GENNET-Toolbox: An Evolving Genetic Algorithm for Neural Network Training

V. Gómez, E. Irigoyen, F. Artaza

Dep. of Systems Engineering and Automatic Control
Intelligent Control Research Group
University of the Basque Country (UPV/EHU)

Outline

- Introduction
- Goals
- GENNET Toolbox
- Tests: Training & Validation
- Conclusions
- Demo

Introduction

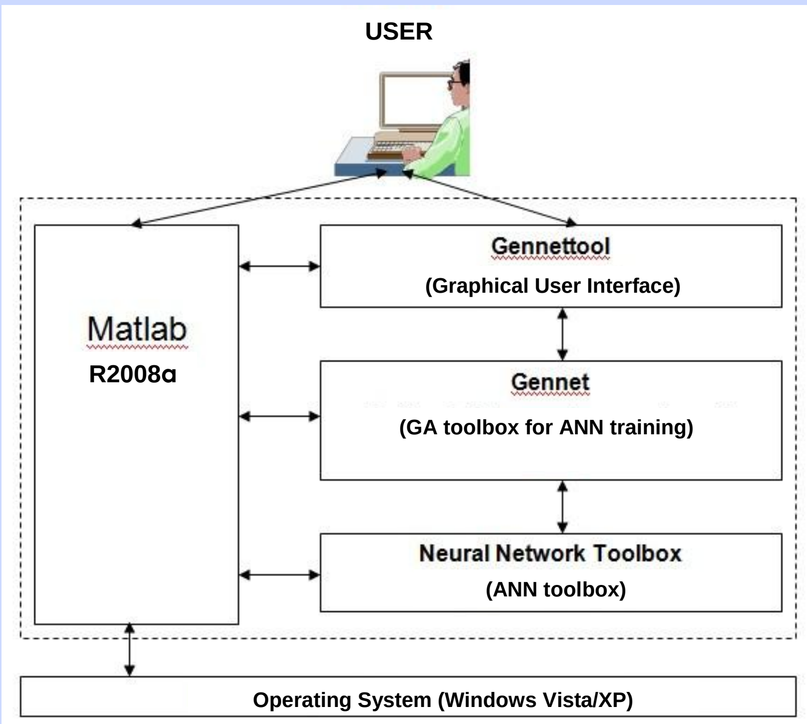
- **ANN: The design and training problem**
  - Selecting ANN architecture: Unknown optimal solvers in order to select an appropriate ANN architecture.
  - Classical training methods: Basically, based on gradient descent learning algorithms without ensuring satisfactory results.
- **GA: An important alternative**
  - High-ability in global searching.
  - High-complex problems: Both reaching good solutions and good analytical proposals.
- Back in 1989, the use of ANN with GA has been increasingly widespread.

