Soft Computing: A Common Framework to Represent Meta-heuristics

Using Knowledge and Reasoning to Control Search and Vice-versa

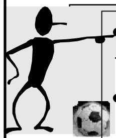
Piero P. Bonissone
GE Global Research Center
bonissone@crd.ge.com

NFL, Meta-Heuristics & Hybrid SC: Outline

- **The NFL**
- **Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics**
- **Soft Computing Overview**
  - SC Components: PR, FL, NN, EA
- **Using SC to implement the Meta-Heuristics: Modeling with FL and EA**
- **Example of Hybrid SC Systems at GE**
  - FLC Parameter Tuning by EA
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- **Conclusions**

IEEE Argentina, 11 Marzo 2004 PPB 2

NFL, Meta-Heuristics & Hybrid SC: Outline



- **The NFL**
- Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics
- Soft Computing Overview
  - SC Components: PR, FL, NN, EA
- Using SC to implement the Meta-Heuristics: Modeling with FL and EA
- Example of Hybrid SC Systems at GE
  - FLC Parameter Tuning by EA
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- Conclusions

Evolutionary Algorithms and the No Free Lunch Theorem (NFL)

- Wolpert and Macready (1997): No Free Lunch Theorems for Optimization - IEEE TEC (1)1: 67-82
  - *Taken over the set of all possible combinatorial optimization problems*, the performance of an two search algorithms** is the same*
- For any two "black-box" optimization algorithms a_1 and a_2:

$$\sum_f P(\vec{c} \mid f, m, a_1) = \sum_f P(\vec{c} \mid f, m, a_2)$$

  - m is the number of time steps
  - d_m is a particular set of m values (for *distinct* visiting points)
  - $\bar{c}$ is the cost value of $\{d_m\}$
  - f is a combinatorial optimization problem
- Danger of comparing algorithms on a small sample of problems
- We must incorporate problem-specific knowledge into the behavior of the algorithm (from weak to strong search, in AI parlance)

* $f : X \to Y$ and $|X|$ *and* $|Y|$ *are finite*

** *Black boxes do not rely explicitly on cost structure of partial solutions, like branch-and-bound*

No Free Lunch Theorem (NFL) – cont.

- For any optimization algorithm, an elevated performance over one class of problems is offset by degraded performance over another class
- The performance average of a given optimization algorithm over the entire class of potential problems is constant.
  - If an algorithm performs better than random search for some problems, it will perform worse than random search for other problems, maintaining the performance average constant.

No Free Lunch Theorem (NFL) – cont.

- For any optimization algorithm, an elevated performance over one class of problems is offset by degraded performance over another class
- The performance average of a given optimization algorithm over the entire class of potential problems is constant.
  - If an algorithm performs better than random search for some problems, it will perform worse than random search for other problems, maintaining the performance average constant.
- Any one set of potentially optimal algorithm parameters can be considered valid only for a limited subset of problems and should not be expected to result in consistently superior performance over the entire space of optimization problems.
- Theoretically this applies only to problems closed under permutation.
- In MOO problems holds over the set of all MOO problems that share the same type of Pareto Front (Corne & Knowles)

No Free Lunch Theorem (NFL): Take away

- General-purpose universal optimization strategy is impossible, and the only way one strategy can outperform another is if it is specialized to the structure of the specific problem under consideration (Ho & Pepyne, 2001).
- Therefore, it is essential to leverage specific problem domain knowledge and incorporate it into the algorithm. Some of this knowledge will be embedded in the design of the algorithms, and some explicitly through meta-heuristics

IEEE Argentina, 11 Marzo 2004 PPB 7

Example of Embedding Knowledge in the Design of an Evolutionary Algorithm

Implicit KR

- **Solution Representation**
  - **Data Structures** (Parsimonious; Avoid unfeasible solutions)
  - **Encoding** (Natural encoding: binary, real, integer)
  - **Constraints** (Static: embedded in DS; Dynamic: penalty functions)

Explicit KR

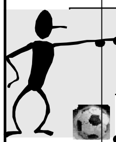
- **Initial Population**
  - **Boundary Conditions** (Exercise boundary conditions)
  - **Best Guesses** (Seed candidate solutions)
- **Variational and Selection Operators**
  - **Customized Mutation** (self-repair)
  - **Customized Crossover**
  - **Variable Selection Pressure** (Increasing from linear to NL)
- **Assignment, Tuning and Control of EA parameters**
  - **Prior Design** (Usually from literature studies of parameters)
  - **Offline Tuning**
  - **On-Line Control**

Meta-Heuristics

IEEE Argentina, 11 Marzo 2004 PPB 8

NFL, Meta-Heuristics & Hybrid SC: Outline

- The NFL



- **Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics**
- Soft Computing Overview
  - SC Components: PR, FL, NN, EA
- Using SC to implement the Meta-Heuristics: Modeling with FL and EA
- Example of Hybrid SC Systems
  - FLC Parameter Tuning by EA
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- Conclusions

Semantic Clarification

- **Meta-heuristics –1**

The use of heuristic procedures to extend search and optimization algorithms to avoid local optima

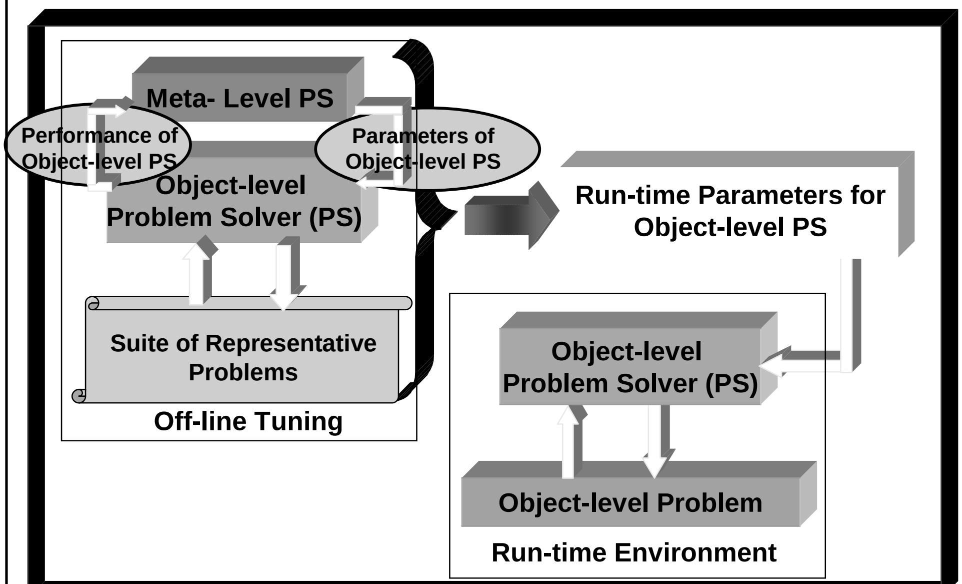
- **Meta-heuristics –2**

The use of heuristic procedures at the ***meta-level*** to:

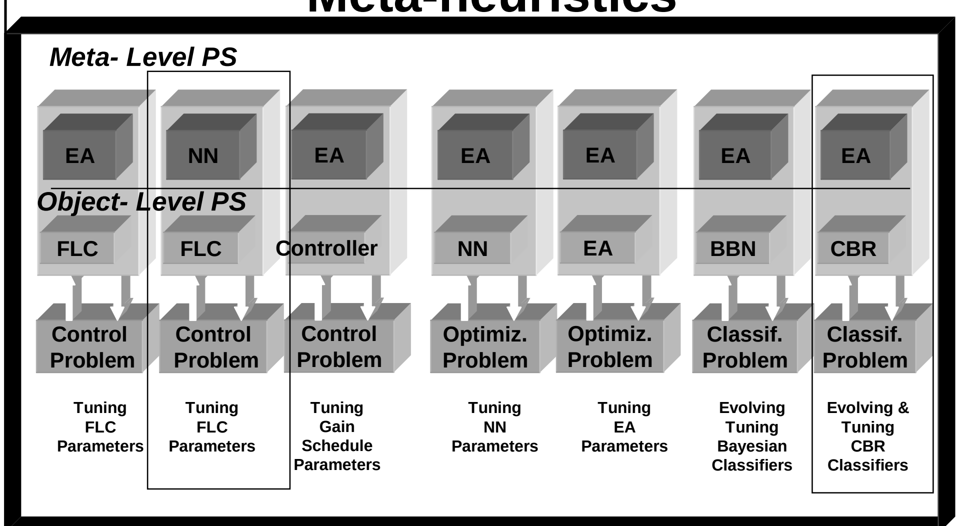
- Control/Guide
- Tune
- Allocate/Reallocate computational resources for
- Reasoning about

the ***object-level*** problem solver to improve the quality, performance, or efficiency of its solution

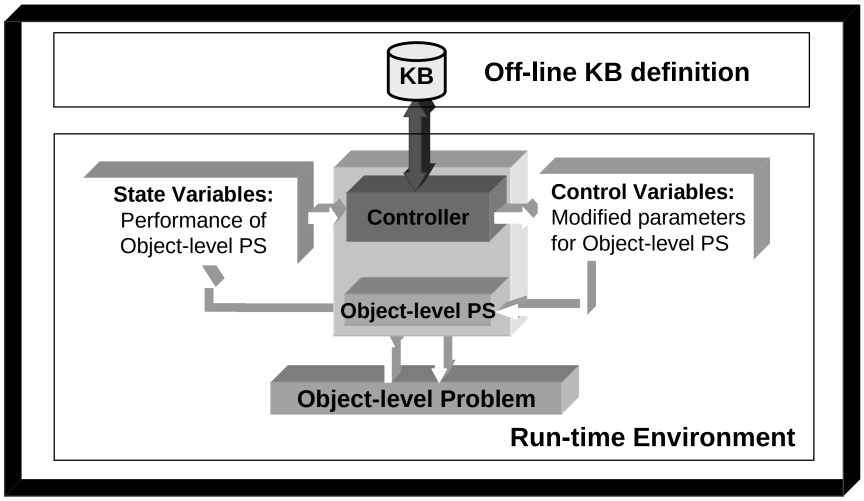
Off-line Meta-Heuristics



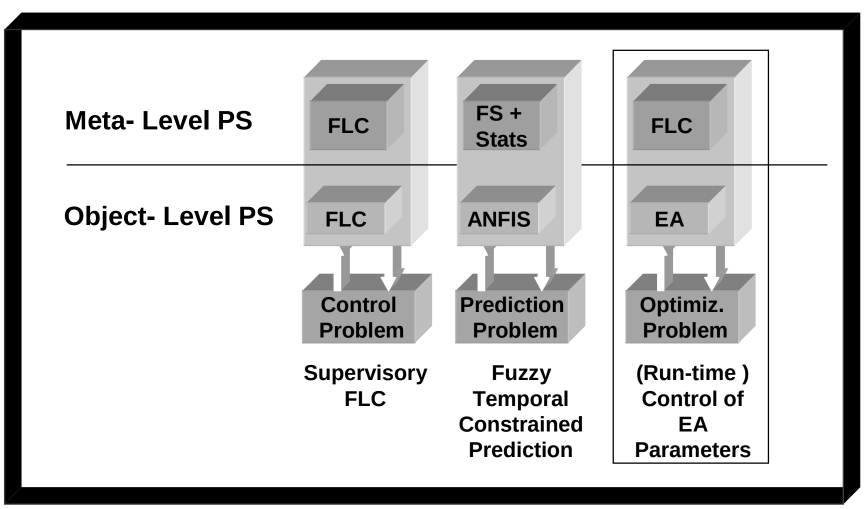
Examples of Off-line Meta-heuristics



On-line Meta-Heuristics:
Knowledge-based Controller for Object-level Problem Solver

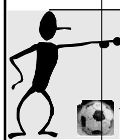


Examples of On-line Meta-heuristics



NFL, Meta-Heuristics & Hybrid SC: Outline

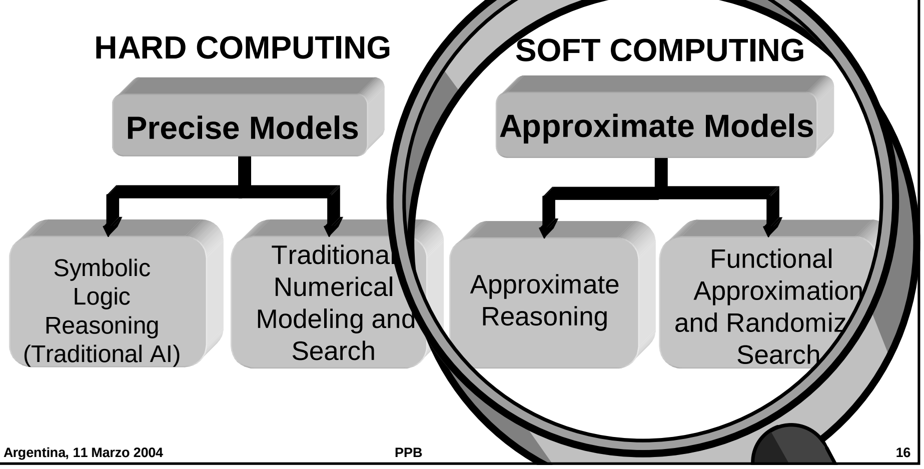
- The NFL
- Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics

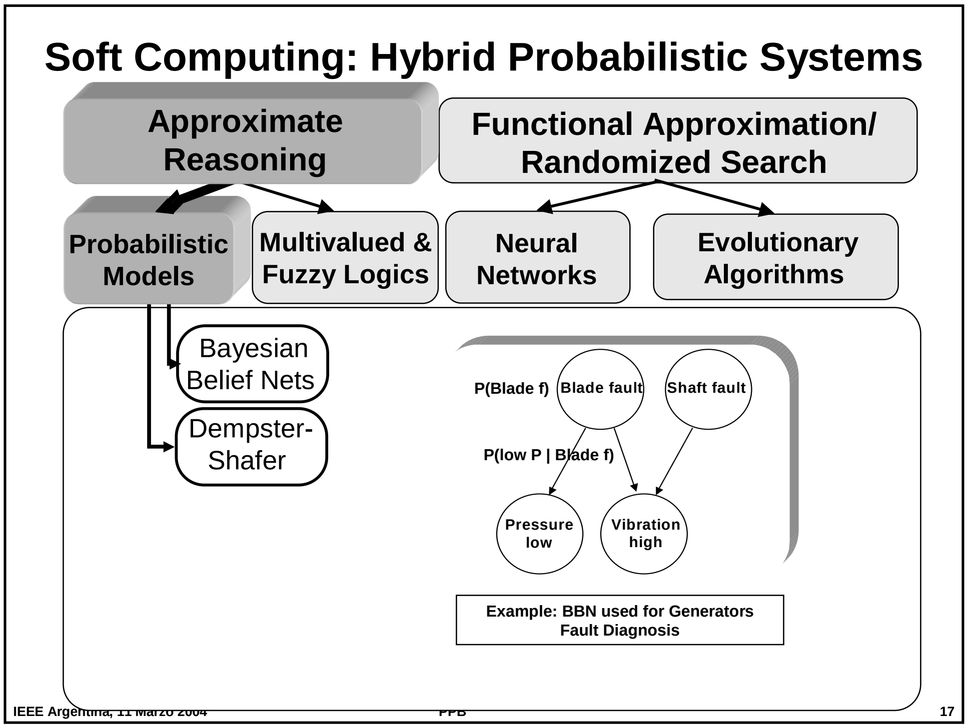


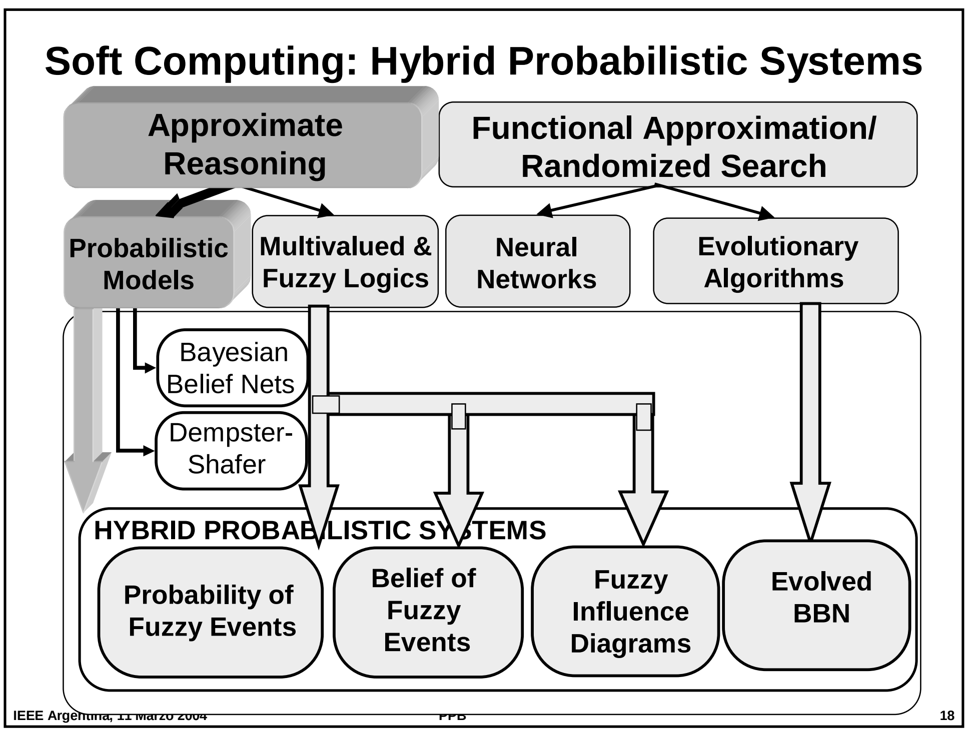
- **Soft Computing Overview**
  - SC Components: PR, FL, NN, EA
- Using SC to implement the Meta-Heuristics: Modeling with FL and EA
- Example of Hybrid SC Systems
  - FLC Parameter Tuning by EA
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- Conclusions

Problem Solving Technologies

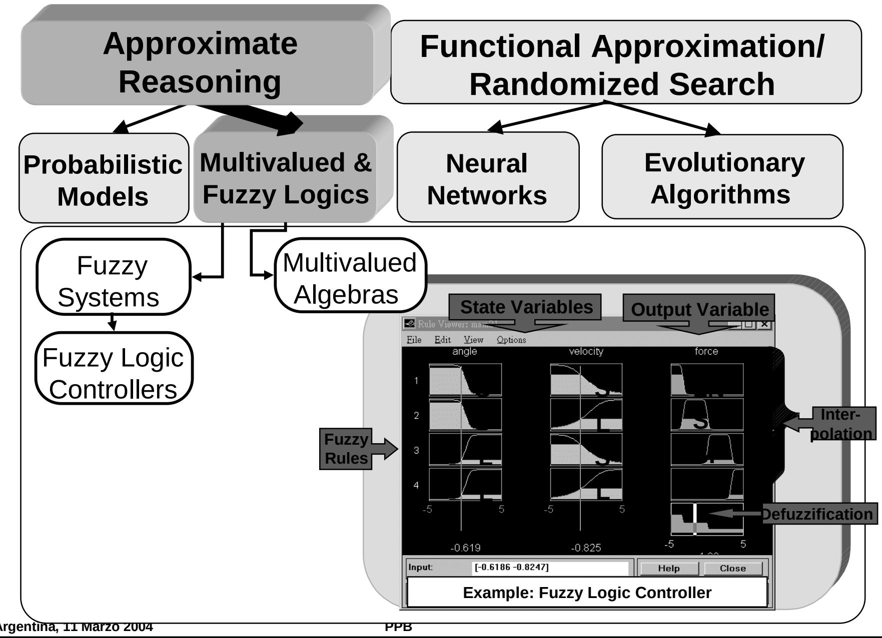
"In contrast to traditional hard computing, soft computing exploits the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, low solution-cost, and better rapport with reality" (Zadeh 1991)



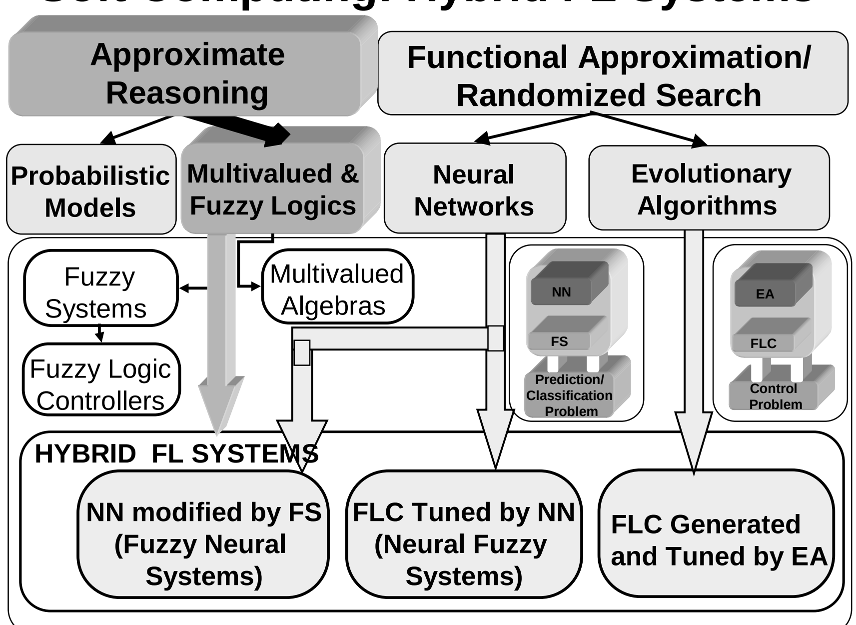

Soft Computing: Hybrid Probabilistic Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Bayesian Belief Nets
Dempster-Shafer
P(Blade f)
Blade fault
Shaft fault
P(low P | Blade f)
Pressure low
Vibration high
Example: BBN used for Generators Fault Diagnosis
17

