Advanced Artificial Intelligence Technologies and Applications

Course organiser: A/Prof. Shihua Zhou

Course presenter

Prof Nikola Kasabov

Visiting Professor at Dalian University

Life FIEEE, FRSNZ, FINNS, DVF RAE UK

Founding Director KEDRI

Professor, Auckland University of Technology, NZ

George Moore Chair/Professor, Ulster University, UK

Honorary Professor, University of Auckland NZ , Peking University China

Visiting Professor IICT/Bulgarian Academy of Sciences and Teesside University UK

Doctor Honoris Causa Obuda University Budapest

Director, Knowledge Engineering Consulting Ltd (https://www.knowledgeengineering.ai)

Assistants

A/Prof. Wei Qi Yan

Director of the CeRV Center, AUT

Weiqi.yan@aut.ac.nz

https://academics.aut.ac.nz/weiqi.yan

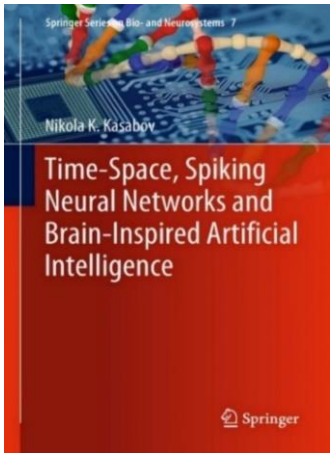
Ms Iman AbouHassan

iabouhassan@tu-sofia.bg

abouhassan.iman@gmail.com

Advanced Artificial Intelligence Technologies and Applications

1. AI and the evolution of its principles. Evolving processes in Time and Space (Ch1, 3-19)
2. From Data and Information to Knowledge. Fuzzy logic. (Ch1,19-33 + extra reading)
3. Artificial neural networks - fundamentals. (Ch2, 39-48). Computational modelling with NN. Tut1: NeuCom.
4. Deep neural networks (Ch.2, 48-50 + extra reading).
5. Evolving connectionist systems (ECOS) (Ch2, 52-78). Tutorial 2: ECOS in NeuCom.
6. Deep learning and deep knowledge representation in the human brain (Ch3)
7. Spiking neural networks (Ch4). Evolving spiking neural networks (Ch5)
8. Brain-inspired SNN. NeuCube. (Ch.6). Tutorial 3: NeuCube software (IA)
9. From von Neuman Machines to Neuromorphic Platforms (Ch20 , 22)
10. Other neurocomputers: Transformers.
11. Evolutionary and quantum inspired computation (Ch.7)
12. AI applications in health (Ch.8-11)
13. AI applications for audio-visual information (Ch.12,13)
14. AI for language modelling. ChatBots (extra reading)
15. AI for brain-computer interfaces (BCI) (Ch.14)
16. AI in bioinformatics and neuroinformatics (Ch15,16, 17,18)
17. AI applications for multisensory environmental data. AI in finance and economics (Ch19)



Course book: N.Kasabov, Time-Space, Spiking Neural Networks and Brain-Inspired Artificial Intelligence Springer, 2019, https://www.springer.com/gp/book/9783662577134

Additional materials: https://www.knowledgeengineering.ai/china

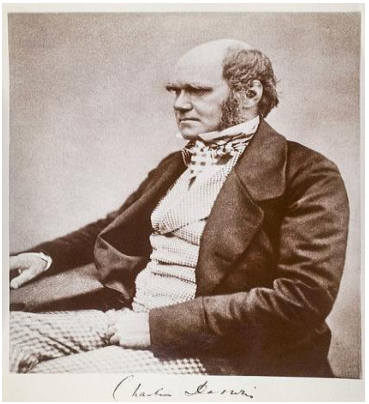
ZOOM link for all lectures: https://us05web.zoom.us/j/4658730662?pwd=eFN0eHRCN3o4K0FaZ0lqQmN1UUgydz09

Lecture 11

Evolutionary and quantum-inspired computation (Ch.7)

1. Evolutionary Computation (EC)
2. Application of EC for parameter optimisation in NeuCom and NeuCube
3. Quantum information principles
4. Quantum-inspired evolutionary algorithms
5. Applications of QiEA for parameter optimisation of spiking neural networks
6. Questions

1. Evolutionary computation



Charles Darwin (1809-1882)

Evolutionary computation: Learning through evolution

- Species learn to adapt through genetic evolution (e.g. crossover and mutation of genes) in populations over generations.
- Genes are carrier of information: stability vs plasticity
- A set of chromosomes define an individual
- Survival of the fittest individuals within a population
- Evolutionary computation (EC) as part of AI is population/generation based optimisation method.

EC can be used to optimise parameters (genes) of learning systems.

GA: Generating populations of individuals, selecting the best ones to go to the next generation etc.

1. Initialize population of possible solutions

2. WHILE a criterion for termination is not reached DO

{

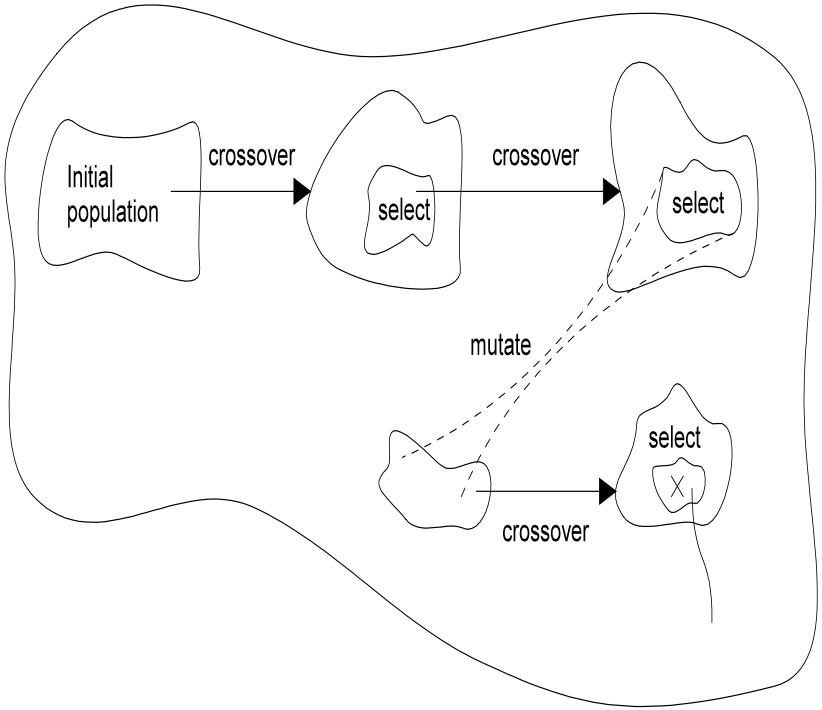
2a. Crossover two specimens ("mother and father") and generate new individuals;

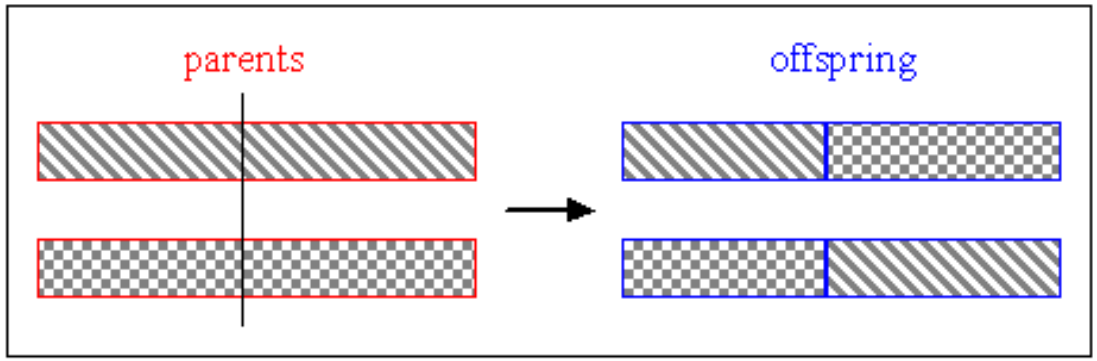
2b. Select the most promising ones, according to a fitness function;

