AUCKLAND UNIVERSITY OF TECHNOLOGY
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

Tutorial, IJCNN 2007, Orlando

Evolving Connectionist Systems: The Knowledge Engineering Approach

Nikola Kasabov
Knowledge Engineering and Discovery Research Institute, KEDRI
nkasabov@aut.ac.nz, www.kedri.info
Auckland, New Zealand

Abstract

- Evolving Connectionist Systems (ECOS) are systems that develop their structure, their functionality and their internal knowledge representation through continuous learning from data and interaction with the environment. ECOS can also evolve through generations of populations using evolutionary computation, but the focus of the tutorial is on the adaptive learning and improvement of each individual system. The learning process can be: on-line, off-line, incremental, supervised, unsupervised, active, sleep/dream, etc. These general principles can be applied to develop different models of computational intelligence - evolving connectionist systems, evolving rule based and fuzzy systems, evolving kernel-based systems, evolving quantum-inspired systems, and some integrated, hybrid models [1].
- The emphasis though is on the knowledge engineering aspect of the systems, ie how to represent human knowledge in a system and to extract interpretable information that can can be turned into knowledge. ECOS are demonstrated on several challenging problems problems from bioinformatics, neuroinformatics, neuro-genetics, medical decision support, autonomous robot control, adaptive multimodal information processing. The tutorial targets computer scientists, neuroscientists, biologists, engineers, both researchers and graduate students.

- [1] N.Kasabov, Evolving connectionist systems: The Knowledge Engineering Approach, Springer, 2007

- **Keywords:** Computational Intelligence, Neuroinformatics, Bioinformatics, Knowledge-based neural networks, Evolving connectionist systems, Data Mining; Knowledge Discovery

Content

nkasabov@aut.ac.nz

1. Evolving Intelligent Systems: Introduction

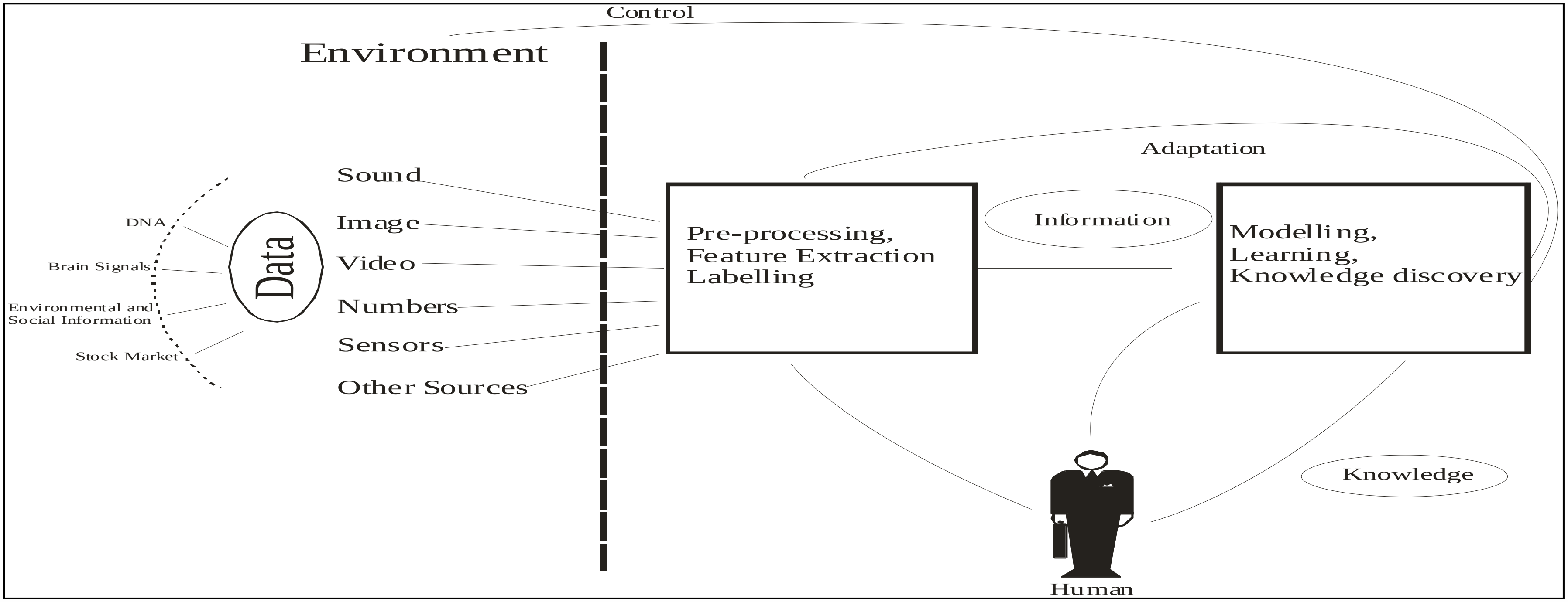
Evolving process: the process is unfolding, developing, revealing, changing over time in a continuous way

EIS: An information system that develops its structure and functionality in a continuous, self-organised, adaptive, interactive way from incoming information, possibly from many sources, and performs intelligent tasks (e.g. adaptive pattern recognition, decision making, concept formation, languages,….).

EIS is characterised by:

- Adaptation in an incremental mode (possibly, on-line, life-long)
- Fast learning from large amount of data, e.g. possibly 'one-pass' training
- Open structure, extendable, adjustable
- Memory-based (add and retrieve information, delete information, trace the system development)
- Active interaction with other systems and with the environment
- Represent adequately space and time at their different scales
- Knowledge-based: rules;
- self-improvement

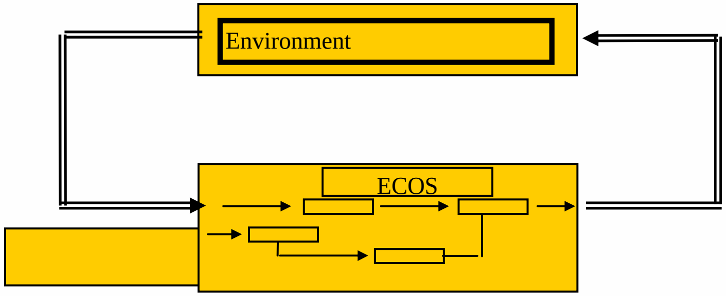
EIS



- Adaptive modelling of complex dynamic processes through incremental learning
- Methods: evolving NN (ECOS) – DENFIS, EFuNN, evolving FS - eTS, EC, statistical learning (e,g. SVM), hybrid systems, quantum inspired EIS
- Extracting relationship rules, knowledge.
- Facilitating applications and discoveries across disciplines – Bioinformatics, Neuroinformatics, Health informatics, Robotics, Business, Environment

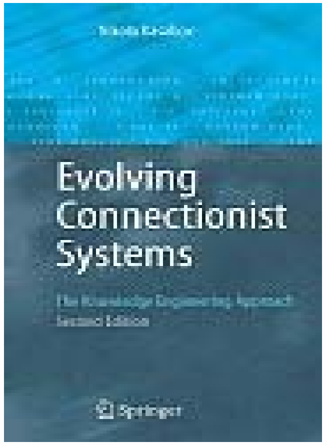
Evolving COnnectionist Systems – ECOS.

- ECOS are modular connectionist-based systems that evolve their structure and functionality in a continuous, self-organised, possibly on-line, adaptive, interactive way from incoming information; they can process both data and knowledge in a supervised and/or unsupervised way.



- Early examples of ECOS:
  - RAN (J.Platt, 1991) – evolving RBF NN
  - RAN with a long term memory – Abe et al, ;
  - Incremental FuzzyARTMAP;
  - Growing gas; etc.

- New developments:
  - EFuNN (Kasabov, 1998, 2001), DENFIS (Kasabov and Song, 2002)
  - EFuRS, eTS (P.Angelov, 2002)
  - SOFNN (McGinnity, Prasad, Leng, 2004)
  - TWNFI (Song and Kasabov, 2005)
  - Many other

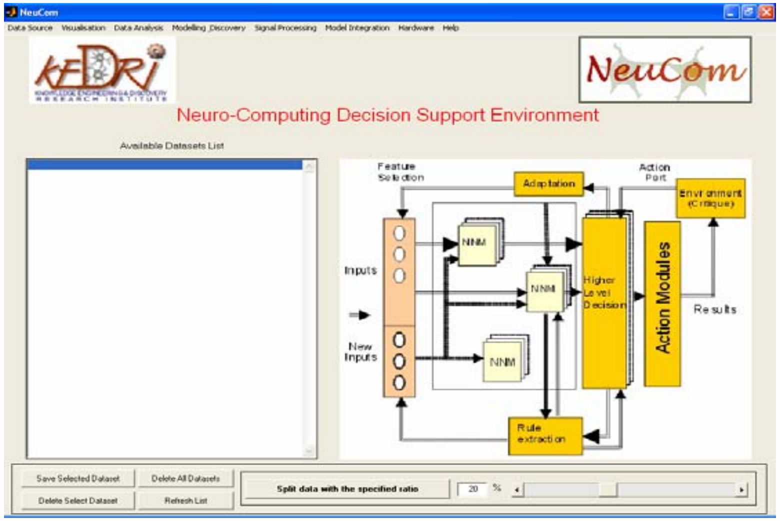


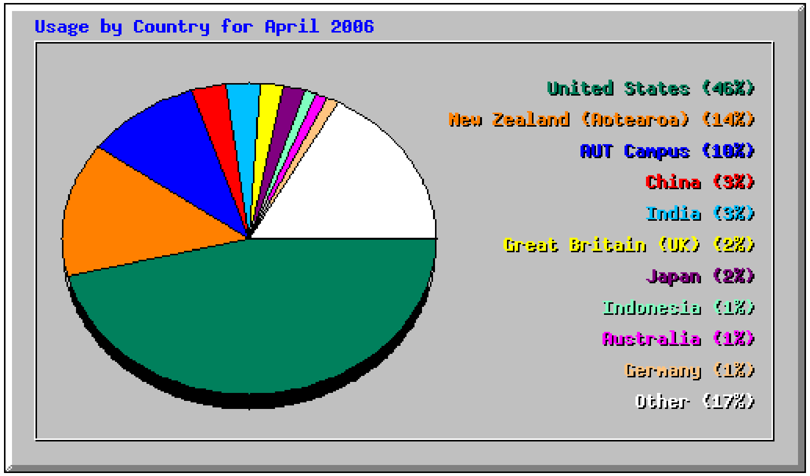
- ‘*Throw the “chemicals” and let the system grow, is that what you are talking about, Nik Walter Freeman, UC at Berkeley, a comment at “Iizuka’”1998 conference*

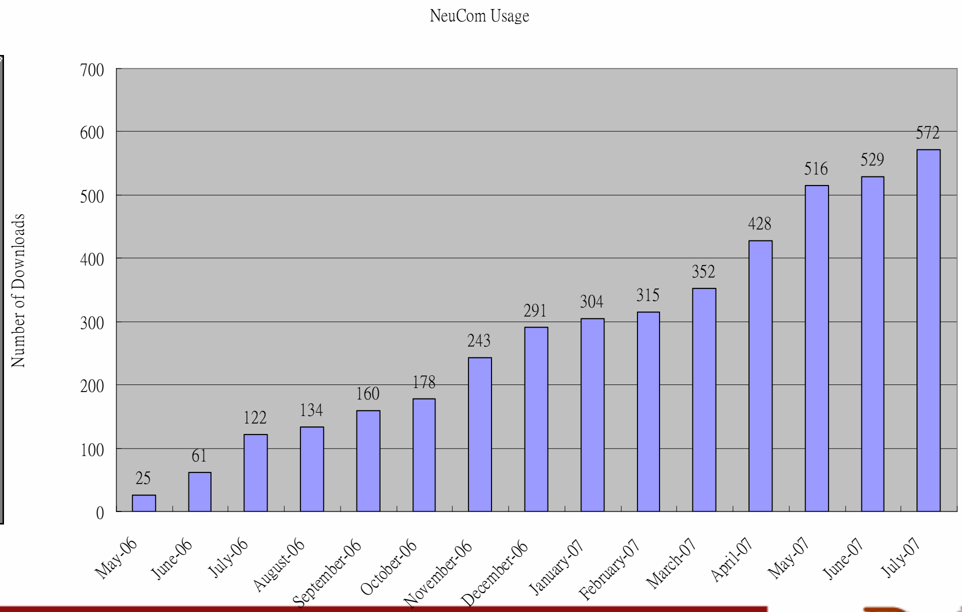
- N.Kasabov, Evolving connectionist systems: The knowledge engineering approach, second edition, Springer, 2007