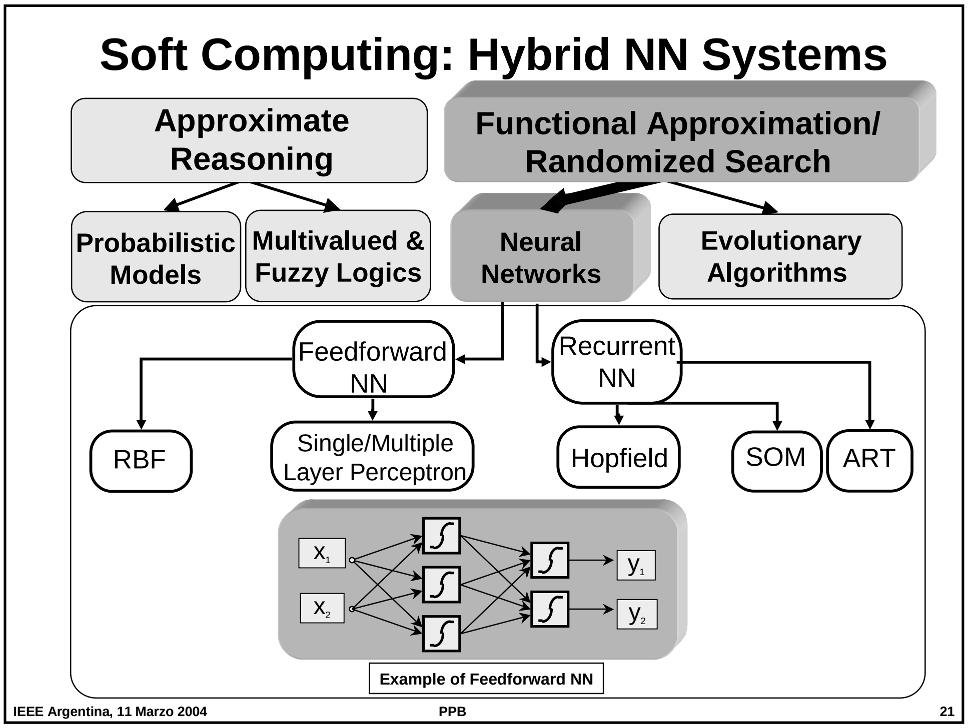

Soft Computing: Hybrid Probabilistic Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Bayesian Belief Nets
Dempster-Shafer
HYBRID PROBABILISTIC SYSTEMS
Probability of Fuzzy Events
Belief of Fuzzy Events
Fuzzy Influence Diagrams
Evolved BBN
18

