supervision. The ROS software architecture [15] is used, with the mission and low level controllers loosely based on the Limnobotics catamaran at the University of Zurich [14].

Fig. 1. Straycat: our autonomous surface vehicle

Straycat has so far been used to validate biogeochemical models from the same area using an on-board fluorometer to sense chlorophyll levels as well as turbidity.

III. Data Sources

Figure 2 shows the relevant environmental data sources we discovered available in South East Tasmania at this stage of the experiments. These sources provide additional real-time sensing information to the catamaran.

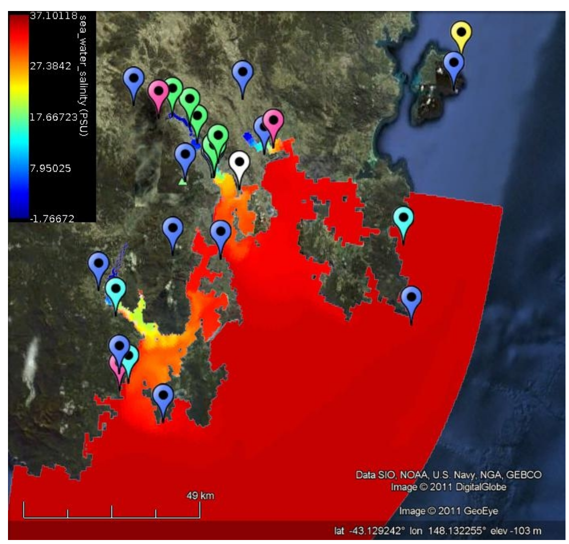


Fig. 2. Sensor Web services in South East Tasmania (42°S, 147°E): Green markers = TasMAN, white markers = CMAR, blue markers = BOM, pink markers = DPIPWE, yellow marker = NRS, cyan markers = Forestry, colour scale = hydrodynamic model forecast for sea surface temperature. Distance scale is in bottom left = 49km

A. CSIRO

The TasMAN low-cost sensor nodes are equipped with temperature, conductivity, and pressure sensors at different depths in the water column. The node may carry optional dissolved oxygen and fluorometer sensors. They are solar powered and communicate via 3G modems and do not require sink nodes or gateways [8]. We have chosen to use existing telecommunication networks because of the sparse nature of the network, as well as exploiting the advantages of using carrier grade telemetry. For a 10 metre sensor string with conductivity and temperature sensors at every metre and a pressure sensor at the bottom, the approximate costs are USD$2500 for a buoy-based node and USD$1500 for a node attached to existing infrastructure. The real-time data is made publicly available through web and mobile applications for science, government and industry; including nodes that are no longer deployed [22].

CSIRO Marine and Atmospheric Research (CMAR) operate two weather stations in South East Tasmania with sensing for air temperature, air pressure, rainfall, wind speed and direction.

We also have access to the historical time series for the sensor nodes. If we detect an anomalous reading then we can search for similar patterns using data mining techniques. In this case we have implemented Dynamic Time Warping [16]. If this pattern has occurred before then it provides more evidence that this is a marine event rather than a sensor fault.

B. Bureau of Meteorology

We also extract data from a service under trial by the Australian Bureau of Meteorology (BOM) [4], which provides 14 weather stations in South East Tasmania with observations for air temperature, air pressure, relative humidity, rainfall, wind speed and direction.

C. Water Resources Observation Network

CSIRO's Water for a Healthy Country flagship is investigating how emerging standards for Sensor Web Enablement (SWE) being developed by the Open Geospatial Consortium (OGC) can contribute to enhanced situation awareness of surface water flows in river catchments. This includes developing a prototype real-time water information system covering a regional river catchment that is made up of two linked sub-systems working in parallel: a continuous flow forecast system, based on emerging SWE standards, and a provenance management system that provides information on how flow forecasts are produced.

The continuous flow forecast system integrates hydrometerolological sensor data from five different agencies. The sensor data is accessed via SOS interfaces and drives a hydrological model that generates flow predictions at key monitoring points in the river catchment. Flow predictions are published to the world-wide web via a SOS interface. This allows current and predicted observations to be visualised using a generic client that understands the SOS interface specification.

In this demonstrator, we use weather stations from Forestry Tasmania and measurements of river flow from the Tasmanian Department of Primary Industry, Parks, Water and Environment (DPIPWE).

D. Hydrodynamic Model

The hydrodynamic model is based on Herzfeld's general purpose model for estuaries to regional ocean domains [7]. It provides three-dimensional distributions of temperature, salinity, current velocity, density, passive tracers, mixing coefficients and sea level. From inputs such as wind, pressure, surface heat and tides, the model calculates momentum, continuity, and conservation of heat and salt. Unfortunately, there is not currently three-dimensional data available for forecasts, but they could potentially be requested for future experiments. In the meantime, the system uses the surface forecasts.

IV. Quality Assurance/Quality Control

One of the issues with automated quality assurance/quality control (QA/QC) techniques is that it is difficult to distinguish between sensor errors and unusual events. This is particularly the case in the TasMAN project, where the concern is that, given the relatively sparse distribution of sensors, unusual events may be flagged as bad data.

In the case of environmental monitoring, it is often these unusual events that we are particularly interested in, as they may have negative effects on the region. Therefore, although QA/QC can build confidence in the data we provide, we also want to be sure that interesting events are not ignored, or filtered out. An autonomous vehicle combined with our real-time data from external parties can assist us to differentiate.

[24] have developed an automated QA/QC system using fuzzy membership functions based on domain knowledge. This approach has since been extended to use Bayesian networks derived from historical data [21]. The system produces error bars so that users of the data can evaluate the uncertainty of the measurement. An example of visualisation of the uncertainty can be seen in Figure 6. A data quality flag is also output:

1) Good data
2) Probably good data
3) Bad data or possible event

The contributing parameters to the quality assessment include:

- Time in water
- Time since calibration
- Rate of change
- Comparison with same phenomenon, same sensor node, different depths
- Comparison with different phenomenon, same sensor node, same depth
- Comparison with same phenomenon, different sensor node, depth

V. Adaptive Marine Monitoring

Figure 3 describes the basic algorithm for making use of data from the sensor web for adaptive marine monitoring. The first step is to identify the geographic context in which the robot will be operating. For some applications, for example a very localised phenomena, we would only be interested in observations from a limited area. In others, we may require a minimum number of data sources, requiring us to augment the geographic context, in which case context establishment would occur after the second stage of finding sensing sources. Ideally this second stage would allow us to use a real-time sensor search engine based on phenomena of interest and/or identify the spatial boundaries. Unfortunately data sources available in our area do not subscribe to such a service. Therefore a database of data sources was created based on the URL.

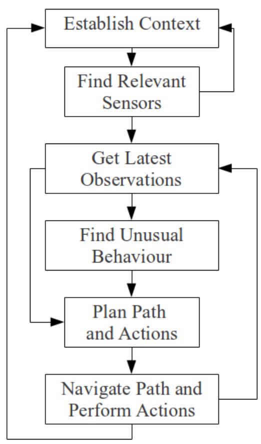


Fig. 3. High level sensor web enabled adaptive algorithm

The sensor discovery relies on standard web services as described by the OGC [13]. The services can be queried with GET or POST requests via HTTP (Hyper Text Transfer Protocol). The web services return an XML schema which may be parsed to extract the readings. These web services provide a query that allows you to find out what sensors are available in what locations, and how observations can be accessed. This request is usually of the form *GetCapabilities* and in most cases we can provide a spatial bounding box to search within. The results of these queries can be filtered to store only relevant phenomena.

Once we have a list of data sources we can then request the latest observations at any time. The requests are usually of the form *GetFeature*, *GetFeatureInfo* or *GetObservations*. In the case of this demonstrator, we use the observations to find our own sensors with suspicious data quality, plus unusual atmospheric conditions.

If it is decided that a TasMAN sensor node has poor quality data and there is little evidence to support that it might be an actual event, then the ASV will visit the node. ESRI (Economic and Social Research Institute) shape files of the South East Tasmanian coast line are used to automatically plan a safe path through the water from the current location.

As the ASV navigates the path it continues to request observations for potential use in creating a more energy efficient path; based on factors such as currents and wind. The context will need to be updated as we reach the boundaries of our current context, which may require new data sources to be discovered. The vehicle will only be able to sense from its current location, but it cannot sense the conditions at its destination, or between the two points. The vehicle may be currently in a slow current area or sheltered from the wind, but strong currents and winds may exist along the shortest path.

VI. Results

Field tests were conducted on the 27th of June 2011 launching at a boat ramp close to our laboratory at -42.896, 147.34. At 12:20 the system evaluated the TasMAN sensor nodes. It discovered poor quality data from the conductivity sensor at -42.886, 147.338; the CSIRO wharf node. The context was selected based on the location of the sensor node. On querying the readings available on the sensor web in close proximity, the air temperatures were above the average high for the time of year (12°C), so this was recorded for later analysis. The water flow was low for the time of year and there was no rainfall recorded for that day, Table I.

Provider	Location	Air Temp	Water flow	Rain
BOM	-42.89,147.33	1.7°C	-	0 mm
BOM	-42.9,147.24	13.4°C	-	0 mm
BOM	-42.71,146.9	14.3°C	-	0 mm
CMAR	-42.89,147.33	14.4°C	-	-
DPIPWE	-42.75,147.44	-	1.65 cumecs	-
DPIPWE	-42.67,147.17	-	1.16 cumecs	-

TABLE I
REAL-TIME SENSOR INFORMATION USED FOR EVENT EVIDENCE AT 12:20PM ON THE 27TH OF JUNE 2011

The next step was to compare with the hydrodynamic model forecast. Figure 4 compares the salinity of the hydrodynamic model as compared and the sensor node salinity, which has been converted from conductivity. The two measurements are similar up until around 1:00am. Figure 6 shows how the uncertainty grew during this time using the automated quality control. If we compare the water temperature from the model and the sensor for this time (Figure 5) we can see that the model and sensor information is very similar. There are also fewer spikes in the uncertainty. Through a rule-based approach it was determined that there was not sufficient evidence of an environmental event to explain the poor quality data and the action to further investigate the CSIRO wharf node was chosen.

A path was created to the sensor using the relevant coastal shape file, Figure 7. As the catamaran travelled to the destination it continued to query real-time sensor data from the sensor web. The context was expanded to include the current location. Figure 8 shows the wind speeds and directions extracted from nearby sensors, over the course of the experiments. Figure 9 shows the winds recorded on the catamaran itself at the

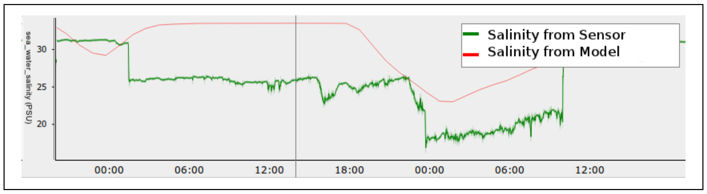


Fig. 4. Comparison of Salinity recorded at CSIRO wharf node and hydrodynamic model for the 27th of June 2011. Black vertical line indicates start time of mission.

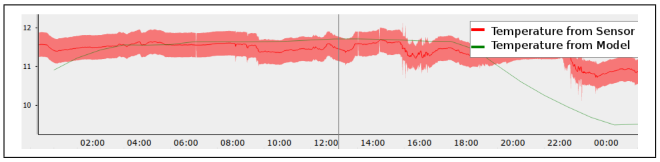


Fig. 5. Comparison of Temperature recorded at CSIRO wharf node (including uncertainty) and hydrodynamic model for the 27th of June 2011. Black vertical line indicates start time of mission.

southernmost point in the path in Figure 7 and at visited static sensor node.

On arrival at the CSIRO wharf node, Straycat took its own sensor measurements. The difference in temperature with the wharf node was just 0.5 of a degree. The difference in conductivity, however, was over 20,000 microsiemens per centimetre. As a result, it was recommended that the sensor node be cleaned and calibrated.

VII. Discussion

Standard names exist to facilitate discovery of specific phenomena [9], however these are not always being delivered by sensor web services. Useful data sources may be missed if they are described in an unexpected way. For the purposes of these experiments, the phenomena names were converted to the climate standard names so they could be compared and combined. We did not encounter issues with combining different measurement units, but this is also something which may well be encountered, especially on global scales.

The bounding box to search within for sensors or model data is difficult to manage and may alter depending on the phenomena. For example, the only real-time wave height sensors that were discoverable in South East Tasmania were approximately 80 kilometres from the nearest TasMAN sensor node. Therefore, there needs to be some assessment about whether it is worth increasing the size of the bounding box or not.

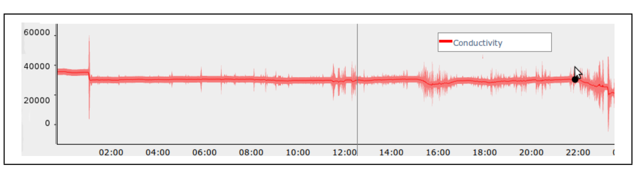


Fig. 6. Uncertainty in conductivity measurements (microsiemens per centimetre) from CSIRO wharf sensor node for the 27th of June 2011. Black vertical line indicates start time of mission.

Fig. 7. Path planned from launch location to sensor node location using coastal shape file.

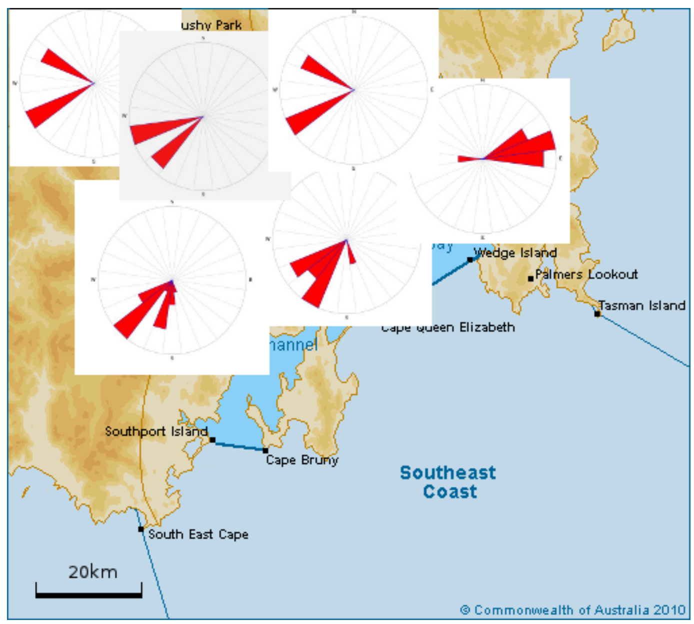


Fig. 8. Wind speed and direction from nearby sensor web enabled devices between 12:20 and 13:05 2/6/2011. Distance scale is in bottom left = 20km

The wind information appears to be appropriate for use in energy efficient path planning. The wind direction from the initial location was not consistent with the wind direction from the weather stations close to the destination. Therefore, the weather stations gave us a prediction of the wind conditions at our destination as can be seen in Figure 9 (bottom). We next intend to use the measurements from both the catamaran and the sensor web to plan paths whilst out on the water. The observations need to be combined with differing measures of uncertainty based on a number of factors including the

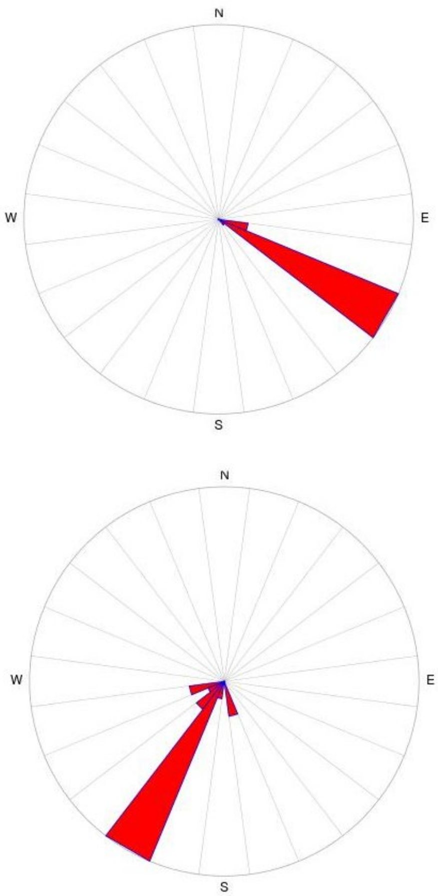


Fig. 9. Wind speed and direction recorded on the catamaran at the commencement of the mission (top) and at the destination (bottom)

distance, the sampling rate, and, ultimately, the reliability of the sensor. There is further work required in this area.

SensorML is designed to be very flexible so that it can be easily applied in a variety mission planning applications. Unfortunately, this flexibility comes at a price of true interoperability, which requires stronger enforcement of encoding rules and well-defined semantics. Though the W3C Semantic Sensor Network Incubator Group (SSN-XG) have developed a sensor ontology that can be used for semantic mark-up of SensorML documents [1], SensorML is not grounded enough for sensor discovery [17].

The CSIRO is promoting the development of a new sensor mark-up language (dubbed Starfish Fungus Language or *FL). *FL is used to describe sensor properties, capabilities, and corresponding deployment aspects. Furthermore, *FL features a clear separation between the physical device (Sensor), its model specific composition (SensorCharacteristics), and the specific procedures running on a physical device or subcomponents respectively (SensingProcedure). By this, it follows its main conceptual ancestor, the sensor ontology developed by the W3C Semantic Sensor Network Incubator Group [1]. The relative simplicity and structure of *FL, as well as its close alignment with O&M, make *FL better suited for sensor discovery. To this end, CSIRO is developing a prototype Sensor Information Service (SIS) that provides a RESTful interface to a sensor catalog. Service queries will return *FL sensor descriptions encoded in either XML, JSON or RDF. The RDF encoding will allow linking of sensor descriptions with associated observation archives and other digital information (and vice versa). The SIS is being designed to enable discovery of sensors and observations fit-for purpose.

VIII. Conclusion

Exploiting observations publicly available via the sensor web allows us to augment the situation awareness of an environmental monitoring robot at limited additional cost. The TasMAN (Tasmanian Marine Analysis Network) project has multiple initiatives for reducing the costs of marine sensor networks [8] including; sourcing sensors, sensor platforms, sensor network design, information delivery and visualisation. This is work is part of an attempt to reduce the costs of operating autonomous marine vehicles.

Preliminary results suggest that interesting autonomous decision making can be achieved by a marine vehicle using currently available technologies. This paper describes how the sensor web might be used to identify marine events, but similar techniques could be applied to many other environmental monitoring challenges.

In addition to the benefits we can envisage for marine monitoring, the development of this system has also highlighted areas for improvement in the delivery of our own real-time sensor data within the Tasmanian ICT Centre, and our initiatives in the area of linked sensor data.

Acknowledgements

The Tasmanian ICT Centre is jointly funded by the Australian Government through the Intelligent Island Program and CSIRO. The Intelligent Island Program is administered by the Tasmanian Department of Economic Development, Tourism and the Arts. This research was conducted as part of the CSIRO Wealth from Oceans National Research Flagship.

References

[1] Payam Barnaghi, Michael Compton, Oscar Corcho, Ral Garca Castro, John Graybeal, Arthur Herzog, Krzysztof Janowicz, Holger Neuhaus, Andriy Nikolov, and Kevin Page. Semantic Sensor Network XG Final Report. Technical report, World-Wide Web Consortium, June 2011.

[2] Mike Botts. OpenGIS Sensor Model Language (SensorML). Document 05-086, Open Geospatial Consortium, Wayland, Massachussets, USA, November 2005. http://portal.opengeospatial.org/files/?artifact_id=12606, Accessed Febuary 2006.

[3] Mike Botts. Sensor web enablement intro and v2.0 directions. http://q2o.whoi.edu/files/SWE_Intro_and_Status_2010.pdf, May 30 2011.

[4] Bureau of Meteorology. Frequently asked questions for the gis2web trial service. http://ows.bom.gov.au/mapserver/public/faq.html, May 30 2011.

[5] Simon Cox. Observations and Measurements. Document 05-087r5, Open Geospatial Consortium, Wayland, Massachussets, USA, April 2005. http://portal.opengeospatial.org/files/?artifact_id=14034, Accessed 5 May 2006.

[6] P. De Souza, G. Timms, A. Davie, B. Howell, and S. Giugni. Marine Monitoring using Fixed and Mobile Sensor Nodes. In *IEEE Oceans*, 2010.

[7] M Herzfeld. An alternative coordinate system for solving finite difference ocean models. *Ocean Modelling*, 14(3-4):174–196, 2006.

[8] D. Hugo, B. Howell, C. D'Este, GP. Timms, C. Sharman, and PA. de Souza. Low cost marine monitoring: From sensors to information delivery. In *IEEE Oceans*, Kona, USA, 2011.

[9] Lawrence Livermore National Laboratory. CF standard names - CF metadata. http://cf-pcmdi.llnl.gov/documents/cf-standard-names/standard-name-table/17/cf-standard-name-table.html, May 31 2011.

[10] S. Liang, S. Chen, Huang C.Y., R.Y. Li, D.Y.C. Chang, J. Badger, and R. Rezel. Capturing the long tail of the sensor web. In *International Workshop on Role of Volunteered Geographic Information in Advancing Science, GIScience 2010*, 2010.

[11] S. H. L. C. Liang and C. V. Tao. A Distributed Geospatial Infrastructure for the Sensor Web. *Computers & Geosciences*, 31(2):221–231, 2005.

[12] Arthur Na and Mark Priest. Sensor Observation Service. Document 05-088r1, Open Geospatial Consortium, Wayland, Massachussets, USA, January 2006. http://portal.opengeospatial.org/files/?artifact_id=12846, Accessed 1 April 2006.

[13] Open Geospatial Consortium. Ogc standards and specifications. http://www.opengeospatial.org/standards, May 30 2011.

[14] Cedric Pradalier. Asl limnobotics. http://www.asl.ethz.ch/research/asl/limnobotics, July 4 2011.

[15] Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Ng Andrew. Ros: an open-source robot operating system. In *International Conference on Robotics and Automation*.

[16] MS. Shahriar, P. de Souza, and G. Timms. Smart information retrieval for marine sensor data. In *European Geosciences Union General Assembly*, Vienna, Austria, 2011.

[17] I. Simonis, C. Malewski, A. Terhorst, P. Taylor, and S. Cox. *FL Starfish Fungus Language for Sensor Description. Technical report, Open Geospatial Consortium, 2011.

[18] Ingo Simonis. Sensor Planning Service. Document 05-089r1, Open Geospatial Consortium, Wayland, Massachussets, USA, December 2005. http://portal.opengeospatial.org/files/?artifact_id=12971, Accessed 1 April 2006.

[19] Ingo Simonis. Sensor Alert Service. Document 06-028, Open Geospatial Consortium, Wayland, Massachussets, USA, April 2006. http://portal.opengeospatial.org/files/?artifact_id=13921, Accessed 4 July 2006.

[20] Ingo Simonis and Andreas Wytzisk. Web Notification Service. Document 03-008r2, Open Geospatial Consortium, Wayland, Massachussets, USA, April 2003. http://portal.opengeospatial.org/files/?artifact_id=1367, Accessed 6 Febuary 2006.

[21] D. Smith, GP. Timms, and PA. de Souza. Automated quality assessment of marine sensing incorporating statistical, causal inference and expert knowledge. In *IEEE Oceans*, Kona, USA, 2011.

[22] Tasmanian ICT Centre. Tasmanian Marine Analysis Network: Web data service. http://www.csiro.au/tasman/WDS/wds?request=GetDeployment&format=xml, May 30 2011.

[23] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz. Web-enabled robots. *Robotics Automation Magazine, IEEE*, 18(2):58 –68, june 2011.

[24] GP. Timms, PA. de Souza, and L. Reznik. Automated assessment of data quality in marine sensor networks. In *IEEE Oceans*, Sydney, Australia, 2010.

[25] A. Underbrink, A. Potter, K. Witt, and J. Stanley. Modeling sensor web autonomy. In *Aerospace Conference, 2011 IEEE*, pages 1 –14, march 2011.

