Design of Optimal Robot User Interfaces

Kris Hauser
School of Informatics and Computing, Indiana University at Bloomington
{hauserk}@indiana.edu

I. Introduction

As teleoperated robots grow in complexity their number of degrees of freedom will outpace the number of simultaneous input channels realistically controlled by a human user. It is therefore important to address the problem of how to map a set of input channels into a higher dimensional space in order to achieve favorable properties of the mapping. To my knowledge this issue has not yet been addressed by the motion planning community, but touches on many issues raised in the control of nonholonomic systems. This abstract presents a formulation of the *optimal user interface design problem* and preliminary musings on three particular instances of the problem.

II. Assumptions and Notation

Let the robot's configuration x be an element of an n-dimensional configuration space $\mathcal{C}$, and let the subset of feasible configurations be denoted $\mathcal{F}$. The inherent dynamics of the robot are given by $\dot{x} = f(x,u)$ where $u \in \mathcal{U} \subseteq \mathbb{R}^m$ is the robot's control.

The user's input is given by $a \in \mathcal{A} \subseteq \mathbb{R}^p$. A *user interface* is a map $g : \mathcal{C} \times \mathcal{A} \mapsto \mathcal{U}$ such that the robot follows the dynamics

$$\dot{x} = f(x, g(x,a)). \quad (1)$$

The question is how to design g so that it allows the user to "effectively" control the robot, in a sense to be made more precise later.

It is appropriate to constrain the form of g to "well-behaved" functions, both for optimization to be mathematically tractable as well as to achieve reasonable interfaces. We are also interested in the case where $p < m \leq n$ because it contains interesting open problems. Otherwise it is almost trivial to map $\mathcal{A}$ to cover $\mathcal{U}$ at every point x. We also want to eliminate the possibility of constructing poorly behaved mappings, such as space-filling curves, which may also cover $\mathcal{U}$ but would be poor as user interfaces.

An instance of an interface g gives rise to an underactuated dynamic system $\dot{x} = h(x,a) \equiv f(x, g(x,a))$, and properties of this system can be analyzed using standard techniques from nonholonomic control. Examples of properties of interest include:

Continuity. Continuity and smoothness of $h(x,a)$ is likely to be essential for an interface to be perceived as "reasonable" by a user. It may also be important to bound the derivative of h to prevent the robot from moving erratically in response to sensorimotor noise.

a-Linearity. For input devices that map naturally to a Cartesian space (e.g., joysticks, mice, pressure sensors) it is reasonable for users to assume that doubling input will result in roughly doubling output, and reversing an input will result in the reversing of output.

Completeness. We say $x' \in \mathcal{F}$ is u-reachable from $x \in \mathcal{F}$ if there exists a sequence of controls $u(t)$ that brings x to x' while remaining in $\mathcal{F}$. Likewise, x' is a-reachable from x if the user can choose actions $a(t)$ that brings x to x' along a feasible trajectory. An interface g is *complete* if all u-reachable pairs of states are also a-reachable. Completeness is important because it indicates that the user can drive the robot to any desired state.

Small-Time Local Controllability. An interface g is small-time locally controllable (STLC) at x if the system $\dot{x} = h(x,a)$ is small-time locally controllable – that is, the neighborhood of reachable states, in the limit as time shrinks to zero, has the same dimensionality as $\mathcal{C}$. g is STLC if it is STLC at every $x \in \mathcal{F}$. This property may be desirable because it allows the user to drive the robot along any *trajectory* arbitrarily closely.

III. Optimal User Interface Design Problems

In its most general form, an **optimal user interface design problem** asks to find an interface g that minimizes an objective functional $J(g)$, possibly subject to certain constraints. The values of n, m, f, and p are held constant. Naturally it is important to consider which objectives and constraints correspond to meaningful properties, such as efficiency in navigating the configuration space and intuitiveness for the user. Below are three concrete problem classes that appear promising.

A. Minimum Distortion Problem

A mapping with low distortion between the action space $\mathcal{A}$ and the space of robot velocities $T_x\mathcal{C}$ may be perceived as more "natural" than one with high distortion.