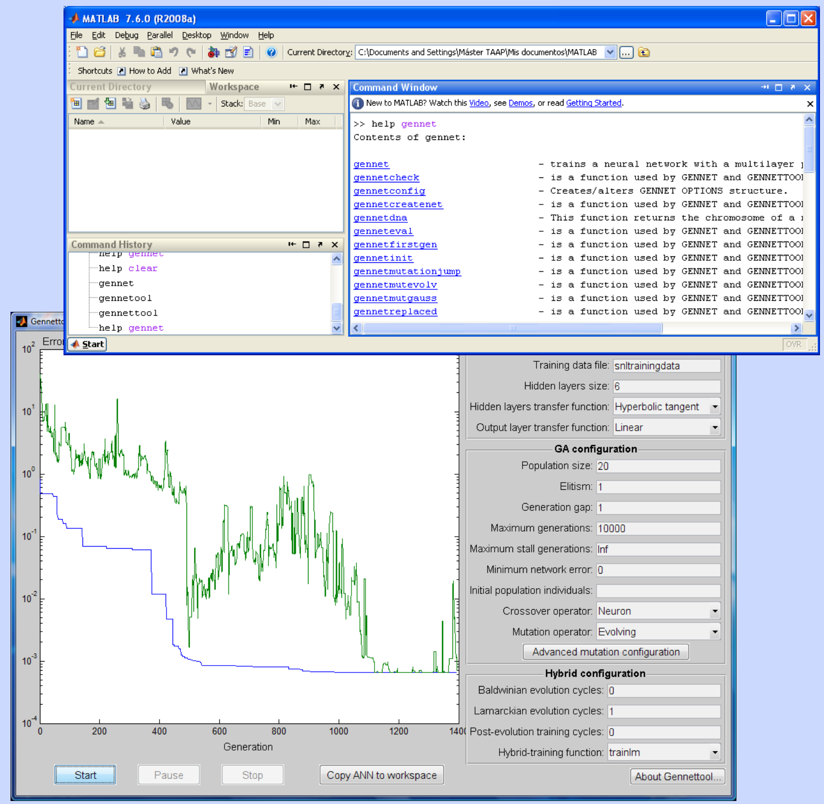
Goals

- **Main Goals:**
  - Developing of a tool based on GA for optimal ANN training.
  - Hybridization of GA and classical algorithms.
  - Using MATLAB as implementation platform.

- **Individual objectives:**
  - Designing of several evolutionary operators focused specifically on ANN training.
  - Testing the tool efficiency by matching against classical learning algorithms.

GENNET – General Scheme





GENNET - Toolbox

- To solve the **ANN training algorithm** problems by introducing a GA.
- This algorithm includes a **mutation operator** working on three different levels:
  - Network
  - Neuron
  - Parameter
- The mutation parameters are encoded and evolved **within each individual**.

GENNET - Special Features

- **Neural crossover operator**.
- **3 level mutation operators** (Network, Neuron, and Parameter):
  - **Gaussian** mutation with adaptive mutation rate
  - **Evolutionary** mutation

Artificial Neural Network Parameters	p_r	m_r	p_n	m_n	p_p	m_p

- **Hybridization**:
  - Lamarckian
  - Baldwinian
  - Post-evolutionary

Tests - Experimental Stage

- **3 representative problems** have been selected to test GENNET tool.
  - Breast cancer diagnostic
    - Diagnostic Wisconsin Breast Cancer Database
  - Reproducing a chaotic time series
    - Prediction of Mackey-Glass Chaotic Time Series
  - Identification of a non-linear system
    - A complex-dynamic system with unstable working points
- GENNET vs. classical algorithms
  - 6 different tool GENNET configurations
- Tests without time-consuming analysis in depth
  - Previous studies show similar computational costs

Tests - Experimental Configuration

- Comparison of 3 problems:
  - Several MLP topologies
  - 7 training different options
  - Similar computational costs
  - Each test is repeated 20 times in order to statistically validate the results

Computational comparison

Config.	Indiv eval	Individual hybrid training cycles	Post-evolution training cycles	Total Training cycles
C_1	***1***	1000	0	***1000***
C_2	***2020***	0	0	***0***
C_3	***1820***	0	100	***100***
C_4	***402***	2	0	***804***
C_5	***362***	2	100	***824***
C_6	***420***	2	0	***840***
C_7	***380***	2	100	***860***

Tests - Experimental Results

- Breast cancer diagnostic problem
  - Diagnostic Wisconsin Breast Cancer Database
    - http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

	Training results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	3,22E-13	0,154970	0,007894	0,034624
C_2	0,015769	0,060014	0,032403	0,011199
C_3	2,73E-13	0,020467	0,001315	0,004596
C_4	3,09E-11	0,011596	0,001452	0,003352
C_5	2,55E-12	0,008771	0,001169	0,002204
C_6	0,009715	0,025655	0,018356	0,004921
C_7	3,02E-13	0,005847	0,00043	0,001430

