Topic 1:

What is AI?

Review and introduction

(we'll get back to different topics later)
Today's Reading:
Luger 1.1.1 – 1.1.3, 1.2 – 1.4
RN sec. 1

Lecture notes by M.Negnevitsky and W.Hsu are used

Questions Addressed

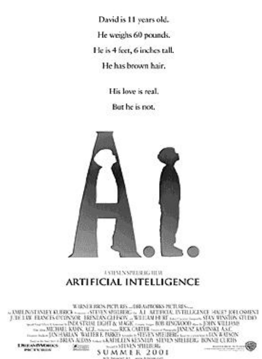
- Problem Area
  - What *are* intelligent systems and agents?
  - Why are we interested in developing them?
- Methodologies
  - What kind of software is involved? What kind of math?
  - How do we develop it (software, repertoire of techniques)?
  - Who uses AI? (Who are practitioners in academia, industry, government?)
- Artificial Intelligence as a Science
  - What is AI?
  - What does it have to do with intelligence? Learning? Problem solving?
  - What are some interesting problems to which intelligent systems can be applied?
  - *Should I be interested in AI (and if so, why)?*
- Today: Brief Tour of AI History
  - Study of intelligence (classical age to present), AI systems (1940-present)
  - Viewpoints: philosophy, math, psychology, engineering, linguistics

What is AI?

- Director:

Steven Spielberg
Stars: Jude Law, Haley Joel Osment, Frances O'Connor

Plot: In the wake of an environmental disaster, a new kind of self-aware computer is created



What is AI again?: descriptive approach

- Although the term of AI has been widely used for quite a long time with steadily increasing amount of research and applications, there is no anonymously accepted definition. AI can mean many things to different people and various techniques are considered as belonging to AI.
- The term coined in 1956 by J. McCarthy at MIT
- Two branches: engineering discipline dealing with the creation of intelligent machines and empirical science concerned with the computational modelling of human intelligence
- The goal of AI is developing methods, which allow producing thinking machines that can solve problems
- Which problems?
- ill-defined and ill-structured
- complicated taxonomy or classifying
- Combinatorial optimisation

What is AI again?:

- The great variety of AI techniques have been developed and applied over the history for solving the problems mentioned above.
- Some of these methodologies are"conventional" or "old" methods (1950s):
- search algorithms,
- Probabilistic reasoning,
- natural language processing,
- belief networks, etc.

- Others are "new" (1960s) – soft computing and computational intelligence

What is AI again? Systematic approach

- Four Categories of Systemic Definitions
  - 1. Think like humans
  - 2. Act like humans
  - 3. Think *rationally*
  - 4. Act *rationally*

What is AI again? Systematic approach

- Thinking Like Humans
  - *Machines with minds* (Haugeland, 1985)
  - Automation of "decision making, problem solving, learning..." (Bellman, 1978)
- Acting Like Humans
  - *Functions* that *require intelligence when performed by people* (Kurzweil, 1990)
  - Making computers do things *people currently do better* (Rich and Knight, 1991)
- Thinking Rationally
  - Computational models of mental faculties (Charniak and McDermott, 1985)
  - Computations that make it possible to *perceive*, *reason*, and *act* (Winston, 1992)
- Acting Rationally
  - Explaining, emulating intelligent behavior via computation (Schalkoff, 1990)
  - Branch of CS concerned with automation of intelligent behavior (Luger and Stubblefield, 1993)

What is AI? Thinking and Acting Like Humans

- Concerns: Human Performance (Figure 1.1 R&N, Left-Hand Side)
  - Top: thought processes and reasoning (learning and inference)
  - Bottom: behavior (interacting with environment)
- Machines With Minds
  - Cognitive modelling
    - Early historical examples: problem solvers (see R&N Section 1.1)
    - Application (and one driving force) of cognitive science
  - Deeper questions
    - What is intelligence?
    - What is consciousness?
- Acting Humanly: The Turing Test Approach
  - Capabilities required
    - Natural language processing
    - Knowledge representation
    - Automated reasoning
    - Machine learning
  - Turing Test: can a machine appear indistinguishable from a human to an experimenter?