Assume that g is smooth, such that its behavior is dominated by its first order Taylor expansion about $(x,0)$: $g(x',a) \approx \frac{\partial h(x,0)}{\partial x}(x'-x) + \frac{\partial h(x,0)}{\partial a} a$. Define local distortion metrics $e(X)$ and $d(X)$ that measures the deviation of X respectively from zero and from an orthonormal matrix. e and d may contain psychophysical components as well as geometric ones. Minimizing the objective functional

$$J(g) = \int_{\mathcal{F}} e\left(\frac{\partial h(x,0)}{\partial x}\right) + d\left(\frac{\partial h(x,0)}{\partial a}\right) dx \quad (2)$$

achieves the desired effect. The benefit of this formulation, and indeed any other in the form $\int_\mathcal{F} L(x, g, g_x, g_a)dx$, is that it can be solved using the method of Lagrange in several variables. That is, given some points where the value of g is fixed, the resulting problem becomes one of solving a partial differential equation on an $n + p$ dimensional space with stationary boundary constraints.

B. Shortest Path Problem

Assuming the user is highly trained, it is reasonable to assume that he/she will choose $a(t)$ to optimally achieve a desired state x'. A second design problem is then to find g that minimizes the shortest path connecting start and goal states x and x'. Assuming x and x' are picked from a distribution $P(x, x')$ the objective functional defining the expected traversal cost is:

$$J(g) = \int_\mathcal{F} \int_\mathcal{F} S(x, x', g)P(x, x')dx'dx \tag{3}$$

where $S(x, x', g)$ denotes the minimum cost path from x to x'. S is defined as the solution to the optimal control problem:

$$S(x, x', g) = \min_{a,y,T} \int_0^T K(y(t), a(t), t)dt \tag{4}$$

$$\text{such that} \tag{5}$$

$$y(0) = x, y(T) = x' \tag{6}$$

$$\dot{y}(t) = f(y(t), g(y(t), a(t))) \text{ for all } t \in [0, T] \tag{7}$$

$$a(t) \in \mathcal{A} \text{ for all } t \in [0, T] \tag{8}$$

where K is the cost function.

The main drawback of this formulation is that it is heavy on computation. Dynamic programming can be applied to solve the Hamilton-Jacobi-Bellman equation corresponding to (8) over the entire state space, which eliminates the need to recompute S for all x'. Nevertheless it is still expensive because it requires minimizing over an integral of a minimization. Efficient methods for reformulating or approximating (3) would be quite valuable.

C. A Local Approximation to Shortest Paths

If h is restricted to the class of STLC interfaces and the distribution $P(x, x')$ vanishes outside of a small neighborhood of x, the objective (3) can be approximated locally. The question is then how to pick a suitable set Δ of vector fields induced by $\mathcal{A}$ to yield a controllable, optimal system. While controllability and steering methods for nonholomic systems have been well studied, much less attention has been focused on the design problem. Here I offer some preliminary thoughts on how this problem might be solved.

Assume h is linear in a, that is, there is a distribution of vector fields $\Delta = \{\beta_1(x), \ldots, \beta_p(x)\}$ such that $h(x, a) = \sum a_k \beta_k(x)$. The STLC constraint indicates that the involutive closure of Δ must cover the tangent space $T_x\mathcal{C}$ at each point x. The involutive closure of a set of vector fields is defined by applying repeated Lie bracketing operations to pairs of elements in the set, and adding the result to the set until the span of the set no longer grows.

In order to satisfy this constraint one approach might enumerate n vector fields (heretofore not yet defined) $h_1, \ldots, h_n$, where $h_k = \beta_k$ for $k = 1, \ldots, p$, and the remaining fields h_k are composed of a Lie bracket $[h_i, h_j]$ where $i, j < k$. Then, fixing the indices of the composite vector fields, adjust the β_k's such that $\beta_k(x) = f(x, u)$ for some u (allowed to vary with x), and such that the set $h_1, \ldots, h_n$ has full rank at every x. This can be done by hand for small systems but it may be possible to device a general-purpose algorithmic construction. But the problem is not yet solved; there still remains the problem of optimizing the cost function over the space of integrable distributions. One line of study would investigate whether it possible to cast the problem as a minimization of the form $\int_\mathcal{F} L(x, g, g_x, g_a)dx$, in which case existing techniques can be applied.

IV. Discussion

The problems presented here are only the tip of the iceberg of the human-robot interaction problem. How can robot symmetries be used to simplify design problems? How can human psychology inform the choice of objective functions? What is the role of human sensing, in particular when the full state of the robot is not observed? How can human learning be exploited and optimized? It is clear that humans in-the-loop pose theoretical, computational, and practical issues that may prove to be fertile for future motion planning research.