Tests - Experimental Results

- Breast cancer diagnostic problem
  - Diagnostic Wisconsin Breast Cancer Database
    - http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

	Validation results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	0,028303	0,180616	0,051577	0,030986
C_2	0,027942	0,090451	0,053064	0,016481
C_3	0,027776	0,063209	0,044163	0,00970
C_4	0,017266	0,056591	0,039754	0,009665
C_5	0,026038	0,07050	0,040807	0,008934
C_6	0,018252	0,054231	0,035087	0,007712
C_7	0,018635	0,056753	0,038487	0,01007

Tests - Experimental Results

- Mackey-Glass chaotic time series problem
  - Sexton R. S., et al, (1998) “Toward Global Optimization…”

$$Y_t = Y_{t-1} + 10.5 \cdot \left[\frac{0.2 \cdot Y_{t-5}}{1 + (Y_{t-5})^{10}} - 0.1 \cdot Y_{t-1}\right]$$

	Training results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	8,59E-06	0,016444	0,004006	0,006286
C_2	0,127702	0,213834	0,16220	0,02250
C_3	0,000209	0,053994	0,011800	0,011846
C_4	0,000323	0,106257	0,018166	0,026101
C_5	5,34E-05	0,023079	0,007885	0,008886
C_6	0,064247	0,118518	0,090686	0,015533
C_7	0,00014	0,018652	0,009747	0,00794

Tests - Experimental Results

- Mackey-Glass chaotic time series problem
  - Sexton R. S., et al, (1998) “Toward Global Optimization…”

$$Y_t = Y_{t\text{-}1} + 10.5 \cdot \left[\frac{0.2 \cdot Y_{t\text{-}5}}{1 + (Y_{t\text{-}5})^{10}} - 0.1 \cdot Y_{t\text{-}1}\right]$$

	Validation results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	4,19E-05	0,043466	0,009871	0,014766
C_2	0,144790	0,254668	0,188347	0,028568
C_3	0,00038	0,104987	0,023135	0,023750
C_4	0,000619	0,178600	0,027546	0,041748
C_5	1,71E-04	0,132440	0,01976	0,030431
C_6	0,072453	0,152860	0,109300	0,021833
C_7	0,000289	0,041877	0,020927	0,017320

Tests - Experimental Results

- Identification of a non-linear system problem
  - Irigoyen, E., et al, (2003) “A Neural Controller …”

$$y_{k+1} = K_1 \cdot u_k^{\ 3} + \frac{K_2 \cdot y_k - K_3 \cdot y_{k-1}}{1 + K_4 \cdot y_k^{\ 2}}$$

	Training results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	0,000555	0,131543	0,022941	0,037819
C_2	0,33939	0,650634	0,485201	0,091436
C_3	0,000245	0,070381	0,009421	0,016263
C_4	0,000403	0,159182	0,021962	0,037472
C_5	0,000603	0,11320	0,012188	0,025619
C_6	0,155258	0,328142	0,233706	0,040322
C_7	0,000314	0,040234	0,009663	0,01167

Tests - Experimental Results

- Identification of a non-linear system problem
  - Irigoyen, E., et al, (2003) "A Neural Controller …"

$$y_{k+1} = K_1 \cdot u_k^{\ 3} + \frac{K_2 \cdot y_k - K_3 \cdot y_{k-1}}{1 + K_4 \cdot y_k^{\ 2}}$$

	Validation results			
Conf.	**Best**	**Worst**	**η**	**σ**
C_1	0,001731	28,00865	1,832883	6,258940
C_2	0,482956	1,045188	0,716132	0,17110
C_3	0,001625	0,850996	0,070516	0,188787
C_4	0,001312	1,340799	0,144020	0,319937
C_5	0,001487	0,732906	0,089525	0,196451
C_6	0,336507	0,997356	0,582253	0,173005
C_7	0,001462	0,156441	0,043948	0,048899