What is AI? Thinking and Acting Rationally

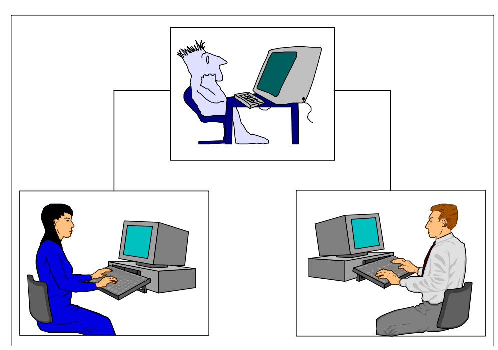
- Concerns: Human Performance (Figure 1.1 R&N, Right-Hand Side)
  - Top: thought processes and reasoning (learning and inference)
  - Bottom: behavior (interacting with environment)
- Computational Cognitive Modelling
  - Rational ideal
    - In this course: rational agents
    - Advanced topics: learning, utility theory, decision theory
  - Basic mathematical, computational models
    - Decisions: automata Search
    - Concept learning
- Acting Rationally: The Rational Agent Approach
  - Rational action: *acting to achieve one's goals, given one's beliefs*
  - Agent: entity that perceives and acts
  - Focus of next topic
    - "Laws of thought" approach to AI: correct inferences (reasoning)
    - Rationality not *limited to* correct inference

One of the most significant papers on machine intelligence, ***"Computing Machinery and Intelligence"***, was written by the British mathematician ***Alan Turing*** over fifty years ago . However, it still stands up well under the test of time, and the Turing's approach remains universal.

He asked: ***Is there thought without experience? Is there mind without communication? Is there language without living? Is there intelligence without life?*** All these questions, as you can see, are just variations on the fundamental question of artificial intelligence, ***Can machines think?***

- Turing did not provide definitions of machines and thinking, he just avoided semantic arguments by inventing a game, the ***Turing Imitation Game***.
- The imitation game originally included two phases. In the first phase, the interrogator, a man and a woman are each placed in separate rooms. The interrogator's objective is to work out who is the man and who is the woman by questioning them. The man should attempt to deceive the interrogator that *he* is the woman, while the woman has to convince the interrogator that *she* is the woman.

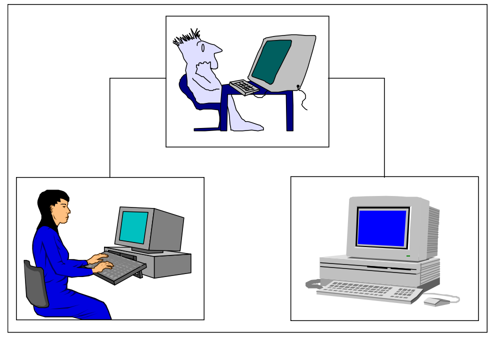
Turing Imitation Game: Phase 1



Turing Imitation Game: Phase 2

- In the second phase of the game, the man is replaced by a computer programmed to deceive the interrogator as the man did. It would even be programmed to make mistakes and provide fuzzy answers in the way a human would. If the computer can fool the interrogator as often as the man did, we may say this computer has passed the intelligent behaviour test.

Turing Imitation Game: Phase 2



The Turing test has two remarkable qualities that make it really universal.

- By maintaining communication between the human and the machine via terminals, the test gives us an objective standard view on intelligence.
- The test itself is quite independent from the details of the experiment. It can be conducted as a two-phase game, or even as a single-phase game when the interrogator needs to choose between the human and the machine from the beginning of the test.

- Turing believed that by the end of the 20th century it would be possible to program a digital computer to play the imitation game. Although modern computers still cannot pass the Turing test, it provides a basis for the verification and validation of knowledge-based systems.
- **A program thought intelligent in some narrow area of expertise is evaluated by comparing its performance with the performance of a human expert.**
- To build an intelligent computer system, we have to capture, organise and use human expert knowledge in some narrow area of expertise.

AI History (old)

- Philosophy Foundations (400 B.C. – present)
  - Mind: dualism (Descartes), materialism (Leibniz), empiricism (Bacon, Locke)
  - Thought: syllogism (Aristotle), induction (Hume), logical positivism (Russell)
  - Rational agentry (Mill)
- Mathematical Foundations (c. 800 – present)
  - Early: algorithms (al-Khowarazmi, 9th century Arab mathematician), Boolean logic
  - Computability (20th century – present)
    - Cantor diagonalization, Gödel's incompleteness theorem
    - Formal computuational models: Hilbert's Entscheidungsproblem, Turing
    - Intractability and NP-completeness

AI History (not too old)

- Computer Engineering (1940 – present)
- Linguistics (1957 – present)
- Stages of AI
  - Gestation (1943 – c. 1956), infancy (c. 1952 – 1969)
  - Disillusioned early (c. 1966 – 1974), later childhood (1969 – 1979)
  - "Early" (1980 – 1988), "middle" adolescence (c. 1985 – present)

The history of artificial intelligence (birth)

The birth of artificial intelligence (1943 – 1956)

The first work recognised in the field of AI was presented by **Warren McCulloch** and **Walter Pitts** in 1943. They proposed a model of an artificial neural network and demonstrated that simple network structures could learn.