[26] Markus Waibel, Michael Beetz, Javier Civera, Raffaello D' andAndrea, Jos Elfring, Dorian G andlvez L andpez, Kai H andussermann, Rob Janssen, J.M.M. Montiel, Alexander Perzylo, Bj andrn Schie andle, Moritz Tenorth, Oliver Zweigle, and Ren and van de Molengraft. Roboearth. *Robotics Automation Magazine, IEEE*, 18(2):69 –82, june 2011.

Exploration of the Gulf of Mexico Oil Spill with the *Sentry* Autonomous Underwater Vehicle

Michael V. Jakuba*, James C. Kinsey†, Dana R. Yoerger†, Richard Camilli†, Christopher A. Murphy†, Daniel Steinberg*, and Asher Bender*

*Australian Centre for Field Robotics
University of Sydney
Sydney, NSW 2041, Australia
{m.jakuba|d.steinberg|a.bender}@acfr.usyd.edu.au

†Deep Submergence Laboratory
Woods Hole Oceanographic Institution
Woods Hole, MA 02543, USA
{jkinsey|dyoerger|rcamilli|cmurphy}@whoi.edu

Abstract*— We report on the use robotic assets in the investigation of subsea hydrocarbon plumes caused by the blowout, on 20 April 2010, and subsequent sinking of the *Deepwater Horizon* drilling platform in the Gulf of Mexico. We employed conventional oceanographic sampling techniques along with the *Sentry* autonomous underwater vehicle (AUV) to confirm the existence of a coherent subsea hydrocarbon plume, then to map the plume's spatial extent out to 35 km down-current from the well head, and finally to collect targeted water samples from within the plume itself for later laboratory analysis. In this paper we focus on the techniques used to coordinate sampling activities between the AUV and conventional instrumentation: geo-referenced navigation of all data, integrative visualization of multi-modal and multi-platform data, real-time telemetry, visualization, and analysis of data, and the real-time adaptation of vehicle trajectory in response. Our results demonstrate that when initial characterization is poor, limited human interaction and feedback can accelerate the study, and improve the analysis, of evolving environmental phenomena. We discuss several lessons learned, particularly as they apply to the future development of limited-interaction autonomy in subsea robotics. Using real data collected during the *Deepwater Horizon* expedition, we present simulations of semi-automated data interpretation and sampling plan adaptation comparable to the real-time actions taken by us during the expedition itself.

Fig. 1. The Woods Hole Oceanographic Institution AUV *Sentry* on the deck of the R/V *Endeavor* between deployments. Drilling platforms and other equipment working at the *Deepwater Horizon* blowout site are visible in the background. The closest to the site that the vehicle was deployed was 3 nautical miles (5 km). Photo by D. Yoerger.

I. Introduction

On 20 April 2010, the *Deepwater Horizon* drilling rig suffered a blowout that resulted in the eventual sinking of the rig and the death of 11 personnel on board. Prior to the successful capping on 15 July 2010, oil from the damaged well head was leaking at a rate whose quantification remains contentious but undoubtedly represents one of the largest accidental releases of oil on record. The environmental impact of the oil spill depends on a number of incompletely understood characteristics of the spill including composition of the oil, its chemical evolution in the environment, the rate and total volume of oil released, and the dynamics of its spread.

Since shortly after the explosion various sources have reported the presence of subsurface plumes of oil [1], [2]. Contrary to the elementary notion that oil and water do not mix, the mixture emanating from the well head is a complex multi-phase mixture of oil and gases that interacts with the surrounding water column as it rises. Both controlled experiments [3] and historical evidence [4] suggest that some constituents of the effluent and/or minute droplets of oil will enter the water column forming a subsurface plume with little or no residual buoyancy. The composition of any subsurface plumes and the fraction of the total oil released that they represent could play a significant role in the ultimate environmental and economic impact of the spill.

In June 2010 the authors were part of a research cruise to the Gulf of Mexico funded by the United States National Science Foundation to identify and characterize any subsurface plumes associated with the *Deepwater Horizon* spill. We employed two principal sampling platforms, a conventional cable-lowered oceanographic conductivity, temperature, and depth (CTD) rosette augmented with a TETHYS in situ mass spectrometer [5] as well as several sensors specifically selected for the cruise; and the Woods Hole Oceanographic Institution's *Sentry* Autonomous Underwater Vehicle (AUV) also equipped with a TETHYS instrument as well as various other water column sensors (Fig. 1). The lowered CTD included the ability to collect water samples — crucial to determining the exact composition of the plume (Fig. 2).

This paper is organized as follows. Sec. II discusses the techniques used to coordinate sampling activities between the CTD and AUV for plume localization, plume characterization, and targeted water sampling: geo-referenced navigation of all data; integrative visualization of multi-modal and multi-platform data, real-time telemetry, visualization, and analysis of data, and the real-time adaptation of vehicle trajectory in response. Sec. III proposes a method for conducting subsea robotic survey that capitalizes on the increasing availability and bandwidth of acoustic communications for real-time human interaction with subsea assets combined with modern machine learning techniques for dimensionality reduction and data pre-processing. The method aims to enable human operators to focus on high-level data interpretation and mission objective formulation when adapting sampling plans. We demonstrate the method on data collected during the *Deepwater Horizon* expedition using a data-denial methodology to simulate real-time vehicle trajectory adaptation and provide a qualitative assessment of the results relative to actions taken by us to adapt vehicle trajectory during the actual expedition. We conclude with a discussion of lessons learned.

II. Subsurface Plume Localization at the *Deepwater Horizon* Blowout Site

Our ultimate objective was to collect targeted water samples from within any subsurface plumes for later analysis in shore-side laboratories. To accomplish this we first had to confirm the existence of plumes by locating and mapping them. Our approach capitalized on the strengths of our two sampling platforms, while respecting the constraints imposed by the sensors on board. The lowered CTD was used to initially locate the plume and then to characterise its vertical structure. The AUV provided a complementary horizontal perspective.

Fig. 3 shows the northeast corner of the Gulf of Mexico and the location of the *Deepwater Horizon* site off the Louisiana coast. With the exception of background water profile measurements all subsurface operations during the 10 days we spent on station took place within the area indicated. We conducted 23 CTD lowerings including 3 extended deployments in which the instrument package was towed slowly while undulating within a prescribed depth interval (a procedure known in the oceanographic community as a tow-yo). The CTD data identified a deep subsurface plume tending to the WSW of the *Deepwater Horizon* site and centered at a depth of 1100 m. *Sentry* dived 3 times, covering approximately 240 kilometers at depths between 1000 m and 1300 m. Two of these dives,

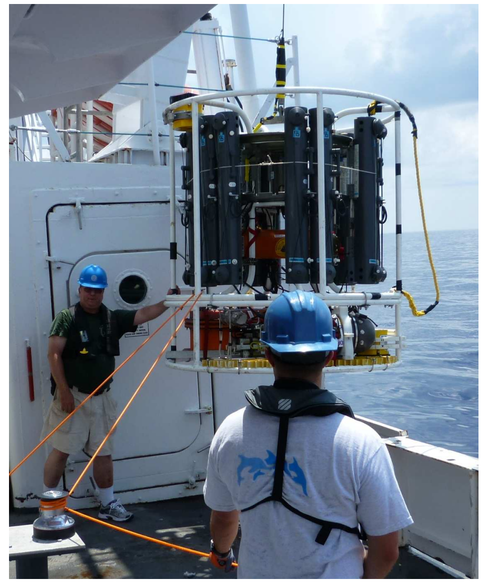


Fig. 2. CTD and rosette being deployed off the R/V *Endeavor*. Non-standard instrumentation including the TETHYS in situ mass spectrometer and an optode affixed to the cage on the bottom of the package. Photo by C. McIntyre, WHOI

both to the WSW of the site (Fig. 5), encountered water enriched with hydrocarbons significantly above background levels. Together, the CTD and *Sentry* resulted in a detailed, multimodal picture of a coherent subsurface plume extending at least 35 km from the *Deepwater Horizon* site. The precise nature of that plume, including constituents, their absolute concentrations, and the fraction of oil confined to the plume are discussed in the scientific literature [6].

A. Acoustic Telemetry

Seawater rapidly attenuates electromagnetic radiation, rendering radio frequency signals of the type commonly employed in terrestrial communications systems ineffective underwater. Acoustic modem systems, e.g. [7], provide a relatively low-bandwidth ($40\ \mathrm{km} \cdot \mathrm{kbps}$ [8]) alternative that nevertheless allows for the real-time downlink of a portion of data collected subsea and for the transmittal of high-level control commands to the vehicle. *Sentry* uses a commercial acoustic modem integrated into the USBL navigation system.

This system enables commands to be sent to the vehicle and data sent from the vehicle to the surface vessel. The amount of data being obtained on the AUV far exceeds the available bandwidth, so we employ a queuing system in which the user sends a message to the vehicle requesting which information should be transmitted back to the surface vessel for human interpretation. In addition to sending information requests to the vehicle, we can transmit mission re-specification commands to achieve tasks such as changing vehicle depth and retasking the vehicle on new trajectories. This architecture allows us

[2] http://mw1.google.com/mw-earth-vectordb/disaster/gulf_oil_spill/kml/noaa/nesdis_anomaly_rs2.kml

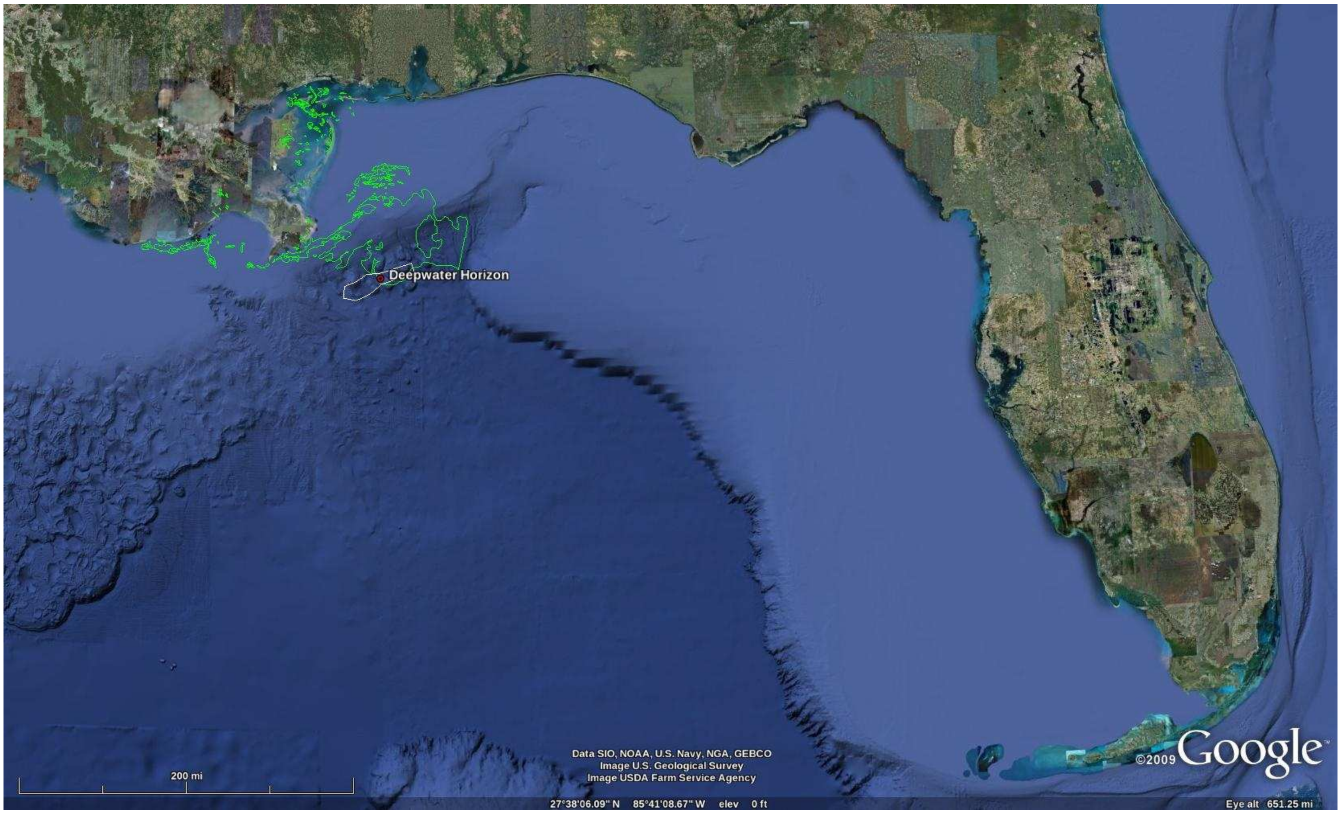


Fig. 3. Work site location in the Gulf of Mexico off the Louisiana coast. All AUV and shipboard operations were carried out within the white bounds indicated, an area about 50 nautical miles in length. For comparison, the light green outlines show the potential oiling footprint of the surface plume observed by NOAA 2010-07-07[2]. The *Deepwater Horizon* site is shown located at $28^o\ 44.3071'$ N, $88^o 21.9611'$ W, based on ship's radar while working near the site.

to receive crucial sensor data and, based on the information obtained, to retask the vehicle in response.

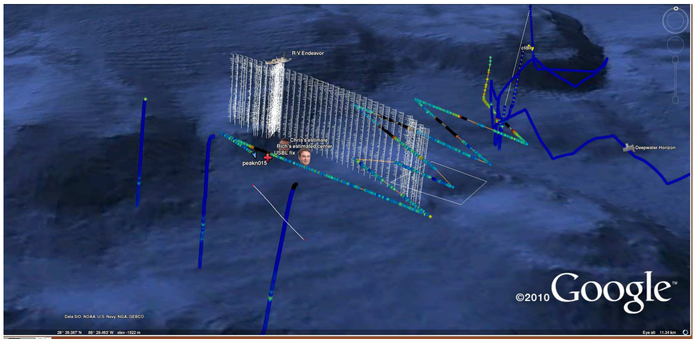
B. Real-time Visualization

Limited cruise duration, limited a priori information about the plume, and the need to obtain precisely targeted water samples from within a dynamic phenomenon required the rapid analysis and visualization of data to devise appropriate sampling strategies and use the available assets efficiently. Leveraging previous work employing the Keyhole Markup Language (KML) for the dissemination and visualization of geo-referenced oceanographic data [9], [10], we provided the science party with near real-time displays of integrated chemical tracer data from all instruments on board the CTD and selected ion peaks from the TETHYS mass spectrometer on board *Sentry*. Fig. 4 shows a screenshot of our data visualization part way through AUV dive sentry064, rendered by Google Earth. The image shows aromatic hydrocarbon fluorimetry from several CTD casts and one tow-yo, real-time normalized methane concentration telemetered acoustically from *Sentry*, and real-time water current profiles generated by the ship's acoustic Doppler current profiler (ADCP). This visualization was instrumental in coordinating the sampling strategies of the CTD and *Sentry*. It aided in site selection and survey design, water sample location selection, real-time survey modification, and provided the first visual confirmation of a coherent subsea plume.

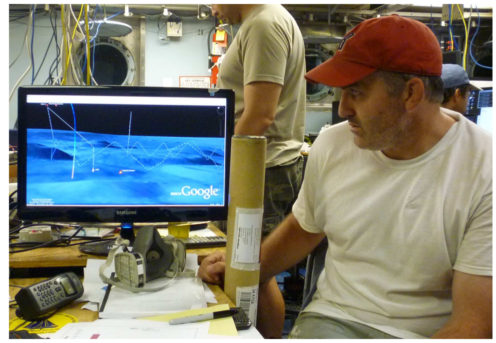
C. Sentry Dives 064 and 065 — Tracking a Subsurface Oil Plume

The first challenge in assessing the extent of the subsurface plume was initially locating it. A CTD tow-yo conducted around the periphery of the *Deepwater Horizon* site registered intense hydrocarbon anomalies at 1100 m depth to the west-southwest of the well head and weaker anomalies at the same depth to the northeast. Based on this and other supporting data we planned a series of AUV surveys aimed at tracking the plume down-current of the well head. The goal of these surveys was to determine the horizontal extent of the plume and provide the necessary reconnaissance for targeted water sampling.

The *Sentry* AUV was deployed on two dives — sentry064 and sentry065 — during which the vehicle tracked the plume over 30 km down-range from the origin of the plume at the well head (Fig. 5). While both dives had identical goals, the manner in which they were conducted differed, and the contrasting survey techniques used in either case each possessed advantages and disadvantages. We planned and executed sentry064 in the conventional manner, with the vehicle following a series of preplanned tracklines designed to repeatedly cross



(a) CTD and *Sentry* data rendered in Google Earth.



(b) Co-chief scientist Dr. Reddy considering the data.

Fig. 4. Real-time data visualization: (a) Screenshot of Google Earth rendering taken during the cruise and showing both fluorometer data collected over the preceding days using CTD casts and tow-yos as well as TETHYS mass spectrometer data being telemetered acoustically from the AUV in real time; (b) inspecting the visualisation. Prompt, effective visualization improved the ability of the science party to coordinate sampling and survey activities as well as to alter survey plans in real time, enhancing the efficiency and effectiveness of operations.

the plume at a constant depth. Real-time acoustic telemetry from the vehicle was used to select the site for a CTD cast that was then conducted during the dive and out of acoustic range of the vehicle.

We designed sentry065 similarly but with the intention of acoustically manipulating the mission plan in real time. We planned to cut tracklines short after mass spectrometry data received acoustically indicated a return to background values following a transect of the plume. This strategy was designed to increase the down-current extent of the survey and was employed successfully early during the dive. Plume intensity on later tracklines exhibited an unexpected decrease in magnitude, prompting us to dramatically alter *Sentry's* mission plan, first to reacquire the plume closer to the well head, and then later to refine the survey depth before continuing with (a modified version of) the original survey plan.

At 30 km from the well head the hydrocarbon anomaly remained well above the detection threshold of the TETHYS instrument on *Sentry*; however deteriorating weather conditions prevented further AUV deployments and ultimately forced an end to scientific operations altogether. In total, dives 64 and 65 spanned a total of 61 hours during which *Sentry* spent 47.4 hours deployed and covered over 170 km.

III. Semi-Autonomous Subsea Robotic Survey

Our real-time interactions with *Sentry* yielded scientifically more productive dives but also required us to engage in low-level data processing and trajectory-level mission re-specification. As subsea robots become more sophisticated and the number of robots concurrently in the water increase, the scope for low-level interactions will decrease commensurately. Human oversight will remain valuable but must transition to higher-level interaction. This will require enhanced autonomy on the part of the robots themselves. In this section we discuss the performance of a semi-autonomous method for subsea robotic survey applied, via data-denial simulation, to dive sentry064.

Our method applies the classical sense-plan-adapt (SPA) approach to robotic decision making but with high-level human input at each stage of the cycle. Our primary aim is to reduce the cognitive load on human operators while still leveraging human skill in high-level decision making. This aim aligns well with the reality of limited bandwidth acoustic communications—data pre-processing carried out autonomously subsea can reduce the bandwidth required to telemeter data to the surface; a robot capable of interpreting high level objectives rather than direct trajectory specification will also likely reduce the bandwidth required to transmit control commands. The motivations behind our particular implementation of each stage of the SPA cycle is discussed subsequently.

a) Sense: Various authors have reported on the use of AUVs to trace and/or map both synthetic and naturally occurring turbulent plumes, e.g. [11]–[15]. A necessary component of any of these methods is a mechanism for deciding what sensor readings represent contaminated plume water versus background water. Such mechanisms become more difficult to construct when, as in our case, the signature of the plume within data from the various sensors available was initially unknown.

The sensor suite on board *Sentry* was chosen by scientists based on expert knowledge of the likely chemical constituents of a subsea hydrocarbon plume; nevertheless, significant uncertainty remained concerning the presence, relative concentrations, and manifestation of these constituents in the sensor data. Ultimately the methane measurement produced by the TETHYS instrument provided the most reliable indication of plume presence; however, this knowledge was unavailable prior to human analysis of all sensor data streams.

Our approach to automated plume detection considers all 11 available scalar sensor data streams together as vector-valued data, sorts these into statistically distinct classes, and relies on human interpretation to provide a semantic label for each class as either plume, background, or other. Parametrized statistical models for each class are learned as part of the procedure, meaning the robot can autonomously apply semantic labels to subsequently acquired data.

The model used for classification in this paper is the

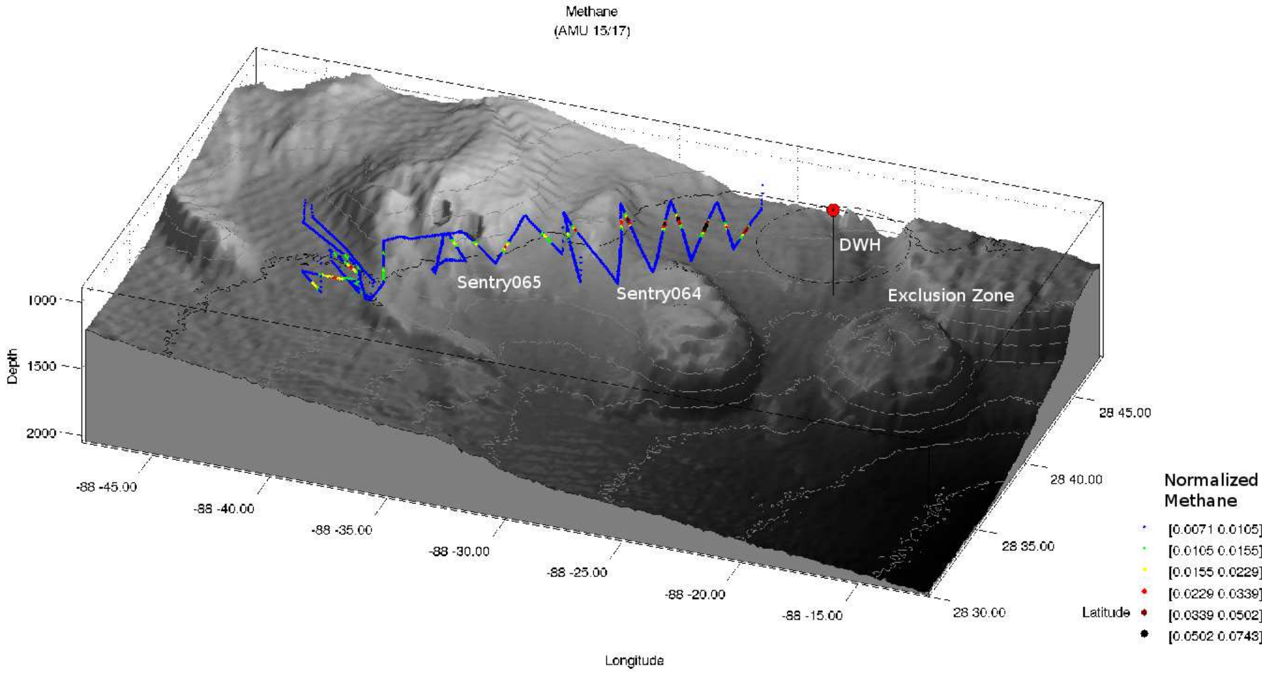


Fig. 5. Normalized Methane observed from the TETHYS mass spectrometer aboard the *Sentry* AUV during two dives, sentry064 and sentry065, to the west of the *Deepwater Horizon* site. The *Deepwater Horizon* site and 5 km exclusion zone are indicated in the perspective view.

Bayesian, non-parametric, Variational Dirichlet Process model (VDP) [16]. This model is a mean-field variational approximation of a Dirichlet Process Mixture Model (DPMM) [17], [18]. Important assumptions made in this paper are that observations are distinctly multimodal, can be represented using a Gaussian Mixture Model (GMM), and are independently and identically distributed (i.i.d.) when conditioned on their class label. Its principal feature, besides rapid execution, is that the method automatically infers the number of classes present in the data. Fig. 6 shows the classified output produced after semantic labeling by a human. The algorithm appears to have implicitly identified methane (mass-to-charge ratio m/z 15) and optical backscatter (OBS) as indicative of a distinct cluster (labeled plume and shown in red), and has also successfully identified two periods of anomalous behavior in the OBS sensor as a distinct cluster (labeled other and shown in green).