2c. Development (if at all);

2d. Possible mutation (rare) }

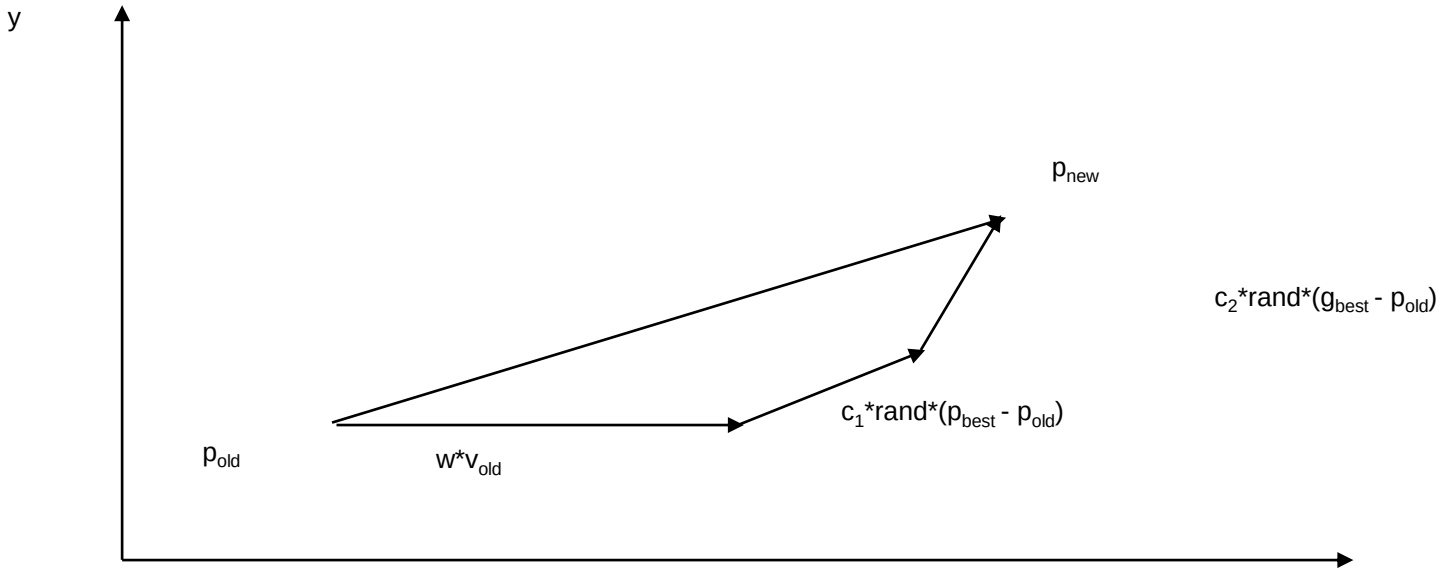
}





Particle swarm optimisation (PSO)

(Kennedy and Eberhard, 1995)



A particle (e.g. chromosome) is a solution (e.g. a set of parameter values of a model)

A particle is evaluated as its local best fitness (pbest) and the whole set of particle is evaluated as global best (gbest)

A particle's velocity and position is updated at every iteration (generation) based on the local and the global best values.

This is typical for swarms of individuals, each of them also following the best performing one at a time moment t.

PSO have been developed for continuous, discrete and binary problems.

The representation of the individuals varies for the different problems.

Binary Particle Swarm Optimization (BPSO) uses a vector of binary digits representation for the positions of the particles.

Box 2. A pseudo code of a PSO algorithm

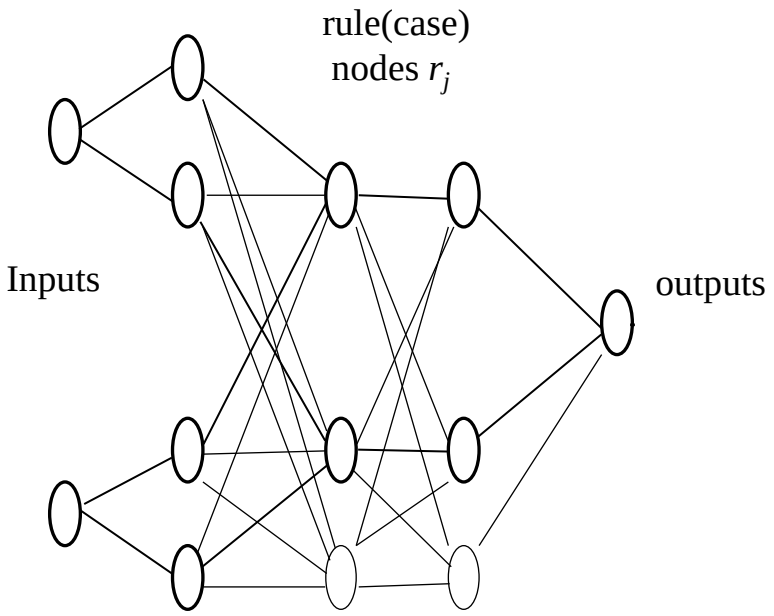
---

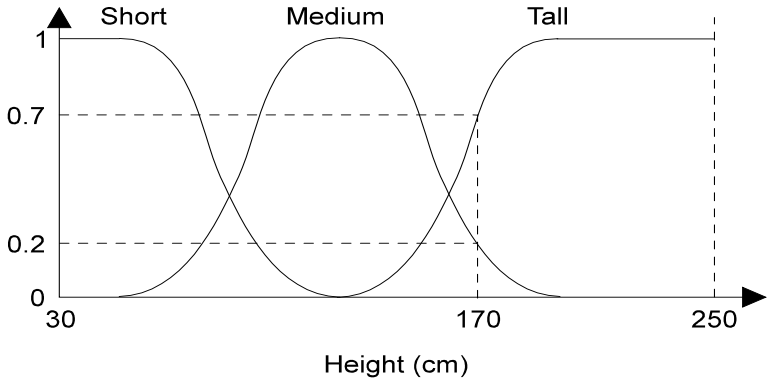
begin
t ← 0 (time variable)
1) Initialize a population with random positions and velocities
2) Evaluate the fitness based on objective function
3) Select the local *pbest* and the global *gbest*
while (termination condition is not met) do
begin
t ← t+1
4) Compute velocity and position updates of each particle
5) Determine the new fitness
6) Update the *pbest* and *gbest* if required
end
end

2. Application of EC for parameter optimisation in NeuCom and NeuCube

Optimisation of the parameters of Evolving Classification Function (ECF) model in NeuCom using GA

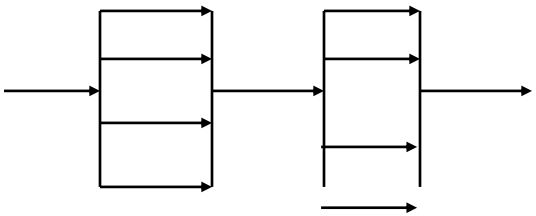
- Hidden nodes evolve, starting from no nodes at all.
- Each hidden node is a cluster center.
- Clusters grow in radius and shrink through a learning algorithm
- Each hidden node represents a local model (a rule) that associates an input cluster with an output function, e.g. a constant label, a linear 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.
- As a general case input and/or output variables can be fuzzy or non-fuzzy (crisp)
- ECF – evolving classifier function – no output MF, only input MF.
- 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, …
- Input features to optimise use
- (N. Kasabov, IEEE Tr SMC, 2001)