NeuCom

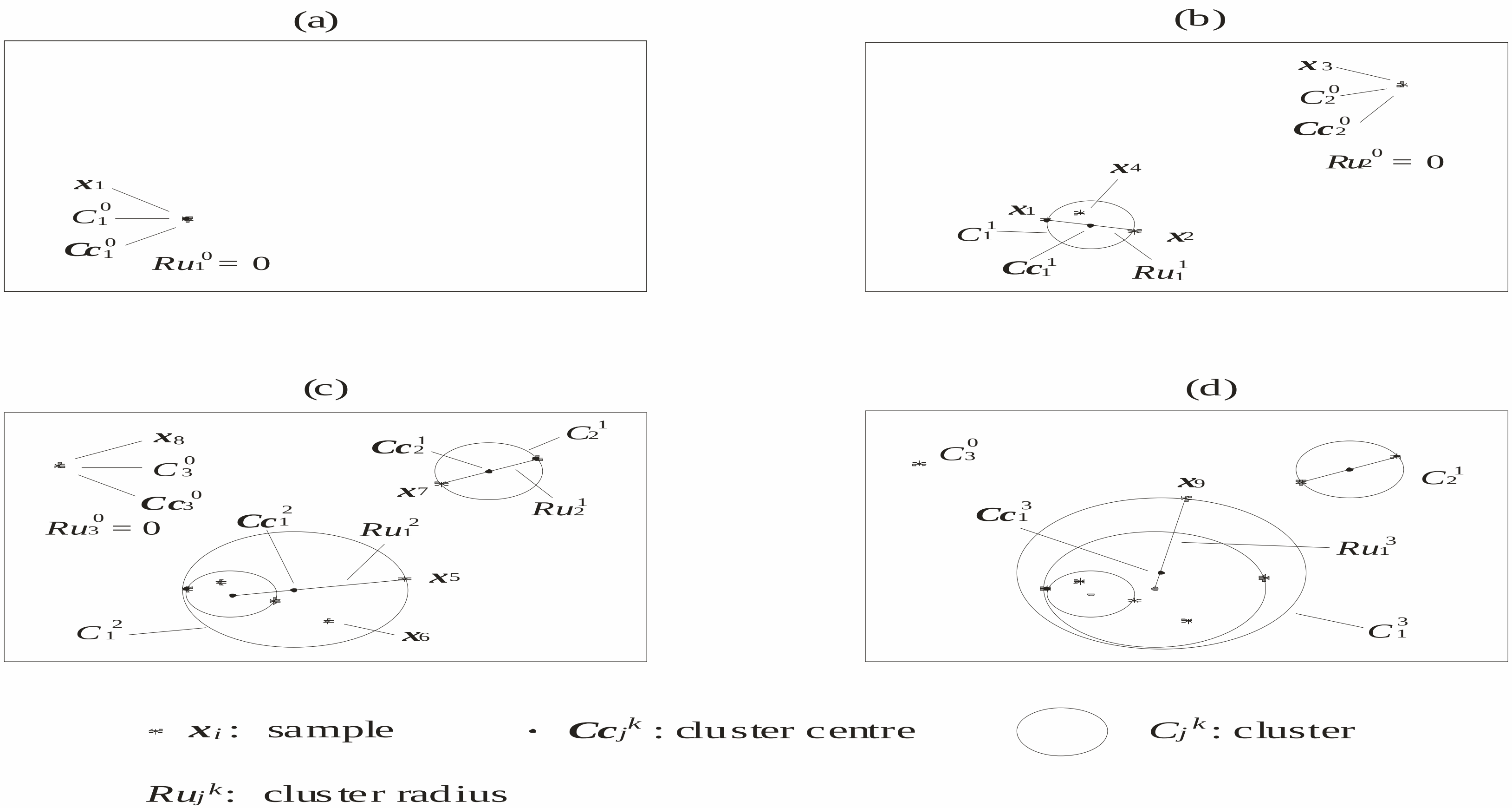
- **NeuCom** (www.theneucom.com): (P.Hwang et al.)
- A Software Environment for NeuroComputing, Data Mining and Intelligent Systems Design
- Incorporates 60 traditional and new techniques for intelligent data analysis and the creation of intelligent systems
- A free copy available for education and research from: www.theneucom.com
- Adopted in 70 research laboratories, institutes and universities from all over the world





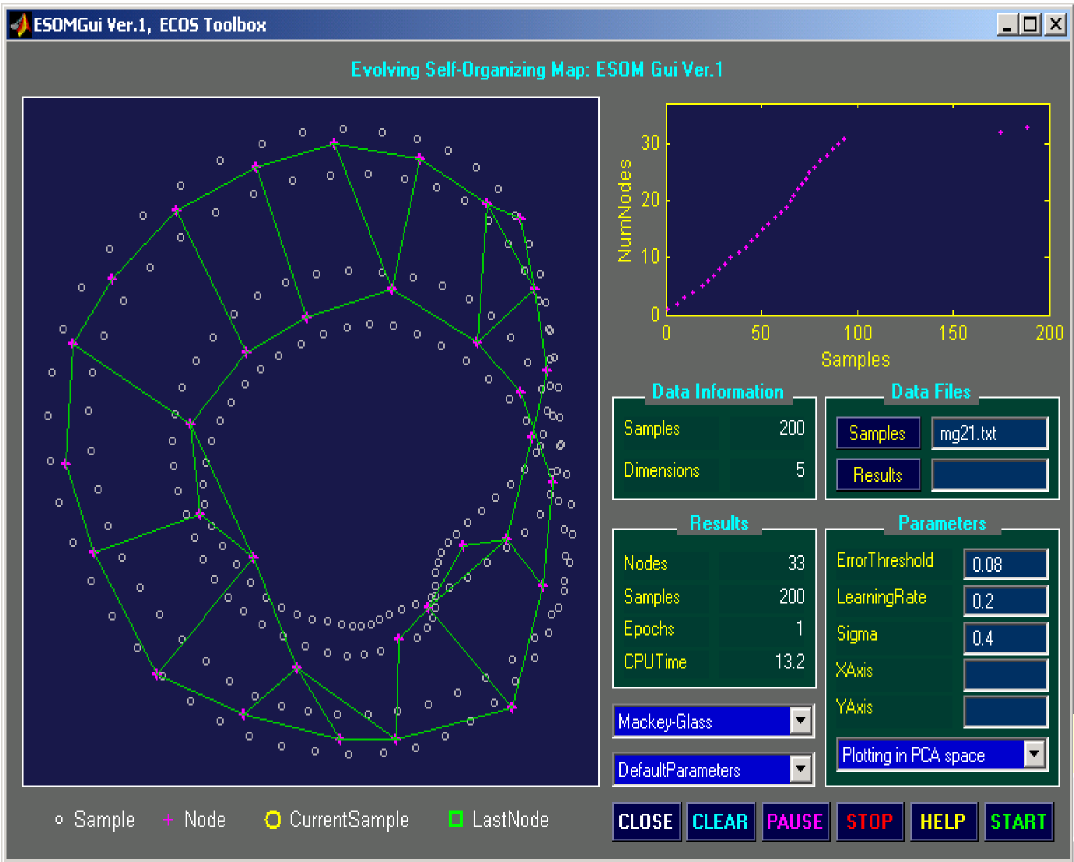


2. Evolving clustering methods. ECM



- ECM: Fast one-pass (or at most - several passes) algorithm for dynamic clustering of a stream of data
- Performs a simple evolving, on-line, maximum distance based clustering
- The figure shows an evolving clustering process using ECM with consecutive examples x1 to x9 in a 2D space
- If the learning is supervised – a local function is evolved in each cluster
- Demo

ESOM



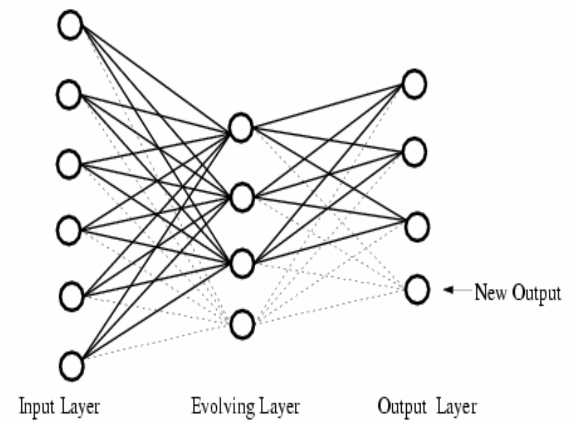
Model	No. of units	Error rate	No. of Epochs
GCS	145	0	180
DCS-GCS	135	0	135
LVQ	114	11.9%	50
SOM	144	22.2%	50
ESOM	105	0	1

3. Simple ECOS and Evolving Fuzzy Neural Networks

- Hidden nodes evolve, starting from no nodes at all.
- Each hidden node is a cluster center.
- Clusters grow in radius and shrink through a learning algorithm
- Each hidden node represents a local model (a rule) that associates an input cluster with an output function, e.g. a constant label, a linear function, a non-linear function, etc
- If a new input vector belongs to a cluster to certain degree, than the corresponding local model applies, otherwise – m of the closest models are used to calculate the output.
- Incremental **supervised clustering** with new input vectors **x**
- First layer of connections: $W1(r_j(t+1))=W1(r_j(t)) + lj.\ D(\mathbf{x}, W1(r_j(t))$
- Second layer: $W2(r_j(t+1)) = W2(r_j(t)) + lj.\ (y - A2).\ A1(r_j(t))$,

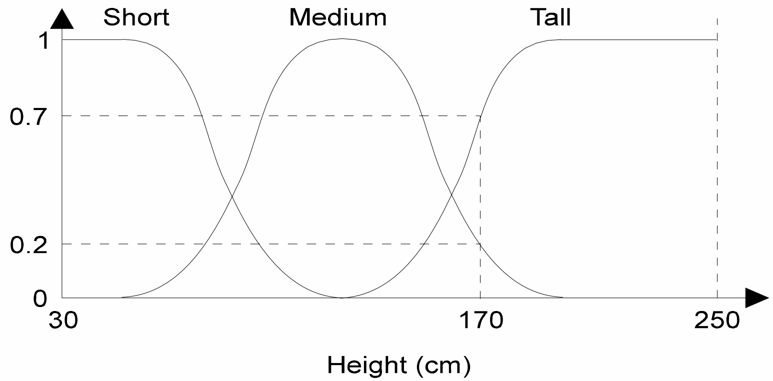
where: r_j is the jth rule node (hidden node); D – distance;
$A2=f2(W2.A1)$ is the activation vector of the output neurons when **x** is presented;

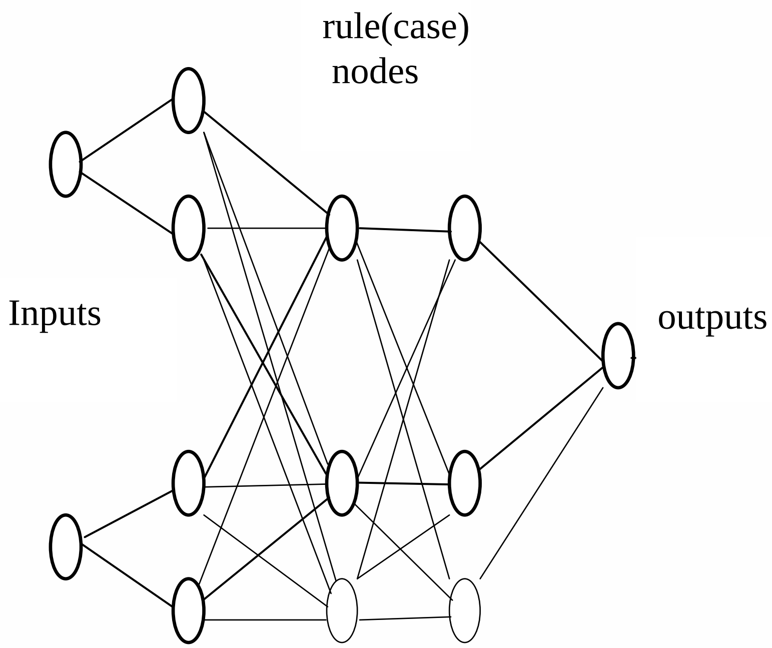
- $A1(r_j(t)) = f2(D(W1(r_j(t)), \mathbf{x}))$ is the activation of the rule node $r_j(t)$;
- a simple linear function can be used for f1 and f2, e.g. $A1(rj(t)) = 1- D(W1(r_j(t)), \mathbf{x})$;
- lj is the current learning rate of the rule node r_j calculated for example as $lj = 1/ Nex(r_j)$, where $Nex(r_j)$ is the number of examples associated with rule node r_j.



Evolving fuzzy neural networks EFuNN and ECF

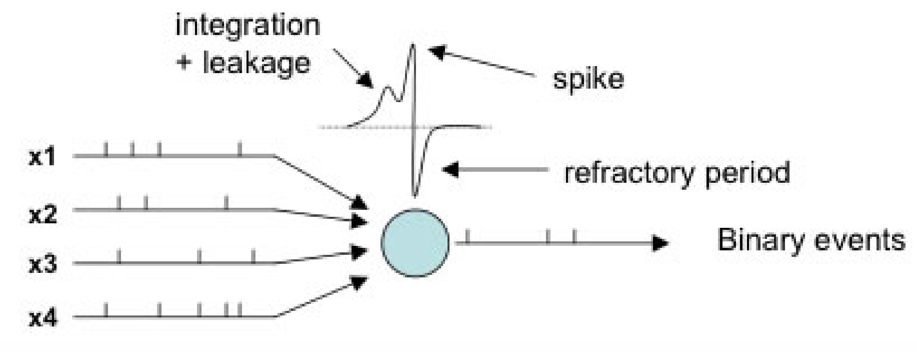
- As a general case, input and/or output variables can be non-fuzzy (crisp) or fuzzy
- Fuzzy variables
- Example of three Gaussian MF