As yet our classification process exploits no notion of spatial coherency in the environmental phenomena of interest. While we have attained promising classification results despite this, to adapt vehicle trajectory some mechanism for performing inference over the spatial domain of the survey area is often necessary.

b) Plan: Several spatial inference methods specific to robotic plume mapping as applied especially to plume source localisation exist [14], [19]–[22]. Like [19] our approach employs a Mixture of Gaussian Processes (MGP) to model the spatial coherence of the plume and background; however, because our output space is 11-dimensional rather than a scalar chemical concentration, we perform a logistic regression over the scalar class labels to avoid learning the parameters of what would otherwise become a multivariate MGP. This is known as Gaussian Process Classification (GPC) [23].

Once the hyperparameters of the mixture components have been learned GPC regression provides a way to extrapolate the probability of observing each semantically labeled class to the spatial domain of the survey (Fig. **??**). On the basis of this map, an agent can plan by evaluating the expected outcome of future actions relative to a specified objective function, for instance, [24] traded off information gain with traversal cost to generate constrained maximum entropy sampling plans.

In practise, developing good objective functions in the dynamic setting of a scientific expedition remains challenging. On the other hand, scientists and operators try to design pre-programmed AUV surveys in a way that encapsulates key objectives, some of which, like coordination with other assets and weather considerations, would be difficult to encode in a useful objective function because they depend on external circumstances not readily sensed by a deployed robot. We propose that limiting the scope of autonomous planning to modifications of the pre-planned mission can retain good performance relative to these hard-to-codify objectives, and if designed with autonomous adaptation in mind, also benefit from autonomous decision making.

c) Act: Dive sentry064 (Fig 5) was designed to provide multiple down-current horizontal crossings through the hydrocarbon plume, under the assumption that the current would cause the plume to spread along isobaths to the WSW. The increasing amplitude of the zigzag trajectory specified in the mission plan reflected our uncertainty about plume spreading rate and the precise direction of the current. The

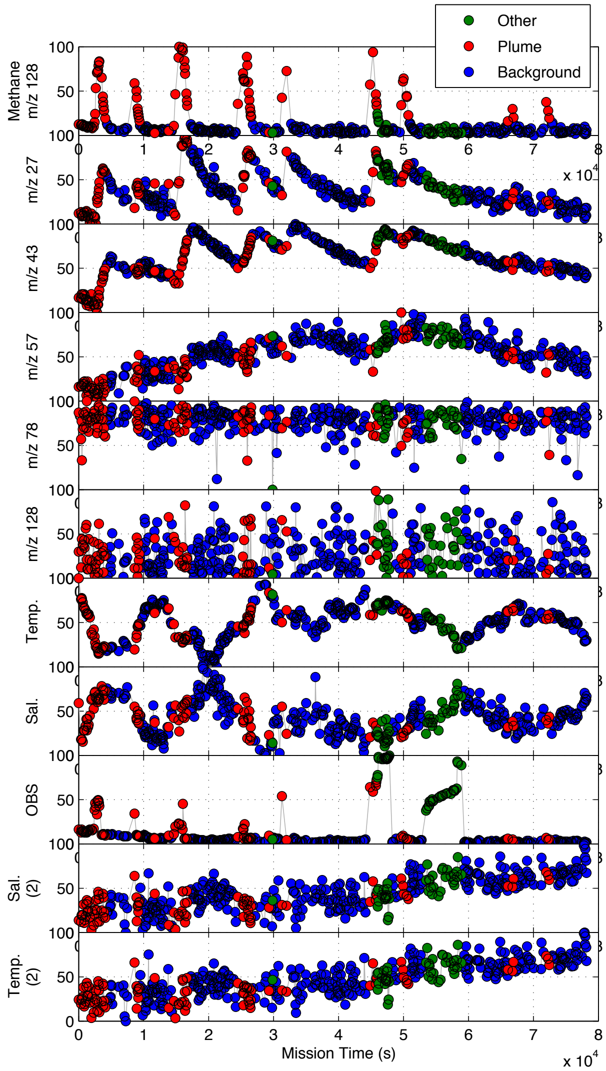


Fig. 6. All 11 scalar chemical sensor data streams interpolated onto the timebase of the TETHYS instrument, normalized to span the range $[0, 100]$, and clustered into distinct components of a Gaussian Mixture Model. Semantic labeling as three classes, plume (red), background (blue), and other (green), was provided by a human.

large amplitude of the survey tracklines in relation to the width of the plume encountered represents an inefficiency that might have been mitigated by terminating tracklines early, as was commanded by human operators via acoustic link on sentry065. This might also have been accomplished autonomously had an appropriate objective function and set of admissible control actions been available.

To test this supposition, we simulated a heuristic that allowed the robot to abandon the remaining portion of a trackline and the corresponding fraction of the next trackline, if, based on the output of the GPC, it was unlikely to encounter plume water on the skipped portions of the survey. We split tracklines into four segments to create four decision points along each trackline. Each time the robot completed a segment, it compared a fixed threshold (0.35) with the maximum probability of plume presence from the GPC for test points arranged along the trackline segments to be skipped. This would enable the robot to trim off the tips of zigzags unlikely to be productive, and consequently to be able to complete more crossings of the plume before exhausting its batteries. The approach is myopic in the sense that the GPC regression is regarded as truth at each iteration.

A good manually-labeled GMM for classification and GPC were attained after completion of the first four tracklines and did not require relearning or relabeling until the anomalous OBS data appeared later in the dive. During this training phase the simulated robot was not allowed to consider skipping trackline segments. Fig. 7 shows the simulation part way through. The GPC has found hyperparameters suggesting long correlation lengths. This enables the robot to correctly decide to skip the remaining trackline segments in the current zigzag.

Fig. 8 shows the simulation at completion. A plume-like structure is visible as a red band in the GPC-regression results. However, the GPC results indicate a correlation length scale shorter than in Fig. 7, and short relative to the length scale imposed by inter-trackline spacing on the survey. This lack of predictive certainty distant from existing data produced conservative behavior with respect to skipping trackline segments, and no further segments were skipped.

IV. Discussion and Conclusions

Several aspects of this work bear directly the current and future use of robotics for environmental monitoring:

1) The time and spatial scales associated with dynamic features in an environment should drive sampling plan design as well as the selection of appropriate instrumentation, including the use of autonomous platforms. In our case an AUV offered maneuverability and speed advantages over a cable-lowered CTD for reconnoitering the horizontal extent of a subsurface hydrocarbon plume, but was most effective in concert with the CTD.
2) Real-time transmission of data from autonomous platforms can augment the effectiveness of these platforms by enabling operators to deploy other assets before conditions change. We used the real-time AUV data to inform the sampling strategy of the CTD and to target water samples.
3) Some environmental monitoring tasks are characterized by relatively large swaths of uninteresting terrain. In these circumstances, we stand to gain the most from adaptive survey. In our case, we adapted the cruise plan to data as it became available, and, on a finer scale, we also adapted the AUV's trajectory to more effectively sample the feature of interest.
4) Human intervention may increase the scientific yield of robotic surveys, but any increased value must be traded off against the opportunity cost of demanding a human's attention. There is a pressing need to develop autonomous and semi-autonomous data process-

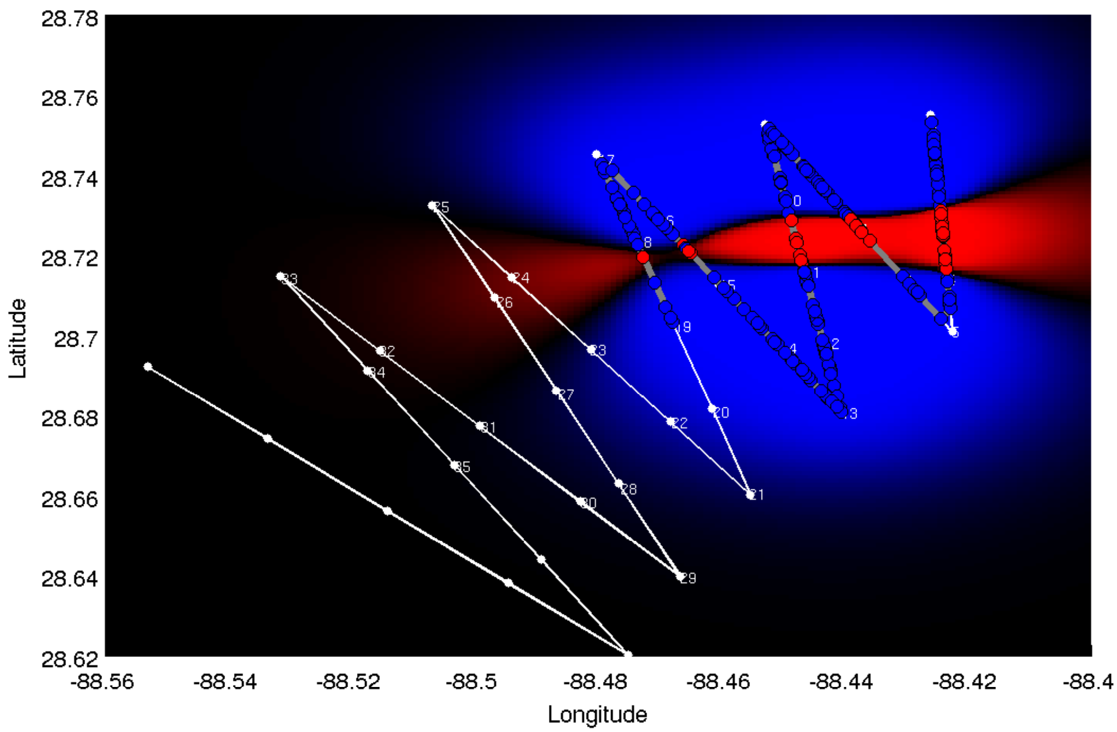


Fig. 7. Data-denial simulation of semi-autonomous adaptive execution of sentry064 part way through the simulation. The robot's path is shown in gray. The colored circles represent data classified as plume (red) and background (blue). The entire domain of the survey is colored according to the GPC regression, with black representing maximum ambiguity, that is an equal chance of either plume or background.

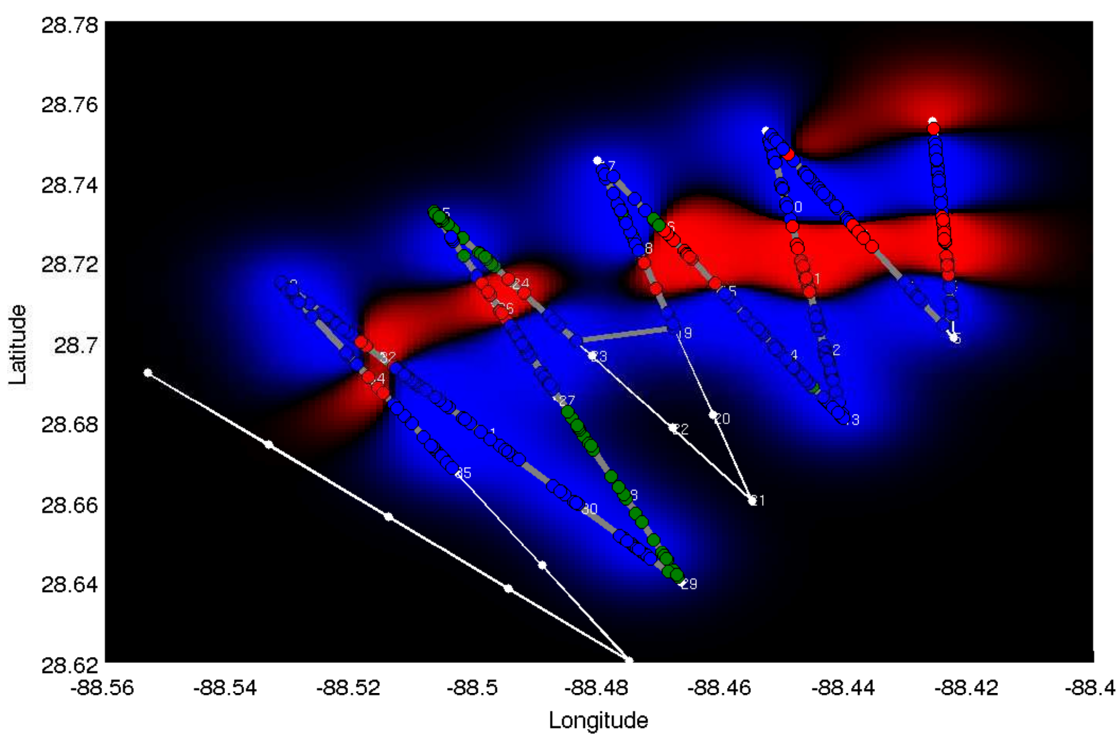


Fig. 8. Data-denial simulation of semi-autonomous adaptive execution of sentry064 at the conclusion of the simulation. Data points from a third class (green), corresponding to the anomalous OBS data, is ignored during the GPC regression.

ing and adaptive survey methods that reflect the real challenges of incompletely characterized environmental phenomena, limited processing power and communications bandwidth, and limited endurance.

Based on these observations, we developed a semi-supervised method for adaptive survey that conceivably could have reduced the time spent by the AUV outside the plume during dive sentry064 without requiring intensive operator interaction. The comparison with sentry065 is instructive. On sentry065 intensive human interaction was required to reacquire the plume signal. Our approach to semi-autonomous adaptive survey relies on a sensible pre-planned mission. Radical changes to the mission plan like that required in sentry065 would require a more complex 3-dimensional environmental model, a far more complete set of admissible control actions, and a consequently much more complex decision process.

Acknowledgment

This work was funded through the National Science Foundation's RAPID program. We acknowledge the captain and crew of the R/V *Endeavor* for their careful handling of *Sentry* during over-the-side operations as well as other members of the scientific party aboard for their management of CTD operations and water sample handling. We also acknowledge the Dorado AUV team from MBARI and the scientific party from the NOAA ship R/V *Thomas Jefferson*, and Prof. Ruoying He of the Dept. of Marine, Earth and Atmospheric Sciences at North Carolina State University, all of whom graciously provided their results from their own investigations into the oil spill. As always, the ABE/*Sentry* team provided excellent vehicle operational support and we are grateful to Andy Billings, Alan Duester, and Scott McCue for their efforts before and during the cruise.

References

[1] J. Gillis, "Giant plumes of oil forming under the gulf," New York Times, May 15 2010, http://www.nytimes.com/2010/05/16/us/16oil.html.

[2] MBARI Dorado AUV group, "Data Report: June 2-3, 2010 Operations of MBARI AUV R/V Gordon Gunter Cruise, Gulf of Mexico," Monterey Bay Aquarium Institute, Tech. Rep., June 11 2010.

[3] P. S. Daling, M. O. Moldestad, O. Johansen, A. Lewis, and J. Rodal, "Norwegian testing of emulsion properties at sea–the importance of oil type and release conditions," *Spill Science and Technology Bulletin*, vol. 8, no. 2, pp. 123–136, Apr. 2003.

[4] J. W. Farrington, "NOAA Ship RESEARCHER/Contract Vessel PIERCE Cruise to IXTOC-1 Oil Spill: Overview and Integrative Data Assessment and Interpretation," Woods Hole Oceanographic Institution, Tech. Rep., 1983.

[5] R. Camilli and A. Duryea, "Characterizing spatial and temporal variability of dissolved gases in aquatic environments with in situ mass spectrometry," *Environmental Science and Technology*, vol. 43, no. 13, pp. 5014–5021, 2009.

[6] R. Camilli, C. Reddy, D. Yoerger, B. V. Mooy, M. Jakuba, J. Kinsey, C. McIntyre, S. Sylva, and J. Maloney, "Tracking hydrocarbon plume transport and biodegradation at Deepwater Horizon," Submitted to *Science*.

[7] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball, "The WHOI micro-modem: an acoustic communications and navigation system for multiple platforms," in *Proceedings IEEE/MTS Oceans Conference and Exhibition*, vol. 2, 2005, pp. 1086–1092.

[8] J. Partan, J. Kurose, and B. N. Levine, "A survey of practial issues in underwater networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 23–33, Oct. 2007, special section on ACM WUWNet 2006.

[9] C. Murphy, "Lossy compression and real-time geovisualization for ultra-low bandwidth telemetry from untethered underwater vehicles," Master's thesis, MIT / WHOI Joint Program in Oceanography, Cambridge, MA, Sep. 2008.

[10] C. Murphy and H. Singh, "Human-guided autonomy for acoustically tethered underwater vehicles," in *Proceedings IEEE/MTS Oceans Conference and Exhibition*, Sep. 2008, pp. 1–8.

[11] J. A. Farrell, S. Pang, and W. Li, "Chemical plume tracing via an autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 428–442, 2005.

[12] D. Yoerger, A. Bradley, C. G. M.V. Jakuba, T. Shank, and M. Tivey, "Autonomous and remotely operated vehicle technology for hydrothermal vent discovery," *Oceanography*, vol. 20, no. 1, pp. 152–161, Jan. 2007.

[13] M. Fisher and M. Dock, *Trace Chemical Sensing of Explosives*. Wiley-Interscience, 2007, ch. Detection of Trace Explosive Signatures in the Environment, pp. 133–150, ronald L. Woodfin, editor.

[14] M. Jakuba and D. Yoerger, "Autonomous search for hydrothermal vent fields with occupancy grid maps," in *Proceedings Australian Conference on Robotics and Automation*, Dec. 2008.

[15] R. N. Smith, J. Das, Y. Chao, D. A. Caron, B. H. Jones, , and G. S. Sukhatme, "Cooperative multi-auv tracking of phytoplankton blooms based on ocean model predictions," in *Proceedings IEEE/MTS Oceans Conference and Exhibition*, Sydney, Australia, May 2010.

[16] K. Kurihara, M. Welling, and N. Vlassis, "Accelerated variational Dirichlet process mixtures," *Advances in Neural Information Processing Systems*, vol. 19, p. 761, 2007.

[17] T. Ferguson, "A Bayesian Analysis of some Nonparametric Problems," *The Annals of Statistics*, vol. 1, no. 2, pp. 209–230, 1973.

[18] C. E. Antoniak, "Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems," *The Annals of Statistics*, vol. 2, no. 6, pp. 1152–1174, 1974.

[19] C. Stachniss, C. Plagemann, and A. Lilienthal, "Learning gas distribution models using sparse Gaussian process mixtures," *Autonomous Robots*, vol. 26, no. 2, pp. 187–202, 2009.

[20] A. J. Lilienthal and T. Duckett, "Gas source localization by constructing concentration gridmaps with a mobile robot," in *Proceedings ECMR 2003*, 2003.

[21] J. A. Farrell, S. Pang, and W. Li, "Plume mapping via hidden markov methods," *IEEE Transactions on System, Man, and Cybernetics*, vol. 33, no. 6, pp. 850–863, 2003.

[22] S. Pang and J. A. Farrell, "Chemical plume source localization," *IEEE Transactions of Systems, Man, and Cybernetics–Part B: Cybernetics*, vol. 36, no. 5, pp. 1068–1080, 2006.

[23] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Massachusetts Institute of Technology: the MIT Press, 2006.

[24] D. Thompson and D. Wettergreen, "Intelligent maps for autonomous kilometer-scale science survey," in *International Symposium on Artificial Intelligence, Robotics and Automation in Space (ISAIRAS)*, 2008.

Lizhbeth: Toward Autonomous Toxic Algae Bloom Monitoring

Gregory Hitz[1], François Pomerleau[1], Marie-Ève Garneau[2],
Cédric Pradalier[1], Thomas Posch[2], Jakob Pernthaler[2], Roland Y. Siegwart[1]

***Abstract*—With the recent increase of toxic algal blooms in lakes, biologists are looking for new sampling methods to analyze their spatiotemporal dynamics. In this paper, we present an Autonomous Surface Vessel (ASV) capable of achieving long-term sampling missions covering up to now a transect plane (1.5 km wide and 20 m deep). The ASV is equipped with a probe which can be lowered via a winch to measure critical environmental parameters allowing more sophisticated prediction models of toxic algal blooms. So far, 21 km of autonomous sampling confirmed the system stability and produced enough basin-wide biological data to gain biologists' interest. We intend to use this ASV as basis to further explore autonomous navigation on inland waterbodies.**

I. Introduction

A. Biological Background

The monitoring of water resources is of increasing importance as Earth's reservoirs of clean, potable water are drained by the growing human population, rapid economic growth, and environmental degradation [1]. One emerging threat to freshwater ecosystems is the growing incidence of mass proliferation (bloom) of toxic cyanobacteria (blue-green algae) [2] caused by rising rates of anthropogenic nutrient inputs [3]. The escalating occurrence of toxic blooms in freshwaters will likely be intensified due to the global increase of air temperatures [4].

For many lakes however, the basin-wide variability of these toxic algae at a given season and/or throughout the year are still poorly documented. Biologists are actively looking for solutions to increase the spatial and temporal data acquisition to improve the monitoring of those toxic algae, and to expand their understanding of lake ecosystems in general.

B. Automated Data Acquisition

Automated sensing technologies are developing into an increasingly important tool both for water quality monitoring and for research in aquatic microbial ecology [5]. So far, most systems for automated data acquisition in freshwaters are stationary buoys that do not allow for the investigation of horizontal heterogeneity in lakes. Academic institutions around the world are actively developing and deploying Autonomous Underwater Vehicles (AUV) and Autonomous Surface Vessels (ASV), but these are devoted to extended observation networks in coastal and marine environments [6]. So far, few studies have applied this technology for limnological work (i.e. the study of inland waters), and none of these deployments aimed for long-term missions on the order of months or years [7].

(1) Autonomous Systems Lab. – ETH Zurich, Switzerland firstname.lastname@mavt.ethz.ch
(2) Limnological Station Kilchberg – University of Zurich, Switzerland

Even if oceanography has been leading researches in remote sensing for many years, new technical approaches must be developed to account for the specific needs of lake monitoring. In this article, we present our ongoing development of an ASV suited for limnological studies (see Fig. 1), recent observations following field tests realized over the last year and, finally, future research orientations for our lake platform.

Fig. 1: Lizhbeth, our limnological ASV, during a sampling mission on Lake Zurich. The probe is in its parking position out of the water.

II. State of the Art

The domain of field robotics has recently gained more interest in AUV and ASV. The three most important uses of these platforms are military applications, structure inspections (particularly in the oil industry) and ecological studies. As a result, many universities around the world have developed or are developing their first prototypes. Whereas research in the field of AUVs is more concerned about localization and communication methods, surface vessels usually use GPS for localization. Military and defense applications deploy ASVs to patrol shorelines or harbors. Elkins et al. [8] have developed a ASV that operates on relatively large motor boats and feature a sophisticated set of sensors to detect obstacles or target boats, which can then be followed.

Applications for environment monitoring usually feature electrically driven ASVs, and aim for long-term autonomy because environmental dynamics occur on monthly to yearly time-scales. Some systems (e.g [9]) also employ solar panels to extend their maximal range. Caccia et al. developed an ASV called Sea Surface Autonomous Modular Unit (SESAMO) [10] to analyze the water quality of coastal marine waters in combination with atmospheric parameters. Other groups have also developed ASV for marine environment monitoring ([11], [7]), but the work by Dunbabin et al. [12] relates better to ours since it also uses sensors to measure physicochemical parameters of lakes. However, their equipment is designed for sampling at maximal depths of 5 m only. The maximal depth of 20 m can only be reached in stationary operation.

Given that toxic cyanobacteria can be found as deep as 25 m [13], new technical approaches are needed.