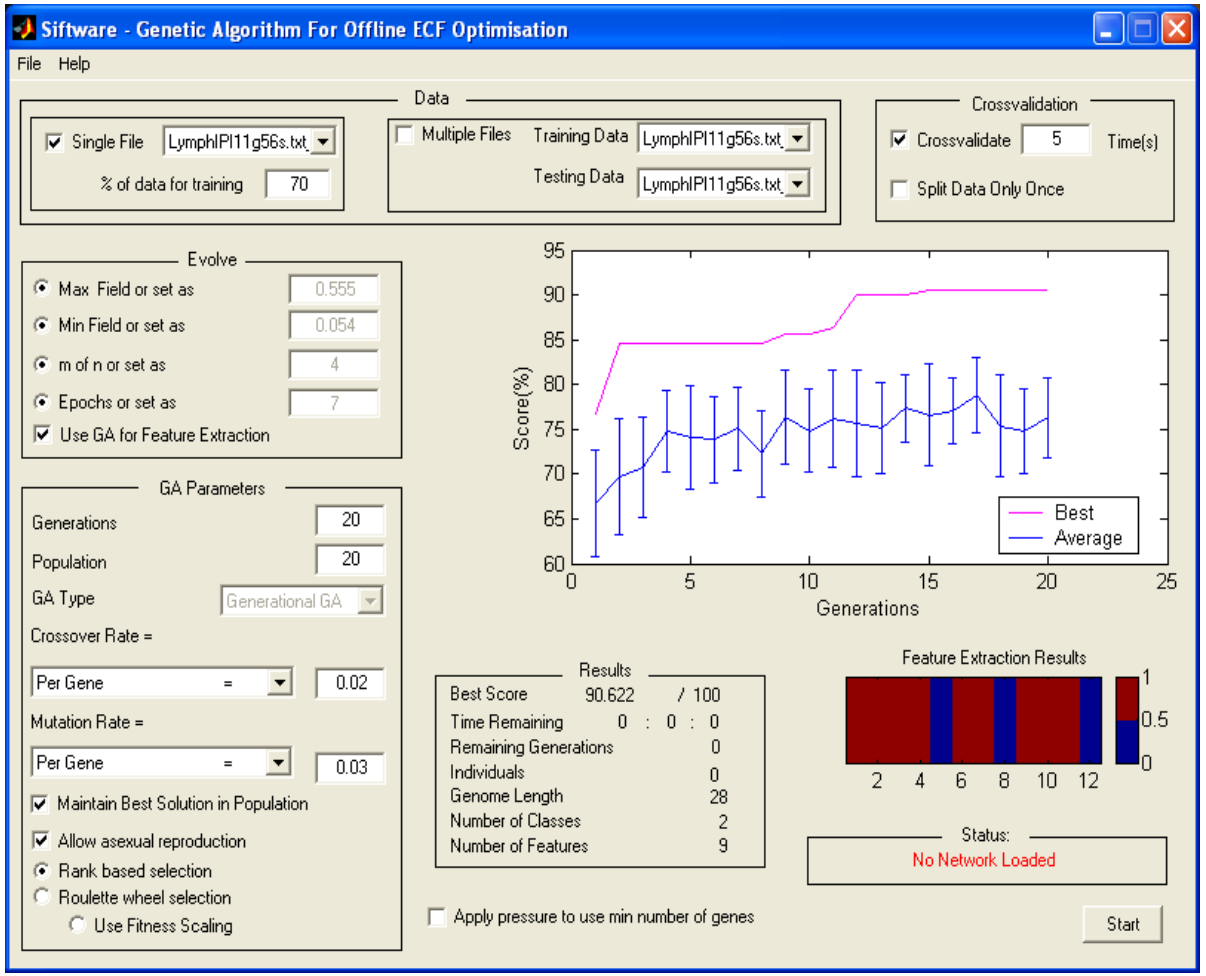




GA feature and parameter optimisation of ECF models in NeuCom

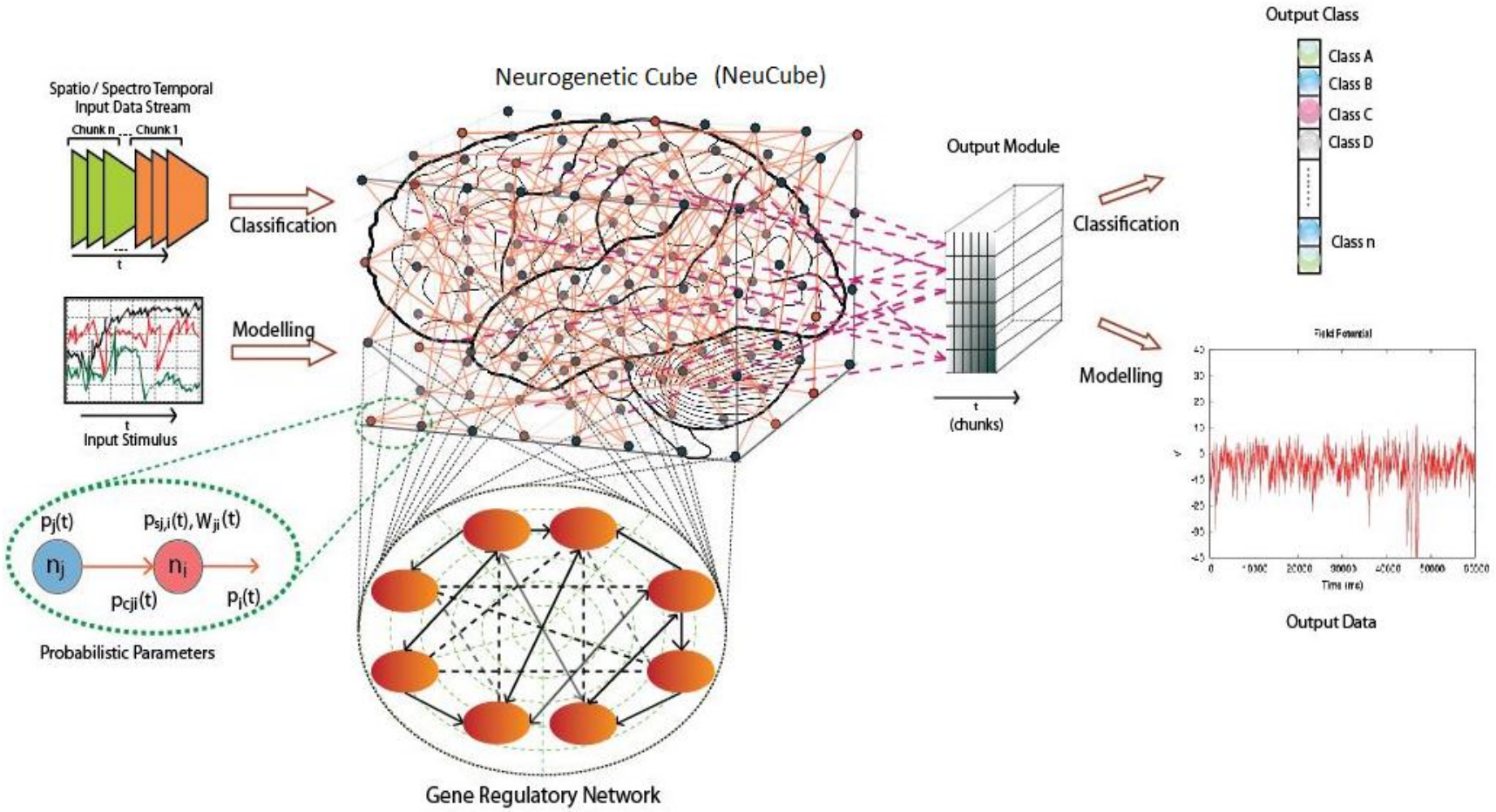
- Optimizing the parameters of the model and the input features
- A chromosome contains as “genes” all model parameters and input features
- Replication of individual ECF models and selection of:
  - The best one
  - The best *m* averaged, etc





NeuCube SNN model parameter optimization

N.Kasabov, NeuCube: A Spiking Neural Network Architecture for Mapping, Learning and Understanding of Spatio-Temporal Brain Data, Neural Networks, vol.52, 2014.