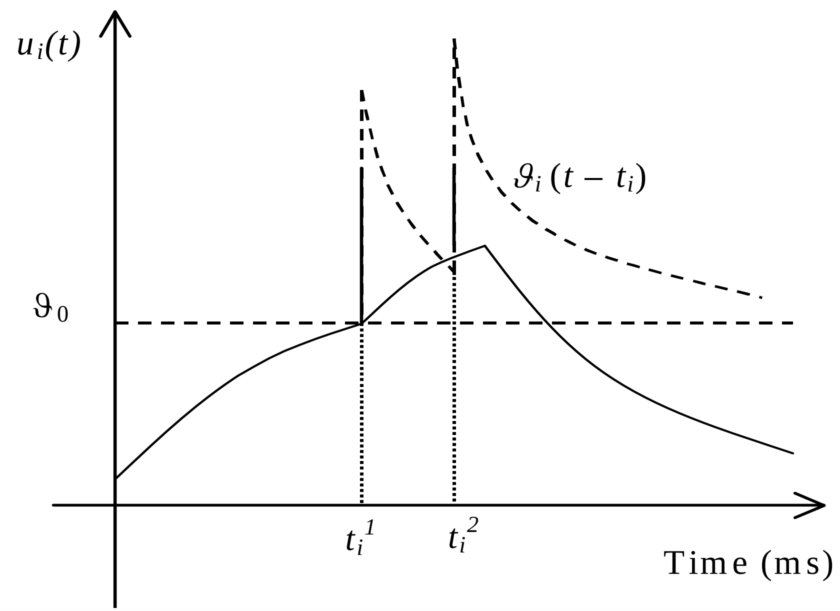




- *EFuNN, N. Kasabov, IEEE Tr SMC, 2001*
- Partial case: ECF – evolving classifier function – no output MF, only input MF.
- Supervised clustering
- Zadeh-Mamdani fuzzy rules
- Simple version – ECF. Parameters: Rmax, Rmin, #input MF (e.g. 1,2,3,...), m-of-n (e.g. 1,2,3,..), # iterations for training (e.g. 1,2,3, …
- Demo

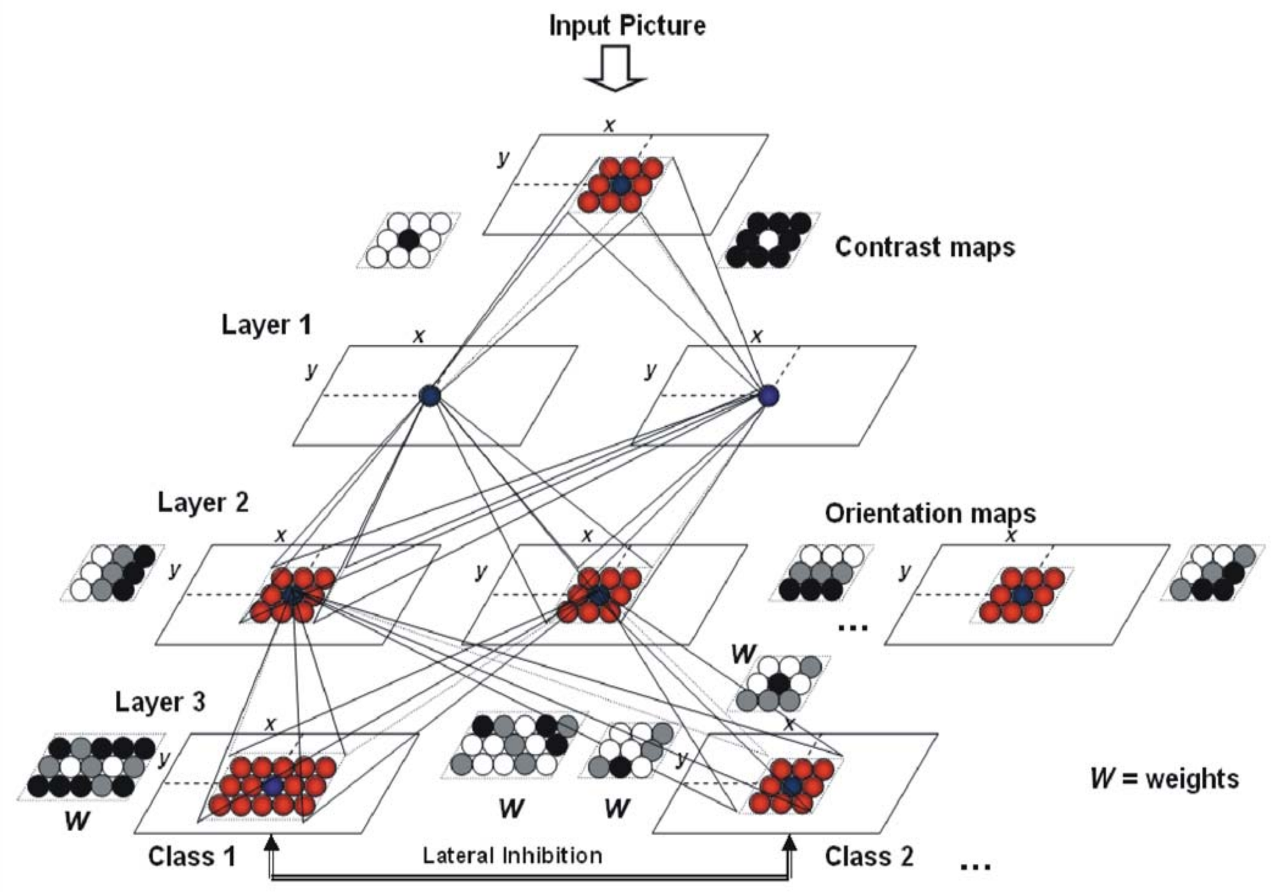
4. Evolving Spiking neural networks – strongly brain-inspired models





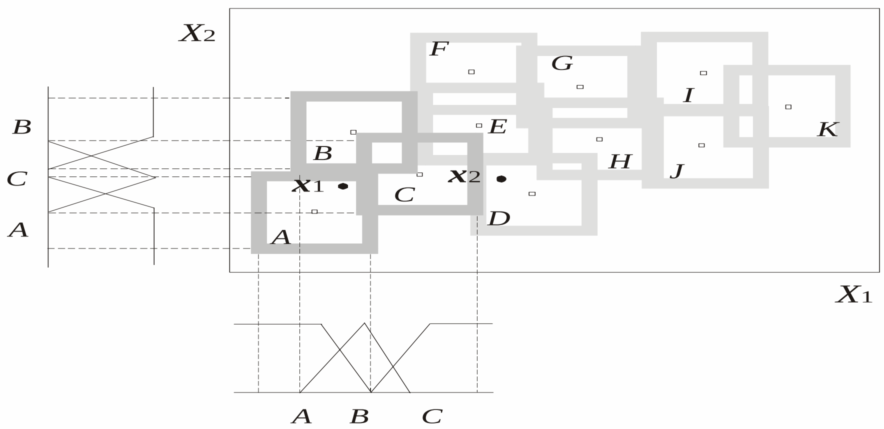
Evolving SNN

(Wysoski, Benuskova, Kasabov, Proc. ICANN, LNCS, Springer2006)

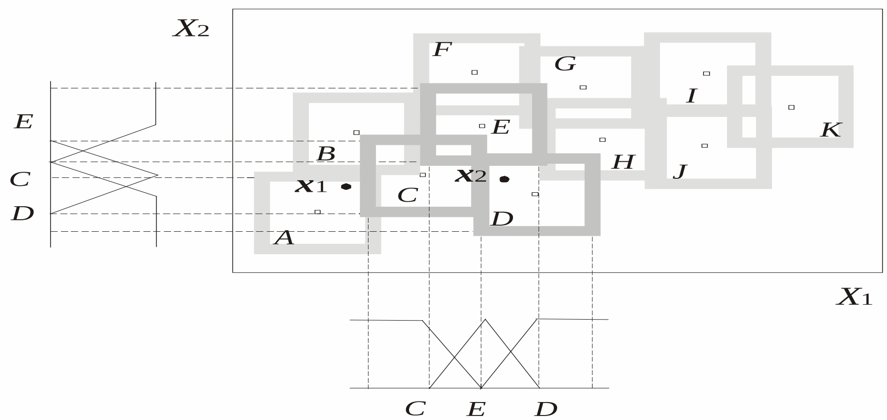


5. Evolving Neuro-Fuzzy Inference Systems
(DENFIS, Kasabov and Song, 2002, IEEE Tr Fuzzy Systems)

(a) Fuzzy rule group 1 for a DENFIS



(b) Fuzzy rule group 2 for a DENFIS



DENFIS algorithm:

(1) Learning:

- Unsupervised, incremental clustering.
- For each cluster there is a Takagi-Sugeno fuzzy rule created: IF x is in cluster Cj THEN yj = fj (x),

where: yi = β0 + β1 x1 + β2 x2 + … + βq

- Incremental learning of the function coefficients and weights of the functions through least square error

(2) Fuzzy inference over fuzzy rules:

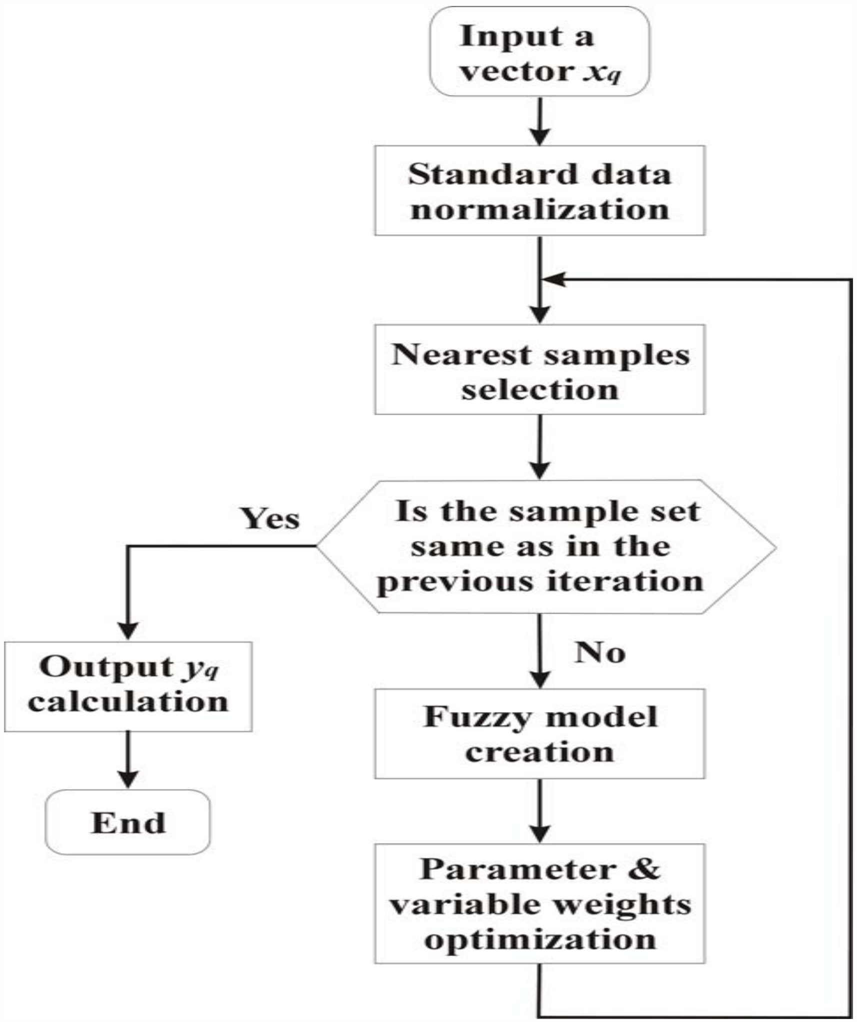
- For a new input vector x = [x1,x2, … , xq] DENFIS chooses m fuzzy rules from the whole fuzzy rule set for forming a current inference system.
- The inference result is:

$$y = \frac{\Sigma_{i=1,m} [\, \omega i\; fi\, (x1, x2, \ldots, xq)]}{\Sigma_{i=1,m}\, \omega i}$$

Two fuzzy rule groups are formed by DENFIS to perform inference for 2 input vectors (Fig 5.7)

Transductive Neuro Fuzzy Inference with Weighted Data Normalisation – TWNFI, for personalised modelling

(Q.Song and N.Kasabov, IEEE Tr FS, December 2005, and Neural Networks, Dec. 2006)



After the nearest samples are selected for an input vector **x,** the samples are clustered using ECM.

Fuzzy rules are created/derived for each cluster:

R_l :If x_1 is F_{l1} and x_2 is F_{l2} and … x_P is F_{lP}, then y is G_l,

where F_{lj} and Gl are fuzzy sets defined by Gaussian type membership functions.