III. Specifications

A. Environment

Planktothrix spp. are among the most important producers of hepatotoxic microcystins in freshwaters [14], and *Planktothrix rubescens* are commonly observed in numerous lakes of the Northern Hemisphere [15]. For these reasons, our ASV is mainly employed to study the spatial and temporal variations of the toxic cyanobacterium *P. rubescens* populations in pre-alpine lakes. Our principal experimental field is Lake Zurich, which has an elongated shape of approximately 36 km long and has a maximal width of 4 km. This large (66.8 km^2) and deep lake (136 m, mean depth = 49.9 m) is subject to the typical seasonal dynamics seen in pre-alpine lakes, i.e. a summer thermal stratification followed by an autumnal mixis. Recurrent blooms of *P. rubescens* observed in Lake Zurich usually occur between late summer and winter. Surface water temperature varies from above 20 °C in summer to around 5 °C in winter. Lake Zurich is a very popular recreational area and is the major source of drinking water for the city of Zurich. The region of interest is a transect of 1.5 km close to the deepest point of the lake.

B. Goals and Objectives

The seasonal dynamics of *P. rubescens* in Lake Zurich have been widely documented over the last decades [13], but measurements have only been taken from vertical profiles at a single point in this large lake. Even though this provides a fair amount of information to describe dynamics with respect to depth and time, it does not allow for conclusions about the horizontal dynamics. Furthermore, the use of such single-point sampling strategy implies that the cyanobacterial population is evenly distributed over the entire lake. Our preliminary observations of Lake Zurich physical parameters suggest that this assumption might not hold.

The main objective of our work is to be able to collect physicochemical data over the transect of interest from the surface to at least 20 m deep. To gather relevant temporal information, especially during period of rapid biological changes such as in the springtime, sampling transects need to be executed once a week over a timespan in the order of months. While sampling, the platform must sail safely in this busy recreational lake that is also used for public transportation via ferries.

C. *The platform* - Lizhbeth

The construction of a surface vehicle that is able to lower a probe at different depths while navigating has been mainly motivated by the simplification of the localization problem. This solution also ensures the visibility of the platform at all time during a mission. The vessel (Fig. 1) has been designed especially for this project and has been optimized with respect to efficiency in forward direction, as well as for pitch and roll stability in the presence of waves. It was manufactured according to these optimized design specifications using a foam core layer strategy. The layers covering the foam core were made from fiberglass to ensure mechanical stability while keeping the weight of the boat as low as possible (130 kg in total, including 40 kg of batteries and 14 kg of motors). Lizhbeth has a catamaran configuration and is 2.5 m long and 1.8 m wide. Each hull is 0.6 m wide and hosts a commercial electrical boat motor (Yamaha M12) at its center. This special setup gives the ASV a differential drive configuration, which allows for rotations on the spot. This is very convenient during maneuvers in narrow passages mainly close to the shore. The hulls contain the necessary electronic equipment and a lead acid battery (12V, 70Ah) each. The option of solar panels has been considered, however, it was calculated that the area that could be used on the boat is not big enough to significantly improve the range of the boat.

A GPS module (UBlox Development Kit) and a digital compass device (HRM3200 from Honeywell) provide position and heading information. The GPS module also provides speed readings, which have proven to be reliable above approximately 0.3 m/s. Differential GPS is not required, as standard GPS readings are precise enough for limnological purposes. The boat features two computers. The Helios Development Kit (by Diamond Systems) is deployed to run hardware drivers as it features 4 RS232 ports, 4 USB ports and also provides analog and digital input and output lines. A second board (pITX-SP 2.5” SBC) featuring an Atom Z510 processor is used for higher level computations. Both devices have low power consumption ($\sim$ 5 W each). Controlling the power levels of the motors is achieved by a commercial motor controller (AX1500 by RoboteQ), which provides an easy-to-use serial interface.

In between the two hulls we installed a custom-designed winch, which allows to lower the commercial limnological sensor (YSI-6600, referred to as probe in this article). The winch it mostly built from aluminum and is driven by an electrical motor (Maxon RE40), which provides maximal force of 15 Nm. The corresponding motor controller (Maxon EPOS 70/10) provides both position and velocity control. The cable drum of the winch is designed to carry 130 m of cable. A special probe cover (see Fig. 2) has been designed to diminish its drag force in water. The tip of the probe is filled with steel to shift the center of mass in front of the fixation point. This shift, in combination with the wings, ensures that the probe is aligned horizontally and thus minimizes the offset of the probe from is desired position vertically below the boat. The wings also stabilize the probe and prevent lateral movements during navigation. The probe works on its own power, therefore the cable is used only for data transmission. To prevent the probe from hitting the ground in shallow regions, a single beam sonar sensor (TMD-1 by CruzPro) is mounted on the boat and provides measurements of the water depth. The range of the sonar (0 - 140 m) covers the maximal length of the cable.

Additionally the boat features a positioning light and a electrical horn. Despite the extensive system monitoring tools from a laptop via the wireless connection, these tools proved to be very useful during field tests for simple feedback to the user. A series of short horn blows, for instance, is used to report the successful completion of a task.

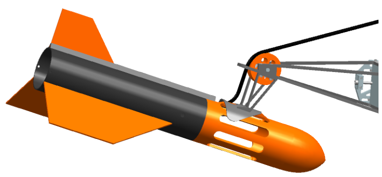


Fig. 2: CAD rendering of the probe supporting structure which reduces its drag.

The software for the boat is based on ROS [16] and features different control modes, such as staying at a given position or following line segments [17] at constant speed. Furthermore, the boat can be controlled remotely or can be asked to follow predefined waypoints. The two main actuators (the propulsion system and the winch) can be commanded to perform tasks either independently or in a synchronized manner. Synchronization is required to achieve the zigzag sampling trajectory of the probe, which will be described in detail below. In order to plan missions that define a sequence of actions, we have developed a mission scripting environment in Python, which enables to use convenient programming techniques. Trajectories for mission paths can be generated in Google Earth by simply drawing lines in order to be accessible to biologists in a near future. Furthermore the boat provides easy access to system monitoring information via a web server, which can also be used to upload mission files. A mission management system detects newly uploaded mission files and executes them. During the execution of a mission the user can pause it and use the remote control to override actuator commands.

D. Sampling missions

The first sampling missions have been chosen to be carried out along predefined straight lines (or transect) for reasons of simplicity and repeatability. While the boat is traveling along such a line with constant speed of 0.7 m/s the probe is being lowered and pulled up between two predefined depth levels. This generates a zigzag trajectory of the probe along which measurements are taken at a constant frequency of 0.5 Hz. The probe measures the following parameters: pressure, temperature, relative fluorescence unit (RFU) of phycoerythrin (a pigment of *P. rubescens*), dissolved oxygen, conductivity, photosynthetic available radiation and chlorophyll fluorescence. Fig. 3 depicts such a trajectory. In a post-processing step, a 2-D interpolation procedure can be applied to the data, as all measurements can be assumed to lie within a vertical transect plane. To account for the different scales in vertical and horizontal directions, an anisotropic distance kernel function has been applied.

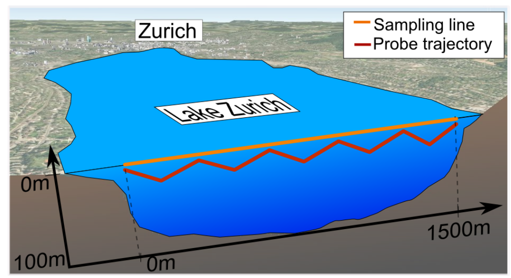


Fig. 3: Schematic representation of the resulting trajectory of the probe during a sampling mission. The total length of the path in horizontal direction is approximately 1.5 km. The map was taken from Google Earth.

IV. Results

A. System dynamics and control

The ASV possesses essential characteristics, as it has proven to be very stable in the presence of waves. It also generates very little drag in forward direction. Nonetheless, only very small rotational speeds can be achieved, due to the fact that the boat has two hulls. When rotating on the spot, the maximal angular velocity has been measured at 4.2 °/s. The line following controller has been applied in multiple test runs and has shown to be reliable, even in the presence of currents and strong winds. Over a distance of 1.5 km a lateral deviation from the target line of 0.67 m in average with a standard deviation of 0.66 m has been measured. Up to now the system has been running in autonomous waypoint navigation mode for more than 21 km in total and also a large total distance in remote controlled mode. Its endurance was found to be around 3 hours of continuous motion. This corresponds to sampling at 0.7 m/s over a distance of approximately 6 km.

B. Data collection

The interpolated plots in Fig. 4 show the temperature distribution and the distribution of *P. rubescens* (assessed from the RFU measurements) along the sampling line across Lake Zurich. The horizontal axis indicates the traveled distance along the sampling line, whereas the vertical axis depicts depth. Besides the obvious and expected vertical temperature gradient, these measurements also show that the temperature is not equally distributed over the entire width of the lake. While the lake is warmer on the north-eastern side (right-hand side of the plot), the RFU values indicate that the density of *P. rubescens* is higher on the opposing side (i.e. the colder side) of the lake. To date, a total 21 km of measurements has been recorded to test the stability of the hardware, software and electronic components. To verify assumptions on the spatiotemporal behavior of *P. rubescens*, measurements at higher frequency have to be obtained over a fixed period. Nonetheless, these preliminary results confirm the capabilities of the ASV, and raise the biologists interest for more measurements to solve this intriguing spatial heterogeneity in *P. rubescens* distribution.

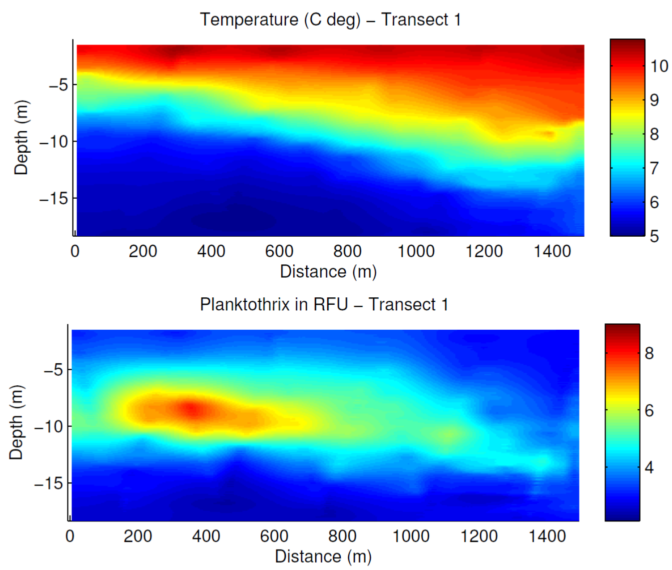


Fig. 4: Plots showing the results of a sampling mission after the application of 2-D interpolation methods. The graphs show the temperature distribution and the presence of *Plankthotrix* (in RFU) respectively.

V. Outlook

The platform that we presented in this paper is ready to be deployed and is capable of taking transect measurements autonomously. We intend to deploy it frequently during summer months in order to generate a large data set. However the system is still blind, i.e. it can not perceive its local environment and can only localize itself with the help of artificial landmarks (GPS references). Especially in summer, Lake Zurich is a popular recreational area and is used both by boats of different sizes and by swimmers. This justifies the need for an obstacle detection and avoidance system. Additionally, more sophisticated methods to collect data in the lake could be applied. Some ideas that will be pursued during the next month are presented in the following.

A. Obstacle detection

As our platform is designed for long-term sampling missions, an energy-efficient solution to detect obstacles should be preferred. A vision-system consisting of a single pan-tilt camera, multiple fixed cameras or an omni-directional camera should be able to operate at low power consumption and to cover a large field of view. This will allow to cover small, close objects (such as buoys or even swimmers) as well as larger boats at distances of several hundred meters. Detecting obstacles on a lake reliably is very difficult to achieve, because weather and lighting conditions can change fast and the water surface can act as a moving mirror. However, it is crucial to improve the ASV's autonomy on Lake Zurich, as it can be dangerous to swimmers.

B. Visual Homing

Preliminary tests have been conducted to test visual homing methods to home the boat into a boat house, where pure GPS localization is not precise enough. Reliably matching previously recorded features is difficult since a camera mounted on the boat is subject to pitch and roll motion. Furthermore, changes in lighting can significantly alter featured distributions and thus, have a negative effect on successful homing procedure. The idea to use normal cameras to apply visual servoing methods is similar.

C. Dynamic sampling

In the first phase of the project, only transect sampling missions will be conducted to gain basic knowledge on the dynamics of the different environmental parameters. Afterwards, dynamic sampling missions can significantly improve the information density of the data representation one mission. By analyzing the measured data in an on-line manner the boat could explore areas of high interest or search for local maxima or minima of certain parameters. The biologists could then be notified about such places of interest. This would allow them to also take water samples from these regions.

In another scenario, the boat could follow a isocline in the parameter space, i.e. try to control the position of the probe such that one parameter remains constant.

D. Bathymetry

Additionally to its primary purpose of sampling for biological research, we would like to deploy our ASV to record bathymetric information. To achieve this we plan to use a multi-beam sonar sensor. In order to fuse the scans of the sonar, we intend to use an IMU in order to retrieve roll and pitch information of the boat. Based on the quality of the scans, we would like to explore the possibility of applying Simultaneous Localization and Mapping Methods (SLAM), which could improve the boats navigation capabilities in narrow regions (such as harbors) where GPS might not provide the required precision. SLAM in combination with bathymetric measurements has been investigated in AUV applications [18] but could be of great support for ASV navigation also.

VI. Conclusion

In this paper we presented a new ASV system and the progress on deploying the platform to execute sampling mission for limnological monitoring tasks. The boat was designed and manufactured to the specifications of these algal bloom monitoring tasks and therefore it is mainly intended to be used on lakes. However, due to its relatively small size (2.5 m by 1.8 m) and its light weight, it is simple to transport it to different testing sites. Also marine environments are possible testing areas, the only limitation is imposed by the strength of external forces (wind and waves) which might exceed the maximal thrust force that the motors can provide. Apart from such hardware related limitations the system is very versatile and has proven to operate in a robust manner in autonomous waypoint navigation. The platform was deployed in several data collection missions during which it covered a total distance of 21 km in autonomous mode. With its ability to sample at a large range of depth in combination with precise GPS localization methods, the ASV provides methods to collect data sets that allow to analyze the horizontal variability of populations of *P. rubescens* in an unprecedented manner.

VII. Acknowledgment

This work is funded by the Swiss National Science Foundation (CR22I2-130023).

References

[1] "World water assessment programme. the united nations world water development report 3: Water in a changing world." Paris: UNESCO, and London: Earthscan, Tech. Rep., 2009, technical report.

[2] I. Stewart, P. Webb, P. Schluter, and G. Shaw, "Recreational and occupational field exposure to freshwater cyanobacteria - a review of anecdotal and case reports, epidemiological studies and the challenges for epidemiologic assessment. environmental health: A global access science source," in *Environmental Health: A Global Access Science Source*, 2006, pp. 5(1):1–13.

[3] J. Heisler, P. Glibert, D. Burkholder, J. Anderson, W. Cochlan, W. Dennison, Q. Dortch, C. Gobler, C. Heil, E. Humphries, A. Lewitus, R. Magnien, H. Marshall, K. Sellner, D. Stockwell, D. Stoecker, and M. Suddleson, "Eutrophication and harmful algal blooms: a scientific consensus," *Harmful Algae*, vol. 8, no. 1, pp. 3–13, 2008.

[4] K. Johnk, J. Huisman, J. Sharples, B. Sommeijer, P. Visser, and J. Stroom, "Summer heatwaves promote blooms of harmful cyanobacteria," *Global Change Biology*, vol. 14, no. 3, pp. 495–512, 2008.

[5] Ashley Shade, Cayelan C Carey, Emily Kara, Stefan Bertilsson, Katherine D McMahon, and Matthew C Smith, "Can the black box be cracked? the augmentation of microbial ecology by high-resolution, automated sensing technologies," in *The ISME Journal*, August 2009, pp. 3(8):881–8.

[6] S. Kröger, E.R. Parker, J.D. Metcalfe, N. Greenwood, R.M. Forster, D.B. Sivyer, and D.J. Pearce, "Sensors for observing ecosystem status," in *Ocean Science*, 2009, pp. 6:765–798.

[7] D.A. Caron, B. Stauffer, S. Moorthi, A. Singh, M. Batalin, E.A. Graham, M. Hansen, W.J. Kaiser, J. Das, A. Pereira, A. Dhariwal, B. Zhang, C. Oberg, and G.S. Sukhatme, "Macro- to fine-scale spatial and temporal distributions and dynamics of phytoplankton and their environmental driving forces in a small montane lake in southern california, usa," in *Limnology and Oceanography*, 2008, pp. 53:2333–2349.

[8] L. Elkins, D. Sellers, and W. R. Monach, "The autonomous maritime navigation (amn) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *Journal of Field Robotics*, vol. 27, no. 6, pp. 790–818, 2010. [Online]. Available: http://dx.doi.org/10.1002/rob.20367

[9] J. Higinbotham, J. Moisan, C. Schirtzinger, M. Linkswiler, J. Yungel, and P. Orton, "Update on the development and testing of a new long duration solar powered autonomous surface vehicle," in *OCEANS 2008*, sept. 2008, pp. 1 –10.

[10] M. Caccia, R. Bono, G. Bruzzone, E. Spirandelli, G. Veruggio, AM Stortini,and G. Capodaglio, "Sampling sea surfaces with sesamo: an autonomous craft for the study of sea-air interactions," in *IEEE Robotics & Automation Magazine*, 2005-2006, pp. 12(3):95–105.

[11] P. Tokekar, D. Bhadauria, A. Studenski, and V. Isler, "A robotic system for monitoring carp in minnesota lakes," *Journal of Field Robotics*, vol. 27, no. 6, pp. 779–789, 2010. [Online]. Available: http://dx.doi.org/10.1002/rob.20364

[12] M. Dunbabin, A. Grinham, and J. Udy, "An autonomous surface vehicle for water quality monitoring," in *Australasian Conference on Robotics and Automation 2009*, 2011.

[13] S. V. den Wyngaert, M. M. Salcher, J. Pernthaler, M. Zeder, and T. Posch, "Quantitative dominance of seasonally persistent filamentous cyanobacteria (Planktothrix rubescens) in the microbial assemblages of a temperate lake," *Limnology and Oceanography*, vol. 56, pp. 97–109, 2011.

[14] R. Kurmayer and M. Gumpenberger, "Diversity of microcystin genotypes among populations of the filamentous cyanobacteria *Planktothrix rubescens* and *Planktothrix agardhii*," *Molecular Ecology*, vol. 15, no. 12, pp. 3849–3861, 2006.

[15] K. Sivonen and G. Jones, "Cyanobacterial toxins," pp. 41–111, 1999.

[16] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *Open-Source Software workshop of the International Conference on Robotics and Automation (ICRA)*, 2009. [Online]. Available: http://www.ros.org

[17] C. Samson and K. Ait-Abderrahim, "Feedback control of a nonholonomic wheeled cart in cartesian space," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, apr 1991, pp. 1136 –1141 vol.2.

[18] S. Barkby, S. B. Williams, O. Pizarro, and M. V. Jakuba, "A featureless approach to efficient bathymetric slam using distributed particle mapping," *Journal of Field Robotics*, vol. 28, no. 1, pp. 19–39, 2011. [Online]. Available: http://dx.doi.org/10.1002/rob.20382

Mapping Greenhouse Gas Emissions on Complex Inland Waterways Using Autonomous Surface Vehicles

Matthew Dunbabin
Autonomous Systems Laboratory
CSIRO ICT Centre
P.O. Box 883, Kenmore, QLD 4069, Australia
matthew.dunbabin@csiro.au

Alistair Grinham
Centre for Water Studies
School of Engineering
University of Queensland
St Lucia, QLD, Australia
a.grinham@uq.edu.au

***Abstract*— Autonomous Surface Vehicles (ASVs) offer a unique opportunity to obtain rich spatially and temporally distributed environmental data sets within ocean and inland water environments. Beneficial aspects include long duration operation, localisation accuracy and ability to carry relatively large sensor payloads. Operational challenges, particularly for inland waterways, relate to the level of allowable autonomy in cluttered environments. Clutter can arise from natural features and obstacles above and below the water line, and introduced factors such as boats, bridges and weirs. This extended abstract provides and overview of an ASV system for the application of quantifying and mapping greenhouse gas emissions to the atmosphere in complex narrow waterways. The system incorporates a spinning laser scanner and sonar systems to generate 3D obstacle and hazard maps in real-time allowing operation in previously unexplored and GPS denied regions of the water storage. Furthermore, a new method of quantifying greenhouse gas emissions via methane ebullition using an Optical Methane Detector has been developed and experimentally evaluated using the CSIRO 16 ft ASV illustrating its ability to continuously map greenhouse gas emission across an entire reservoir containing many navigation hazards.**

I. Introduction

Quantification of greenhouse gas emissions to atmosphere is becoming an increasingly important requirement for scientists and managers to understand their total carbon footprint. Methane in particular is a powerful greenhouse gas, approximately 21 times higher global warming potential than carbon dioxide. Water storages are known emitters of methane to atmosphere [1]. The spatio-temporal variation of release is dependent on many environmental and biogeochemical parameters. Therefore, in order to accurately quantify this greenhouse gas release requires long duration and repeat monitoring of the entire water body. This is where robotics can play a significant role.

There are limited examples of ASVs designed for long-duration, large-scale unsupervised environmental monitoring [2]–[4]. These systems have primarily been designed for oceanographic surveys and are not particularly suitable for relatively unexplored inland waterways with complex and often varying navigational requirements. An emerging research area for Autonomous Surface Vehicles is that of mobile adaptive sampling where the ASV can alter its trajectory to improve measurement resolution in space and time (e.g [5]). These studies have demonstrated the ability to capture and track various parameter distributions over relatively small areas, however, they do not consider obstacle avoidance, physically changing environments or large-scale survey requirements. A study by Ferreira et al [6] has demonstrated the use of an ASV for mapping hazards below the waterline, however, these maps are not used in real-time to guide the vehicle for exploration and other parameter (e.g. scientific) measurements.

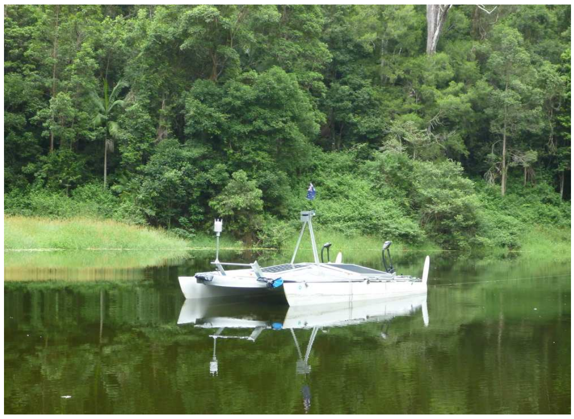


Fig. 1. The solar-powered ASV on Little Nerang Dam during methane quantification experiments.

This work addresses a need to improve the spatio-temporal resolution of data collected on inland water storages and lakes with the development of a unique Autonomous Surface Vehicle (ASV) (see Figure 1). A key requirement in our application is the ability to navigate and explore narrow, complex and dynamic waterways unsupervised whilst taking precision measurements of water quality and greenhouse gas emissions.

II. 3D Laser Scanner for Obstacle Detection

Prior to this work, the ASV shown in Figure 1 used a single forward facing laser range scanner to detect the presence of obstacles at a distance of approximately 7-15 m in front of the vehicle. While this has been successful in avoiding small obstacles such as buoys it is not considered sufficient for navigating complex environments such as narrow rivers where higher resolution 360^o coverage is required.

Based on previous work with rotating scanning laser scanners [7], a new scanning system based on a Hokuyo URG scanner was developed (shown in Figure 2). The laser is inclined 15^o and spins around a vertical axis giving a maximum sensing volume (when mounted on the existing laser mast) relative to the ASV as shown in Figure 3.

Fig. 2. The prototype rotating 3D laser system as configured for operation on the ASV.

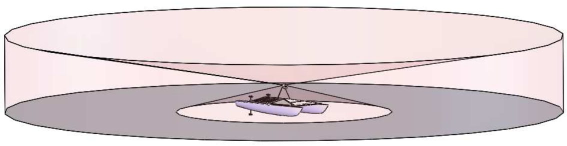


Fig. 3. Sensing volume of the rotating scanner scaled to the size of the ASV.