NeuCube SNN model parameter optimization using GA

ParameterOptimizationPanel

Options

Optimization Tool: Exhausted grid search | Cross Validation Number: 2

Optimization Parameters

		Minimum	Step number	Maximum
☑	AER Threshold	0.1	5	3
☐	Small world radius	1.5	5	6
☐	STDP rate	0.01	5	0.1
☑	Threshold of firing	0.2	5	0.8
☐	Refactory time	2	9	10
☑	Training time	1	9	9
☐	Mod	0.1	8	0.5
☐	Drift	0.1	8	0.5

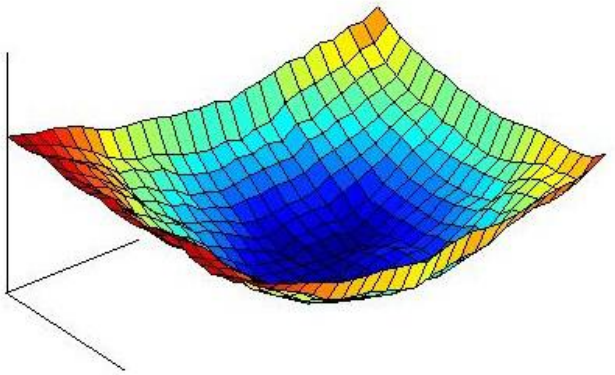
GA Parameters

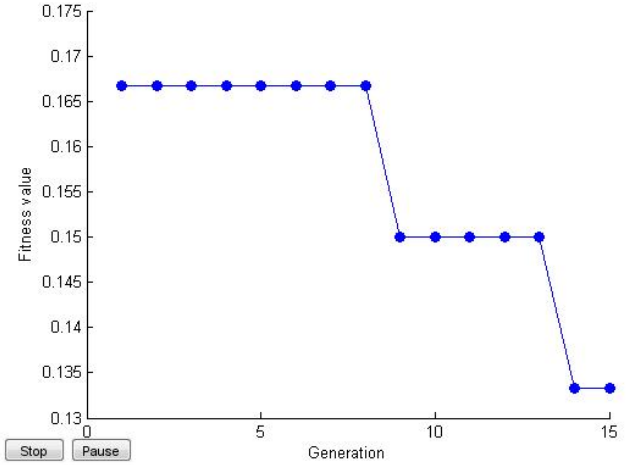
Crossover Function: Scattered | Selection Function: Stochastic Unifo...

Population Size: 20 | Crossover Fraction: 0.2

Generation number: 15 | Elite Count: 2

Start | Cancel

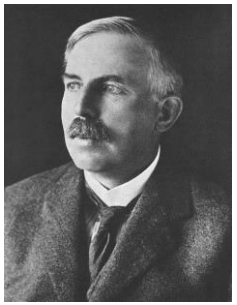




nkasabov@aut.ac.nz

3. Quantum information principles (section 7.2)

Quantum information principles: superposition; entanglement, interference, parallelism (M.Planck, A.Einstein, Niels Bohr, W.Heisenberg, John von Neumann, **E. Rutherford**)



Ernest Rutherford (1871-1937)

- *Quantum bit (qu-bit) – probabilistic superposition of 1 and 0 at time* t

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad |\alpha|^2 + |\beta|^2 = 1$$

- *Quantum vector (qu-vector) – quantum chromosome*

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{bmatrix}$$

- *A measured qu-bit (collapsed) has crisp values of 1 or 0 depending on the highest probability*

- *Define the fitness of the crisp chromosome according to an objective function (e.g. accuracy of classification) and the*

- *Quantum gates – how to define the qu-bit values for the next generation*

$$\begin{bmatrix} \alpha_i^j(t+1) \\ \beta_i^j(t+1) \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \begin{bmatrix} \alpha_i^j(t) \\ \beta_i^j(t) \end{bmatrix}$$

- **Applications:**
  - Specific algorithms with polynomial time complexity for NP-complete problems (e.g. factorising large numbers, Shor, 1997; cryptography)
  - Search algorithms (Grover, 1996), $O(N^{1/2})$ vs O(N) complexity)
  - Quantum associative memories
  - Quantum-inspired EC/EA

More about quantum information processing (superposion and entanglement)

- ***Supersposition:*** A quantum bit (*qubit)* may be in a state ‘1’or ‘0’, but also in a *superposition* of both to different probabilities.

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where α and β are complex numbers that are used to define the *probability* of which of the corresponding states is likely to appear when a *qubit* is read (collapsed).

- Normalization of the states to unity guarantees: $|\alpha|^2 + |\beta|^2 = 1$
- When we measure the state of a qbit (*collapse* the superpostion), then it is in a fixed state.
- *Examples: 2 ordinary bits represent at each time only one of 4 numbers, while a 2 Qbit represents all fours numbers at any time with their allocated probabilities. One vector of 1,000 ordinary bits can represent at any time one of the 2 to the power of 1000 states, while a 1000 qbit represents* ***all*** *these states at any time moment.*

- A popular example is the famous analogy of Schrodinger's Cat. First, we have a living cat and place it in a lead box. At this stage, there is no question that the cat is alive. Then throw in a capsule of cyanide and seal the box. We do not know if the cat is alive or if it has broken the cyanide capsule and died. Since we do not know, the cat is both alive and dead, according to quantum law -- in a superposition of states. It is only when we break open the box and see what condition the cat is in that the superposition is lost, and the cat must be either alive or dead (from WhatIs.com).

- ***Entanglement*** - two or more particles, regardless of their location, can be viewed as “correlated”, undistinguishable, “synchronized”, coherent. This is the basic principle used in the so called “teleportation” in space and time.

4. Quantum inspired Evolutionary Algorithms (QiEA)

- QEA (e.g, QiGA) use a q-bit representation of a chromosome of n "genes" at a time t:

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

- Each q-bit is defined as a pair of numbers (α, β) – probability density amplitude.
- A n element q-bit vector can represent probabilistically 2^n states at any time

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

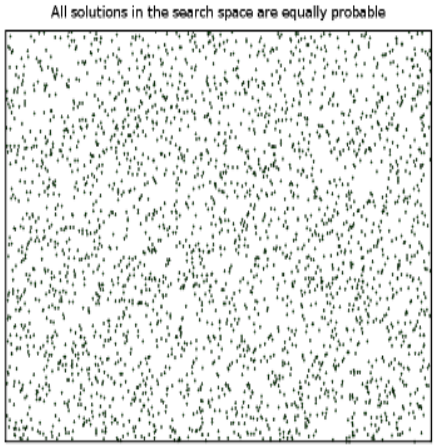
- The output is obtained after the q-bit vector is collapsed into a single state
- Changing probability density with quantum gates, e.g. rotation gate:

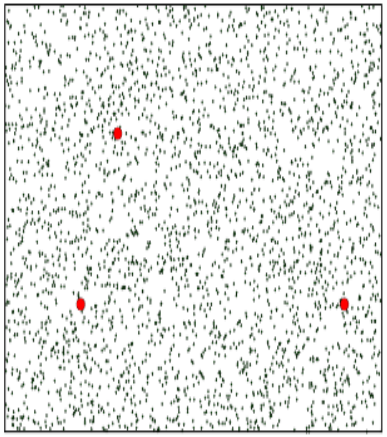
$$U(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

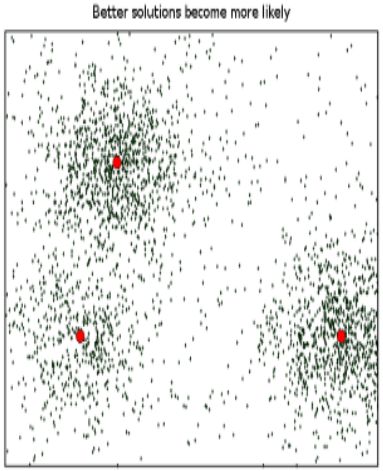
- Evolutionary computing with q-bit representation has a better characteristic of population diversity than other representations, since it can represent linear superposition of states probabilistically .