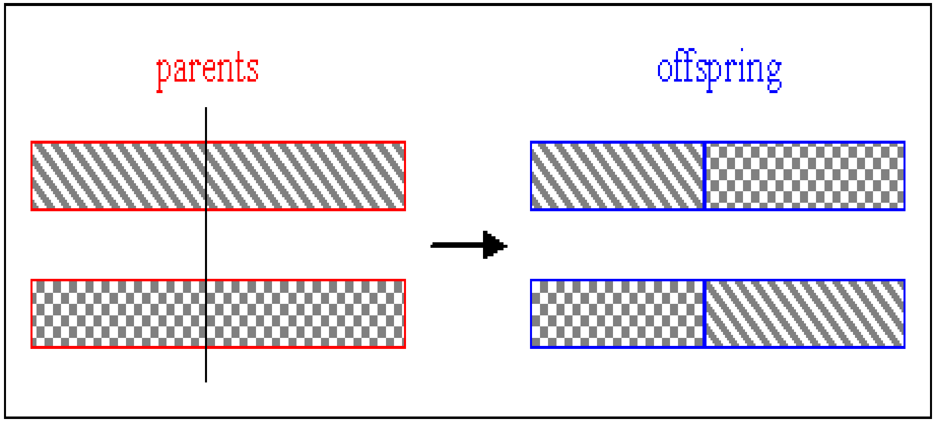
Input variable weights w_j and fuzzy rule parameters are optimized through the steepest descent algorithm.

$$f(x_i) = \frac{\sum_{l=1}^{M} \frac{n_l}{\delta_l^2} \prod_{j=1}^{P} \alpha_{lj} \exp\left[-\frac{w_j^2 (x_{ij} - m_{lj})^2}{2\sigma_{lj}^2}\right]}{\sum_{l=1}^{M} \frac{1}{\delta_l^2} \prod_{j=1}^{P} \alpha_{lj} \exp\left[-\frac{w_j^2 (x_{ij} - m_{lj})^2}{2\sigma_{lj}^2}\right]}$$

6. Evolutionary Computation for feature, parameter and structure optimisation of ECOS

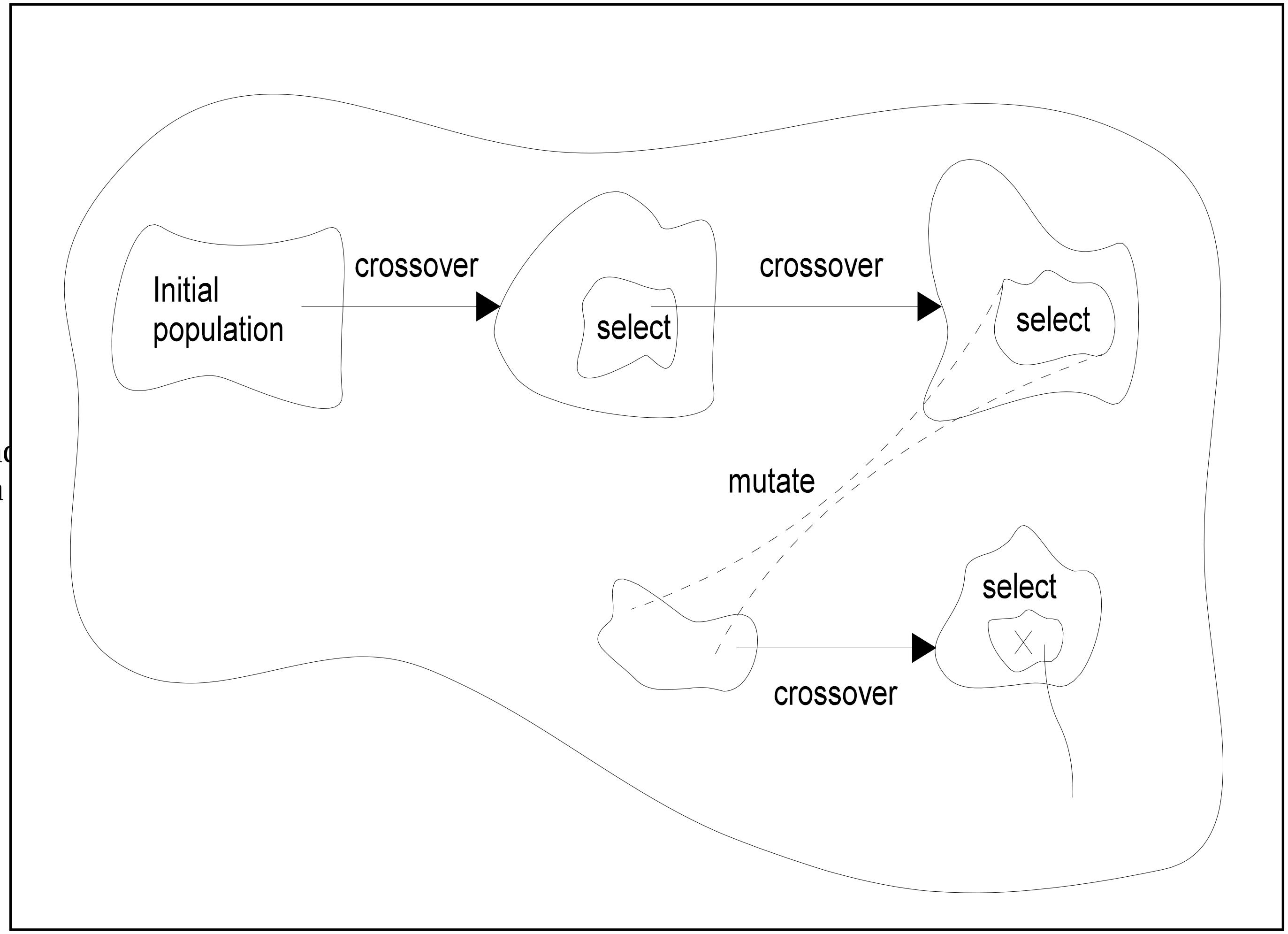
Evolutionary computation.
Terminology:

- *Gene*
- *Chromosome*
- *Population*
- *Crossover*
- *Mutation*
- *Fitness function*
- *Selection*



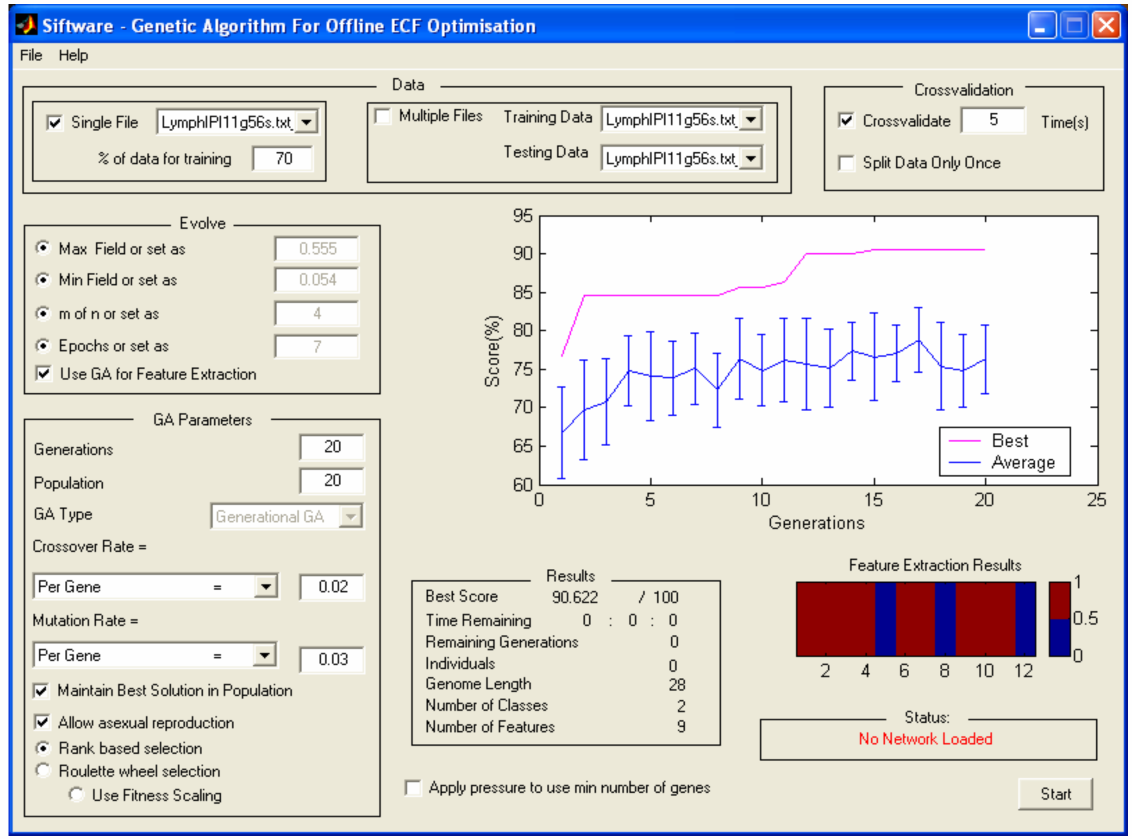
GA and ECOS

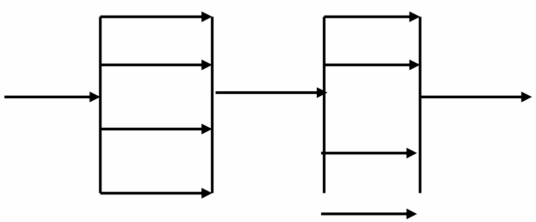
- Many individual ECOS are evolved simultaneously on the same data through a GA method
- A chromosome represents each individual ECOS parameters
- Individuals are evaluated and the best one is selected for a further development
- Mutation



EC for feature, parameter and structure optimisation of ECF ECOS

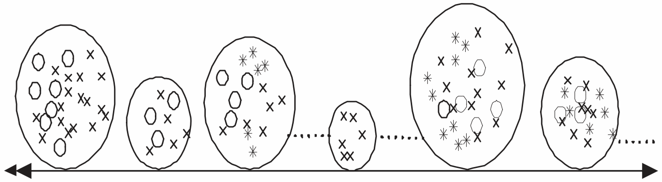
- GA optimisation of the parameters of the model and the input features
- A chromosome contains as "genes" all model parameters and input features (yes, no)
- Replication of individual models and selection of:
    - The best one
    - The best *m* averaged, etc





Online feature selection for EIS with incremental PCA and LDA

- Let us consider the case that the (*N+1*)*th* training sample is presented. The addition of this new sample will lead to the changes in both of the mean vector and covariance matrix; therefore, the eigenvectors and eigenvalues should also be recalculated. The mean input vector is easily updated.



- If the new sample has almost all energy in the current eigenspace, dimensional augmentation is not needed. However, if it has some energy in the complementary space to the current eigenspace, the dimensional augmentation cannot be avoided. When the norm of the residue vector is larger than a threshold value, it must allow the number of dimensions to increase from *k* to *k+1*, and the current eigenspace must be expanded.
- Application: Face recognition
- Recent publications:
  - S. Ozawa, S.Too, S.Abe, S. Pang and N. Kasabov, *Incremental Learning of Feature Space and Classifier for Online Face Recognition*, Neural Networks, August, 2005
  - S. Pang, S. Ozawa and N. Kasabov, *Incremental Linear Discriminant Analysis for Classification of Data Streams*, IEEE Trans. SMC-B, vol. 35, No. 4, 2005

7. Multimodel ECOS.
Model and data integration through ECOS

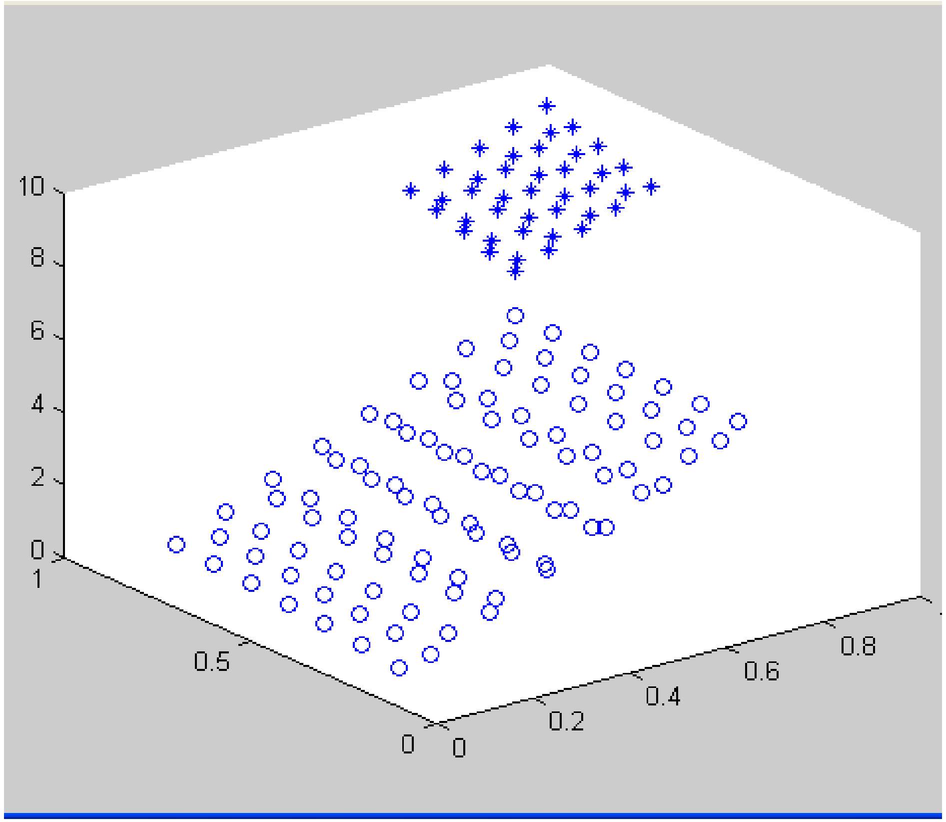
-**Inserting initial rules** (existing knowledge) and training with new data

- **Generating data from existing model** and training an ECOS on both old and new data

- **New rules evolve continuously**

Example: A 3D plot of data D_0 (data samples denoted as "o") generated from a model *M* (formula) $y = 5.1x_1+0.345x_1^2 - 0.83x_1 \log_{10} x_2 + 0.45x_2 +0.57 \exp(x_2^{0.2})$ in the sub-space of the problem space defined by x_1 and x_2 both having values between 0 and 0.7, and

• New data *D* (samples denoted as "*") defined by x_1 and x_2 having values between 0.7 and 1;



Prototype rules evolved through DENFIS and EFuNN after model and new data integration

Takagi-Sugeno fuzzy rules (DENFIS):