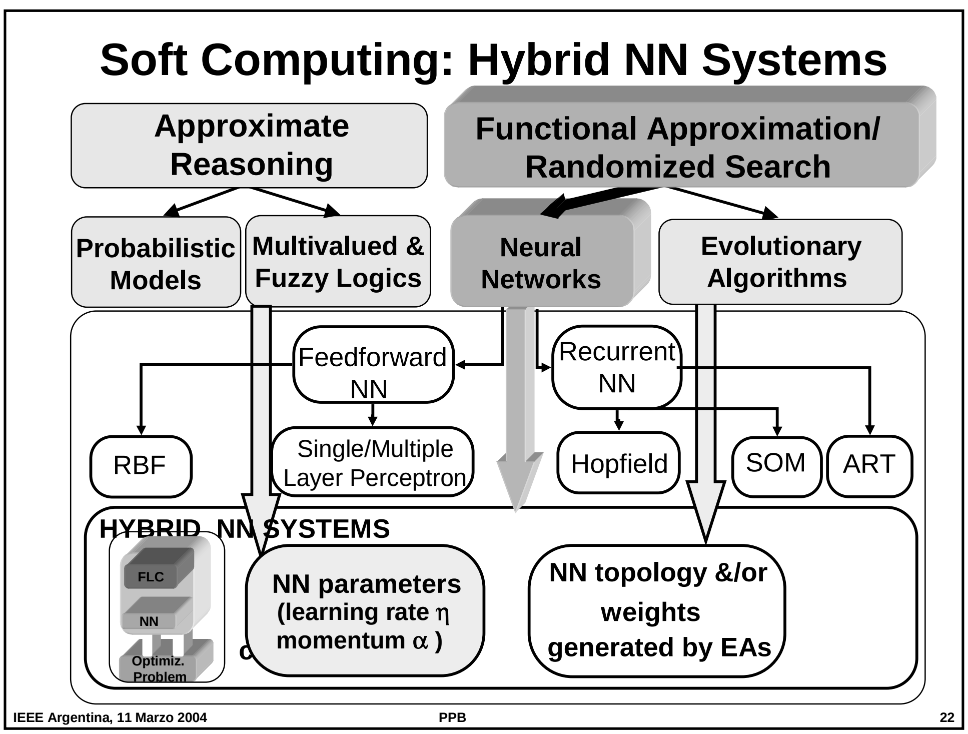
Soft Computing: Hybrid FL Systems

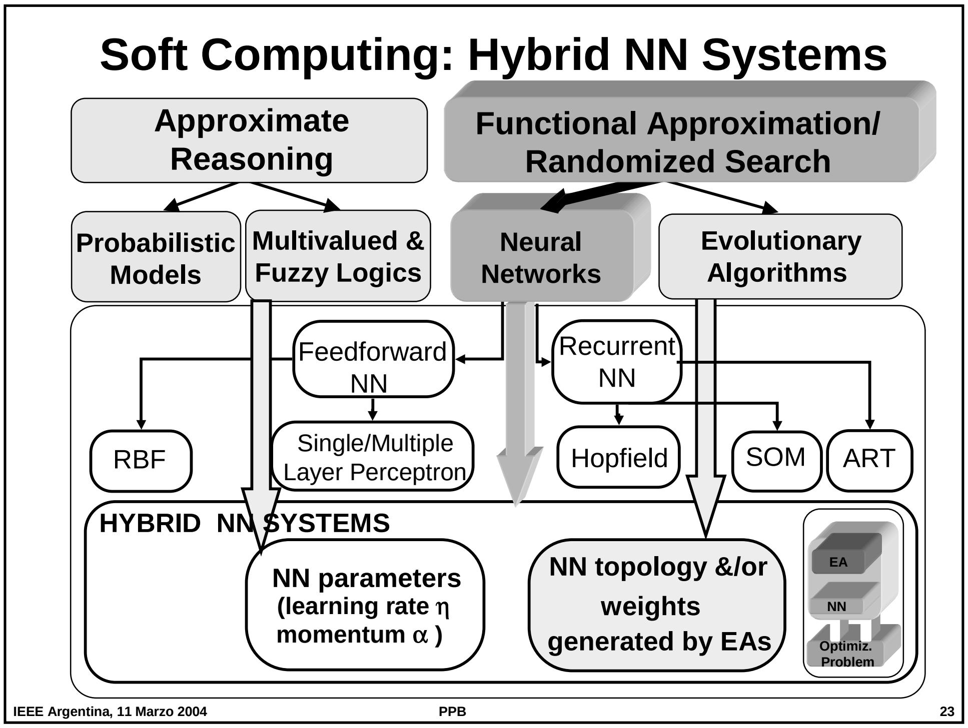


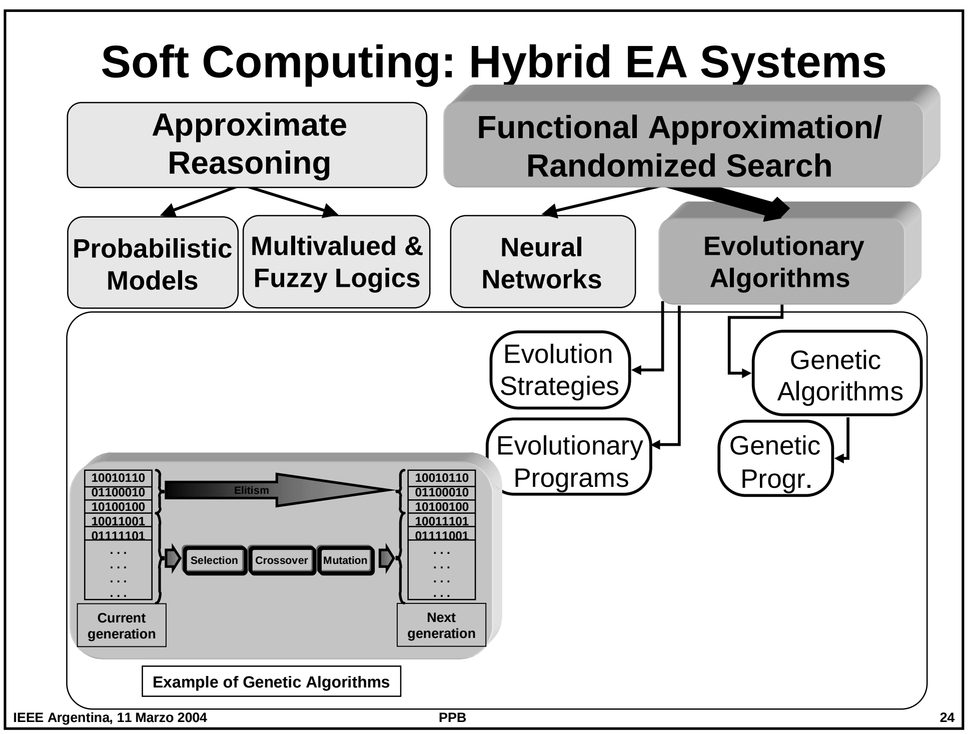
Soft Computing: Hybrid FL Systems



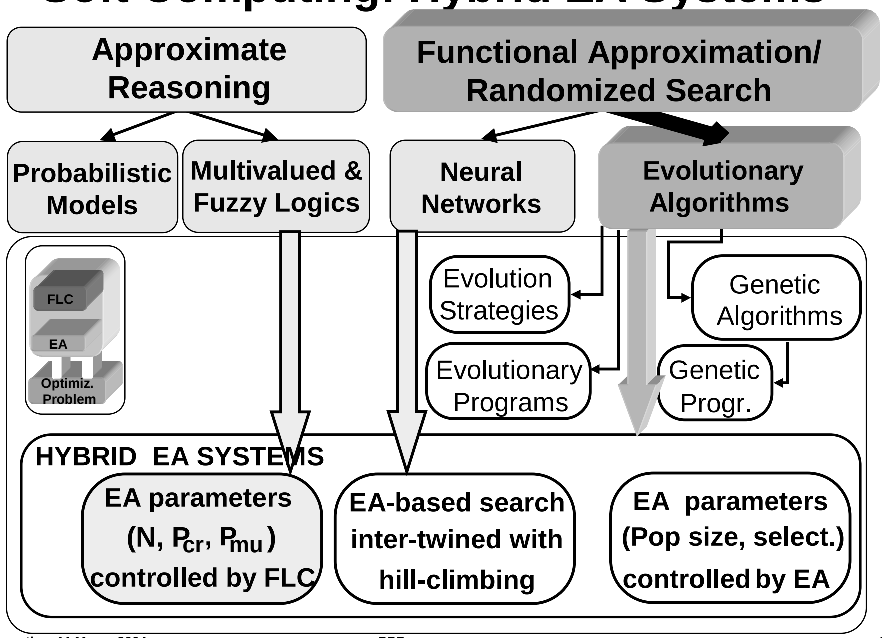

Soft Computing: Hybrid NN Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Feedforward NN
Recurrent NN
RBF
Single/Multiple Layer Perceptron
Hopfield
SOM
ART
x1
x2
y1
y2
Example of Feedforward NN
IEEE Argentina, 11 Marzo 2004
PPB
21


Soft Computing: Hybrid NN Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Feedforward NN
Recurrent NN
RBF
Single/Multiple Layer Perceptron
Hopfield
SOM
ART
HYBRID NN SYSTEMS
FLC
NN
Optimiz. Problem
NN parameters (learning rate η momentum α)
NN topology &/or weights generated by EAs
IEEE Argentina, 11 Marzo 2004
PPB
22