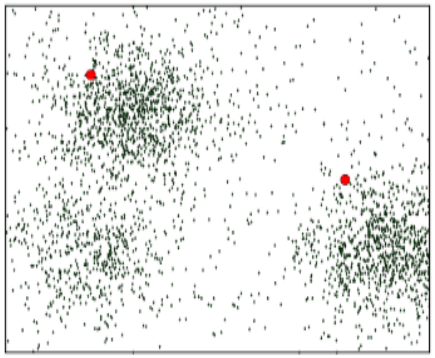
Parameter optimisation in a high dimensional feature (variable) space – every point denotes a model with different parameter values (“genes”). The problem is to select the best model according to a given optimisation criteria (fitness function) for a minimum iterations (generations)

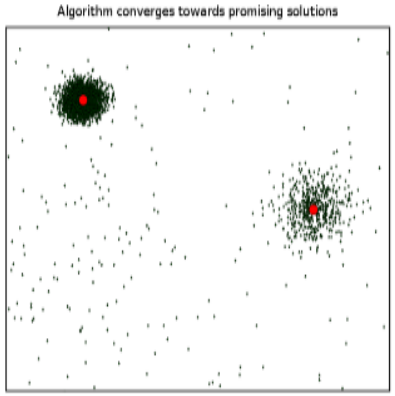
Local and global optimum points

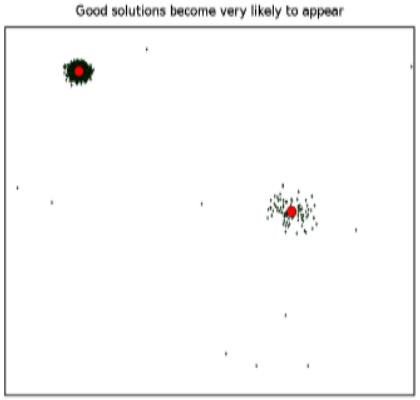




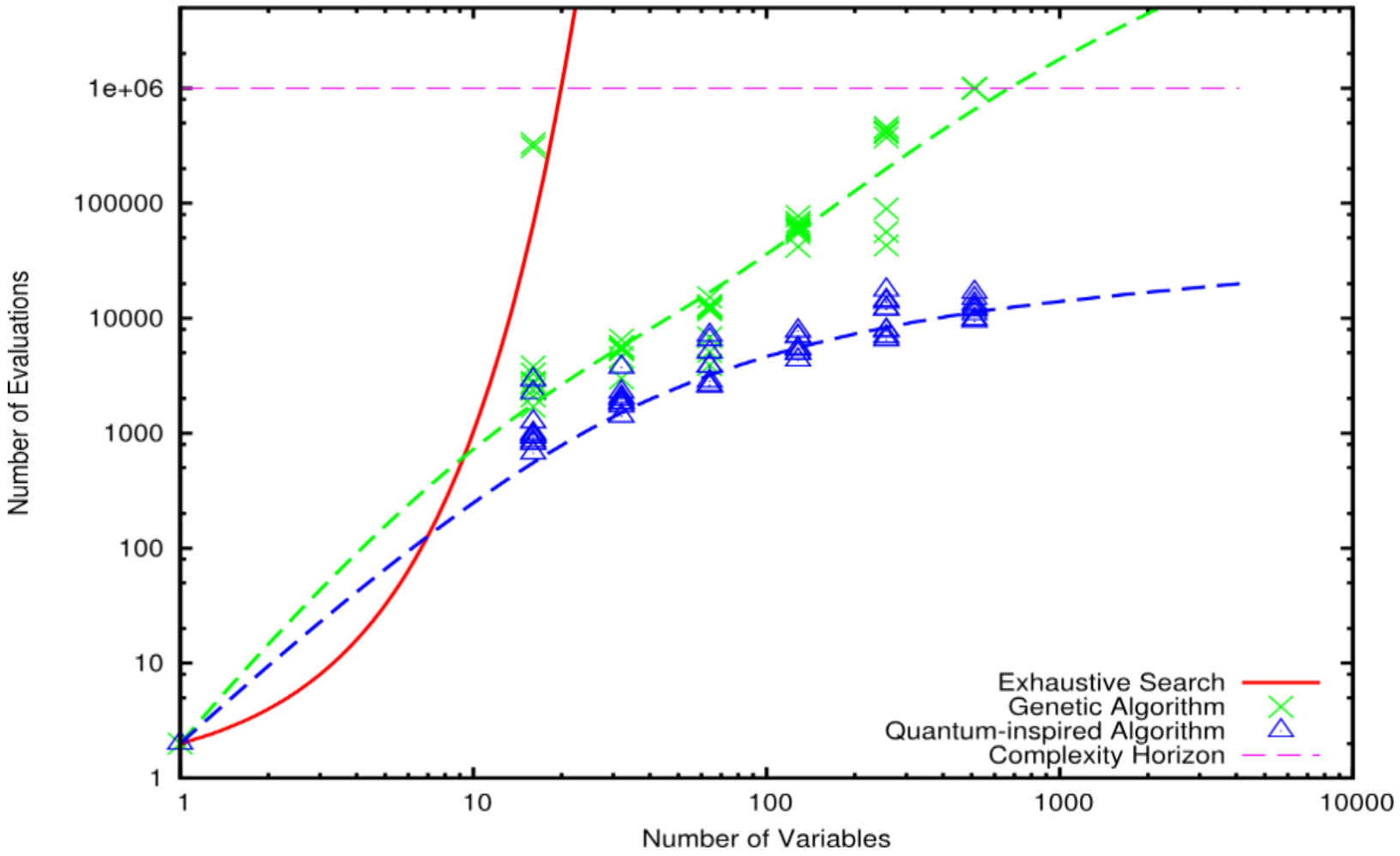






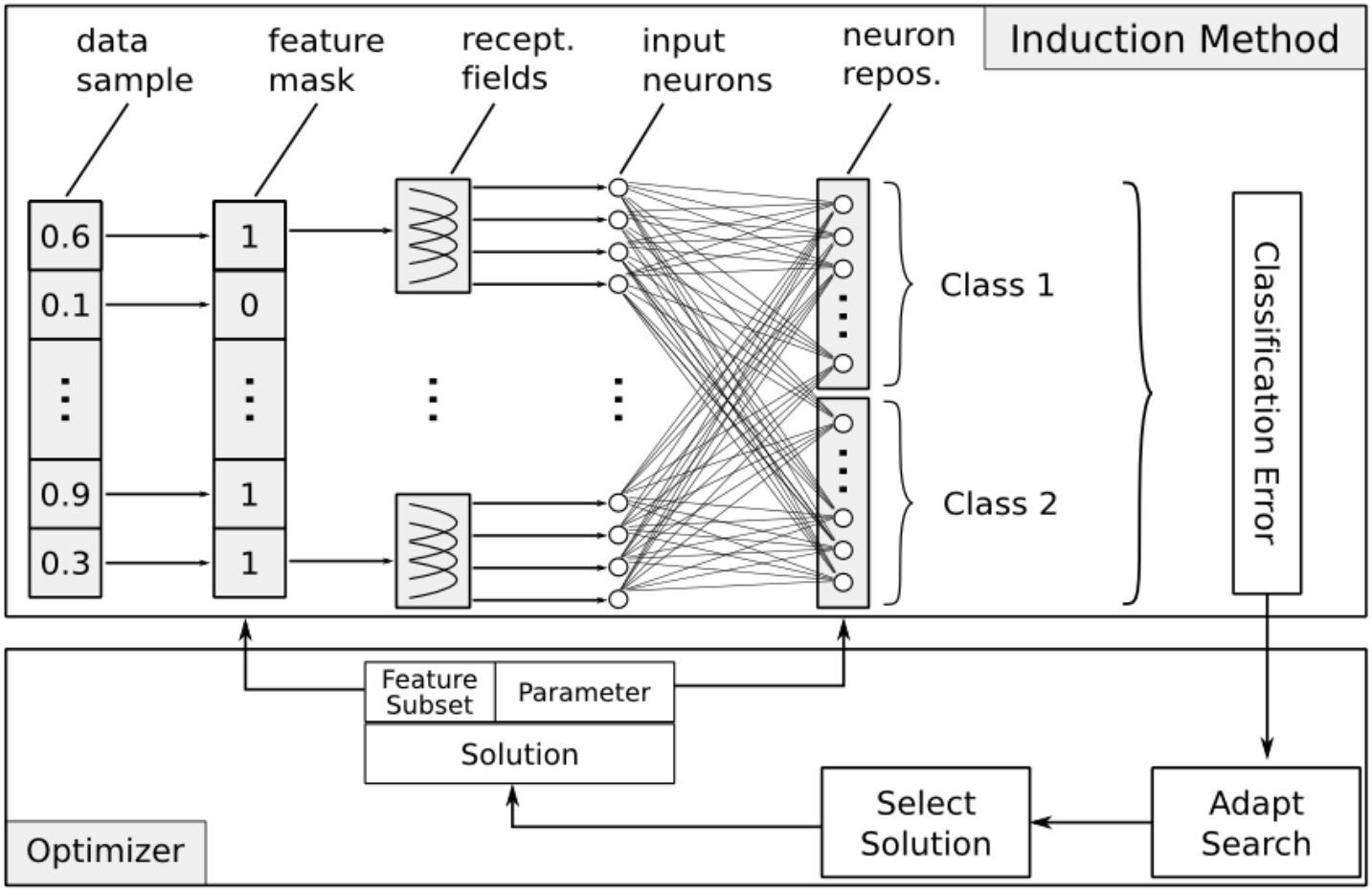


Comparison between different algorithms in terms of the number of iterations to optimise the values of variables (parameters) of a benchmark function