- Rule 1: IF x_1 is (-0.05, 0.05, 0.14) and x_2 is (0.15,0.25,0.35) THEN $y = 0.01 + 0.7x_1 + 0.12x_2$
- Rule 2: IF x_1 is (0.02, 0.11, 0.21) and x_2 is (0.45,0.55, 0.65) THEN $y = 0.03+ 0.67x_1+ 0.09\ _{x2}$
- Rule 3: IF x_1 is (0.07, 0.17, 0.27) and x_2 is (0.08,0.18,0.28) THEN $y = 0.01 +0.71x_1 + 0.11x_2$
- Rule 4: IF $x_1$is (0.26, 0.36, 0.46) and x_2 is (0.44,0.53,0.63) THEN $y = 0.03+ 0.68x_1+ 0.07x_2$
- Rule 5: IF $x_1$is (0.35, 0.45, 0.55) and x_2 is (0.08,0.18,0.28) THEN $y = 0.02 + 0.73x_1+ 0.06x_2$
- Rule 6: IF $x_1$is (0.52, 0.62, 0.72) and x_2 is (0.45,0.55,0.65) THEN $y = -0.21 + 0.95x_1 + 0.28x_2$
- Rule 7: IF $x_1$is (0.60, 0.69,0.79) and x_2 is (0.10,0.20,0.30) THEN $y = 0.01+ 0.75x_1+ 0.03x_2$

- **New rules:**
- **Rule 8: IF $x_1$is (0.65,0.75,0.85) and x_2 is (0.70,0.80,0.90) THEN $y =-0.22+0.75x_1+0.51x_2$**
- **Rule 9: IF $x_1$is (0.86,0.95,1.05) and x_2 is (0.71,0.81,0.91) THEN $y =0.03 + 0.59x_1+0.37x_2$**

Zhade-Mamdani fuzzy rules (ECF, EFuNN):

Rule 1: IF x_1 is (Low 0.8) and x_2 is (Low 0.8) THEN y is (Low 0.8), radius R_1=0.24; N_{1ex}= 6
Rule 2: IF x_1 is (Low 0.8) and x_2 is (Medium 0.7) THEN y is (Small 0.7), R_2=0.26, N_{2ex}= 9

Rule 3: IF x_1 is (Medium 0.7) and x_2 is (Medium 0.6) THEN y is (Medium 0.6), R_3 = 0.17,N_{3ex}=17

Rule 4: IF x_1 is (Medium 0.9) and x_2 is (Medium 0.7) THEN y is (Medium 0.9), R_4 = 0.08, N_{4ex}=10

Rule 5: IF x_1 is (Medium 0.8) and x_2 is (Low 0.6) THEN y is (Medium 0.9), R_5= 0.1, N_{5ex} = 11

Rule 6: IF x_1 is (Medium 0.5) and x_2 is (Medium 0.7) THEN y is (Medium 0.7), R_6= 0.07,N_{6ex}= 5

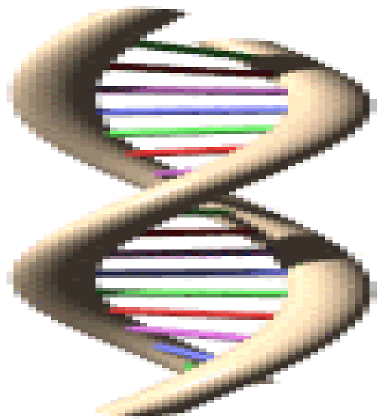
New rules:
Rule 7: IF x1 is (High 0.6) and x2 is (High 0.7) THEN y is (High 0.6), R_7 = 0.2, N_{7ex} = 12

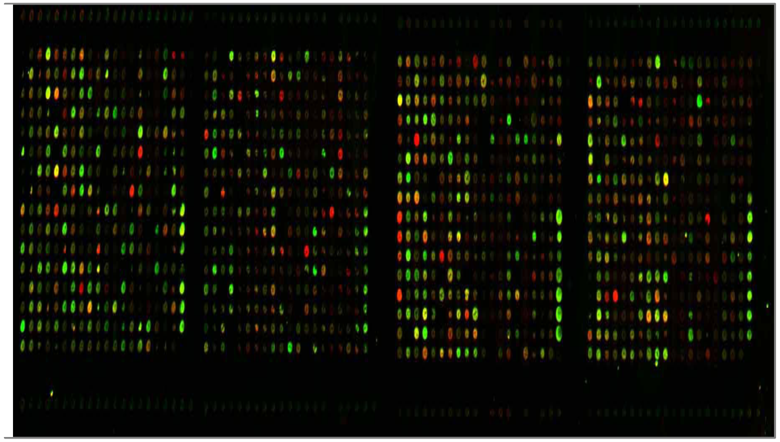
Rule 8: IF x1 is (High 0.8) and x2 is (Medium 0.6) THEN y is (High 0.6), R_8=0.1,N_{8ex}= 5

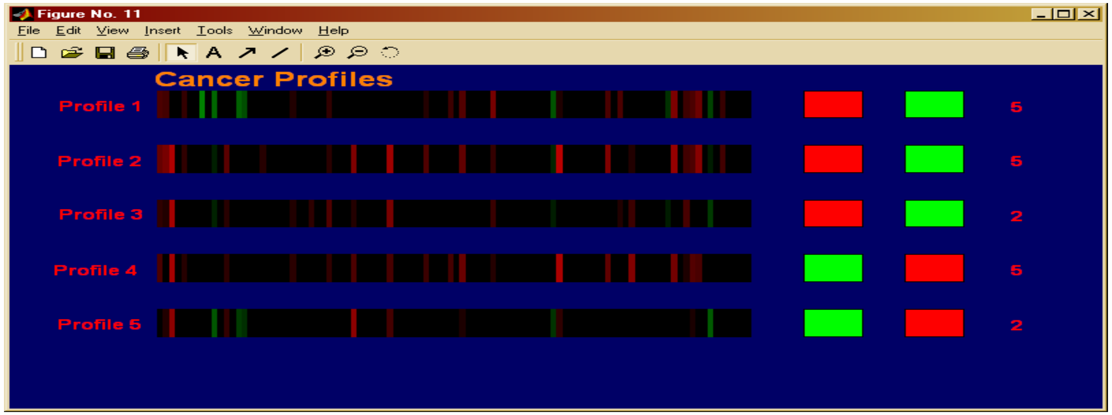
Rule 9: IF x1 is (High 0.8) and x2 is (High 0.8) THEN y is (High3 0.8), R_9= 0.1, N_{9ex} = 6

8. Applications in Bioinformatics
Gene expression data profiling

- DNA analysis - large data bases; data always being added and modified; different sources of information
- Markers and drug discoveries:
  - Gastric cancer
  - Bladder cancer
  - CRC
  - www.pedblnz.com
- Specialised software SIFTWARE

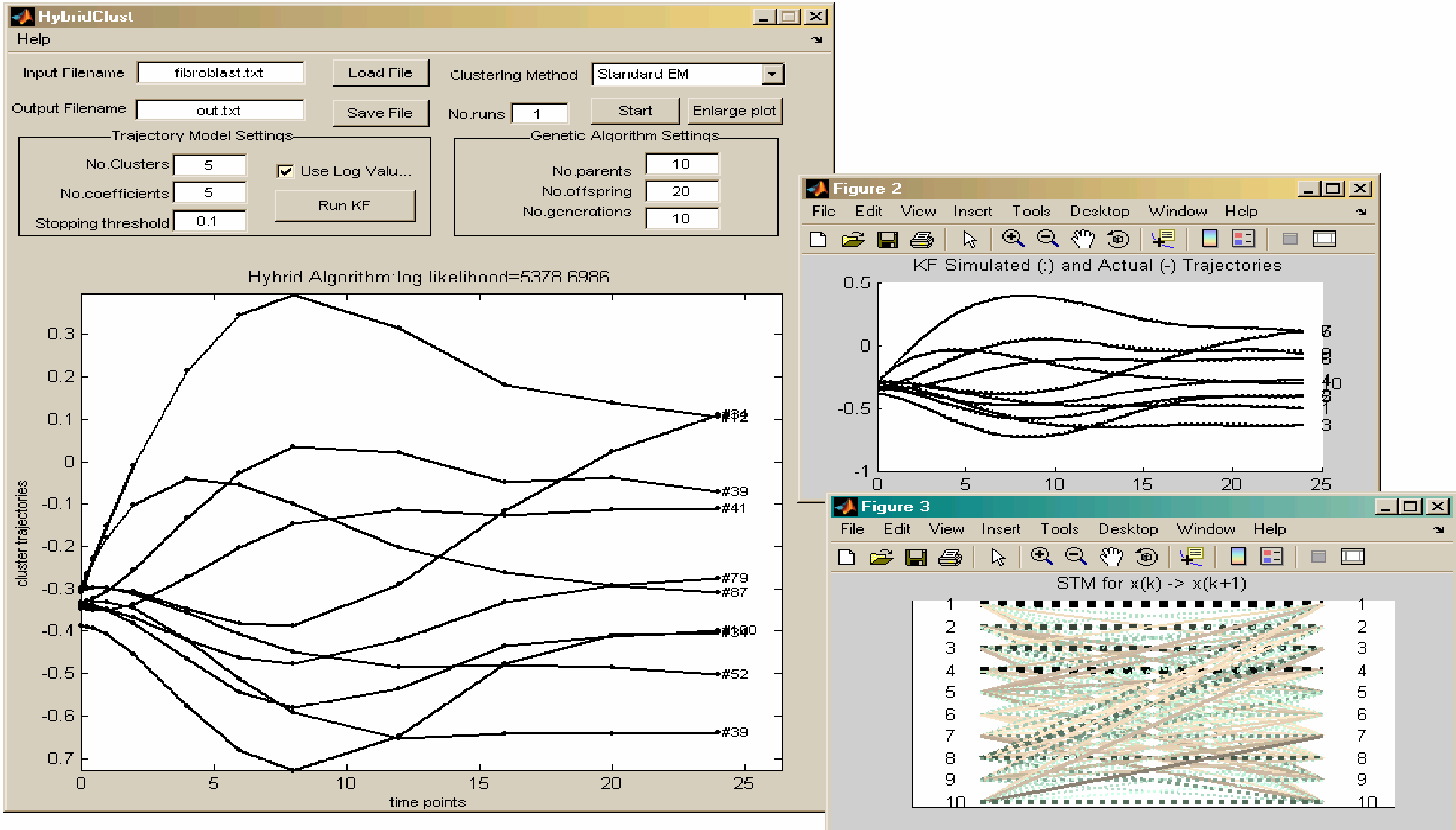






- Kasabov, N., *Modelling and profile discovery in Bioinformatics: Global, local and personalised approach,* Pattern Recognition Letters, Jan. 2007

A gene regulatory network modelling software system, GeneNetXP (Chan, Jain et al, 2006)



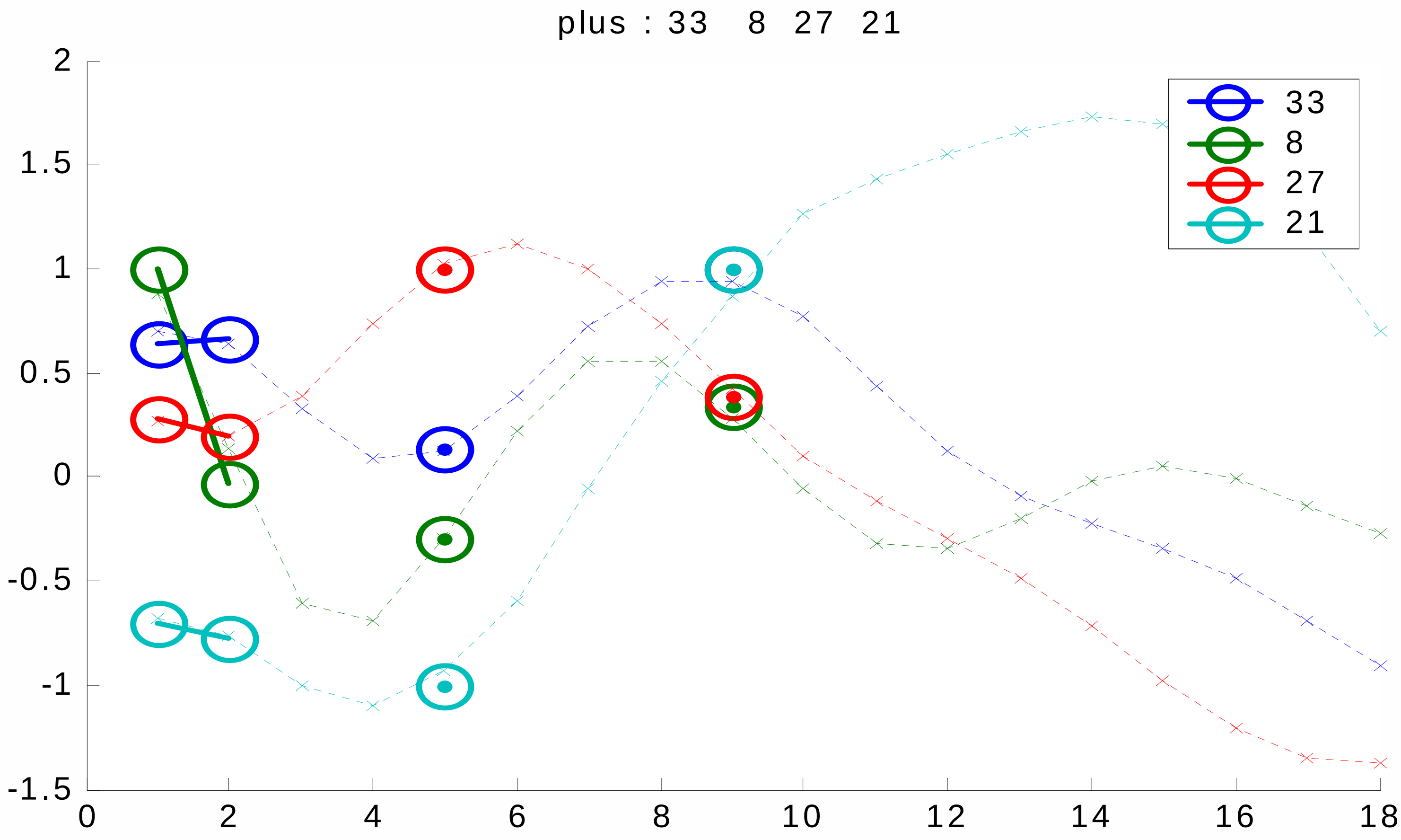
Evolving fuzzy neural networks for GRN modeling