The laser rotates at approximately 180 degrees per second, and with a survey speed of 1 ms^{-1}, this provides the ASV with a complete update of its obstacle map every 2 m of travel.

III. Environmental sensing

At the highest level, missions are specified as a series of waypoints and segment velocities with functionality tags (such as profile, station keep, dock). The vehicle attempts to maintain a straight path between successive waypoints, however, this can be modified with the detection of obstacles and shallow non-traversable water.

A typical sensor payload for the ASV consists of an Optical Methane Detector (OMD, Heath Consultants, Texas), YSI Sonde (measuring temperature, conductivity, chlorophyll, turbidity, dissolved oxygen), wind sensor and a profiling sonar. Details of the ASV systems and further functionality are described in [8].

The final paper will detail the methodology of producing whole-of-reservoir methane efflux estimates using the measured OMD data.

IV. Experimental Results

The ASV has undergone extensive field trials to evaluate the tracking and obstacle avoidance performance in a variety of weather conditions and operational scenarios as described in [9]. Environmental monitoring trials were conducted on Little Nerang Dam (south of Brisbane, Australia) in which the CSIRO ASV was used to autonomously and repeatably map environmental parameters, in particular greenhouse gas emission release across the entire 49 hectare water reservoir.

Figure 4 shows a example result of the data produced by the 3D spinning laser sensor for obstacle avoidance over one full revolution of the sensor. This example shows both sides of the ravine wall and a floating obstacle (an instrument for detecting methane). Since the laser beam is absorbed by water, there are no range measurements to the water surface which appears as clear space.

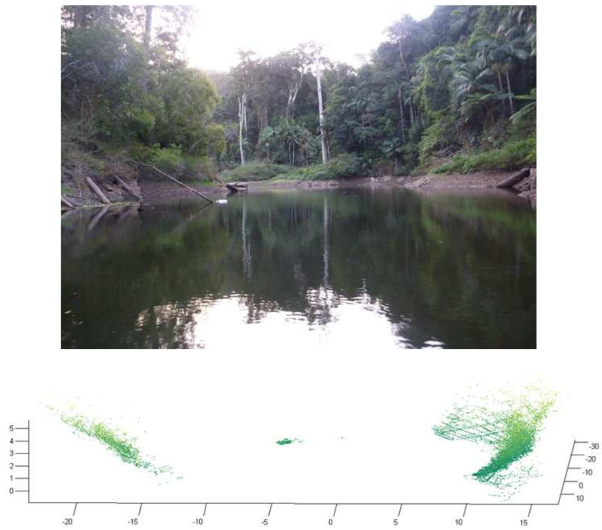


Fig. 4. Ravine at Little Nerang Dam (taken prior to experiments). The white box is a floating methane detector. (bottom) Two seconds of scan data from the same location showing the detection of both the ravine walls and the methane detector which is now located in the centre of the ravine.

In addition to the navigational sensors used to detect obstacles and allow traversal into the southern distal arms, the ASV was fitted with an Optical Methane Detector (OMD) to measure surface methane concentration levels. A high level path was presented to the ASV to traverse the entire storage. This path was executed whilst being modified in real-time based on the hazard maps generated to avoid running aground against the side of the storage or the southern shallow arms and to avoid colliding with obstacles such as logs. This was achieved despite the extremely variable wind conditions with gusts approaching 70 km h^{-1}. The ability

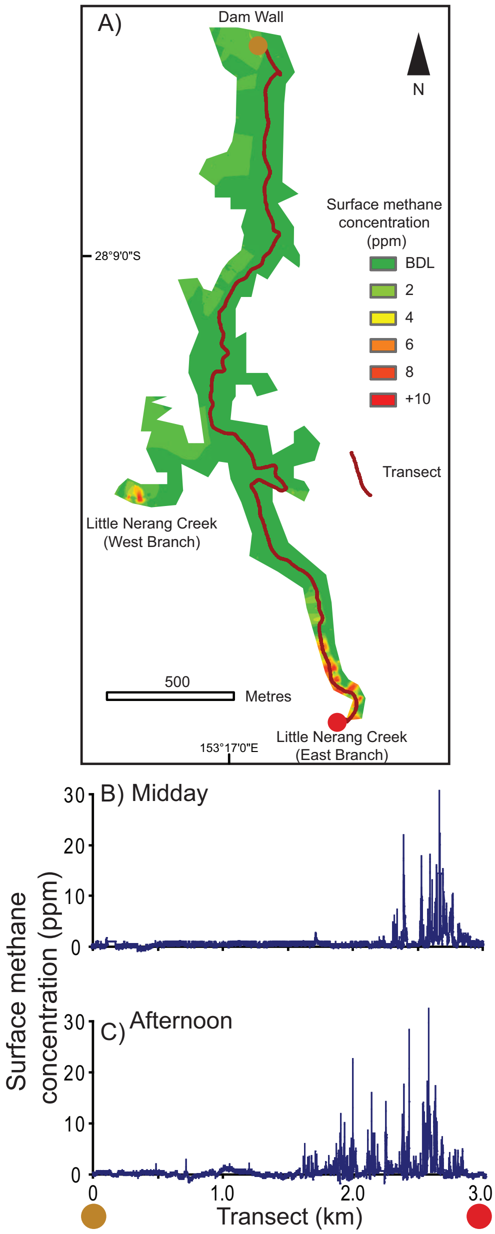


Fig. 5. A measured atmospheric methane distribution graph and variation along the same track with time of day using the Optical Methane Detector on the ASV for Little Nerrang Dam in Queensland Australia.

to generate spatio-temporal maps of greenhouse gas emissions using the ASV were evaluated on Little Nerang Dam. Figure 5(a) shows a measured surface methane concentration distribution using the OMD attached to the ASV. Figures 5(b) and (c) show the measured methane concentration along a repeat ASV transect (shown in Figure 5(a)) during a day and night survey illustrating a spatial-temporal variation of methane flux.

V. Conclusions

This paper provides and overview of a novel Autonomous Surface Vehicle used to quantify greenhouse gas release to atmosphere across an entire water storage. The ASV incorporates a 3D spinning laser scanning system to allow obstacle detection and avoidance in previously unmapped shallow water environments. A method to quantify methane ebullition has been developed and validated experimentally illustrating the ASV's its ability to continuously map the spatio-temporal greenhouse gas emissions across an entire reservoir containing many navigation hazards.

Acknowledgements

This work was jointly funded by Seqwater, the CSIRO Sensor & Sensor Network Transformational Capability Platform and Water for a Healthy Country National Research Flagship. The authors would like to thank David Prasser, Brett Wood, Les Overs and John Whitham from the CSIRO ICT Centre Autonomous Systems Laboratory, and Deborah Gale from Seqwater for their contributions during construction and field evaluation of the system.

References

[1] V. S. Louis, C. Kelly, E. Duchemin, J. Rudd, and D. Rosenberg, "Reservoir surfaces as sources of greenhouse gases to the atmosphere: A global estimate," *Bioscience*, vol. 50, pp. 766–775, 2000.

[2] J. Higinbotham, J. Moisan, C. Schirtzinger, M. Linkswiler, J. Yungel, and P. Orton, "Update on the development and testing of a new long duration solar powered autonomous surface vehicle," in *Proc. IEEE OCEANS 2008*, September 2008, pp. 1–10.

[3] P. Rynne and K. von Ellenrider, "A wind and solar-powered autonomous surface vehicle for sea surface measurements," in *Proc. IEEE OCEANS 2008*, September 2008, pp. 1–6.

[4] J. Wang, W. Gu, and J. Zhu, "Design of an autonomous surface vehicle used for marine environmental monitoring," in *Proc. International Conference on Advanced Computer Control (ICACC09)*, January 2008, pp. 405–409.

[5] B. Zhang and G. Sukhatme, "Adaptive sampling for estimating a scalar field using robotic boat and a sensor network," in *Proc. International Conference on Robotics and Automation*, April 2007, pp. 3673–3680.

[6] C. Ferreira, A. Almeida, J. Martins, N. Almeida, E. Dias, and E. Silva, "Autonomous bathymetry for risk assessment with ROAZ robotic surface vehicle," in *Proc. Maritime Systems and Technology Americas 2010*, 2010.

[7] M. Bosse and R. Zlot, "Continuous 3d scan-matching with a spinning 2d laser," in *Proc. International Conference on Robotics and Automation*, 2009, pp. 4312–4319.

[8] M. Dunbabin, A. Grinham, and J. Udy, "An autonomous surface vehicle for water quality monitoring," in *Proc. Australasian Conference on Robotics and Automation*, December 2009.

[9] M. Dunbabin and A. Grinham, "Experimental evaluation of an autonomous surface vehicle for water quality and greenhouse gas monitoring," in *Proc. International Conference on Robotics and Automation*, May 2010, pp. 5268–5274.

Aquatic and Land-based Robotic Telemetry for Tracking Invasive Fish

Elliot Branson, Volkan Isler, Pratap Tokekar and Josh Vander Hook

***Abstract*— Carp is a highly invasive, bottom-feeding fish which pollutes and dominates lakes by releasing harmful nutrients. Recently, environmental scientists started studying carp behavior by tagging the fish with radio-emitters. The radio-tagged fish are tracked manually using GPS and a directional antenna. We have been working on developing a novel robotic sensor system in which the human effort is replaced by autonomous robots to find and track carp. During the summer months, we use robotic boats whereas in the winter, mobile robots track the fish on frozen lakes.**

In this extended abstract, we report the current state of our system including system architecture, coverage and active tracking algorithms. We also present results from field experiments including coverage experiments in which our boat travels 2.4 km in Lake Keller in Minnesota.

I. INTRODUCTION

Invasive fish such as the common carp pose a major threat to the ecological integrity freshwater ecosystems around the world. Presently, the only way to control these fish is through the use of non-specific toxins which are expensive, ecologically damaging, and impractical in large rivers and lakes. Recent studies in small lakes have established that some of these fishes aggregate densely at certain times and places and can be controlled by targeting these aggregations using netting. Therefore, biologists started using a new technology based on tracking radio-tagged carp to accurately predict the presence of large carp populations.

Unfortunately, carp aggregations are unpredictable. Manually locating tagged fish in large, turbid bodies of water remains a difficult task. Our goal is to replace this manual effort with robots. Toward this goal, we developed an autonomous robotic boat (Figure 1(a)) capable of localizing tagged fish in lakes and a field robot (Figure 1(b)) performing the same task on frozen lakes.

In this work, we build on our previous system [1] and present the following improvements: (1) **Coverage:** We have recently developed a new coverage algorithm to detect the presence of fish. The algorithm takes regions that are likely to contain fish as input and computes a path to cover these regions. This allows for incorporating scientists' domain knowledge. (2) **Active Localization:** After detecting the fish, the goal is to accurately estimate its location. We use multiple measurements taken at various locations for estimation. The problem we address is how to choose measurement locations in an online fashion so as to accurately localize the fish with a small number of measurements. We report recent results on active localization.

We also report results from field experiments for both the problems. We begin with the coverage problem.

(a) Robotic boat at during coverage experiments at Lake Keller, MN.

(b) Robot (with tracking equipment and antenna) on frozen Lake Casey, MN.

Fig. 1. Robotic system for monitoring radio-tagged carp during field trials.

II. SEARCH AND COVERAGE

In this section, we present our coverage algorithm for finding fish. We say that location x on the lake is *covered* if the boat moves to a location y from where the tag on the fish when located at x can be heard. While the fish move significantly throughout the day, they are expected to remain within a certain area for shorter periods of time. If we assume that a fish is approximately stationary during the search phase, the searching task reduces to a coverage task: find the shortest trajectory which ensures that all possible locations of the fish are covered.

We can speed-up the coverage task by incorporating domain knowledge. Suppose we are given a set of regions which are likely to contain the fish. For example, these can be areas rich in vegetation. We assume that these regions are connected in the sense that there is a path between any two points. We model the *regional contiguity* property as follows:

When the robot visits a region, it must cover it completely before visiting another region.

With the regional contiguity requirement, the coverage problem can be defined as follows: Given a set of connected regions $R = \{R_1, R_2, \ldots, R_n\}$, find a minimum length tour with the regional contiguity property which covers every point in each region $R_i \in R$.

We propose an approach composed of two steps: First, we compute an α approximation tour τ_R that visits all the regions in R. We say that region R_i is *visited* if *any* point in R_i is visited by the tour. The tour, τ_R, imposes an ordering on the regions. Next, we compute a β approximation coverage

The work presented in this extended abstract is being considered for publication in IROS 2011. This work is supported by NSF Awards #0916209, #0917676, #0936710 and a fellowship from the Institute on the Environment at the University of Minnesota. The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA. {elliot,isler,tokekar,jvander}@cs.umn.edu

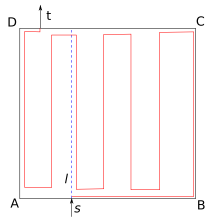


Fig. 2. Covering a rectangle with given entry and exit points.

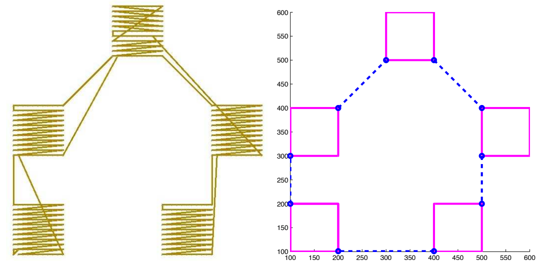


Fig. 3. Complete TSP path. Using the same regions the TSPN graphs show the entry and exit points for the region along with the the paths between the regions. The coverage paths are not shown for clarity.

tour C_{R_i} for each region $R_i \in R$ independently. The final tour τ is constructed by adding the coverage tours of each region to τ_R. We prove that imposing regional contiguity costs at most a factor $(\alpha+\beta)$ deviation from the unrestricted optimal solution.

We now present algorithms for the two components of the algorithm: Computing a tour that visits the regions and covering the regions.

A. Visiting the regions: TSPN and the Zookeeper Problems

The computation of the tour τ_R depends on the geometric properties of the regions. If the regions are convex polygons touching the boundary of a (simply-connected) lake then the tour can be computed optimally by computing the so-called zookeeper's route [2]. In this case $\alpha = 1$. If the regions are arbitrarily placed, we can use algorithms for TSP with neighborhoods (TSPN) such as [3].

Most geometric instances of the TSPN problem are NP-Hard. In our application, it is reasonable to model the lake as a simply-connected region. Further, areas of interest where the fish may lie are usually close to the shore because of vegetation and oxygen levels. This special instance of TSPN known as the zoo-keeper problem can be solved in polynomial time due to the following lemma.

Lemma 1 ([2]): Let $R = \{R_1, R_2, \ldots, R_i, \ldots, R_n\}$ be a set of convex regions located along the perimeter of a simply connected polygon P. There exists an optimal solution for visiting the regions in R which visits them in the order they appear along the boundary of P.

Once the ordering of the regions is known, the shortest tour visiting all regions can be calculated which yields entry and exit points for each region. To turn these tours into coverage paths, we need a way to cover a region with given entry and exit points. The computation of these paths is presented next.

B. Coverage

In the second step of our algorithm, a coverage tour for each region is computed. In our application, we represent regions with rectangles with arbitrary orientations since they are easy to specify on one hand and general enough for practical purposes on the other.

The algorithm presented in Section II-A generates an entry and exit point for each region. Our coverage problem is to travel through every vertex in a given rectangular graph with a given starting and ending point. The following lemma shows that we can cover the entire rectangle efficiently even with this constraint.

	TSP	TPSN	Coverage	Total
Environment 1	8,585	500	4,970	5,470
Environment 2	11,361	600	5,940	6,540
Environment 3	2,522.8	302.2	1,450	1,752.2
Environment 4	3,278.1	371.44	1,825	2,196.4
Environment 5 (Fig. 3)	11,856	682.84	4,950	5,632.8

TABLE I

A COMPARISON OF THE TSP PATH VS. COMBINED TSPN/COVERAGE PATH FOR DIFFERENT INPUT SETUPS.

Lemma 2: Let R be a rectangle with a grid imposed on top. Let s and t be two grid points on the boundary specified as entry and exit points. There exists a tour T which starts at s, visits every grid point and exits at t such that the length of T is at most twice the optimal tour which visits every point.

Proof: Given start and end points s and t respectively, we construct a coverage path as shown in Figure 2. This path consists of three parts: an optimal part that covers the rectangle with length equal to that of the OPT, and parts that connect s and t to the start and end points of this optimal part. We can prove that the two connecting parts are non-overlapping and giving us a 2-approximation. The details of the proof are deferred for the full paper. ■

We also show that this analysis is tight: there are instances where we cover the region twice when s and t are fixed.

We now evaluate our proposed algorithms through simulations and field experiments for covering the lake.

C. Simulations and Field Experiments

We first compare the performance of our algorithms with the standard TSP solution. The TSP solution uses all grid points to be covered independent of the regions. For computing the TSP solution, we use the heuristic by Christofides [4] which yields a 3/2-approximation. We ran the two algorithms for the environments given in Figure 3. The results are reported in Table I, whose first column is the length of the TSP tour and the last column is the length of our solution.

As these results show, in addition to enforcing regional contiguity, our algorithm is more efficient than the Christofides heuristic in these instances. It seems that the matching component of the Christofides heuristic sometimes yields long tours. For example, in Figure 3, the TSP path is almost twice as long as our solution.

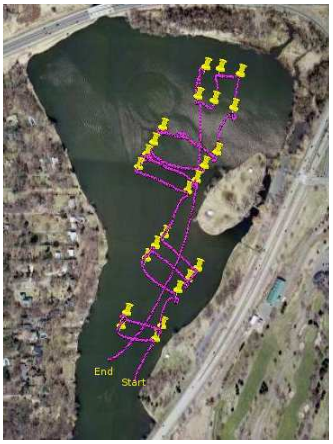


Fig. 4. The GPS trace of the path taken by the boat during the experiment. The trails shows that the boat covered all four regions by visiting all the waypoints robustly. Also the trace suggest that the navigation algorithm negotiated well with the drift caused by wind. The boat traveled approximately 2.5 km in 36 minutes of the run.

We conducted field experiments in Lake Keller, Maplewood, MN to test the coverage algorithm and the navigation performance of the system. The size of the lake is approximately $900m \times 350m$.

We fixed four regions of interest in the lake. The dimension of the four regions are approximately $71m \times 23m$, $100m \times 70m$, $118m \times 100m$ and $93m \times 83m$ with a total area of $28,150m^2$. During the experiment the boat traveled approximately 2.5 kilometers in 36 minutes until all the waypoints were covered Figure 4 shows the boat's path.

From the experiment we conclude that the coverage algorithm proposed in this paper is useful for real applications. The experiment also demonstrates that our robotic system is capable of robustly navigating using waypoints for long periods of time.

Using the above algorithm, we can efficiently search the lake for stationary fish. However, the radio antenna has a large range (about $30m$) and we get only coarse estimate for the fish. To accurately localize the fish, we must combine multiple measurements. The following section proposes three strategies to obtain these measurement locations so that the resulting uncertainty in fish is minimized.

III. Active Localization

We first describe how we obtain bearing measurements from the radio antenna. Then, we propose three active localization strategies followed by their evaluation through simulations and field experiments.

A. Measurement Model

The radio antenna used to detect the tag is direction sensitive: the signal strength output from the antenna depends on the relative angle between antenna and the tag. Hence, we take a coarse sampling of signal strength by rotating the antenna in steps of $15°$. We can then fit sine waves and third-degree polynomials using least squares, estimate the maxima accordingly (Figure 5).

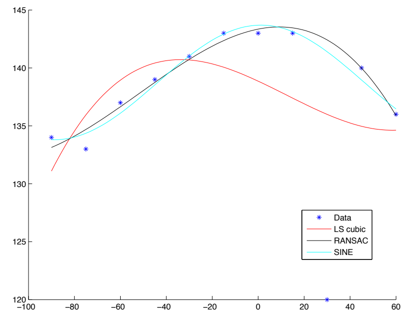


Fig. 5. A coarse sampling (signal strength versus bearing) and various least-squares fitting. RANSAC estimation of a cubic polynomial typically provided the best estimates. The true bearing is $15°$.

This maxima gives us a bearing measurement towards the target. Our objective is to estimate the location (x_f, y_f) of the fish using these bearing measurements. We use an Extended Kalman Filter with the combined robot and target state $X(t) = (x_r, y_r, \theta_r, x_f, y_f)$ to be estimated. The onboard GPS and compass measurements are used to perform EKF updates for the robot state, while bearing measurements are used to update the entire state. Since the bearing measurement function is a non-linear equation, we linearize the measurement about the current state estimate. The resulting uncertainty (as show by the determinant of state covariance) depends on the locations from where the measurements were obtained. Hence, we can optimize these measurement locations to minimize the final uncertainty.

The underlying telemetry technology used by the fisheries researchers introduces another constraint: each tag emits a signal at a dedicated frequency once every second. Since each bearing measurement requires sampling the antenna in multiple directions, we restrict the total measurements to k discrete locations as opposed to obtaining continuous measurements.

For the discussion that follows next, we assume that the initial fish location and covariance estimates are known, propose three strategies for optimization and evaluate them with simulations and field experiments, then discuss the initialization procedure at the end of the section.

B. Active Localization

1) Cramer-Rao Lower Bound: The Cramer-Rao lower bound for an unbiased estimator $\hat{X}$ of state X is a lower bound on the estimation error covariance matrix P_k and is given as the inverse of the Fisher Information Matrix (FIM) I. For k bearing measurements with zero-mean Gaussian noise, determinant of I is inversely proportional to the square of the area of the 1-σ uncertainty ellipse and can be expressed as,

$$|I| = \frac{1}{\sigma^4} \sum_{i=1}^{k} \sum_{j=1}^{k} \left[\frac{\sin(\theta_i - \theta_j)}{d_i d_j} \right]^2 . \qquad (1)$$

where $\Delta x_i = (x_r(i) - x_t)$, $\Delta y_i = (y_r(i) - y_t)$, and $d_i^2 = \Delta x_i^2 + \Delta y_i^2$. Here, $(x_r(i), y_r(i))$ is the location of the robot for the i^{th} measurement, and (x_t, y_t) is the true target location.

To compute the k locations, we impose a grid about the current position of the robot of size $n \times n$. The total number of candidate points for measurement locations are n^2. Hence, to compute the k measurement locations, we consider each of the $C(n^2, k)$ combinations as a candidate trajectory and compute the FIM given by 1.

2) Greedy: Instead of computing a fixed path for the k measurements, we can instead use a greedy strategy which picks the next measurement location based on the current estimate and uncertainty of the target. Given the current robot position and target position, Greedy looks at all neighboring locations of the robot. At every location, we simulate all candidate measurements (e.g. by uniformly picking s samples between 0 to 360°). Using the current state and covariance, we can estimate the posterior covariance by simulating an EKF update using each of these candidate measurements. Thus, for every neighboring location, we will have s posterior covariances. Greedy then picks the candidate location where the maximum determinant of the s posteriors is minimum. This ensures best worst-case uncertainty for the target's position in a greedy fashion. Instead of the best worst-case uncertainty, we can choose some other heuristic for the greedy.

3) Enumeration tree: We extend the objective function of Greedy here, to minimize the worst-case uncertainty obtained by the EKF after k measurements. We use a min-max tree to achieve this objective.

The tree is built by assigning each adjacent measurement location to an action node, and the corresponding measurements to bearing nodes. We recursively define the uncertainty of the actions and bearings and build the tree to depth $2k$, which would correspond to the k desired measurement locations and k measurements. Each bearing node holds a worst-case estimate of the measurement uncertainty, as calculated by the EKF propagation.

Since we use discrete measurement samples while building the tree, we need to find that child node which is closest to the current measurement. As there is some uncertainty associated with the position of the robot itself, we instead use the *Bhattacharya Distance* to find that child node, whose posterior covariance is closest to the current robot covariance (after the measurement update). The robot then repeats the above steps until it reaches the leaf nodes (corresponding to the k^{th} measurement location).