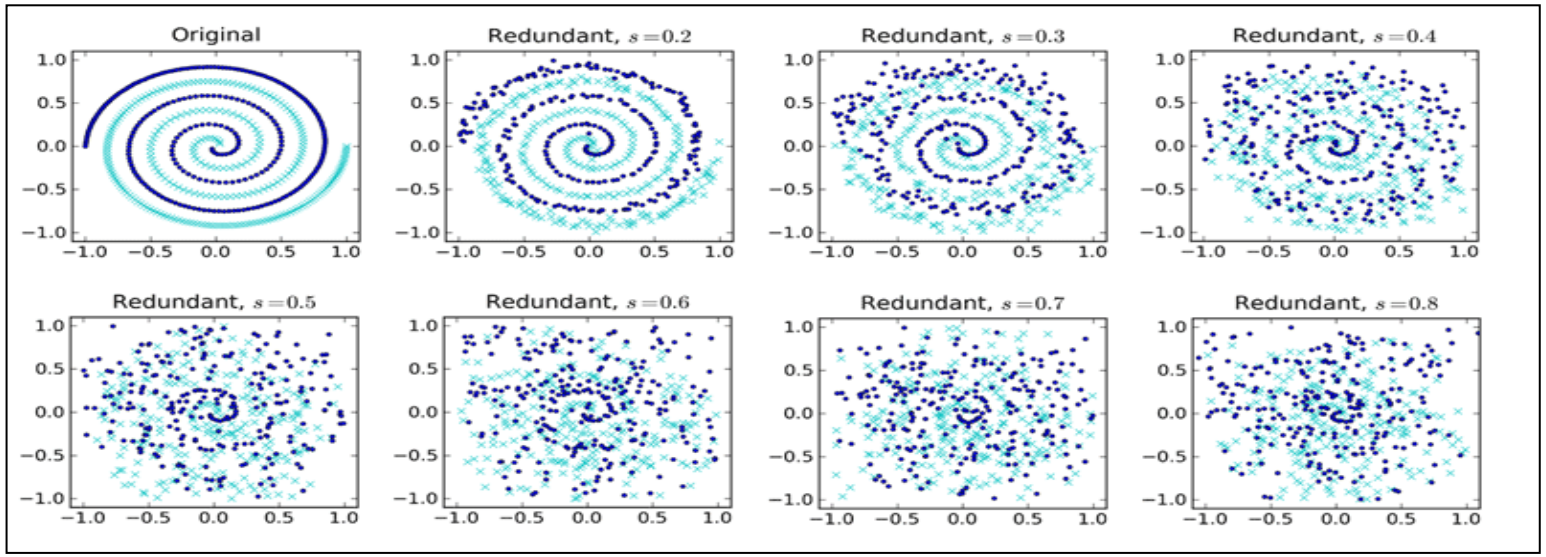


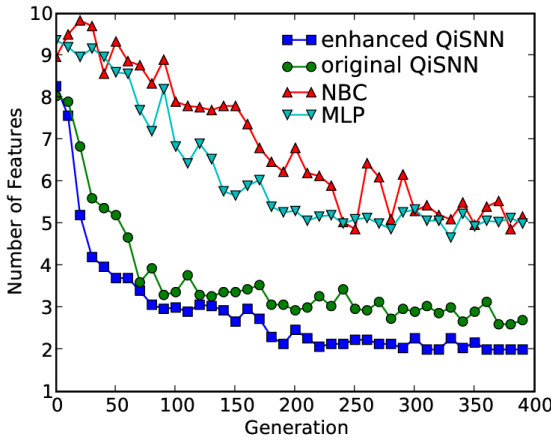
5. Quantum-inspired optimisation of eSNN

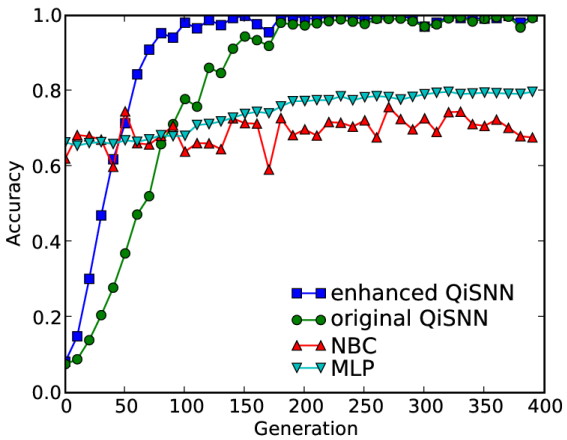
(Kasabov, 2007-2008; S.Schliebs, M.Defoin-Platel and N.Kasabov, 2008; Haza Nuzly, 2010))