(Kasabov and Dimitrov, ICONIP, 2002)

- On-line, incremental learning of a GRN
- Adding new inputs/outputs (new genes)
- The rule nodes capture clusters of input genes that are related to the output genes
- Rules can be extracted that explain the relationship between G(t) and G(t+dt), e.g.: *IF g13(t) is High (0.87) and g23(t) is Low (0.9)*

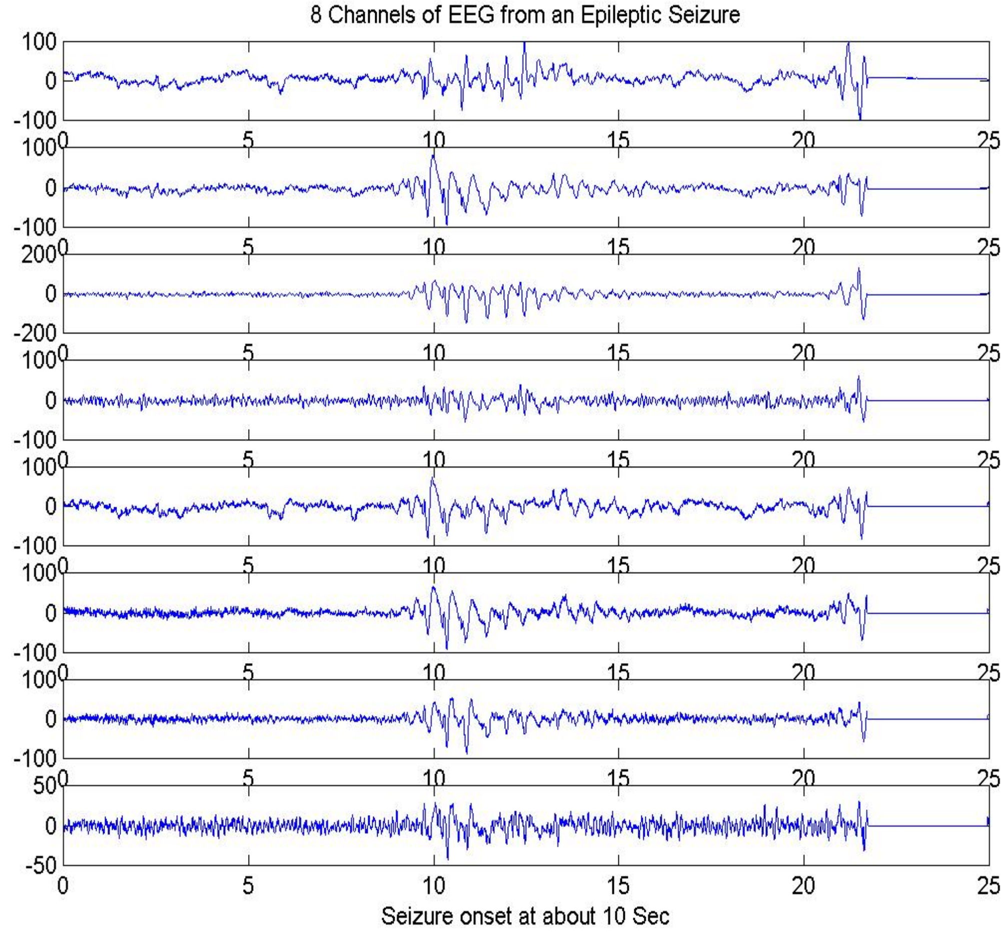
  THEN g87 (t+dt) is High (0.6) and g103(t+dt) is Low
- Playing with the threshold will give stronger or weaker patterns of relationship

Using a GRN model to predict the expression of genes in a future time



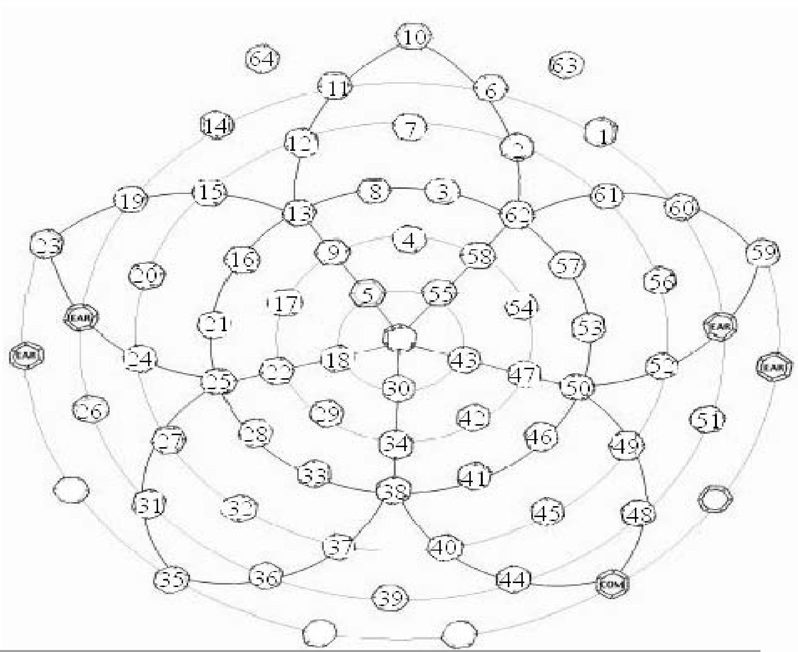
9. Applications in Neuroinformatics EEG data modelling

- Why ECOS for brain study'
- Modeling brain states of an individual or groups of individuals from EEG, fMR and other information
- Example: epileptic seizure a patient; 8 EEG channels data is shown
- Future direction: building dynamic models, e.g. brain- regulatory networks
- Applications:
  - Neuroscience: Understanding and discoveries
  - Engineering: Brain- computer interfaces



ECOS for modeling perception states based on EEG data

- Standard EEG electrode systems
- In the experiment here, four classes of brain perception states are used with 37 single trials each of them including the following stimuli:
  - Class1 - Auditory Stimulus;
  - Class2 - Visual Stimulus;
  - Class3 - Mixed Auditory and visual stimuli;
  - Class 4 - No stimulus.
- (With van Leewen et al, RIKEN, BSI, Tokyo)

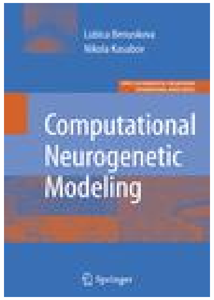


Stimulus	A	V	AV	No	Accuracy
A	81.2	1.3	0.1	0.2	98%
V	1.1	82.4	2.9	1.8	93.4%
AV	0.6	3.3	75	1.4	93.4%
No	0.4	1.5	1.3	80.5	96.2%

nkasabov@aut.ac.nz

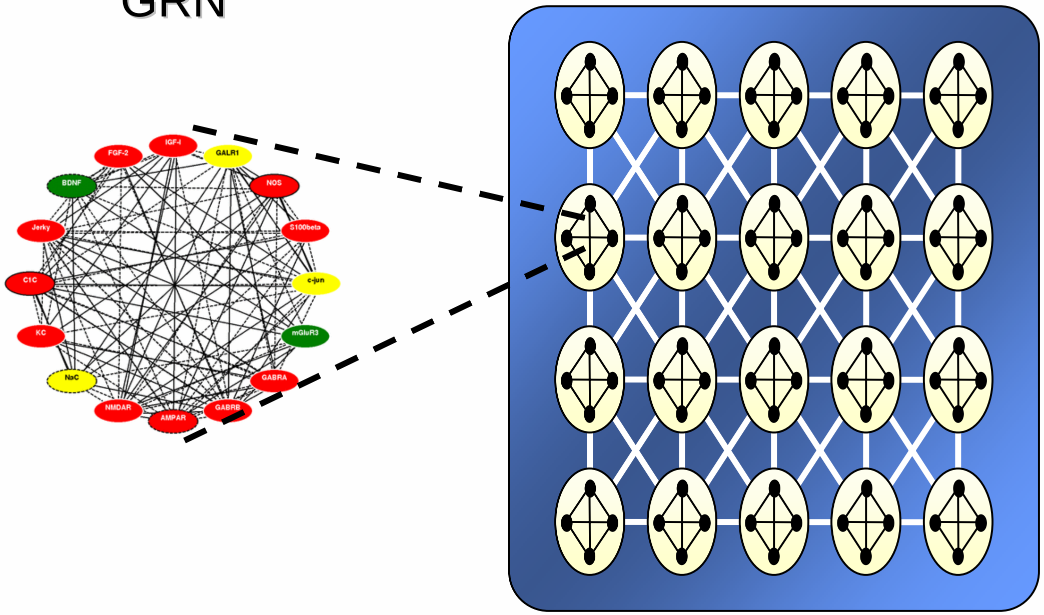
Computational Neurogenetic Modelling (CNGM)

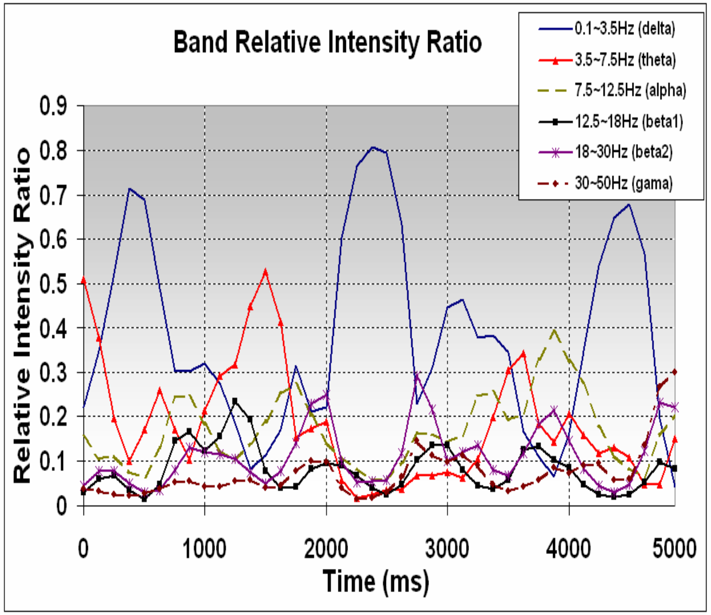
L.Benuskova and N.Kasabov, Springer, 2007