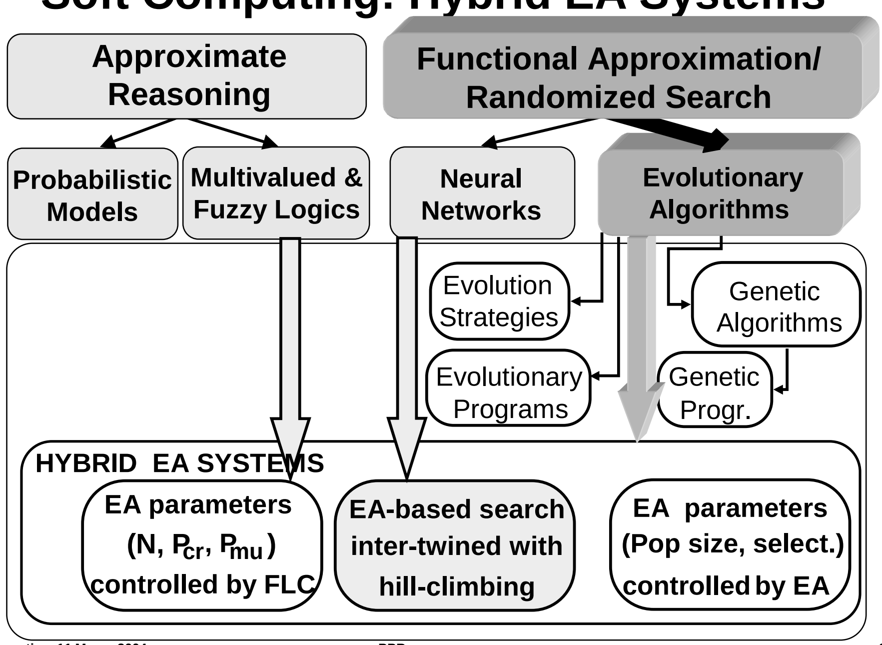

Soft Computing: Hybrid NN Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Feedforward NN
Recurrent NN
RBF
Single/Multiple Layer Perceptron
Hopfield
SOM
ART
HYBRID NN SYSTEMS
NN parameters (learning rate η momentum α)
NN topology &/or weights generated by EAs
EA
NN
Optimiz. Problem
IEEE Argentina, 11 Marzo 2004
PPB
23


Soft Computing: Hybrid EA Systems
Approximate Reasoning
Functional Approximation/ Randomized Search
Probabilistic Models
Multivalued & Fuzzy Logics
Neural Networks
Evolutionary Algorithms
Evolution Strategies
Genetic Algorithms
Evolutionary Programs
Genetic Progr.
10010110
01100010
10100100
10011001
01111101
Elitism
Selection
Crossover
Mutation
10010110
01100010
10100100
10011101
01111001
Current generation
Next generation
Example of Genetic Algorithms
IEEE Argentina, 11 Marzo 2004
PPB
24

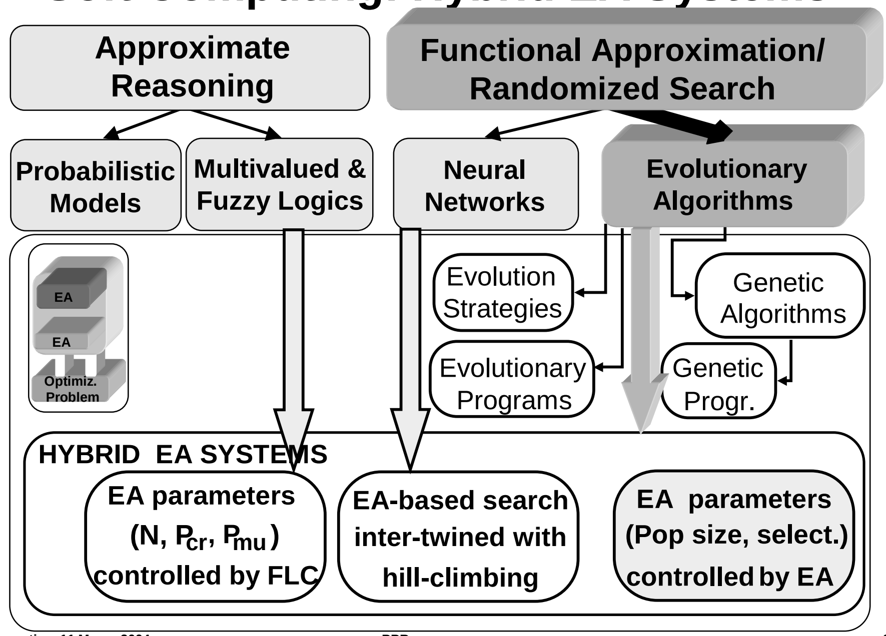
Soft Computing: Hybrid EA Systems



Soft Computing: Hybrid EA Systems



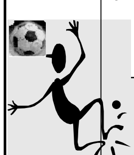
Soft Computing: Hybrid EA Systems



IEEE Argentina, 11 Marzo 2004 PPB 27

NFL, Meta-Heuristics & Hybrid SC: Outline

- The NFL
- Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics
- Soft Computing Overview
  - SC Components: PR, FL, NN, EA



- **Using SC to implement the Meta-Heuristics: Modeling with FL and EA**
- Example of Hybrid SC Systems at GE
  - FLC Parameter Tuning by EA
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- Conclusions

IEEE Argentina, 11 Marzo 2004 PPB 28

Fuzzy Logic Genealogy

- **Origins: MVL for treatment of imprecision and vagueness**
  - **1930s: Post, Kleene, and Lukasiewicz attempted to represent *undetermined*, *unknown,* and other possible intermediate truth-values.**
  - **1937: Max Black suggested the use of a *consistency profile* to represent vague (ambiguous) concepts**
  - **1965: Zadeh proposed a complete theory of fuzzy sets (and its isomorphic fuzzy logic), to represent and manipulate ill-defined concepts**

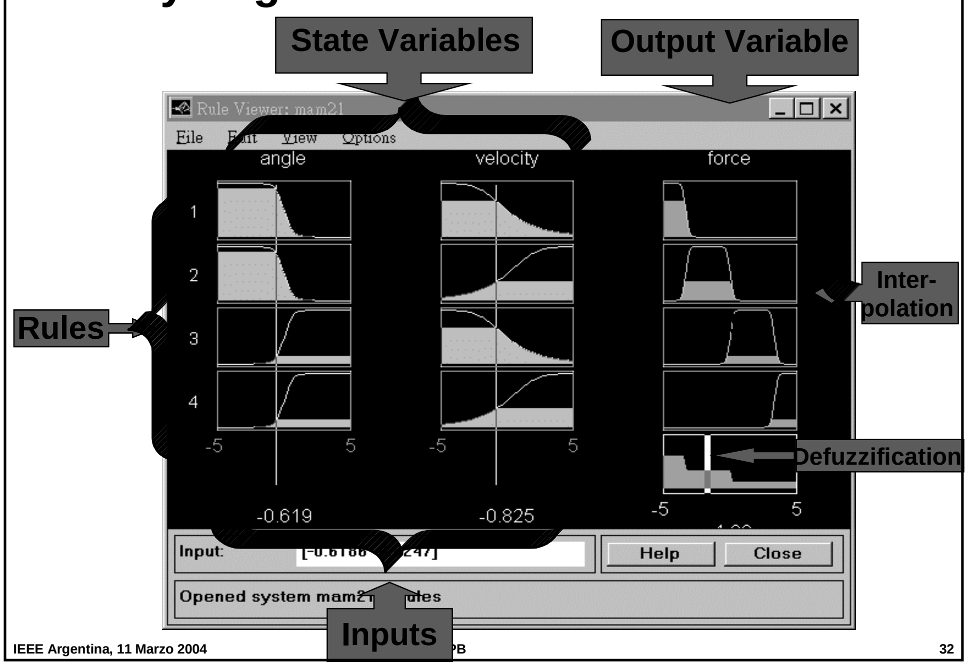
Fuzzy Logic : Linguistic Variables

- **Fuzzy logic give us a language (with syntax and local semantics), in which we can translate our qualitative domain knowledge.**
- ***Linguistic variables* to model dynamic systems**
- **These variables take *linguistic values* that are characterized by:**
  - **a *label* - a sentence generated from the syntax**
  - **a *meaning* - a membership function determined by a local semantic procedure**

Fuzzy Logic : Reasoning Methods

- The meaning of a linguistic variable may be interpreted as a elastic constraint on its value.
- These constraints are propagated by fuzzy inference operations, based on the *generalized modus-ponens*.
- A FL Controller (FLC) applies this reasoning system to a Knowledge Base (KB) containing the problem domain heuristics.
- The inference is the result of interpolating among the outputs of all relevant rules.
- The outcome is a membership distribution on the output space, which is defuzzified to produce a crisp output.