In each of the three strategies proposed above, we assume initial estimates for the fish position and covariance were known. We use the following lemma to pick the first two bearing measurement locations before beginning the strategies.

Lemma 3: Let r_{min} and r_{max} be the minimum and maximum sensing range of the sensor. Assume w.l.o.g the first measurement taken from the origin is along the X-axis. Then, if the second measurement is taken from $(\frac{r_{max}+r_{min}}{2}, \pm\frac{r_{max}-r_{min}}{2})$, the worst-case uncertainty in the target's position after two measurements is minimized.

C. Simulations and Experiments

We ran 100 random trials for each strategy using the same initial conditions, target locations, and random seed to generate measurement noise. The result of the simulations is presented in Table II, and the corresponding histograms of final error and determinant of the final covariance matrix are shown in Figures 6(a) and 6(b) respectively. The outliers with large error resulted from poor initial estimates.

TABLE II
SIMULATION RESULTS FOR 100 TRIALS

Method	Mean final error	Mean final uncertainty
Enumeration tree	$5.7275m$	48.36
Greedy	$5.9809m$	40.59
FIM	$6.2975m$	54.81

The two best closed-loop (online) strategies, Enumeration tree and Greedy were then evaluated in field experiments using the Husky and tracking equipment (Figure 1(b)). Two results are shown in Figures 6(c) and 6(d). In both plots the robot's mean estimated positions are labeled by green circles, while estimates of fish locations are blue marks. The rest of the results are presented in Table III. Similar to the simulation results, we can see that the Enumeration tree performs better than the Greedy strategy.

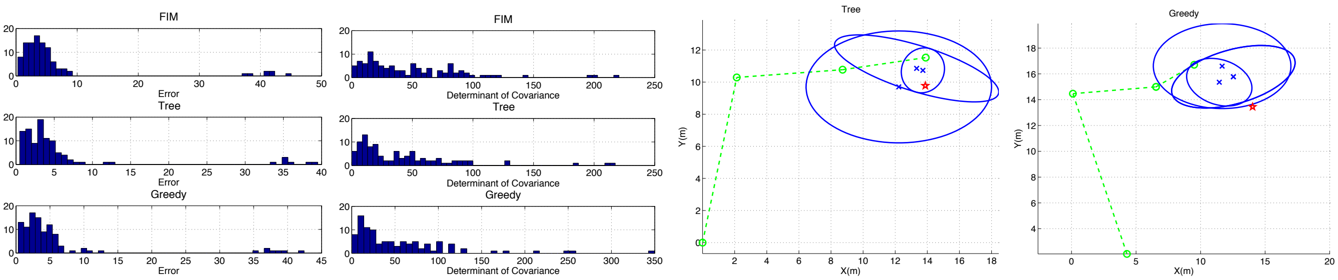
TABLE III
EXPERIMENTAL RESULTS WITH DEPTH 2

Method	Final error	Final uncertainty
Enumeration Tree	0.97	3.53
	3.32	8.57
	5.35	6.04
Greedy	3.21	20.52
	3.29	11.93
	8.65	11.34

From the results we observe that both the mean final error and final uncertainty (determinant) is better for the Enumeration Tree, where as the FIM strategy performs the worst of the three. This result is not surprising, for two main reasons: (1) Since the true target location is unknown, we compute the FIM using the initial estimate of the target's location. (2) The FIM strategy computes locations which minimize the lower bound on the final uncertainty of an "efficient estimator". Since the Extended Kalman Filter is not an efficient filter, there is no guarantee that it would achieve this lower bound. On the other hand, the Enumeration tree and the Greedy actually compute the covariance of the EKF estimator and pick the location which would minimize its determinant.

IV. CONCLUSION

In this paper, we focused on a novel application in which a robotic boat equipped with a directional antenna searches for radio-tagged invasive fish and picks measurement locations to precisely localize the fish. We presented a new coverage



(a) Final errors.

(b) Determinant of the final covariance.

(c) Enumeration tree experiment.

(d) Greedy experiment.

Fig. 6. **Simulations (a) & (b):** We conducted 100 trials with $k = 3$ for each strategy. The mean final error for FIM, Enumeration tree and Greedy was $6.30m$, $5.73m$ and $5.98m$ respectively, and determinant of final covariance was 48.36, 40.59 and 54.81 respectively. **Experiments (c) & (d):** .The true location of the tag is marked by a star. The initial estimate along with estimates after first and second measurement is shown with 1-σ bounds in blue. The measurement locations are shown in green.

algorithm with regional contiguity properties. After presenting theoretical results on the performance of the algorithm, we compared it with a standard TSP solution. In field experiments, we showed that the boat can cover large areas efficiently using our algorithm. For the localization problem, we proposed three strategies, compared them in simulations and reported results from field experiments which show that our system is capable of localizing the target within a meter of the true location.

There are a number of directions we have identified for future work. Since the fish move very little for long periods of time in the winter, in our algorithms we make the assumption that the target is stationary. To handle violations of this assumption, we are working on strategies to localize moving fish. We are also planning to acquire additional boats. Having multiple robots is useful because robots can localize the fish as well as each other more accurately. New algorithms for multi-robot coordination are being developed and will be tested on the field.

V. ACKNOWLEDGMENT

We thank the members of the Sorensen lab at the Department of Fisheries at the University of Minnesota for many useful discussions and sharing equipment. The Husky Robot is provided by ClearPath Robotics through their PartnerBot program.

REFERENCES

[1] P. Tokekar, D. Bhadauria, A. Studenski, and V. Isler, "A robotic system for monitoring carp in minnesota lakes," *Journal of Field Robotics*, vol. 27, no. 6, pp. 779–789, 2010.

[2] W. Chin and S. Ntafos, "The zookeeper route problem," *Information Sciences: an International Journal*, vol. 63, no. 3, pp. 245–259, 1992.

[3] J. S. Mitchell, "A constant-factor approximation algorithm for tsp with pairwise-disjoint connected neighborhoods in the plane," in *SoCG '10: Proceedings of the 2010 annual symposium on Computational geometry*. New York, NY, USA: ACM, 2010, pp. 183–191.

[4] N. Christofides, "Worst-Case Analysis of a New Heuristic for the Travelling Salesman Problem." CARNEGIE-MELLON UNIV PITTSBURGH PA MANAGEMENT SCIENCES RESEARCH GROUP, Tech. Rep., 1976.

Long-Life High-Energy Fuel Cell Power for Robots and Sensor Networks in Environmental Monitoring

Jekanthan Thangavelautham
Mechanical Engineering Department
Massachusetts Institute of Technology
Cambridge, MA
jekan@mit.edu

Steven Dubowsky
Mechanical Engineering Department
Massachusetts Institute of Technology
Cambridge, MA
dubowsky@mit.edu

Daniel Strawser
Mechanical Engineering Department
Massachusetts Institute of Technology
Cambridge, MA
dstrawse@mit.edu

Keywords: power supply, fuel cells, hybrid system, hydrogen storage, robots, sensor networks.

I. INTRODUCTION

Small mobile robots and sensor networks can perform many important tasks in forestry, farming, climate change monitoring and disaster management [1], [2]. These devices may need to work unattended and continuously for months and years requiring high energy in remote field environments. These power supplies may be subject to design constraints such as mass and volume limits.

High energy internal combustion engines have been used to power mobile robots but their exhaust, noise and strong thermal signatures make them inappropriate for many applications. Solar panels are generally not applicable to either field robots (except space) or sensor networks. Hence batteries are the power system of choice. Current batteries can provide relatively high power for short periods of time; however they have low energy densities. For applications that use current rechargeable batteries, the system needs to stop and recharge every few hours, making them ineffective for continuous, long duration missions. New and effective power supplies with high energy densities are required for long duration missions.

Current research into batteries has the potential to increase energy densities [3]. Fuel cells are a promising alternative to batteries and are the subject of this work. They are electrochemical energy conversion devices that convert chemical energy directly into electricity. They have two major advantages, high operating efficiencies and high energy densities. Here the fuel cell power supply presented has a cell operating efficiency of 65 % and theoretical fuel energy density of 4950 Wh/kg (40 folds higher than lithium ion batteries).

Fuel cells have been suggested in the past and in some cases tested for robotics systems and sensor network modules. However they have not been practically applied (See Figure 1) [7], [8], [9], [10]. In spite of their advantages, fuel cells face four fundamental challenges that limit their widespread use. These are:

- Fuel cells have limited life, because they are delicate and degrade under field conditions [11], [12], [13].
- Fuel cells typically have low power densities [4], [5].
- Fuel cells are complex, costly and not ruggedized for field use [4], [5], [12].
- Conventional storage of hydrogen fuel (required for the most common types of fuel cells) is bulky and inefficient [14].

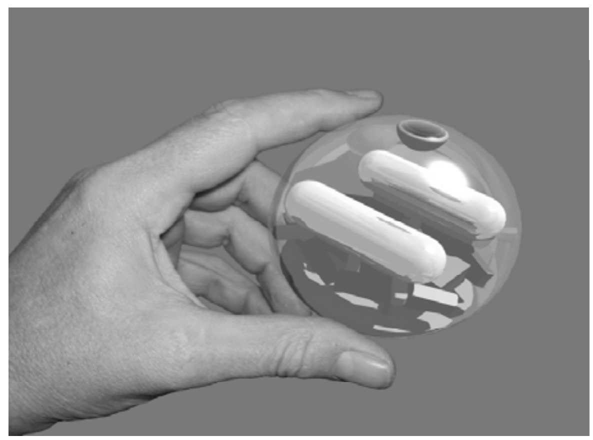


Figure 1. A fuel cell powered micro device concept [10].

This work focuses on Polymer Electrolyte Membrane (PEM) fuel cells that consume hydrogen and oxygen from air to produce electricity and water. These fuel cells are highly efficient with operating efficiencies of 50-70 %, operate at standard temperature, are quiet, have low thermal signatures and produce clean exhaust [4], [5].

Alternatives such as Direct Methanol Fuel Cells (DMFC) consume methanol and are air breathing, produce electricity, water and carbon dioxide. However the carbon dioxide can reduce efficiency and at high concentrations can poison the catalyst and shorten its life, making DMFCs unreliable [6].

II. RESEARCH

This research identifies the critical problems of fuel cells, solutions and technology pathways to applying these devices in challenging field environments [15], [16], [17], [18].

A. Fuel Cell Reliability and Life

As discussed above, PEM fuel cells lack reliability due to degradation of their components resulting in short lives. It has been shown that degradation of the PEM fuel cell platinum catalyst is the principle cause of their premature failure [12] and can significantly decrease power output.

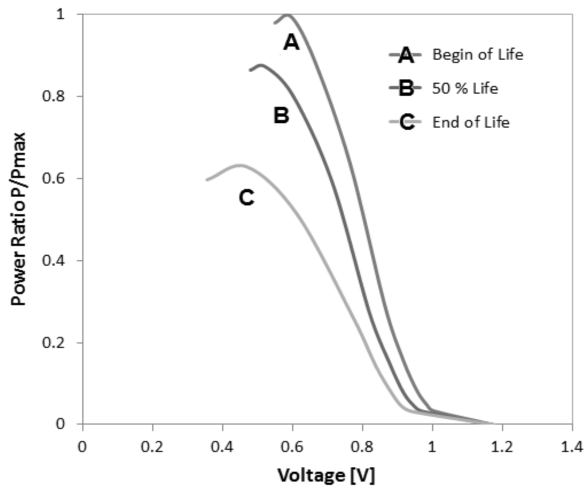


Figure 2. Power Output of Fuel Cell before and after Degradation [12].

Physical models have been developed that describe dissolution of the catalyst, thought to be the primary cause of catalyst degradation [19], [20]. However current models do not match well with experimental data [19].

In this research, physical models have been developed that account for both the dissolution and migration of the fuel cell catalyst causing degradation in field conditions [12]. This is shown to match experimental data, permitting long term prediction of the life of fuel cells, their power output and failure modes. The studies show that high operating voltages, varying power demands and high temperatures can exponentially shorten the life of fuel cells (Figure 3 and 4). In addition their lives are reduced linearly by electrical noise and low humidity [12].

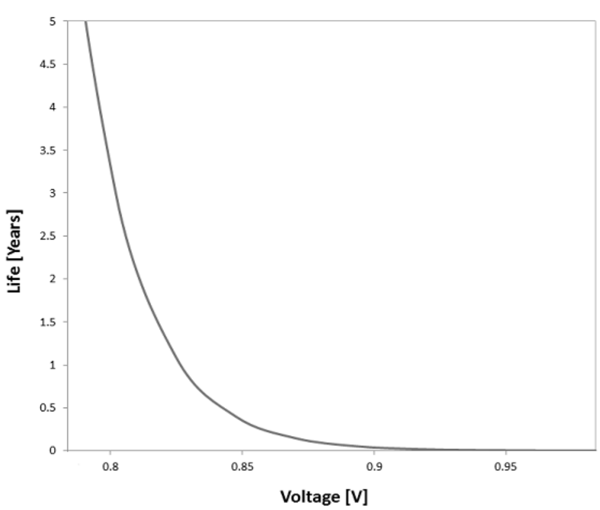


Figure 3. Model Predicted Effect of Operating Voltage on Fuel Cell Life [12].

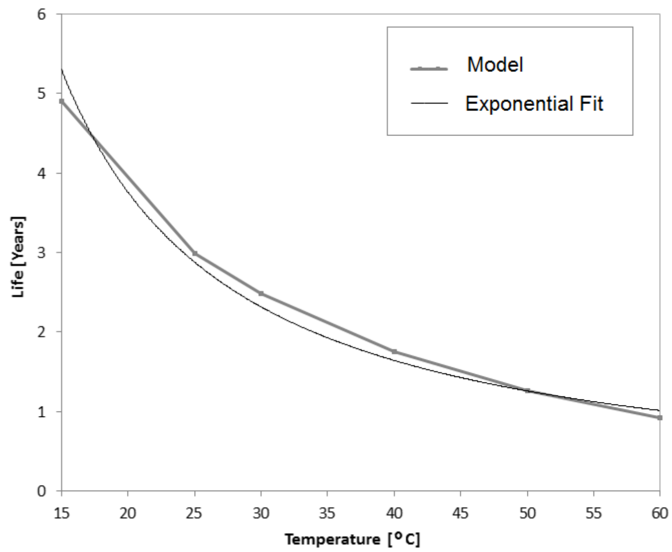


Figure 4. Model Predicted Effect of Temperature on Fuel Cell Life [12].

A fuel cell battery hybrid system has been developed and discussed below that minimizes degradation due to varying system power demands and electrical noise (Figure 5) [17], [21].

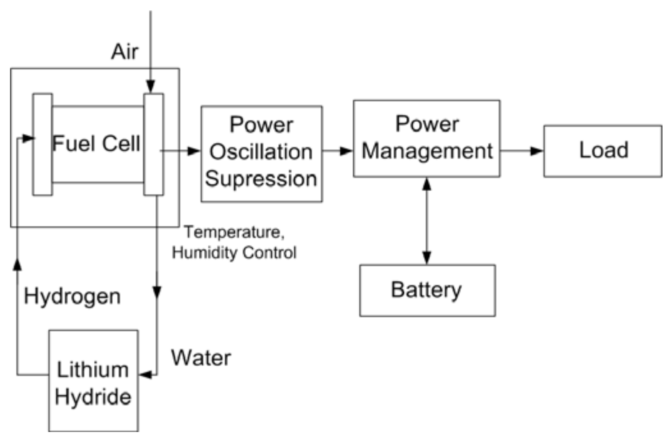


Figure 5. Hybrid Fuel Cell Power Supply.

B. Fuel Cell Power Supply Design

A typical long-life system consists of a hydrogen generator supplying fuel, a fuel cell operating at constant operating voltage, charging a rechargeable battery. A noise isolation circuitry prevents high frequency noise from degrading the fuel cell. The rechargeable battery handles the varying and high power demands from the load.

A PEM fuel cell requires the management of air, water and temperature to maximize the life and performance. For example too little water leads to drying causing accelerated degradation of fuel cells, while too much water causes flooding and loss of power. The development of sensor network modules for extreme desert environments has yielded simple, novels concepts [10], [15], [16]. This includes burying a micro fuel cell power supply below ground to shelter it from extreme temperature swings and using air permeable vapor barrier to maintain an ideal humidity inside the PEM fuel cell, while minimizing water loss [15]. This enables the fuel cell power supply to effectively reuse water produced by the fuel cell for generating hydrogen and for humidification of input gases.

The complexity challenges of fuel cells remain. However it has been shown that water management within PEM fuel cells can be greatly simplified [28]. Simple, passive methods of recapturing water from a micro fuel cell have been developed allowing efficient humidification [28]. This avoids the complexity of active control systems.

C. Hydrogen Storage

Current hydrogen fuel storage mechanisms limit the effectiveness of PEM fuel cell power supplies resulting in a marginal advantage over batteries. Pressurized hydrogen gas storage or cryogenic storage of liquid hydrogen is not practical due to size limitations and energy expended. Metals hydrides are a better solution but current storage methods are inefficient and unreliable, requiring high temperatures or high pressures.

An appealing alternative is using water activated hydrides, where a hydride is exposed to water to release hydrogen. The reaction occurs at standard temperature and pressure. However current water activated metal hydrides including sodium borohydride ($NaBH_4$) [24] and magnesium hydride (MgH_2) [25] have low weight efficiencies, reliability and require expensive catalysts. Magnesium hydride has a theoretical 15.4 % storage efficiency, but less than 8 % has been achieved.

Here it is shown that lithium hydride is ideal for storage and release of hydrogen. Hydrogen can be released by exposing the hydride to water or water vapor releasing the hydrogen from the hydride and stripping water of its hydrogen according to the following reaction [23], [26].

$$LiH + H_2O \rightarrow LiOH + H_2$$

Lithium hydride unlike other water activated hydrides requires no complex mechanisms or catalysts to start, control and complete the reaction [26], [27]. Our experimental studies show that water activated lithium hydride can achieve 100 % reaction completion rates (see Figure 6). Another appealing feature of water activated lithium hydride, for PEM fuel cells is that the fuel cell produces enough waste water for activating the lithium hydride. When exhaust water from a fuel cell is reused for producing more hydrogen using a lithium hydride generator, the reaction achieves a theoretical 25 % hydrogen storage efficiency or 4950 Wh/kg energy density [27] (40 folds higher than lithium ion batteries).

Several control strategies have been developed to produce the required hydrogen at high operating efficiencies. Active control strategies had been initially pursued to achieve a desired hydrogen pressure. A small peristaltic pump drawing an average power of 10^{-2} mW is use to periodically dispense droplets of water exposed to the hydride to produce hydrogen for a 50 mW system (see Figure 7).

However for low power sensor network applications there is a need to simplify the system and increase its reliability by minimizing control electronics and actuators. Passive lithium hydride hydrogen release concepts are being developed that produce hydrogen at a desired pressure by exposure to water as seen in Figure 8. This 75 cm^3 passive concept has shown very promising results and can produce hydrogen for a 140 mW fuel cell system.

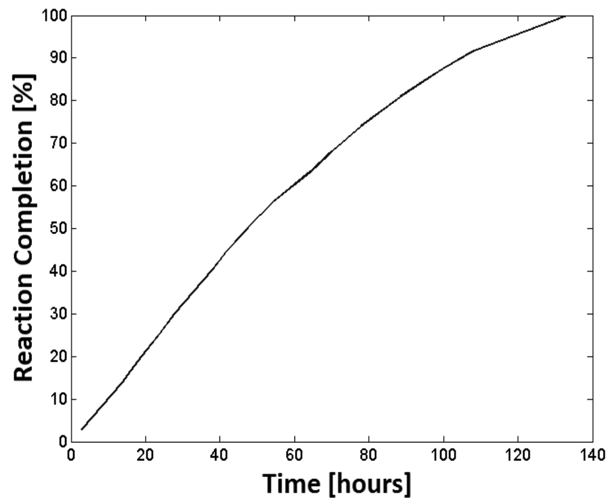


Figure 6. Experimental Results showing percent Hydrogen Produced from Lithium Hydride Reaction with Water.

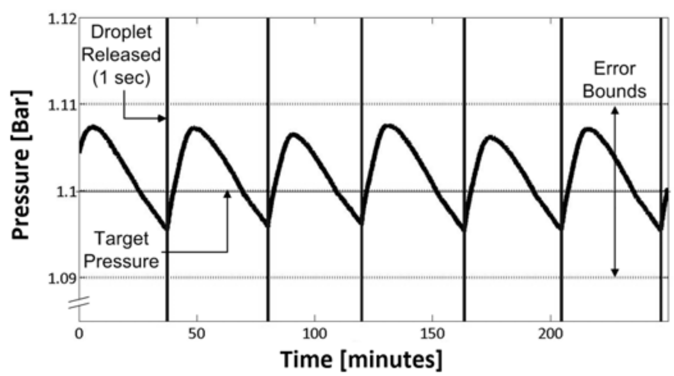


Figure 7. Graph of Pressure versus Time of Hydrogen Release Chamber for 1.1 bar Target Pressure. Vertical Lines represent Times when Droplets of Water Released into the Chamber.

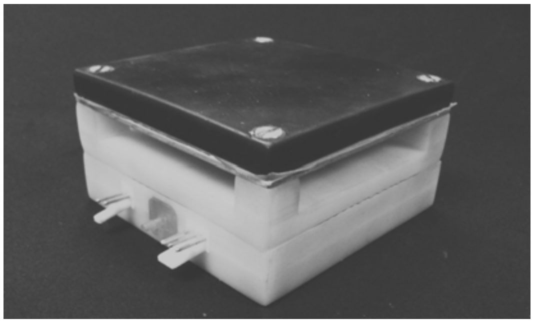


Figure 8. Passive 140 mW Lithium Hydride Hydrogen Generator.

III. Case Study: Sensor Networks

Consider a sensor network module for environmental monitoring. These devices ideally need to operate for 3-5 years or more. Such modules may contain temperature, humidity/moisture, vibration, accelerometers, chemical and light sensors. Their data may periodically be transmitted to a base station. Hence they will typically be low power devices that operate intermittently at higher power. Here a buried sensor might detect the presence of intruders [15]. Sensor can also be mobile, equipped with a polymer actuator to hop, roll and bounce, exploring rugged terrains and caves [10].

Consider a sensor module as shown in Figure 9. It consists of a power supply, an accelerometer to detect vibration, electronics to interface with the sensor, a wireless radio for communication and polymer Dielectric Elastomer Actuator (DEA) for hopping [29]. The hopping mechanism is light weight and compact, weighing in the order of tens of grams and consumes 50 % of total energy required by the sensor module. The sensor module consumes 50 mW average, 100 mW peak at a 0.5 duty cycle. The total mass of the module is assumed to be 1.5 times the mass of the power system.

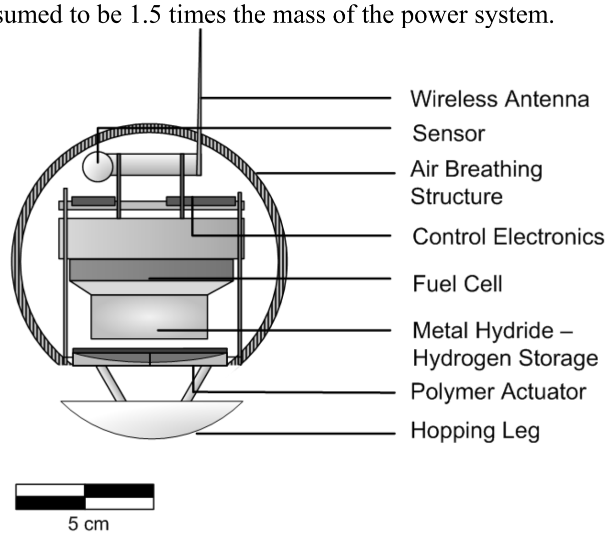


Figure 9: Field Sensor Module.

It is assumed that the field sensor is deployed in a desert location, operating continuously, where temperature varies between 15 and 40 ^{o}C and humidity varies between 0.15 and 0.75.