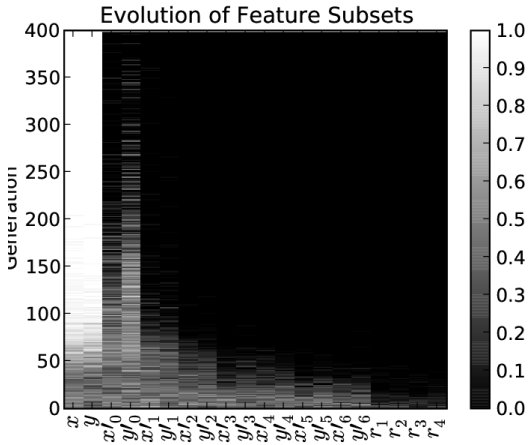


Benchmark Analysis – The two spiral problem classification





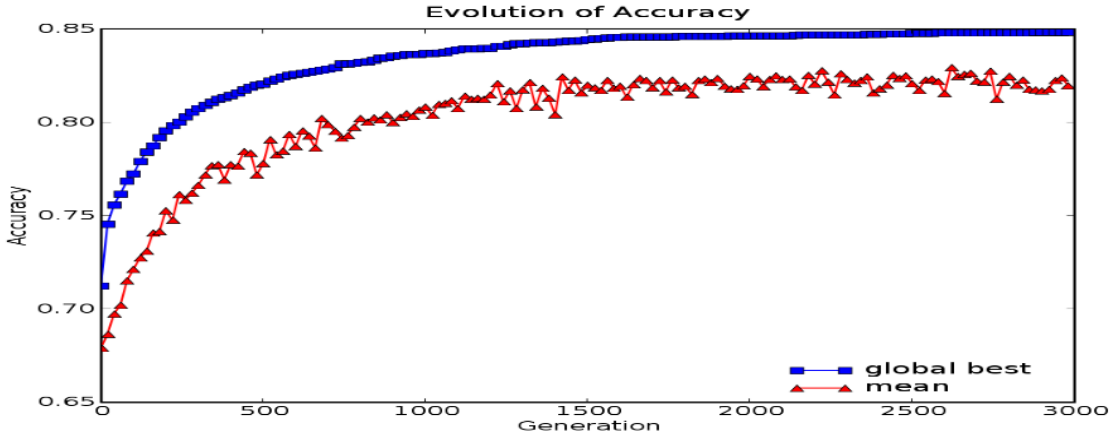


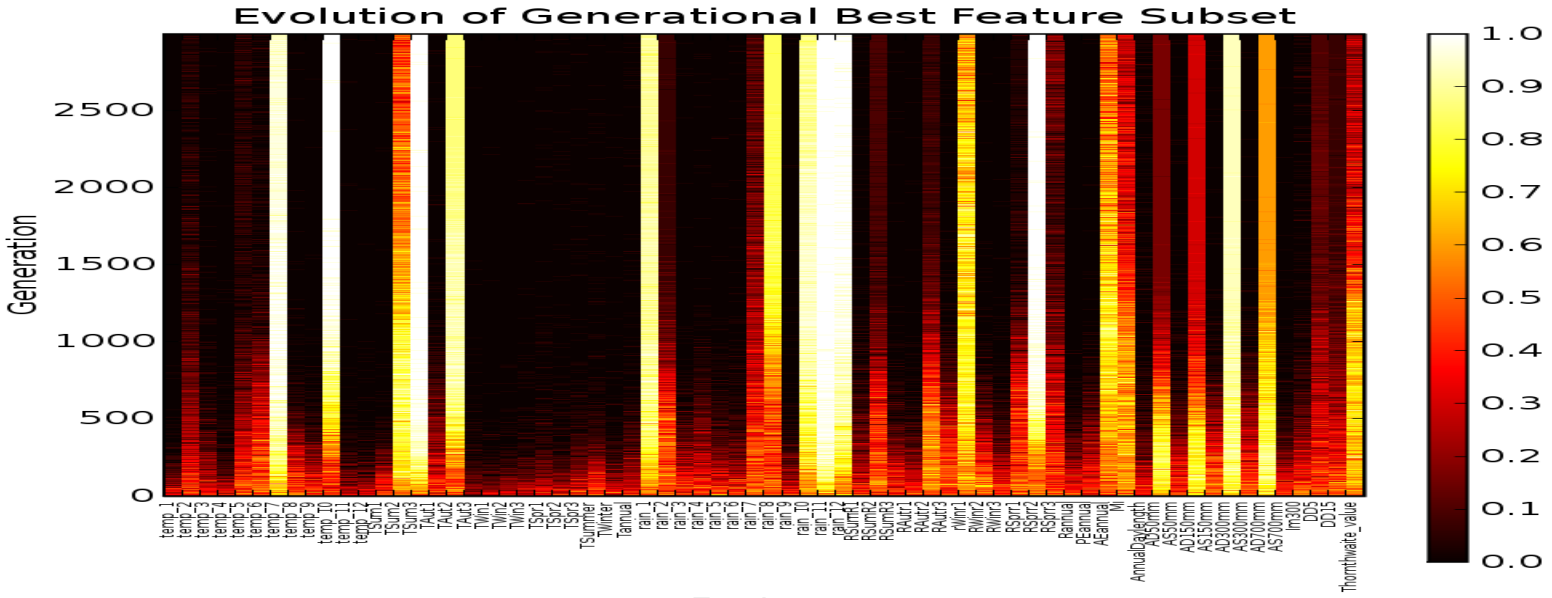


nkasabov@aut.ac.nz, sschlieb@aut.ac.nz

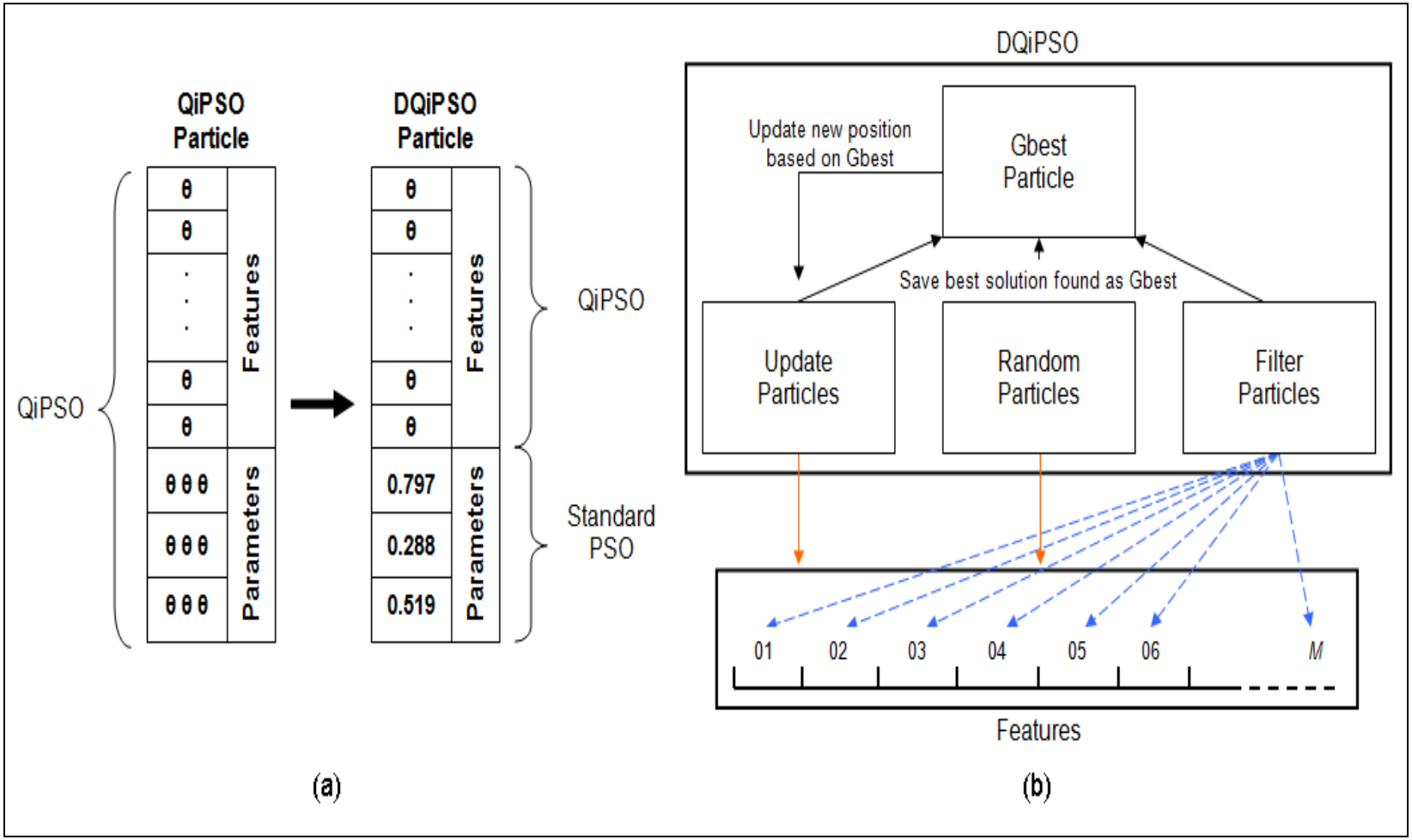
QiEA for the optimisation of features and parameters of a SNN for environmental prediction (section 7.3.2)