Fuzzy Logic Control : Inference Method



FLC Inference Method (cont.)

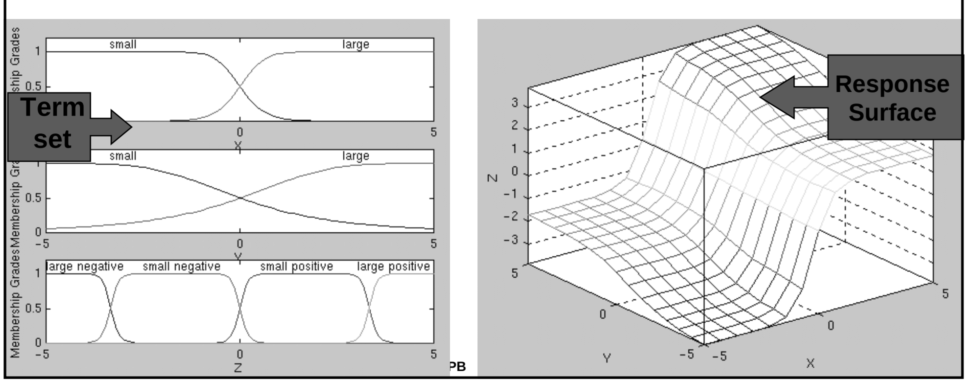
- **A FLC (KB + Reasoning Mechanism) defines a deterministic response surface in the cross product of state and output spaces, which approximates the original relationship.**

- **The FLC leverages the interpolation properties of this reasoning mechanism, to exhibit robustness with respect to parameter variations, disturbances, etc.**

Example (MISO): Max-min Composition with Centroid Defuzzification

Rules set

- **If X is SMALL and Y is SMALL then Z is NEG. LARGE**
- **If X is SMALL and Y is LARGE the Z is NEG. SMALL**
- **If X is LARGE and Y is SMALL the Z is POS. SMALL**
- **If X is LARGE and Y is LARGE then Z is POS. LARGE**



Evolutionary Algorithms (EA)

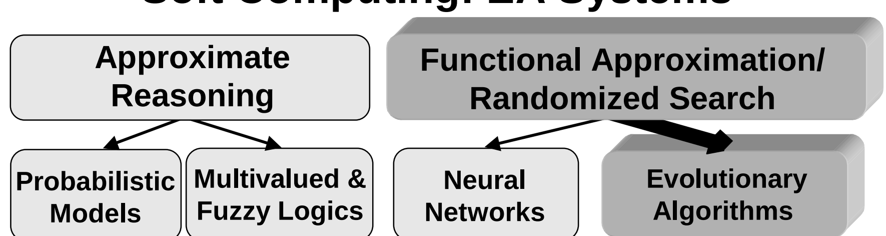
EA are part of the Derivative-Free Optimization and Search Methods:

- Evolutionary Algorithms
- Simulated annealing (SA)
- Random search
- Downhill simplex search
- Tabu search

EA consists of:

- Evolution Strategies (ES)
- Evolutionary Programming (EP)
- Genetic Algorithms (GA)
  - Genetic Programming (GP)

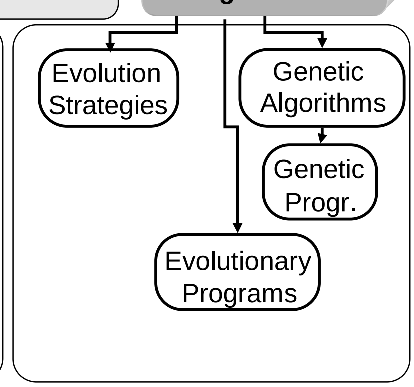
Soft Computing: EA Systems



- **Most Evolutionary Algorithms (EAs) can be described by:**

$$\boxed{\mathbf{x}[t+1] = s(v(\mathbf{x}[t]))}$$

- **$\mathbf{x}[t]$: the population at time t under representation x**
- **v : is the variation operator(s)**
- **s : is the selection operator**



Evolutionary Algorithms: ES

- **<u>Evolutionary Strategies (ES)</u>**
  - **Originally proposed for the optimization of continuous functions**
  - **(μ , λ)-ES and (μ + λ)-ES**
    - » A population of μ parents generate λ offspring
    - » Best μ offspring are selected in the next generation
    - » (μ , λ)-ES: parents are **excluded** from selection
    - » (μ + λ)-ES: parents are **included** in selection
  - **Started as (1+1)-ES** *(Reschenberg)* **and evolved to (μ + λ)-ES** *(Schwefel)*
  - **Started with Mutation only (with individual mutation operator) and later added a recombination operator**
  - **Focus on behavior of individuals**

IEEE Argentina, 11 Marzo 2004 PPB 37

Evolutionary Algorithms: EP

- **<u>Evolutionary Programming (EP)</u>**
  - **Originally proposed for sequence predictiom and optimal gaming strategies**
  - **Currently focused on continuous parameter optimization and training of NNs**
  - **Could be considered a special case of (μ + μ) -ES without recombination operator**
  - **Focus on behavior of species (hence no crossover)**
  - **Proposed by** *Larry Fogel (1963)*

IEEE Argentina, 11 Marzo 2004 PPB 38

Evolutionary Algorithms: GA

•Genetic Algorithms (GA)

- **Perform a randomized search in solution space using a genotypic rather than a phenotypic**
- **Each solution is encoded as a chromosome in a population (a binary, integer, or real-valued string)**
  - » Each string's element represents a particular feature of the solution
- **The string is evaluated by a fitness function to determine the solution's quality**
  - » Better-fit solutions survive and produce offspring
  - » Less-fit solutions are culled from the population
- Strings are evolved using mutation & recombination operators.
- New individuals created by these operators form next generation of solutions
- Started by *Holland (1962; 1975)*

Evolutionary Algorithms: GP

•Genetic Programming (GP)

- **A special case of Genetic Algorithms**
  - »Chromosomes have a **hierarchical** rather than a **linear** structure
  - »Their sizes are not predefined
  - »Individuals are tree-structured programs
  - »Modified operators are applied to sub-trees or single nodes
- Proposed by *Koza (1992)*

A Trend Toward Convergence among EAs

• EA components have increasingly shared their typical traits:
  - ES have added recombination operators similar to GAs,
  - GAs have been extended by the use of real-number-encoded chromosomes, adaptive mutation rates, and additive mutation operators (similar to ES).
  - EP is similar to a (μ+μ)-ES without recombination operator

- EA exhibit an *adaptive behavior* that allows them to handle non-linear, high dimensional problems without requiring differentiability or explicit knowledge of the problem structure.
- EA are very robust to time-varying behavior, even though they may exhibit low speed of convergence.