McCulloch, the second "founding father" of AI after Alan Turing, had created the corner stone of neural computing and artificial neural networks (ANN).

- The third founder of AI was **John von Neumann**, the brilliant Hungarian-born mathematician. In 1930, he joined the Princeton University, lecturing in mathematical physics. He was an adviser for the Electronic Numerical Integrator and Calculator project at the University of Pennsylvania and helped to design the **Electronic Discrete Variable Calculator**. He was influenced by McCulloch and Pitts's neural network model. When **Marvin Minsky** and **Dean Edmonds**, two graduate students in the Princeton mathematics department, built the first neural network computer in 1951, von Neumann encouraged and supported them.

- Another of the first generation researchers was **Claude Shannon**. He graduated from MIT and joined Bell Telephone Laboratories in 1941. Shannon shared Alan Turing's ideas on the possibility of machine intelligence. In 1950, he published a paper on chess-playing machines, which pointed out that a typical chess game involved about 10^{120} possible moves (Shannon, 1950). Even if the new von Neumann-type computer could examine one move per microsecond, it would take 3×10^{106} years to make its first move. Thus Shannon demonstrated the need to use heuristics in the search for the solution.

- In 1956, **John McCarthy**, **Marvin Minsky** and **Claude Shannon** organised a summer workshop at Dartmouth College. They brought together researchers interested in the study of machine intelligence, artificial neural nets and automata theory. Although there were just ten researchers, this workshop gave birth to a new science called ***artificial intelligence***.

The rise of artificial intelligence, or the era of great expectations (1956 – late 1960s)

- The early works on neural computing and artificial neural networks started by McCulloch and Pitts was continued. Learning methods were improved and **Frank Rosenblatt** proved the ***perceptron convergence theorem***, demonstrating that his learning algorithm could adjust the connection strengths of a perceptron.

- One of the most ambitious projects of the era of great expectations was the **General Problem Solver (GPS)**. **Allen Newell** and **Herbert Simon** from the Carnegie Mellon University developed a general-purpose program to simulate human-solving methods.
- Newell and Simon postulated that a problem to be solved could be defined in terms of ***states***. They used the mean-end analysis to determine a difference between the current and desirable or *goal state* of the problem, and to choose and apply *operators* to reach the goal state. The set of operators determined the solution plan.

- However, GPS failed to solve complex problems. The program was based on formal logic and could generate an infinite number of possible operators. The amount of computer time and memory that GPS required to solve real-world problems led to the project being abandoned.
- In the sixties, AI researchers attempted to simulate the thinking process by inventing *general methods* for solving *broad classes of problems*. They used the general-purpose search mechanism to find a solution to the problem. Such approaches, now referred to as ***weak methods***, applied weak information about the problem domain.

- By 1970, the euphoria about AI was gone, and most government funding for AI projects was cancelled. AI was still a relatively new field, academic in nature, with few practical applications apart from playing games. So, to the outsider, the achieved results would be seen as toys, as no AI system at that time could manage real-world problems.

Unfulfilled promises, or the impact of reality (late 1960s – early 1970s)

The main difficulties for AI in the late 1960s were:

- Because AI researchers were developing general methods for broad classes of problems, early programs contained little or even no knowledge about a problem domain. To solve problems, programs applied a search strategy by trying out different combinations of small steps, until the right one was found. This approach was quite feasible for simple **toy problems**, so it seemed reasonable that, if the programs could be "scaled up" to solve large problems, they would finally succeed.

Many of the problems that AI attempted to solve were **too broad and too difficult**. A typical task for early AI was machine translation. For example, the National Research Council, USA, funded the translation of Russian scientific papers after the launch of the first artificial satellite (Sputnik) in 1957.