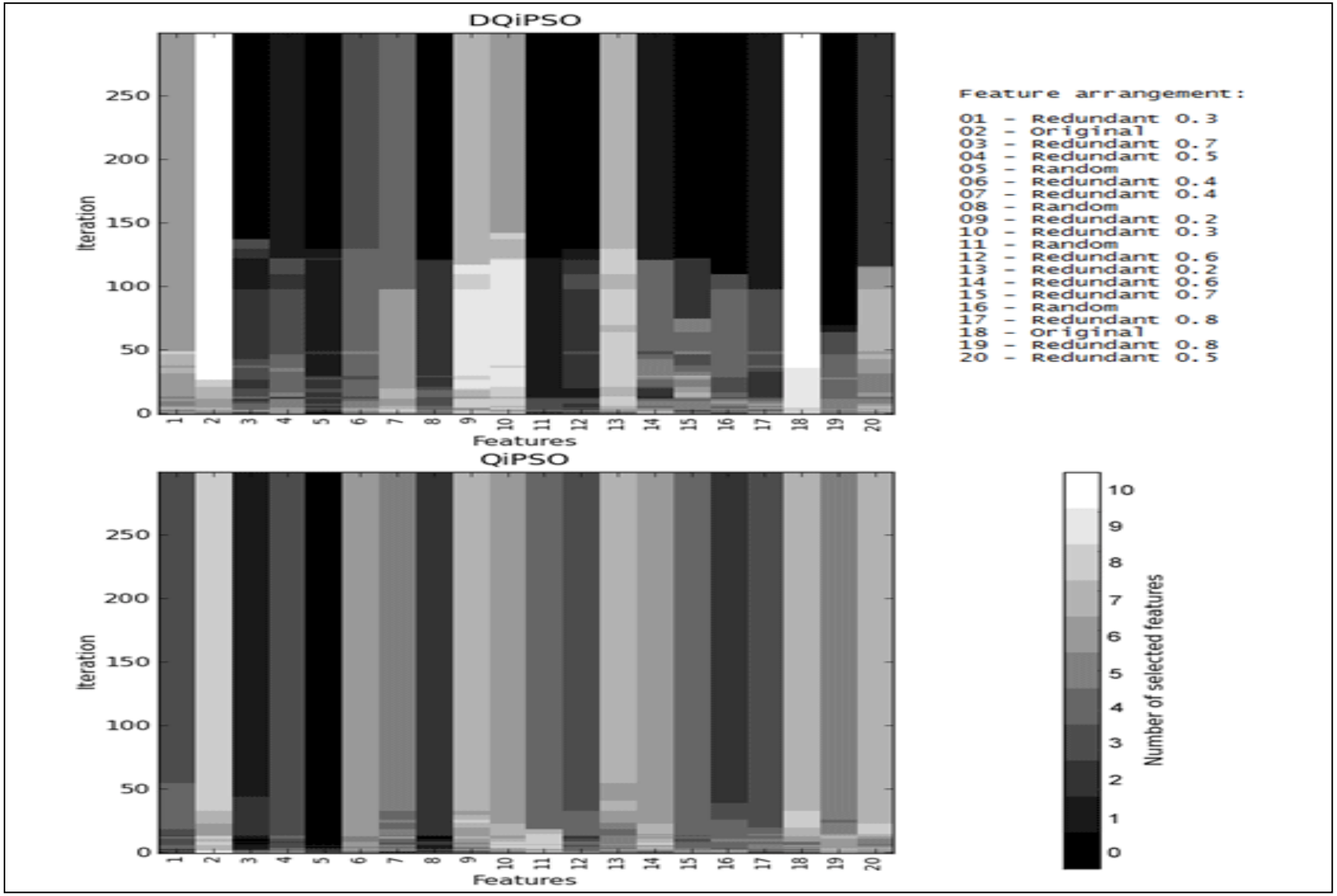
S. Schliebs, M. Defoin-Platel, S. Worner and N. Kasabov (2009), Quantum-inspired Feature and Parameter Optimization of Evolving Spiking Neural Networks with a Case Study from Ecological Modeling. In Proc. of the IJCNN 2009, Atlanta, 11-19 June: IEEE Press, 2009.





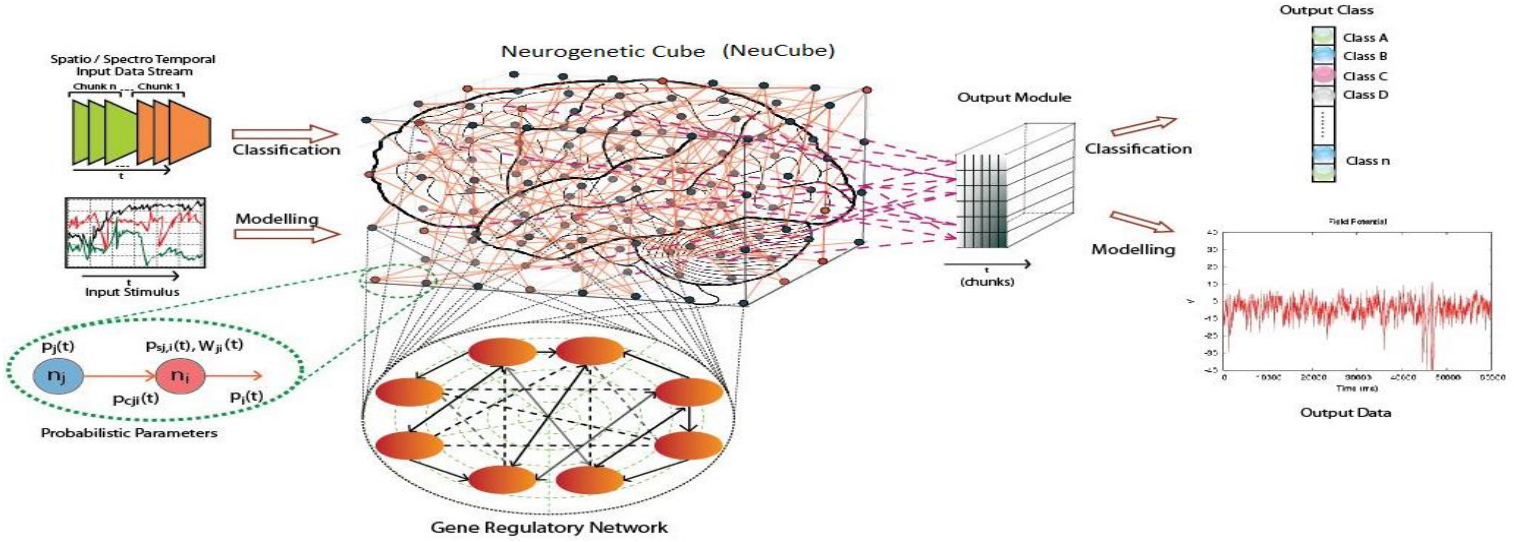
QiPSO (section 7.4)

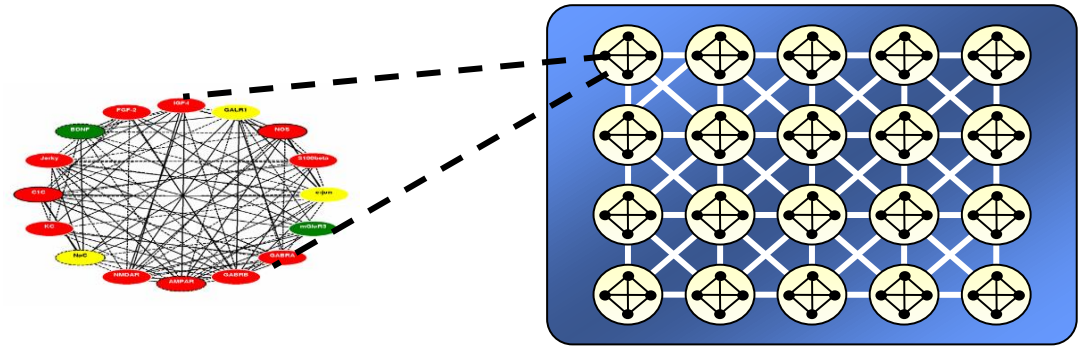



DQiPSO
Iteration
Features
Feature arrangement:
01 - Redundant 0.3
02 - Original
03 - Redundant 0.7
04 - Redundant 0.5
05 - Random
06 - Redundant 0.4
07 - Redundant 0.4
08 - Random
09 - Redundant 0.2
10 - Redundant 0.3
11 - Random
12 - Redundant 0.6
13 - Redundant 0.2
14 - Redundant 0.6
15 - Redundant 0.7
16 - Random
17 - Redundant 0.8
18 - Original
19 - Redundant 0.8
20 - Redundant 0.5
QiPSO
Iteration
Features
Number of selected features

Qi computational neurogenetic models (QiCNGM) (section 7.3.3)

N.Kasabov, NeuCube: A Spiking Neural Network Architecture for Mapping, Learning and Understanding of Spatio-Temporal Brain Data, Neural Networks, vol.52, 2014.





Quantum vector to optimise: Input features / SNN parameter/ Genes on/off and their connections

nkasabov@aut.ac.nz

6. Questions

1. What are EC algorithms?

2. How EC can be applied for parameter optimisation of neural networks (e.g. EFuNN from NeuCom: http://theneucom.com) ?

3. What are the main principles of quantum computing?

4. What are quantum -inspired evolutionary algorithms (QiEA)?

5. How QiEA can be applied for the optimisation of feature and parameters of spiking neural ntworks?