Tests - Experimental Results

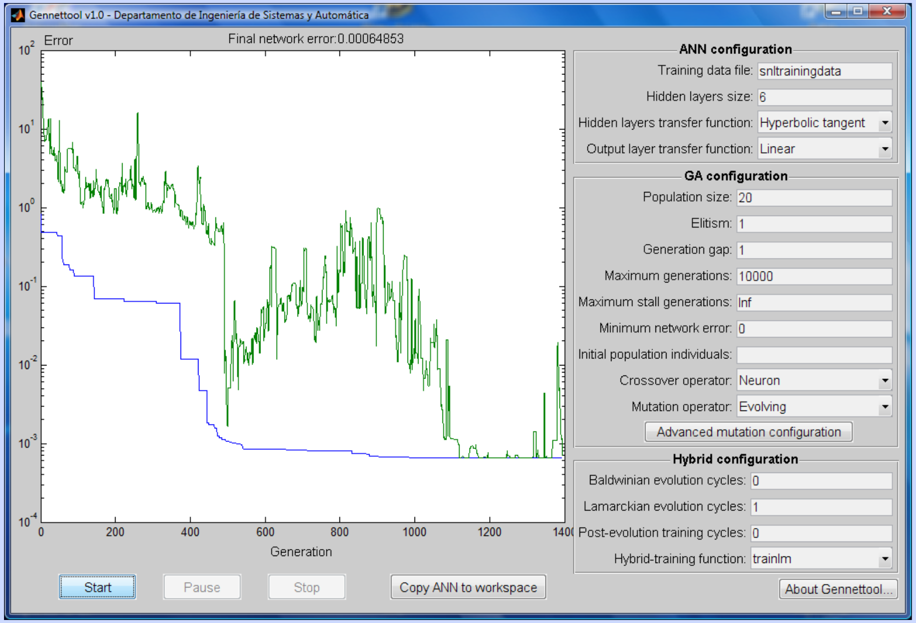
- Breast cancer diagnostic problem
  - Hybridization gives better performance than classical ones
  - Owing to the overtraining, results have not been enough relevant
- Mackey-Glass chaotic time series problem
  - Better results have been obtained with classical algorithms
    - The local minimal problem is not so relevant
  - Finding good results both with taking account computational cost and without time considerations
- Identification of a non-linear system problem
  - Higher performance of GENNET Hybridization vs. classical methods
  - GENNET avoids the local minima problem that classical algorithms suffer

Conclusions

- We have obtained a specific GA tool for ANN training.
- The designed tool has shown a high performance in very complex problems.
- The hybridized proposals have allowed to reach better solutions.

- The hybridization process has shown higher time consuming than the classical algorithms.
- The ANN overtraining problem has to be included into GENNET in order to improve stop criteria.