CNGM as an ANN

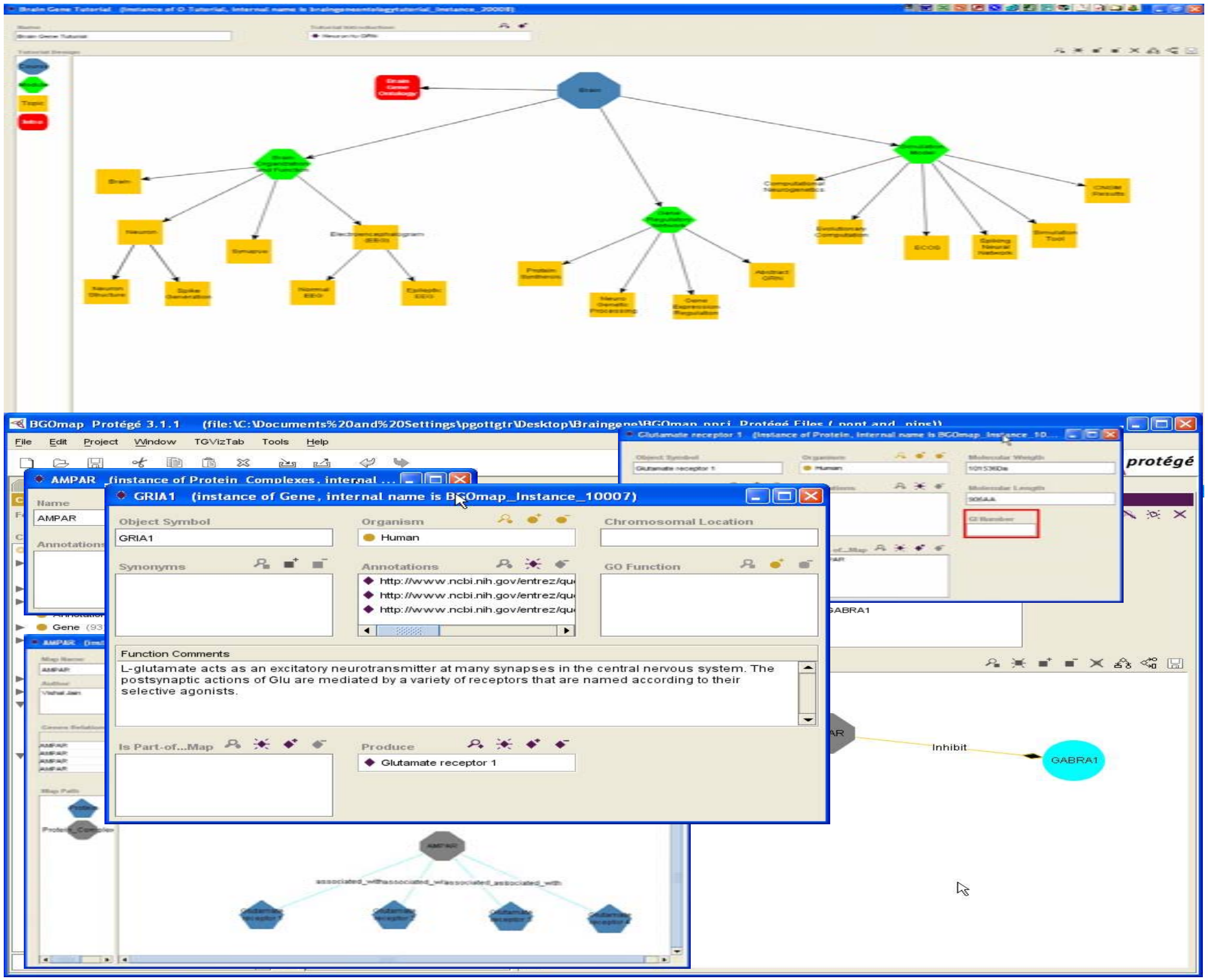
GRN

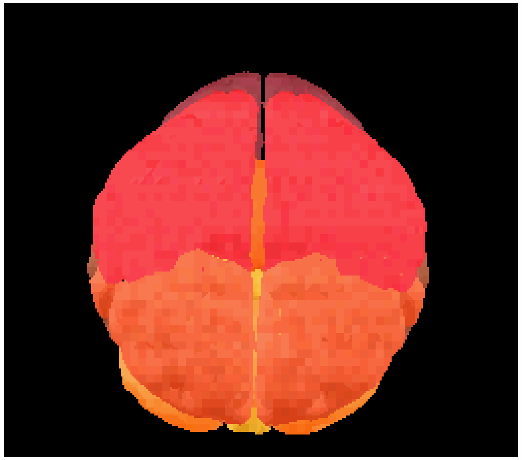


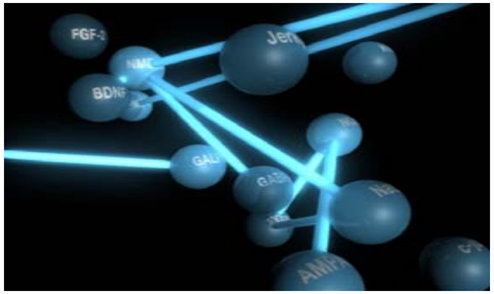


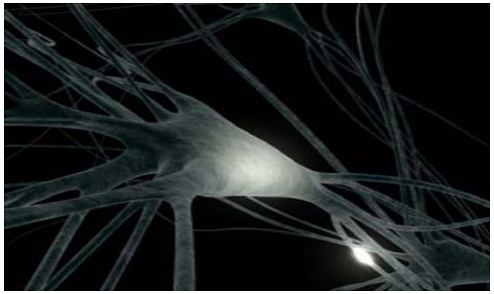
nkasabov@aut.ac.nz

BGO: Brain-gene Ontology System
(IJCNN 2007)



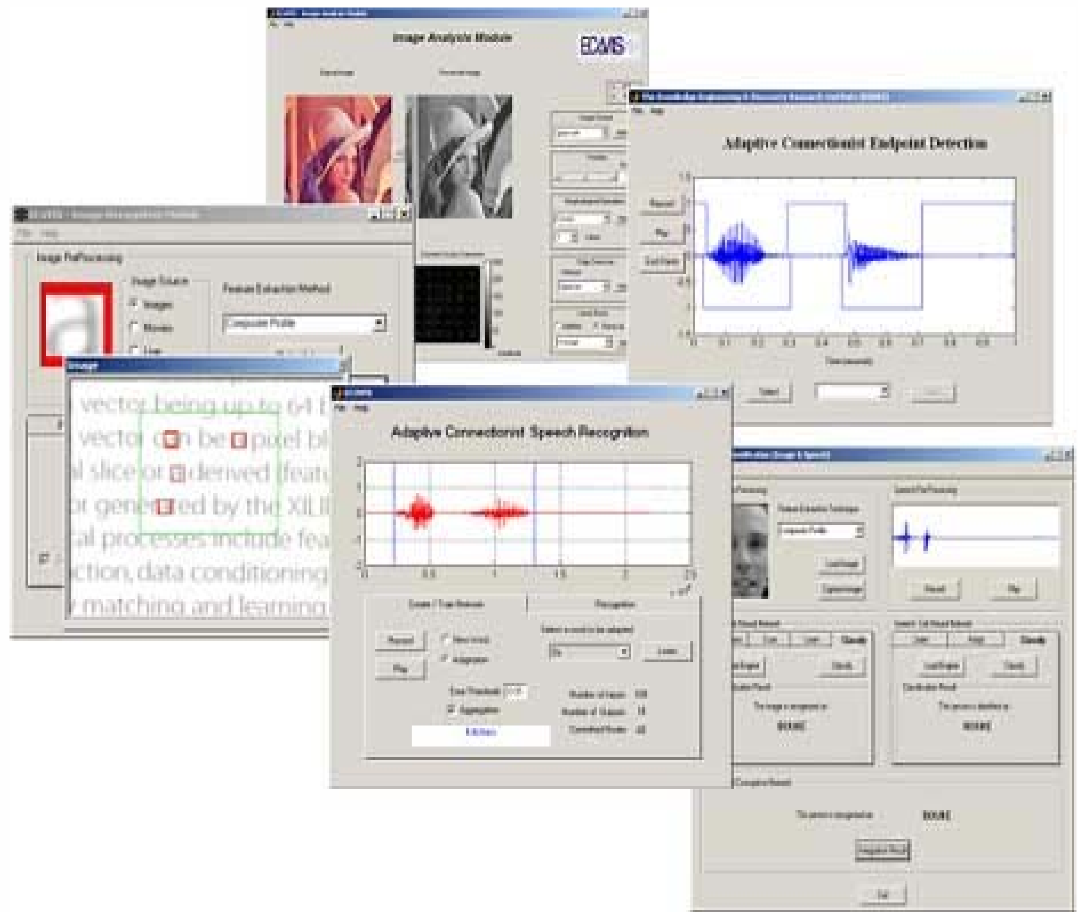






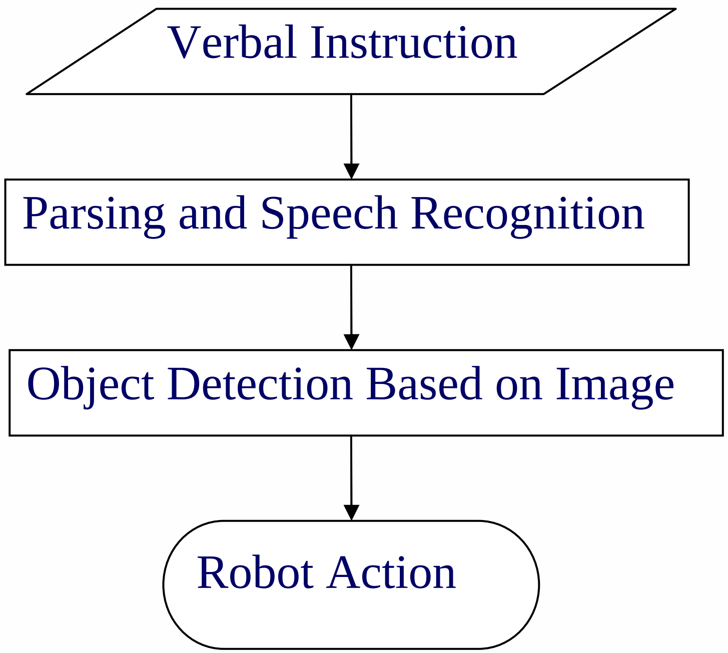
10. Adaptive Pattern Recognition and Robotics

- Adaptive speech recognition, image and video data processing
- ECOVIS prototype system
- Multimodal (face-, finger print-, iris-, and voice) person verification system
- Future research: Other modality recognition models and systems such as: odour-, DNA-, waves-, personal magnetic fields.



ECOS for adaptive multimodal signal processing, speech and image

A simple case study: ECOS-based, adaptive, voice controlled object recognition system



	Speech Utterance
1	"Pen"
2	"Rubber"
3	"Cup"
4	"Orange"
5	"Circle"
6	"Ellipse"
7	"Rectangle"

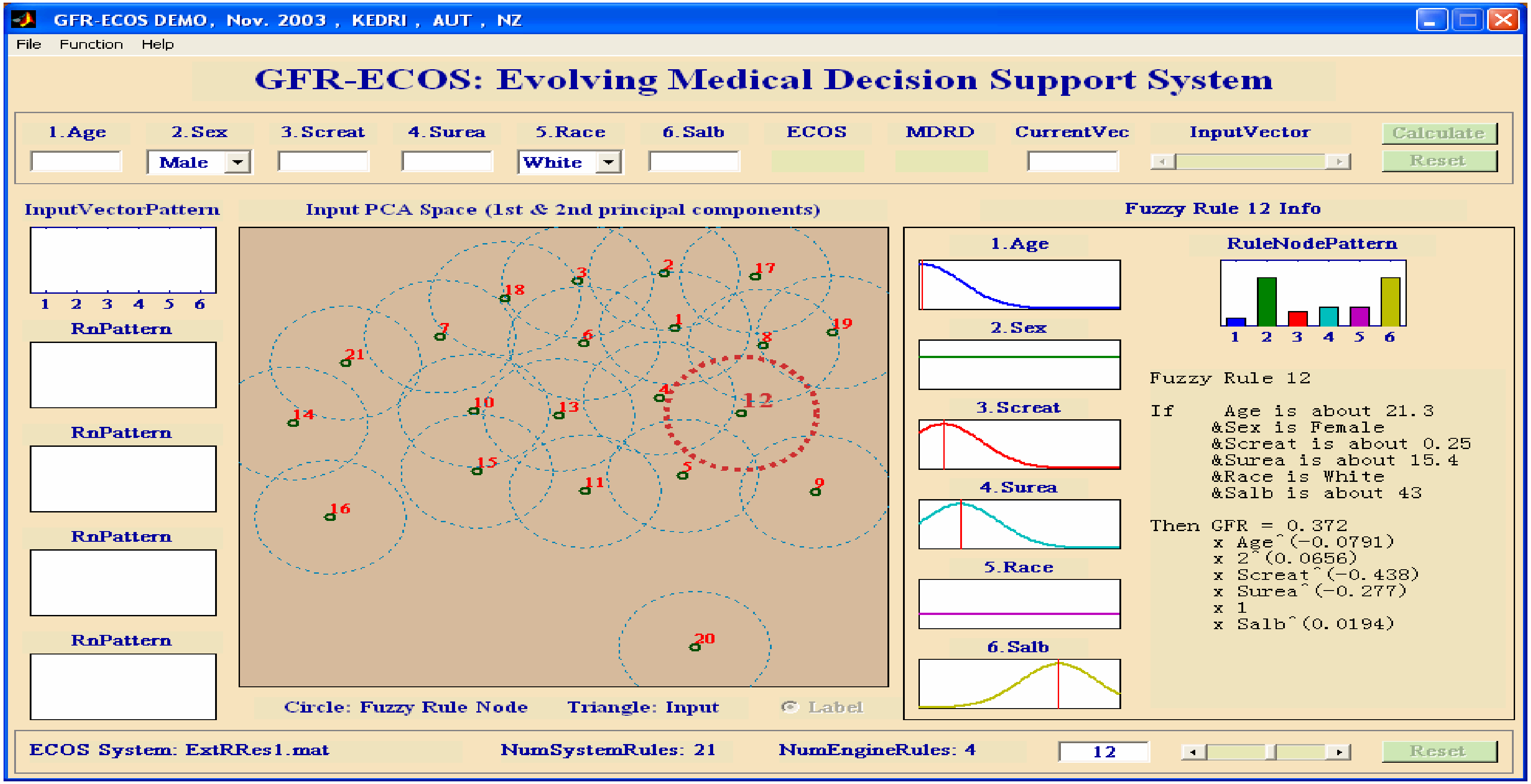
Robocup

1 gold and 2 silver medals for the Singapore Polytechnic at the World Robo-cup in Seoul, 2004, using ECOS; Bronze medal in Japan, 2005 (Loulin)