A. Battery

First batteries are considered as power supplies for these modules. Batteries exhibit self-discharge where stored energy is lost at a fixed rate modeled as a geometric series. Also, it is assumed that the last 20 % of the energy that cannot be used. The mass of a battery power supply required is:

$$M_{bat} = \frac{\alpha \rho_{bat} E(T)\left(1 - r^{T}\right)}{1 - r}$$

where M_{bat} is the total mass of the battery power supply required for T years of life, α is the capacity margin, r is the self-discharge rate, ρ_{bat} is the energy density of the battery, $E(T)$ is the energy required to power a payload device for T years according to a given duty cycle. The energy densities, self-discharge rates and mass of the battery power supplies are shown in Table 4. A sensor module weighing a few kilograms or more lacks scalability to hundreds or thousands of modules owing to high cost and logistics required for deploying them.

B. PEM Fuel Cell

The mass of a PEM fuel cell power supply for this case is given by:

$$M_{fuel} = \frac{\rho_{fuel} E(T)}{r} \cdot \ln\left| \frac{0.5 - \frac{1}{r}}{T + 0.5 - \frac{1}{r}} \right|$$

where M_{fuel} is the total mass of the fuel for T years of life, ρ_{bat} is the energy density of the fuel, $E(T)$ is the energy required to power a payload device for T years for a given a power profile, r is the power degradation rate of the fuel cell power supply for a specified operating point.

The mass calculated accounts for the extra fuel required due to losses from degradation and to ensure the fuel cell provides the energy required at the end of T years. Table 1 shows the mass breakdown of the fuel cell power supply. The power supply consists of the fuel cell, lithium hydride fuel storage, electronics and other components. The lithium hydride fuel produces hydrogen with the addition of water generated from the fuel cell [27].

TABLE 1. Mass Breakdown of Fuel Cell Power Supply for Network of Sensors (3 year Life)

Component	Mass
Fuel Cell	10 g
Lithium Hydride Fuel + Storage	360 g
Electronics	10 g
Other Components	20 g
Total	400 g

C. System Comparison between Batteries and Fuel Cells

The performance of a battery and the PEM fuel cell power supply for field sensor modules are compared in Table 4. Rechargeable batteries such as lithium ion are substantially heavier owing to their high rate of self-discharge. The non-rechargeable batteries such as the alkalines fare better, although they have a low energy density. Other non-rechargeable batteries particularly the lithium CR and Lithium thionyl chloride batteries have relatively high energy density and even lower rates of self-discharge.

Operating the PEM fuel cells at 65 % conversion efficiency (LHV) provides a suitable trade-off between maximizing life and operating efficiency. This results in higher rates of degradation in power output than the other storage options. However unlike the batteries, the fuel cell power supply using lithium hydride as fuel is not prone to energy discharge. A DC-DC convertor can be avoided by connecting several cells in series to obtain a high enough voltage for charging the batteries and thus minimizing electronic conversion losses. In combination, the system is 10 folds lighter than the best batteries (Table 2) making fuel cells a promising option for long life power source for sensor network modules.

TABLE 2. Power Supply Comparison for 3 Years Life

Power Supply	Energy Density (Wh/kg)	Self-Discharge or Degradation % per month	Mass (3 years Life) [kg]
Alkaline	110	0.5	15.6
Lithium Ion	140	5	33
Lithium CR	270	0.17	6.0
Lithium Thionyl Chloride	420	0.08	3.8
Fuel Cell Hybrid - LiH (65 % Eff.)	4,950	1.2	0.4

D. Mobility Comparison between Batteries and Fuel Cells

Assume in this scenario that 50 % of the energy consumed by the sensor is used for mobility, such as hopping. The sensor module is assumed to hop at a 45 ^{o} angle, with an initial velocity of 2 m/s and maximum height of 0.1 m to yield 0.41 m distance traversed per hop. The electrical energy, E_{hop} required per hop (neglecting air resistance) is approximated as follows:

$$E_{hop} = \frac{1}{2\eta_{EM}} mv^2$$

where η_{EM} is the electrical to mobility efficiency of the DEA hopping mechanism and is 13 % [10], [29], m is the mass of the sensor module and v is the initial velocity of the sensor module. The maximum range, l per hop is then:

$$l = \frac{v^2}{g} \sin 2\theta$$

where v is the initial velocity of the sensor module, g is the gravitational acceleration and is 9.8 m·s^{-2}, θ is the hopping angle and is 45 ^{o}.

Savings in mass can be significant if the sensor module needs to be mobile (i.e. hop) as shown in Table 3. Battery powered sensor modules, particularly using alkalines, lithium ion and lithium CR batteries have very short range owing to their high mass. Lithium thionyl chloride batteries have the greatest range of the all the batteries compared but have nearly 10 folds shorter range than the proposed fuel cell power supply. These results show that a fuel cell power supply has the potential to be light weight and efficient power source for sensor network modules.

TABLE 3. Power Supply and Range of Mobile Sensor Module

Power Supply	Mass [kg] (3 years Life)	Number of Hops	Range [km]
Alkaline	15.6	6,600	2.7
Lithium Ion	33	3,100	1.3
Lithium CR	6	17,000	7.1
Lithium Thionyl Chloride	3.8	27,000	11
Fuel Cell Hybrid - LiH (65 % Eff.)	0.4	260,000	104

IV. CONCLUSIONS

Robots and sensor networks have many important practical applications, but current battery technology is often a limiting factor. Current research into batteries has the potential to increase energy densities. Fuel cells are a promising alternative. However fuel cells face several critical challenges that limit their practical use, including their premature degradation and failure, inefficient and bulky hydrogen storage, low power output and high cost. In this study, the degradation of the PEM fuel cell under field conditions can be traced to the catalyst. Degradation is due to improper operating conditions. A novel fuel cell hybrid power supply has been developed, minimizing stresses on the fuel cell permitting long-life. Also a hydrogen storage system has been developed using water activated lithium hydride that has a theoretical energy density of 4950 Wh/kg (40 folds higher than lithium ion batteries). Current work on passive air, water and thermal management promises relatively simple fuel cell power supplies with an objective to make these devices practical for small robots and sensor devices.

V. FUTURE WORK

Work is underway in developing a bench top experimental system of a long-life, 250 mW lithium hydride PEM fuel cell power supply. In addition, a passive lithium hydride hydrogen generator is being developed for a fuel cell powered robotics systems.

ACKNOWLEDGMENTS

The support for this work by Israel's MAFAT Basic Science Office of the MOD is gratefully acknowledged as is the contributions of Dr. Igal Klein and Ling Ling Deng.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankasubramanium and E. Cayirci (2002) "A Survey of Sensor Networks," *IEEE Communications Magazine*, Vol. 49, No. 115 pp. 102 – 114.

[2] M. Hägele, "Contribution to World Robotics 2005," Technical Report, *European Robotics Network*, 2005.

[3] A. G. Ritchie (2005) "Recent developments and likely advances in lithium rechargeable batteries," *Journal of Power Sources*, Vol. 136, No. 2, pp. 285-289.

[4] F. Barbir, PEM Fuel Cells: Theory and Practice, Academic Press, 2005.

[5] R. O'Hayre, S. Cha, W. Colella, W., and F. B. Prinz, *Fuel Cell Fundamentals*, New York, NY: Wiley, 2005.

[6] A. Casalegno, R. Marchesi, (2008) "DMFC performance and methanol cross-over: Experimental analysis and model validation", *Journal of Power Sources*, 185:1, pp. 318-330.

[7] A. Taro, et al., "PEFC Deep Cruising AUV URASHIMA," *Fuel Cell Symposium Proceedings*, pp. 90 – 95, 2003.

[8] H. Joh, et al., (2010) "A direct methanol fuel cell system to power a humanoid robot," *Journal of Power Sources*, Vol. 195, pp. 293-298.

[9] S. Lee, et al. (2010) "Development of a 600 W Proton Exchange Membrane Fuel Cell Power System for the Hazardous Mission Robot," *Journal of Fuel Cell Science and Technology*, pp. 0310061 - 0310067.

[10] S.B. Kesner, J.S. Plante, P. Boston, T. Fabian, S. Dubowsky, "Mobility and Power Feasibility of a Microbot Team System for Extraterrestrial Cave Exploration," *Proceedings of the 2007 IEEE International Conference Robotics and Automation*, Rome, Italy, April 2007.

[11] Y. S. Horn, W. C. Sheng, S. Chen, P. J. Ferreira, E. Holby, D. Morgan (2007) "Instability of Supported Platinum Nanoparticles in Low-Temperature Fuel Cells," *Topics in Catalysis*, Vol. 46, pp. 285-305.

[12] J. Thangavelautham and S. Dubowsky (2011) "Controlling Degradation in Fuel Cell Power Supplies for Long Life Field Applications," Submitted to *Journal of Power Sources*, pp. 1-28.

[13] M.A. Rubio, A. Urquia, and S. Dormida (2009) "Diagnosis of Performance Degradation Phenomena in PEM Fuel Cells," *International Journal of Hydrogen Energy*, pp.1-5.

[14] L. Schlapbach and A. Zuettel (2001) Hydrogen storage materials for mobile applications, *Nature*, Vol. 414, pp. 353-358.

[15] K. Manyapu, (2010), "Feasibility Study of Long-Life Micro Fuel Cell Power Supply for Sensor Networks for Space and Terrestrial Applications," *Master's Thesis*, Department of Aeronautics and Astronautic, MIT, Cambridge, MA.

[16] K. Manyapu, J. Thangavelautham, P. Schenker, and S. Dubowsky. "On the feasibility of fuel cell powered sensor modules for deployment at the Lunar poles," To Appear in the *Proceedings of the International Astronautical Conference (IAC)*, October, 2011 Cape Town, South Africa.

[17] D. Gallardo, (2010) "Power Management for Micro PEM Fuel Cells," *Master's Thesis*, Mechanical Engineering Department, Politecnico di Torino.

[18] B. Burg, S. Dubowsky, J. Lienhard and D. Poulikakos, "Thermal Control Architecture for a Planetary and Lunar Surface Exploration Micro-Robot," *Proceedings of the Space Technology and Applications International Forum*, Albuquerque, NM, February 11-15, 2007.

[19] W. Bi and T.F. Fuller (2008) "Modeling of PEM Fuel Cell Pt/C Catalyst Degradation," *Journal of Power Sources*, Vol. 178, pp. 188-196.

[20] R.M. Darling and J.M. Meyers (2003) "Kinetic Model of Platinum Dissolution in PEMFCs," *Journal of the Electrochemical Society*, Vol 150, No. 11 A1523-1527.

[21] J. Thangavelautham, D. Gallardo, D. Strawser and S. Dubowsky, "Hybrid Fuel Cells Power for Long Duration Robot Missions in Field Environments," *Proceedings of the 14th International Conference on Climbing and Walking Robots*, 2011.

[22] D. Chandra, "Intermetallics for Hydrogen Storage," in Solid State Hydrogen Storage: Materials and Chemistry, Woodhead Publishing Limited, UK, 2008, Ch. 12, pp. 317-324.

[23] T. Newton, et al., "Miscellaneous Information on the Properties and Handling of LiH and Li(D,T)," Los Alamos Scientific Laboratory report, LAMS-1074, Los Alamos Scientific Laboratory, 1954.

[24] H. Schlesinger, et al. (1953) "Sodium Borohydride, Its Hydrolysis and its Use as a Reducing Agent and in the Generation of Hydrogen," *Journal of the American Chemical Society*, Vol. 75, No. 1, pp. 215-219.

[25] Y. Kojima, K.I. Suzuki and Y. Kawai (2004) "Hydrogen generation by hydrolysis reaction of magnesium hydride," *Journal of Materials Science*, Vol. 39, pp. 2227-2229.

[26] V. C.Y. Kong, D.W. Kirk, F.R. Foulkes and J.T. Hinatsu. (2003) "Development of hydrogen storage for fuel cell generators II: utilization of calcium hydride and lithium hydride," *International Journal of Hydrogen Energy,* Vol. 28, pp. 205-214.

[27] J. Thangavelautham, D. Strawser, M. Cheung, S. Dubowsky, "Lithium Hydride Powered PEM Fuel Cells for Long-Duration Mobile Robotic Missions," *Field and Space Robotics Laboratory Technical Report*, FSRL-2011-1, pp. 1-8.

[28] H. Deng, "Passive Control of Water and Humidity with Applications to PEM Fuel Cells," *Field and Space Robotics Laboratory Technical Report*, FSRL-2011-2, 2011, pp. 1-8.

[29] J. S. Plante, M. Santer, S. Pellegrino, and S. Dubowsky, "Compliant Bistable Dielectric Elastomer Actuators for Binary Mechatronic Systems," in *Proceedings of the ASME Conference on Mechanisms and Robotics*, Long Beach, CA, 2005.

Low Energy Consumption Sensor Node Leaping Mechanism for Enhancing Coverage and Connectivity in Wireless Sensor Networks

Yuki Noguchi, Geunho Lee, Yasuhiro Nishimura, Nak Young Chong, Sang Hoon Ji, and Young-Jo Cho

***Abstract*— This paper presents the design, performance analysis, and preliminary implementation of a one-time-use leaping mechanism for energy-saving sensor node relocation, resulting in larger coverage and higher connectivity in wireless sensor networks. It is known to be challenging and difficult to achieve controllable mobility management for energy-constrained sensor nodes that ensures an autonomous, robust, and dependable relocation even in adverse environmental conditions. What is the most important aspect from a practical point of view is to develop a simple yet efficient omni-directional mobility system with the minimum amount of energy required. We propose a new leaping actuating system that utilizes the ground reaction force generated when releasing selectively multiple pre-compressed springs mounted underneath the sensor node. We are in the preliminary stage of developing a final prototype. In this paper, we investigate through simulations and experiments the technical features of our working prototype whose distance and direction are finitely controllable.**

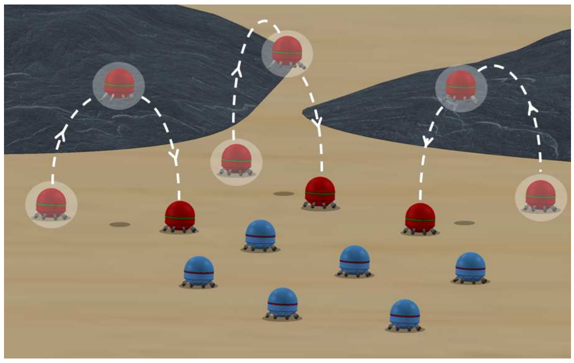


Fig. 1. Relocation and deployment of sensor nodes: network connectivity and coverage enhancement for environmental monitoring

I. INTRODUCTION

With recent advances in electronics and communication technologies, there has been increasing interest in wireless sensor networks in a variety of applications such as environmental or habitat monitoring [1]. One of the most important issues that can be raised in such applications is how to cover an area as large as possible while maintaining network connectivity [2]. Since sensor nodes can be scattered in an area from an aircraft, their self-relocation strategies must accompany initial node distribution to enhance network connectivity and area coverage as shown in Fig. 1. Regarding the self-relocation issue, most researches done to date have focused on developing scalable distributed algorithms for computing target locations [3][4], but the nodes were unrealistically assumed to have unlimited energy resources. In practice, energy consumption caused by node movement accounts for a significant portion of battery lifetime. Considering such practical limitations, in this work, we attempt to propose a new omni-directional mobility design that can minimize node energy consumption.

Recently, node mobility has been gaining increasing attention. To extend the lifetime of a heterogeneous mobile sensor network, a mobile relay strategy is proposed in [5], enabling mobile sensors to help relieve static sensors with burden by high network traffic. To minimize energy consumption for surveillance and data transmission, a minimal energy path planning method is presented [6]. Meanwhile, various prototypes have been developed for mobile sensor networks such as omni-ball [7], evacuation robot [8], jumping robot [9] and so on. Specifically, there exist some notable leaping mechanisms [10]-[12] inspired by springtail, locust, or flea to move at a low energetic cost and jump relatively large obstacles. In spite of impressive leaping performance, their complicated structure and/or bulky size are difficult to be used for tiny wireless sensor nodes.

The main purpose of this work is to present our design of a new instantaneous mobility mechanism well suited for tiny wireless sensor nodes. Toward enhancing coverage and connectivity from initial random distributions of senor nodes, we propose a novel one-time-use spring-powered leaping mechanism that utilizes the ground reaction force generated when selectively releasing multiple actuators (*i.e.*, pre-compressed springs) mounted underneath individual sensor nodes. One significant advantage is to minimize the amount of energy required for relocation and deployment by the proposed robotic leaper. What is the most important aspect from a practical point of view is how to control the leaping distance and direction in order to improve the capability of fixed sensor networks. Another practical issue in the robotic leaper design is how to cope with aerodynamic disturbances over the geographic area. For the purpose, several analysis processes are integrated and automated, and an optimization technique is implemented. Toward building our final prototype, a working prototype is developed and tested through simulations and experiments.

Y. Noguchi, G. Lee, Y. Nishimura, and N. Y. Chong are with the School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa, Japan {ynoguchi, geun-lee, y.nishimura, nakyoung}@jaist.ac.jp

S. H. Ji is with the Korea Institute of Industrial Technology, Ansan, Korea robot91@kitech.re.kr

Y.-J. Cho is with the Electronics and Telecommunications Research Institute, Daejeon, Korea youngjo@etri.re.kr

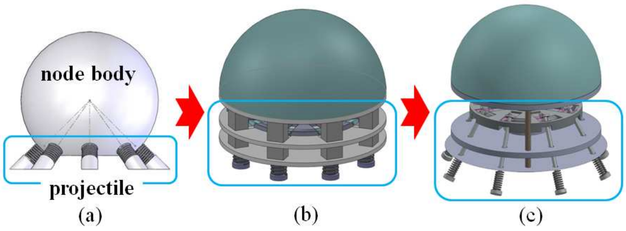


Fig. 2. Development progress of robotic leaper ((a) a proof-of-concept prototype, (b) concept prototype v.1, (c) concept prototype v.2)

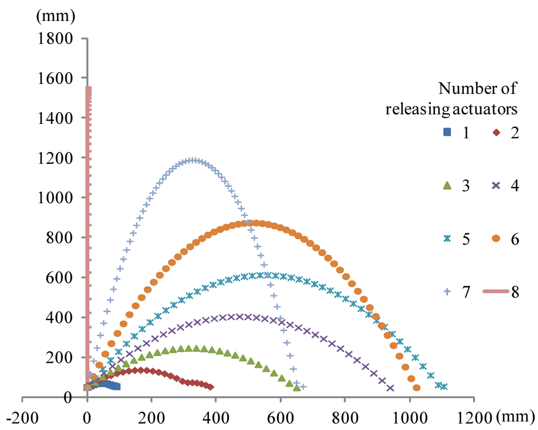


Fig. 3. Variation in leaping distance according to the number of actuators triggered

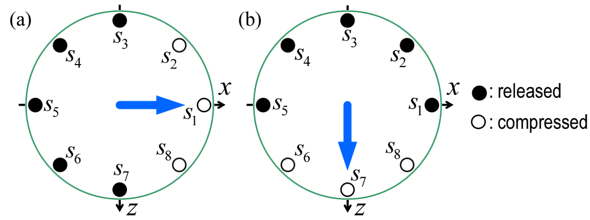


Fig. 4. Leaping direction control: each actuator is denoted as s_i. ((a) $\vec{x}$ direction movement, (b) $\vec{z}$ direction movement)

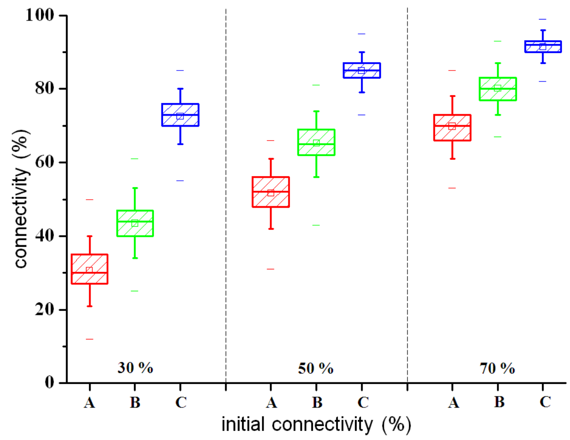


Fig. 5. Connectivity of sensor networks with (A) static nodes, (B) randomly moving nodes, and (C) direction-controlled moving nodes: error bars represent the 95% confidence intervals, and boxes indicate distributions of measured data in the 25% to 75% range.

II. One-Time-Use Leaping Mechanism

Fig. 2 illustrates several one-time-use robotic leaper prototypes. Our idea is to utilize the ground reaction force generated by selectively releasing multiple actuators (*e.g.*, pre-compressed springs) mounted underneath the sensor node. The node body has a globular shape to lessen the air resistance when it flies in the air. The node body encompasses all electronic components for wireless communications and a trigger mechanism for controlled release of actuators. Further, the proposed one-time-use, spring-powered actuators are designed to be suited for leaping in open ground conditions possibly characterized by variable elevations. In detail, the eight pre-compressed springs are installed with the same interval to control the leaping direction across an area as evenly as possible through a selective combination of actuators released.

A. Leaping Direction and Distance Control

The dynamic simulation model of the sensor node is created with SolidWorks [13] and incorporated into a well-known multi-body dynamics analysis software RecurDyn [14]. Analytic studies are conducted to examine how all the possible combinations of releasing actuators affect the leaping distance: changes in the leaping distance with respect to the number and combination of releasing actuators. Fig. 3 shows the changes in the leaping distance according to the number of releasing actuators. When five actuators, out of eight actuators, positioned continuously are released at the same time, the maximum leaping distance can be obtained. Since the impact force by individual actuators s_i released is added to a specific direction and/or conterbalanced each other, the possible combinations of releasing actuators affect the leaping direction. As illustrated in Fig. 4-(a), by releasing from s_3 to s_7, the node's direction of motion is achieved along the positive x-axis direction. Similarly, in Fig. 4-(b), the node's z-axis direction motion is controlled by releasing from s_1 to s_5.

Next, to investigate the effectiveness of the proposed leaping mechanism in improving network connectivity, a network simulation testbed is developed using Microsoft Visual C++. In our simulation, we tested three sets of 100 sensor nodes initially randomly distributed, where the initial connectivity of each set is 30%, 50%, and 70%, respectively. 1000 random distribution patterns are given for each of the three sets, respectively. Under these simulation conditions, the network connectivity of the following cases is investigated. Case A is the static sensor network that only includes non-mobile sensor nodes. In Case B and Case C, mobile sensor networks are tested, where isolated sensor nodes can be relocated using the proposed leaping mechanism. Specifically, isolated sensor nodes move randomly in Case B, and move along the desired directions using an estimation algorithm for direction of arrival of the signal. The results for network connectivity are presented in Fig. 5. Compared with Case A, the connectivity in Case B increased more than

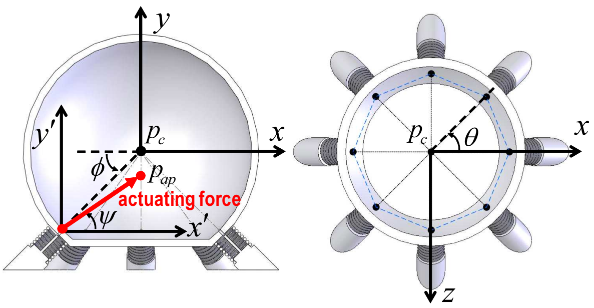


Fig. 6. Angular parameters ϕ and ψ of robotic leaper

10%. The connectivity of CASE C has increased remarkably. Despite minimal energy consumption, these results are very promising to enhance network connectivity in wireless sensor networks.

B. Parametric Representation of Actuating Force

The impact forces applied to the sensor node body by the proposed spring-powered actuators can be parameterized by the locations of acting points and acting directions. The angular parameters are graphically depicted in Fig. 6. Since the leaping mechanism has an axisymmetric structure, the location of acting point can be calculated using the downward angle ϕ measured from the x-axis of the node body center of mass p_c, and the attaching angle θ of each actuator. Moreover, the acting direction can be calculated by ϕ, θ, and another angular parameter ψ, which is the upward angle measured from the x'-axis of the coordinate system defined at the acting point p_{ap} (=(x_{ap}, y_{ap}, z_{ap})).