Demo

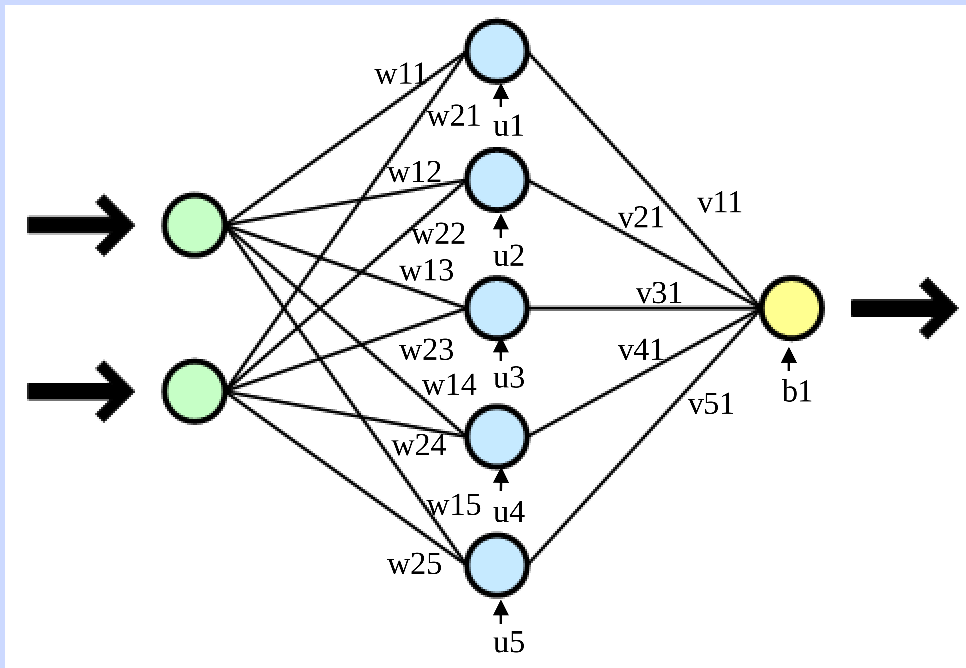

Genettool v1.0 - Departamento de Ingeniería de Sistemas y Automática
Error
Final network error:0.00064853
Generation
Start
Pause
Stop
Copy ANN to workspace
ANN configuration
Training data file: snltrainingdata
Hidden layers size: 6
Hidden layers transfer function: Hyperbolic tangent
Output layer transfer function: Linear
GA configuration
Population size: 20
Elitism: 1
Generation gap: 1
Maximum generations: 10000
Maximum stall generations: Inf
Minimum network error: 0
Initial population individuals:
Crossover operator: Neuron
Mutation operator: Evolving
Advanced mutation configuration
Hybrid configuration
Baldwinian evolution cycles: 0
Lamarckian evolution cycles: 1
Post-evolution training cycles: 0
Hybrid-training function: trainlm
About Gennettool...

GENNET-Toolbox: An Evolving Genetic Algorithm for Neural Network Training

Thank you for your attention!

eloy.irigoyen@ehu.es, vicente.gomez@ehu.es, fernando.artaza@ehu.es

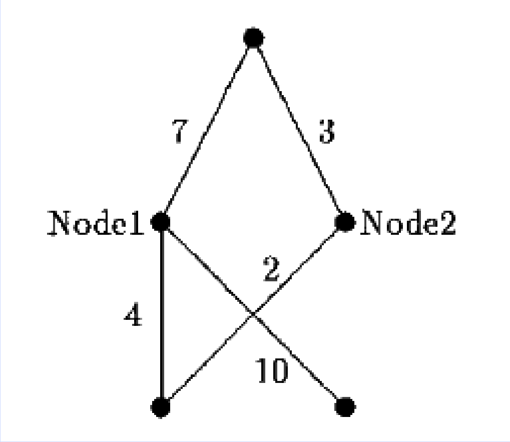
Parameters-coding



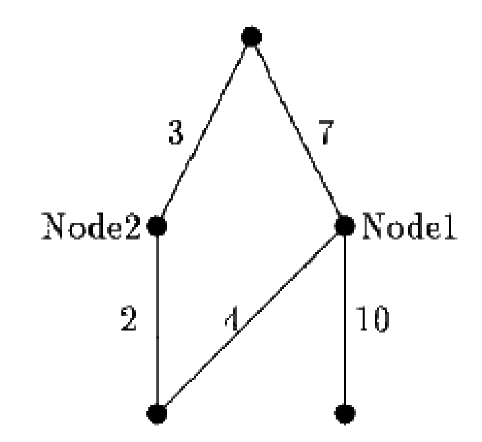
w11	w21	u1	w12	w22	u2	w13	w23	u3	w14	w24	u4	w15	w25	u5	v11	v21	v31	v41	v51	b1

Permutation problem

- Two ANN functionally identical, but with different genotype.



0100 1010 0010 0000 0111 0011



0010 0000 0100 1010 0011 0111