Modeling

• ***Model =***

Structure + Parameters + Search Method

• **Classical control theory:**
  - **Structure: order of the differential equations**
  - **Parameters: coefficients of differential equation.**
  - **Search method: LMSE, Pole-placement, etc.**

Modeling Using FLC (Mamdani type)

Approximate Reasoning | Functional Approximation/ Randomized Search

Probabilistic Models | Multivalued & Fuzzy Logics | Neural Networks | Evolutionary Algorithms

Fuzzy Systems

Fuzzy Logic Controllers

- **A Mamdani- type FLC approximates a relationship between a state *X* and an output *Y* by using a KB and a reasoning mechanism (*generalized modus-ponens*).**
- **The Knowledge Base (KB) is defined by:**
  - ***Scaling factors*** **(SF): ranges of values of state and output variables**
  - ***Termset*** **(TS): membership functions of values**
  - ***Ruleset*** **(RS): a syntactic mapping of symbols from X to Y**

IEEE Argentina, 11 Marzo 2004 PPB 43

Soft Computing: EA Systems

Approximate Reasoning | Functional Approximation/ Randomized Search

Probabilistic Models | Multivalued & Fuzzy Logics | Neural Networks | Evolutionary Algorithms

-The structure of the model is the *representation of an individual in the population (e.g., binary string, vector, parse tree, Finite State Machine).*

-The parameters of the model are the *Population Size, Probability of Mutation, Prob. of Recombination, Generation Gap, etc.*

-The search method is a global search based on maximization of population fitness function

Evolution Strategies

Genetic Algorithms

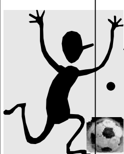
Genetic Progr.

Evolutionary Programs

IEEE Argentina, 11 Marzo 2004 PPB 44

NFL, Meta-Heuristics & Hybrid SC: Outline

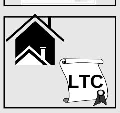
- The NFL
- Tuning or Controlling the Object-Level Problem Solver (PS) with Meta-Heuristics
- Soft Computing Overview
  - SC Components: PR, FL, NN, EA
- Using SC to implement the Meta-Heuristics: Modeling with FL and EA



- **Example of Hybrid SC Systems at GE**
  - **FLC Parameter Tuning by EA**
  - FLR and F-CBR Parameter Tuning by EA
  - EA Parameter Setting (by EA) or Control (by FL)
- Conclusions

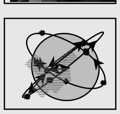
Soft Computing & Related Technologies at GE

GE Appliances
- Preferred Service Contracts (Stat.)
- Call Center Support (CBR)



GE Capital Services
- GE Mortgages
  Collateral Evaluation (Fusion/FL/CBR)
- GE Insurance
  Digital Underwriting (FL/Stat/EA/NN)

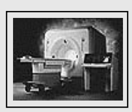
GE Plastics
- Automated Color Matching (CBR)



LM Fed. Systems
- Scheduling Maintenance for Constellation of Satellites (GA)

LM SkunkWorks
- UAV Mission Manager (AI/FL/GA)

LM ORSS
- Vessel Management Syst. (AI/GA)

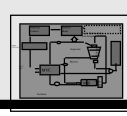


GE Medical Systems
- SPT Auto Analysis for MRI (FL)
- Reverse Engineering of Picker (FL)
- FE Analysis tool (FL)
- X-Ray error Logs Analysis (CBR)

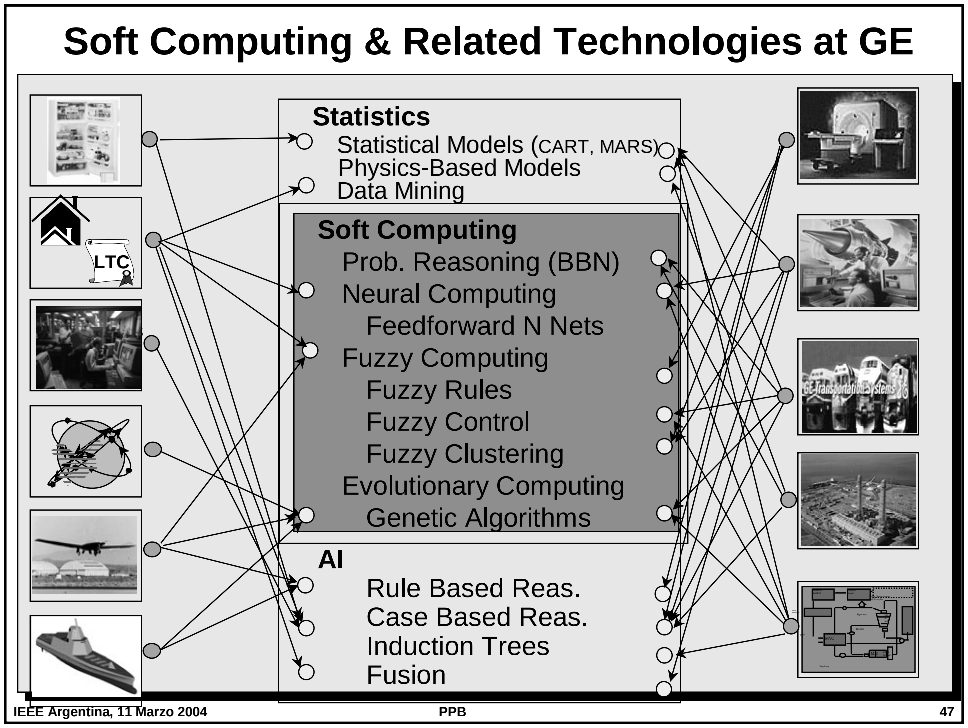
GE Aircraft Engines
- Center for Remote Diagn. (CBR)
- Customer Response Center (CBR)
- Anomaly Detection (FL/Stat.)
- IMATE - Maintenance Advisor (NN/FL)
- Resolver Drift - Sensor Fusion (FL)

GE Transportation Systems
- Log from Transportation DB (CBR)
- *Prototype Train Handling Cntrl. (FL/EA)*
- Prototype Trend Analysis (Stat.)
- Embedded/Remote Diagnostics (BBN)

GE Power Gen. Systems
- Remote Anomaly Detection (Stat.)
- Embedded/Remote Diagnostics (BBN)
- Call Center Problem/Solution (CBR)



GE Ind. Syst. /GE Trading
- Paper Web Breakage Prediction (NN/Stat./Induction)
- Control Mixing of Cement (FL/GA)


Soft Computing & Related Technologies at GE
LTC
Statistics
Statistical Models (CART, MARS)
Physics-Based Models
Data Mining
Soft Computing
Prob. Reasoning (BBN)
Neural Computing
Feedforward N Nets
Fuzzy Computing
Fuzzy Rules
Fuzzy Control
Fuzzy Clustering
Evolutionary Computing
Genetic Algorithms
AI
Rule Based Reas.
Case Based Reas.
Induction Trees
Fusion
IEEE Argentina, 11 Marzo 2004
PPB
47