11 .Applications of ECOS for DSS

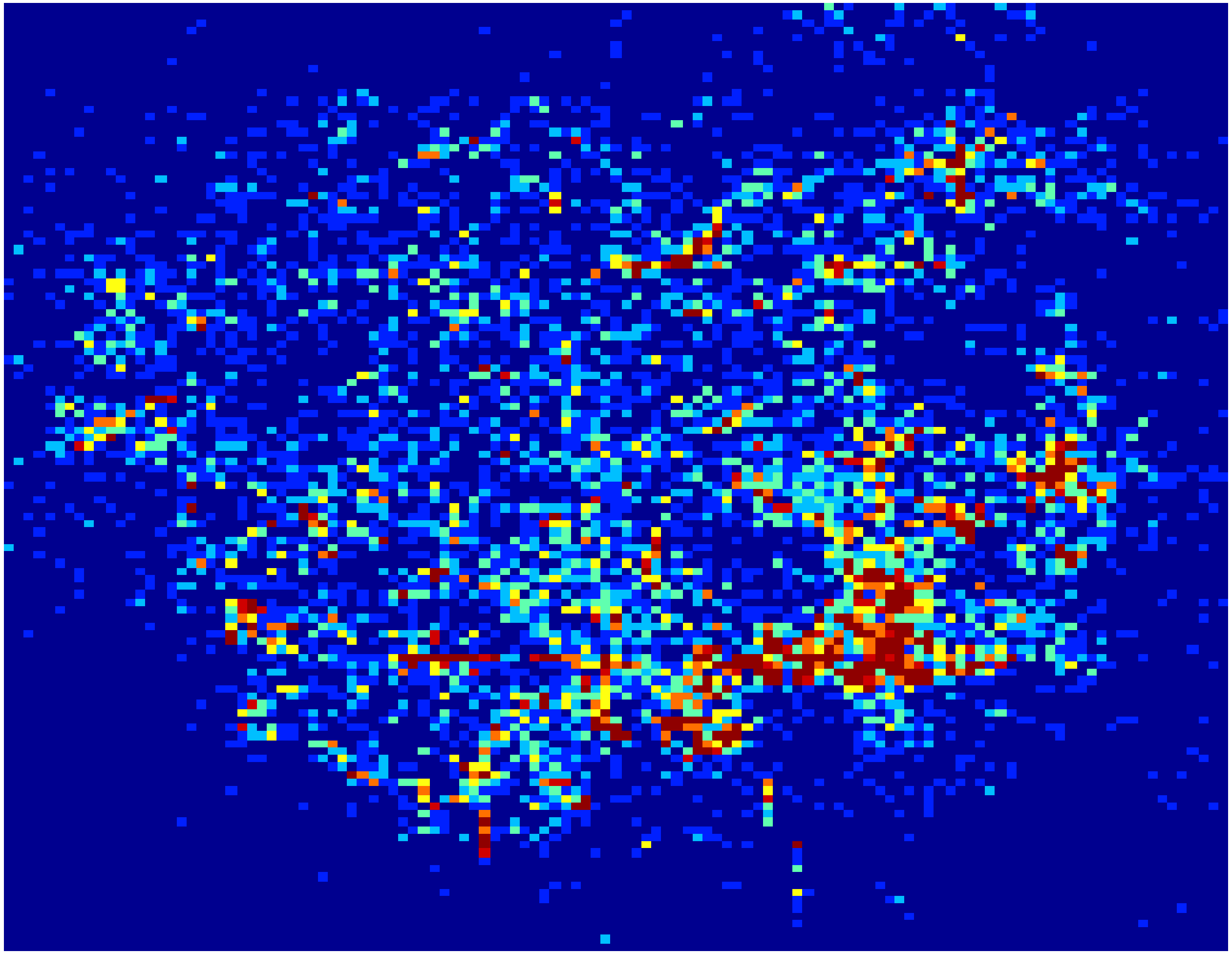
Local, adaptive renal function evaluation system based on DENFIS:

(Marshal, Song, Ma, McDonell and Kasabov, Kidney International, May 2005)



- **New method:** Song, Q. , N. Kasabov, T. Ma, M. Marshall, *Integrating regression formulas and kernel functions into locally adaptive knowledge-based neural networks: a case study on renal function evaluation*, Artificial Intelligence in Medicine, 2006, Volume 36, pp 235-244

KEDRI-TELECOM Mobile Calls Traffic Optimization System



12. Future directions: Quantum inspired CI

- Quantum principles: superposition; entangelment, interference, parallelism

- QI methods for EIS:
  - QI clustering
  - Quantum neuron with a recurrent connection (Li et al, 2006): the output and the input variables are represented as particle waves
  - QI neural networks (Ezov and Ventura, 2000)
  - QI associative memories (e.g. Ventura and Martinez, $m=O(2^n)$ patterns stored in $2n+1$ neurons , while in a Hopfield NN it is $m<0.5n$ patterns in n neurons
  - QI fuzzy systems
  - QI GA
  - QI swarm intelligence

- Applications:
  - Specific algorithms with polynomial time complexity for NP-complete problems (e.g. factorising large numbers, Shor, 1997)
  - Search algorithms (Grover, 1996), $O(N^{1/2})$ vs $O(N)$ complexity)
  - Memorizing large number of patterns

Example: Quantum inspired evolutionary algorithms (QEA)

- The representation of individuals is usually done in the form of bit-strings, real-valued vectors, symbols etc. QEA uses a q-bit representation based on the concept of q-bits in Quantum Computing. Each q-bit is defined as a pair of numbers (α, β). A Q-bit individual as a string of m q-bits is represented as:

$$\left[\begin{array}{c|c|c|c} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{array}\right]$$

for i = 1, 2, …, m:

$$|\alpha_i|^2 + |\beta_i|^2 = 1$$

- Evolutionary computing with Q-bit representation has a better characteristic of population diversity than other representations, since it can represent linear superposition of states probabilistically .

- Here, only one Q-bit individual with m q-bits is enough to represent 2^m states whereas in binary representation, 2^m individuals will be required for the same.
- The Q-bit representation leads to quantum parallelism in the system as it is able to evaluate the function on a superposition of possible inputs. The output obtained is also in the form of superposition which needs to be collapsed to get the actual solution.

- In QEA, the population of Q-bit individuals at time t can be represented as:

$$Q(t) = \{q_1^t, q_2^t, \ldots, q_n^t\}$$

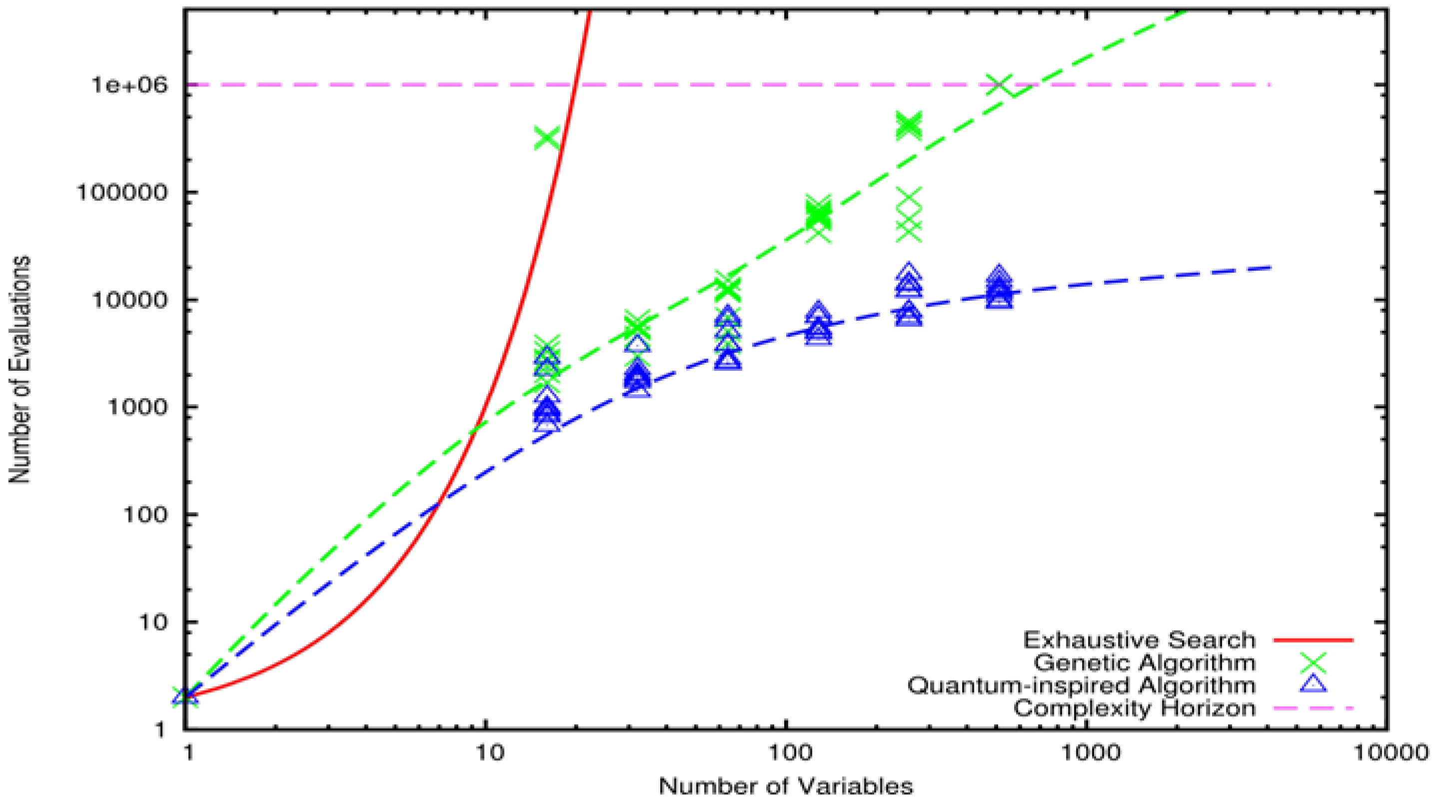
where n is the size of the population.

- The rotation gate, used as Q-gate is represented as:

$$U(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

Quantum Inspired Computational Intelligence

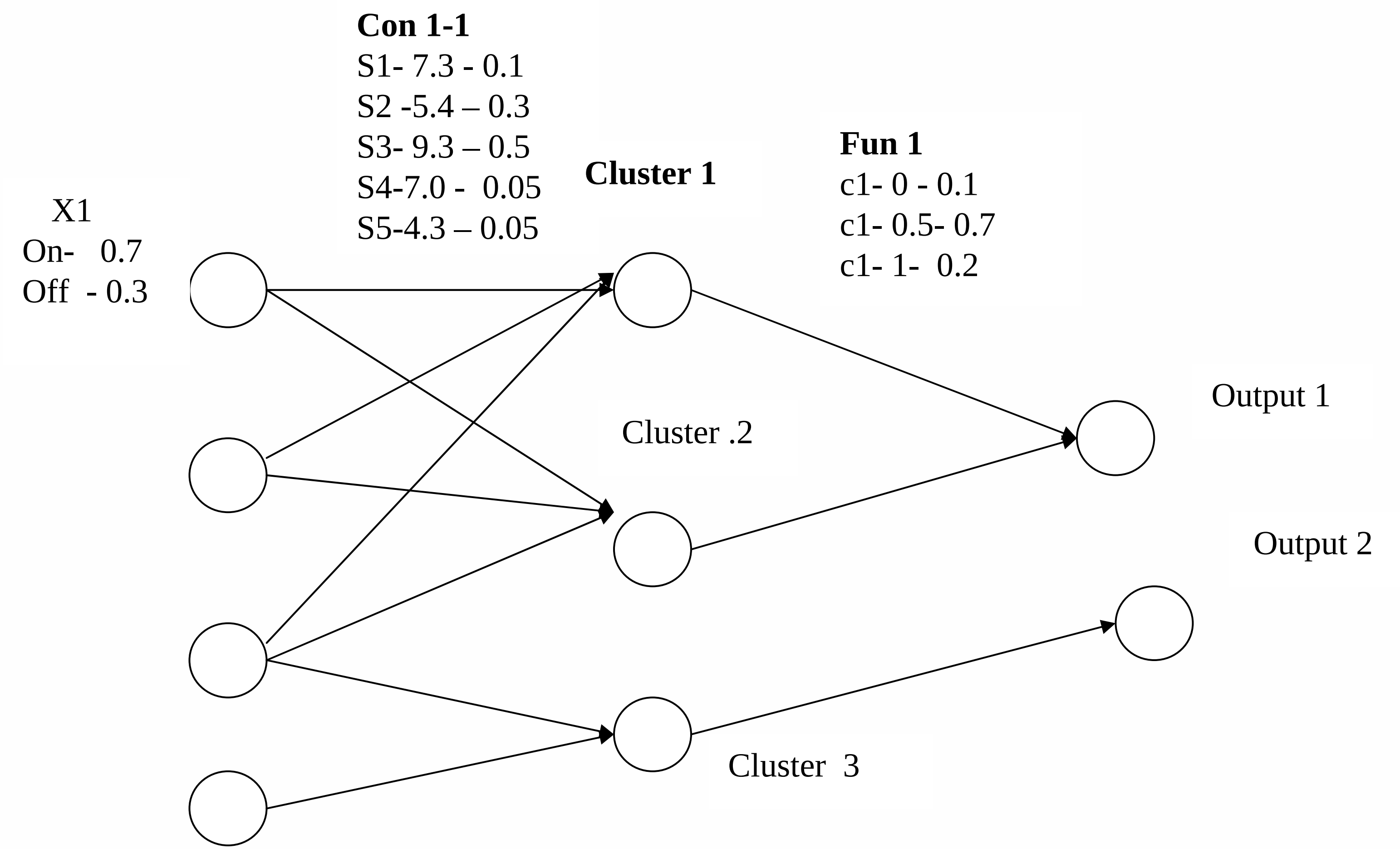
(M Defoin-Platel, S.Shliebs, et al)



The KEDRI quantum inspired evolutionary algorithm performs exponentially faster and more accurately than the classical algorithms when evaluating combinations of variables for a modelling task

nkasabov@aut.ac.nz

QI-ECOS
(Natural Computing, 2007)



13. Conclusions

- Brain-, gene- and quantum principles are useful for the creation of new types of EIS for :
  - Solving problems and making discoveries in bioinformatics, neuroinformatics, medicine, chemistry, physics
  - Solving hard AI and NP-complete problems
  - At the nano-level of microelectronic devices, quantum processes may have a significant impact.
- How much “inspiration”? –Depends on the problem in hand.
- Integrating different levels of information processing through general information theory - a challenge for information science
- New algorithms and models, e.g. quantum inspired CNGM
- Starting to use these models as a further inspiration for new computer devices – million times faster and more accurate
- Impact on the hardware – parallel, ubiquitous
- How do we implement the BGQI computational intelligence in order to benefit from their high speed and accuracy? Should we wait for the quantum computers to be realised many years from now, or we can implement them efficiently on specialised computing devices based on classical principles of physics?

KEDRI: The Knowledge Engineering and Discovery Research Institute

- Established June 2002
- Funded by AUT, NERF (FRST), NZ industry
- External funds approx NZ$3.8 mln.
- 6 senior research fellows and post-docs
- 20 PhD and Masters students;
- 25 associated researchers
- Both fundamental and applied research (theory + practice)
- 150 refereed publications
- 2 PCT patents
- Multicultural environment (9 ethnic origins)
- Strong national and international collaboration