From now, the locations of p_{ap} and acting directions can be calculated by the following three angular parameters: θ, ϕ, and ψ. Since θ is fixed, by adjusting ϕ and ψ, the impact force direction can be controlled. Based on the geometric structure of the sensor node, the location of p_{ap} is calculated by the following equations.

$$\begin{cases} x_{ap} &= -L\cos\phi\cos\theta \\ y_{ap} &= H + L\sin\phi \\ z_{ap} &= -L\cos\phi\sin\theta \end{cases}, \tag{1}$$

where L and H are the characteristic length for calculating the locations and the height of the node body center of mass, respectively, and are set to 49.5 mm and 51.0374 mm. In the same manner, the acting direction (x_{ad}, y_{ad}, z_{ad}) is given by

$$\begin{cases} x_{ad} &= -L\cos\phi\cos\theta - L\cos\psi\cos\theta \\ y_{ad} &= H - L\sin\phi + L\sin\psi \\ z_{ad} &= -L\cos\phi\sin\theta - L\cos\psi\sin\theta \end{cases}. \tag{2}$$

III. DESIGN OPTIMIZATION AND SIMULATION

A. Optimization Problem Formulation

The leaping distance and direction are the two most important performance features in the design of the proposed leaper. The leaping distance should be secured as long as possible to realize sufficient relocation capability. Further, the leaping direction with a high level of accuracy should be maintained for any intended directions. Various attempts can be made to find a design solution that satisfies these requirements. We prefer to find the most robust solutions to uncertain real-world operating conditions, as well as to achieve the easiness of finding solutions. A properly designed leaper should be able to support reliable, accurate positioning of sensor nodes in adverse operating conditions.

We assume an area where the wind blows consistently. To reflect the effect of aerodynamics, the drag force [15] is applied to the sensor node body as a virtual translational force over the area given by $F_D = \frac{1}{2}\rho v^2 C_d A$, where F_D, ρ, v, C_d, and A indicate the drag force, the density of air, the speed of the wind relative to the object, the drag coefficient, and the reference area, respectively. ρ of air is 1.2 kg/m^3. The parametric value of C_d of the node is assumed to be 0.47, the same as that of the sphere [16]. The speed of the wind is set to 10 m/s. F_D is automatically calculated for the relative velocity between the node and the wind.

The interval distance between actuators and releasing combination of actuators can be determined by extensive simulations. Two remaining angular parameters ϕ and ψ, which can represent the actuating force, are chosen as the design variables to optimize the leaping distance and direction accuracy. In this paper, the leaping distance is considered as the objective function to be maximized. For the leaping direction control, a certain level of tolerable error, 5 degrees, is considered as a constraint. Now our design optimization problem is formulated as

$$\begin{array}{ll} \mathit{Find} & \phi \text{ and } \psi \\ \mathit{to\ maximize} & \text{Leaping Distance} \\ \mathit{subject\ to} & \text{Angular error} \leq \pm\, 5^\circ \end{array}. \tag{3}$$

Among the various optimization techniques available, the *progressive quadratic response surface method* (PQRSM) [17] is selected. The overall algorithmic procedure of PQRSM goes as follows: 1) applying the *design of experiments* (DOE) in a trust region, 2) building a metamodel, 3) approximating optimization, 4) updating the trust region, and 5) checking whether the optimization has converged. If this termination criterion is satisfied, PQRSM is finished. Otherwise, the procedure goes back to the first step.

B. Progress Integration and Design Optimization

Several analysis and simulation processes for evaluating the leaping performance are integrated and automated using a commercial *progress integration and design optimization* [18] (PIDO) software PIAnO [19] to perform the design more efficiently. By the use of PIAnO, the optimum solution is obtained in a systematic way. When the aerodynamic drag force is applied, to identify the effect that the optimum design has on the leaping distance and directional accuracy, simulations are performed for the area over which the wind blows in an essentially constant direction. Although the leaping distance is decreased by about 10% (from 1122.4139 mm to 900.2469 mm), the accuracy of direction control

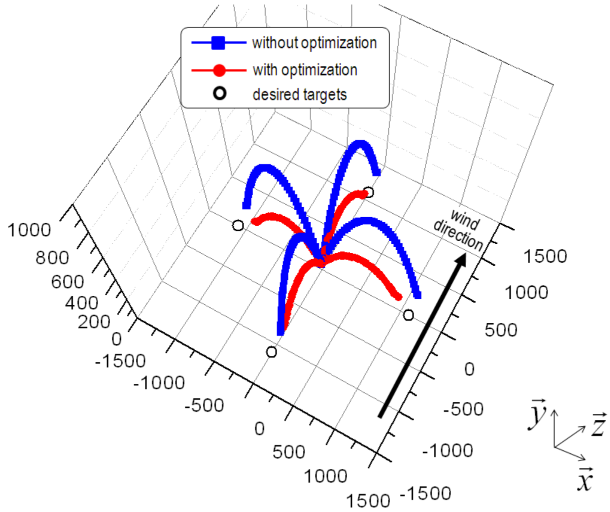
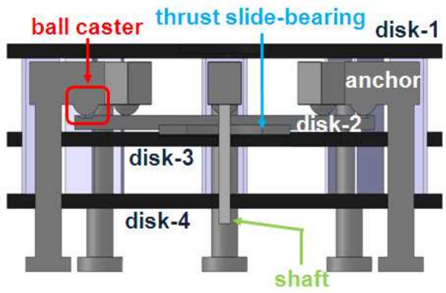
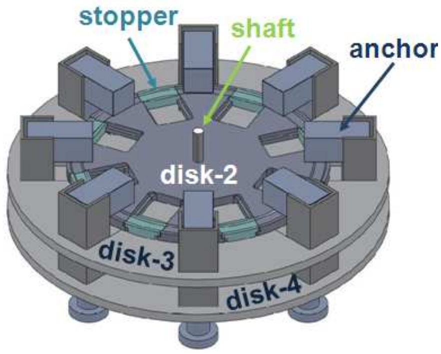


Fig. 7. Leaping direction and distance control against winds



(a) cross-sectional view



(b) bird's eye view

Fig. 8. Schematic view of the leaper triggering device

satisfies the imposed constraint (from 10.2133 degrees to 4.979 degrees). Fig. 7 shows the leaping behaviors under the constant directional wind. The wind blows in the z-direction and leaping behaviors were tested for four intended directions (every 90 degrees) in the x-z plane. From the results, it was verified that the accuracy of direction control could be enhanced. We were able to further verify that an optimum solution can be obtained for most real world situations, for instance, where the nodes fly against uncertain, changing wind directions.

IV. Prototype Development

This section describes a preliminary test of our working prototype, and presents experimental results to demonstrate its feasibility and performance. Prior to developing a final prototype based on analytical and simulation results, we also need to examine whether the arbitrarily-selected pre-compressed actuators can be self-triggered individually or in groups simultaneously. This is necessary to convince ourselves that we can implement the proposed idea properly in our final prototype.

A. Mechanical Configuration of Concept Prototype

Fig. 8 shows the schematic view of our concept prototype version 1, largely divided into the following three components: gear drive, triggering device, and anchor parts. The gear drive includes one DC motor and a 2500:1 gear reduction unit. The triggering device is composed of four disks: Disk-1, Disk-2, Disk-3, and Disk-4, respectively, from the top down to the bottom. Disk-1 and Disk-4 are the outer layers of the triggering device. In detail, the gear drive is fixed to Disk-1, and Disk-4 has eight brackets protruding from its surface where individual anchor parts are fixed. Disk-2 retains the pre-compressed springs in the anchor parts, and selectively releases the springs while rotating. Disk-3 supports Disk-2. Each anchor is composed of a Γ-shaped rod surrounded by a spring (see Fig. 11). As illustrated in Fig. 8-(a), the tip of each rod is equipped with a ball (bearing) caster to slide smoothly over the surface of Disk-2.

B. Concept Prototype Analysis Results

Now we need to calculate the DC motor torque required to rotate Disk-2. A dynamic simulation model is developed with RecurDyn considering the friction and material properties as shown in Fig. 8. Fig. 9 shows how the torque required varies as Disk-2 rotates. From the obtained torque requirement of 800 Nmm, we can determine a suitable DC motor required to rotate Disk-2, allowing pre-compressed springs retained in individual anchors to be released. Fig. 10 presents the simulation result for the node leaping behavior when the obtained torque is applied. This simulation shows that the leaping height was 545.37 mm, when the spring constant K was set to 4.51 N/mm. With $K = 9.02$, the required torque and the leaping height were 1481.499 Nmm and 1136.905 mm, respectively. We have observed that the leaping height linearly increases with K to a certain degree.

We can verify the proposed triggering system by comparing the period of time needed to apply the proper amount of torque in Fig. 9 and the time the node starts to leap in Fig. 10. They are exactly coincident with each other. The resultant force exerted on the node body by releasing the pre-compressed springs is considered as the rate of change of linear momentum. In other words, the force is relatively large that acts over a small interval time. Therefore, it would be very important to synchronize the release time among all of the individual springs.

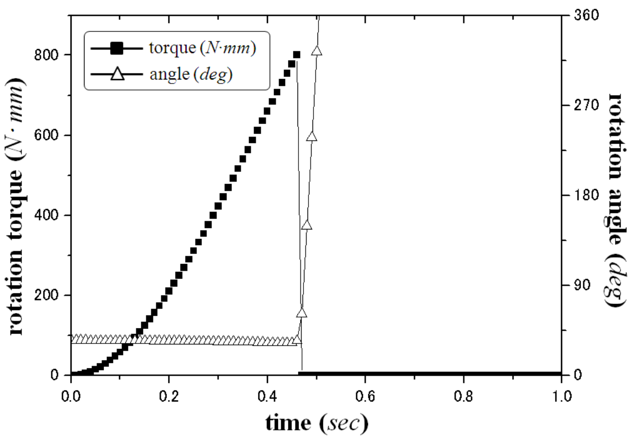


Fig. 9. Simulation result for the motor torque required to rotate DISK-2

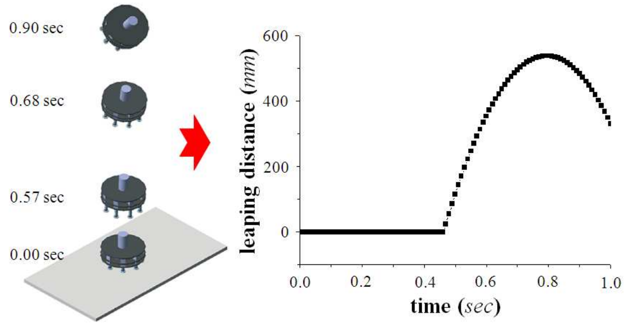


Fig. 10. Leaping height trajectory of our concept prototype

C. Working Prototype Test and Results

Based on extensive simulations, we have developed our first working prototype as shown in Fig. 11. TABLE I shows the mechanical specification of the prototype. Here, we describe in more detail how the triggering device is being designed and modified. In our very first test, DISK-2 could not rotate due to the excessive amount of static friction encountered when DISK-2 and DISK-3 contact. To reduce the friction, as illustrated in Fig. 8, a thrust slide bearing is put between the faces of DISK-2 and DISK-3. Further, we make a circular groove over the surface of DISK-2. The individual rods with the ball (bearing) caster slide along the groove.

Next, it is very important to ensure that the selected pre-compressed springs are released at the same time. If this simultaneous triggering is not supported, we cannot properly control the leaping direction and distance of each node. To this purpose, we built a test bed for triggering and synchronization verification as shown in Fig. 12-(a). This test bed is equipped with eight push-button switches on its top plane. Whenever each compressed spring is released, the test bed detects the instant that each spring pushes its corresponding switch. Fig. 12-(b) presents the experimental results for the releasing synchronization of the actuators placed on the test bed. The synchronization verification tests were performed 10 times, and we examined the release times for individual anchors. An average time difference between the first release and the last release is 0.763 $msec$. We were concerned about the effects of possible synchronization deviations practically not avoidable. We have verified that such an amount of deviation can be tolerated by our working prototype. Fig. 13 presents the snapshots of the leaping behavior of our working prototype that can only leap vertically upward off the ground. The leaping height was measured as 537.98 mm for K = 4.51 N/mm. From the results so far, our working prototype can be considered quite satisfactory toward building our final prototype currently under way.

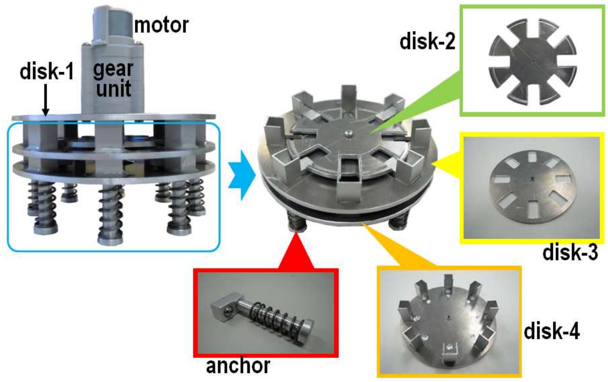


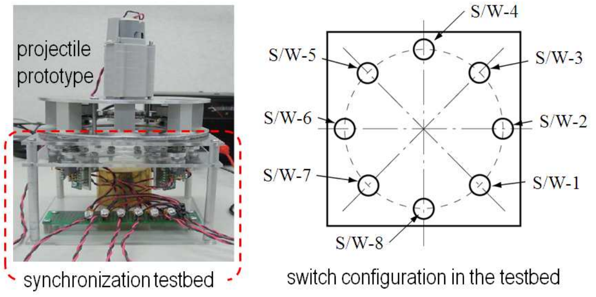
Fig. 11. Mechanical structure of the working prototype

TABLE I
PROTOTYPE MECHANICAL SPECIFICATION

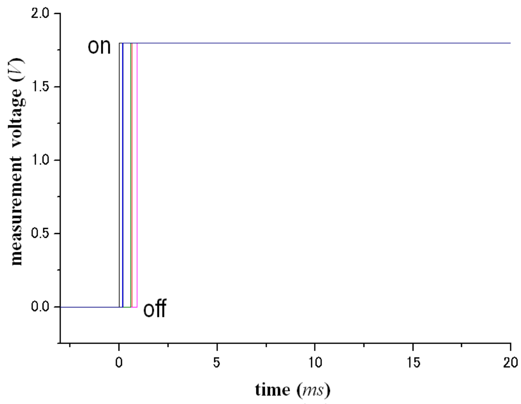
unit	size (mm)	weight (g)	material
disk-1	$t3$ $\phi 116$	70	aluminium alloy
disk-2	$t3$ $\phi 80$	20	aluminium alloy
disk-3	$t3$ $\phi 116$	80	aluminium alloy
disk-4	$t3$ $\phi 116$	110	aluminium alloy
thrust slide bearing	$t2$ $\phi 35$	36	steel
rod	44 (length)	7	aluminium alloy
spring	29.5 $\phi 11.25$ (compressed: 16)	20	steel

V. SUMMARY AND FUTURE DIRECTIONS

This paper presented a one-time-use, spring-powered robotic leaper to support intentional mobility for sensor nodes expected to enhance network connectivity and coverage in wireless senor networks. This controllable mobility offers an efficient way to monitor physical environments. The proposed robotic leaper utilizes the ground reaction force generated when releasing multiple pre-compressed springs mounted underneath the sensor nodes. In particular, it can control the leaping direction as well as the leaping distance by selectively releasing the springs. The design optimization was performed for leaper configuration considering actual operating conditions such as the aerodynamic drag force caused by the wind. The obtained optimum solution was verified by extensive simulations, where the accuracy constraint of the leaping direction was satisfied. Moreover, experimental results were presented to investigate the validity and performance of our working prototype. A more advanced



(a) synchronization testbed



(b) result for the releasing synchronization

Fig. 12. Synchronization testbed: result for simultaneous trigerring

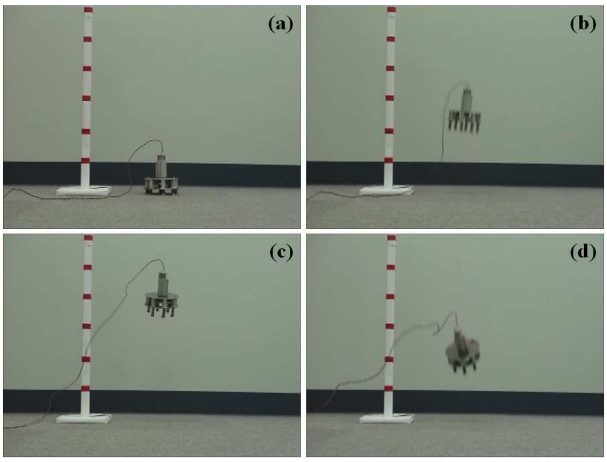


Fig. 13. Experimental result: prototype leaping

mechanism is under development for maintaining a stable attitude and executing a soft landing.

References

[1] J. Yicka, B. Mukherjeea, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol.52, no.12, pp.2292-2330, 2008.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol.40, no.8, pp.102-114, 2002.

[3] G. Lee and N. Y. Chong, "A geometric approach to deploying robot swarms", *Annals of Mathematics and Artificial Intelligence*, vol.52, no.2-4, pp.257-280, 2009.

[4] W. Spears, D. Spears, J. Hamann, and R. Heil, "Distributed, physics-based control of swarms of vehicles," *Autonomous Robots*, vol.17, no.2-3, pp.137-162, 2004.

[5] W. Wang, V. Srinivasan, and K. Chua, "Extending the lifetime of wireless sensor networks through mobile relays," *IEEE/ACM Trans. Networking*, vol.16, no.5, pp.1108-1120, 2008.

[6] C. C. Ooi and C. Schindelhauer, "Minimal energy path planning for wireless robots," *Mobile Networks and Applications*, vol.14, no.3, pp.309-321, 2009.

[7] K. Tadakuma, R. Tadakuma, and J. Berengueres, "Tetrahedral mobile robot with spherical omnidirectional wheel," *Journal of Robotics and Mechatronics*, vol.20, no.1, pp.309-321, 2008.

[8] Y.-D. Kim, Y.-G. Kim, S.-H. Lee, J.-H. Kang, and J. An, "Portable fire evacuation guide robot system," *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp.2789-2794, 2009.

[9] J. Zhao, N. Xi, B. Gao, M. W. Mutka, and L. Xiao, "Development of a contrllable and continuous jumping robot," *Proc. IEEE Int. Conf. Robotics and Automation*, pp.4614-4619, 2011.

[10] Y. Sugiyama, M. Yamanaka, and S. Hirai, "Circular/spherical robots for crawling and jumping," *Proc. IEEE Int. Conf. Robotics and Automation*, pp.3595-3600, 2005.

[11] U. Scarfogliero, C. Stefanini, and P. Dario, "Design and development of the long-jumping 'grillo' mini robot," *Proc. IEEE Int. Conf. Robotics and Automation*, pp.467-472, 2007.

[12] M. Kovac, M. Schlegel, J.-C. Zufferey, and D. Floreano, "A miniature jumping robot with self-recovery capability," *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp.583-588, 2009.

[13] Solidworks User's Manual, http://www.solidworks.co.jp/

[14] RecurDyn User's Manual, http://www.functionbay.co.jp

[15] A. P. French, *Newtonian mechanics* (1st ed.), New York: W. W. Norton and Company Inc., 1970.

[16] B. W. McCormick, *Aerodynamics, aeronautics, and flight mechanics*, New York: John Wiley and Sons Inc., 1979.

[17] K.-J. Hong, M.-S. Kim, and D.-H. Choi, "Efficient approximation method for constructing quadratic response surface model," *Journal of Mechanical Science and Technology*, vol.15, no.7, pp.876-888, 2001.

[18] I.-J. Hwang and G.-J. Park, "A multi-objective optimization using distribution characteristics of reference data for reverse engineering," *Int. J. Numerical Methods in Engineering*, vol.85, no.10, pp.1323-1340, 2011.

[19] PIAnO User's Manual, Version 3.3, PIDOTECH Inc., 2011

Knowledge Representation in the GeRT Project

Alessandro Saffiotti, Lars Karlsson, Fabien Lagriffoul, Federico Pecora
Center for Applied Autonomous Sensor Systems, Örebro University, Örebro, Sweden
firstname.lastname@oru.se

Extended Abstract

A growing number of robots include techniques for knowledge representation and reasoning into their skill repertoire, and the question of which techniques are best suited for robotic systems is still far from being answered – perhaps even far from being properly posed. In this talk, I will discuss the current work related performed in the GeRT project from a knowledge representation point of view.

GeRT (Generalizing Robot manipulation Task) is a European "7th Framework" project involving the German Aerospace Center (DLR, project coordinator), the University of Birmingham in the UK, the University of Örebro in Sweden, and the Max Planck Institute for Biological Cybernetics in Germany. The aim of the GeRT project is to develop artificial intelligence (AI) techniques for improving the ability of robots to cope with novelty in manipulation tasks. The platform used is DLR's Justin platform, a two-armed humanoid robot with 44 degrees of freedom (www.dlr.de/rm) – see Figure 1. Like most other complex robotic platforms, Justin's tasks are entirely pre-programmed for the specific objects and initial configuration. Simply substituting a mug for a glass in a task that involves pouring liquid from the mug into another object would entail different pouring and grasping positions. Similarly, a different initial configuration of the work area may entail the need to remove obstacles before a grasping action can be performed. The GeRT project aims to provide Justin with the ability to automatically exploit existing programs conceived for specific tasks as examples to perform similar tasks with objects of different shapes and in different initial configurations. These tasks will consist of the same types of basic operations, but which may be combined in novel ways and have a different physical execution.

The GeRT approach involves research in planning, learning, and machine perception. In this presentation, I will focus on the planning part, and especially on the type of knowledge needed for that. This knowledge is heterogeneous: we need accurate geometric knowledge to determine if an action is feasible ("can I grasp this cup from the top using my left arm?") as well as causal knowledge for goal-directed planning ("what sequence of actions will lead to the achievement of the goal?). Taking into account both geometric and causal requirements during planning is difficult because of mutual constraints imposed by both types of knowledge. For instance, preparing tea entails that a cup must be filled with water and transferred to a tray (causal requirement), but the fact that the cup is full of liquid requires that it be moved horizontally (geometric constraint) so as to avoid spillage. If horizontal motion is not possible due to the obstalcle configuration, the robot may decide to either remove some obstacles before the move or to fill the cup after the move.

Treating both types of knowledge separately is inefficient and may lead to incompleteness. In GeRT, we study forms of hybrid knowledge representation and related algorithms that are capable of tackling realistically sized problems on real platforms like Justin.

Acknowledgements

This work is supported by the European Community 7th Framework Programme under grant agreement ICT-248273 GeRT. (www.gert-project.eu)

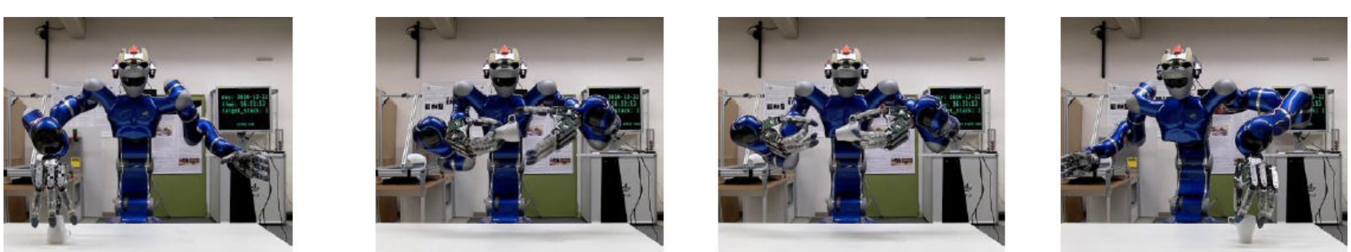


Fig. 1. The Justin robot performs a task involving re-grasping of a cup.