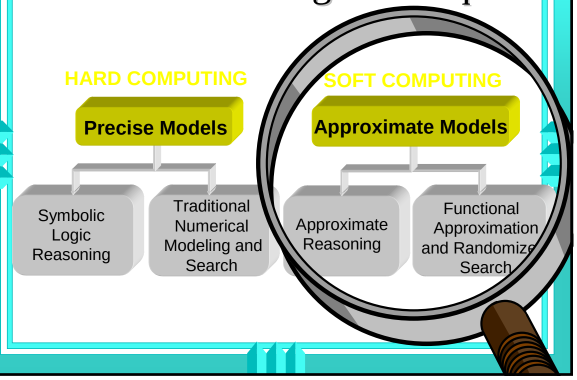
- The spirit is willing but the flesh is weak
- The vodka is good but the meat is rotten

- In 1971, the British government also suspended support for AI research. Sir **James Lighthill** had been commissioned by the Science Research Council of Great Britain to review the current state of AI. He did not find any major or even significant results from AI research, and therefore saw no need to have a separate science called "artificial intelligence".

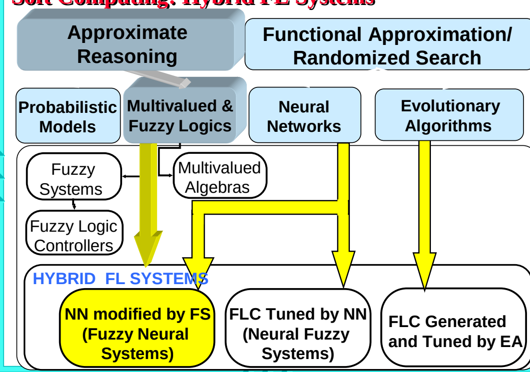
Soft Computing

- Soft Computing (SC): the symbiotic use of many emerging problem-solving disciplines.
- According to Prof. Zadeh:
  "...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"
- Soft Computing Main Components:
  - -Approximate Reasoning:
    - » Probabilistic Reasoning, Fuzzy Logic
  - -Search & Optimization:
    - » Neural Networks, Evolutionary Algorithms

Problem Solving Techniques



Soft Computing: Hybrid FL Systems



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

This is fuzzy math!

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

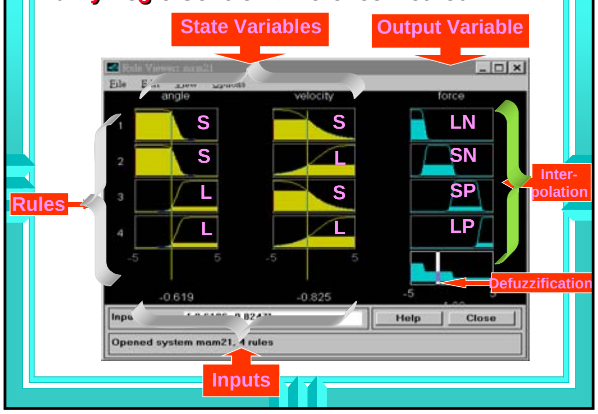
- The meaning of a linguistic variable may be interpreted as a elastic constraint on its value.
- These constraints are propagated by fuzzy inference operations, based on the generalized

•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.

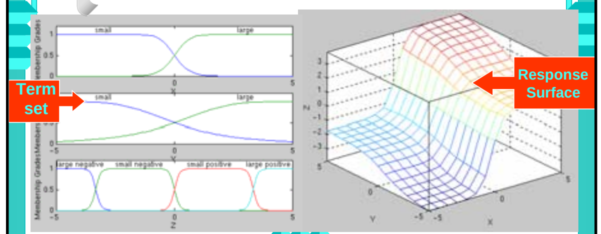
Fuzzy Logic Control : Inference Method



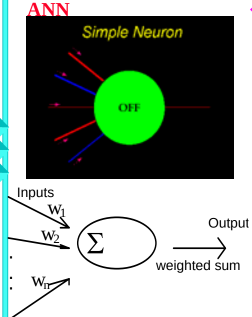
Example (MISO): Max-min Composition with Centroid Defuzzification

- 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

Rules set

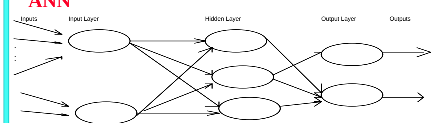


ANN



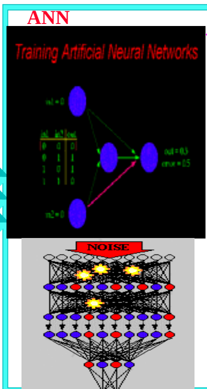
- The artificial neuron is a mathematical construct that emulates the more salient function of biological neurons, namely this signal integration and threshold firing behavior. Just as in the biological case, such neurons are bound together by various connection weights that determine how the outputs from one neuron are to be algebraically weighted before arriving at receiving neurons. The intelligence within these collective structures of artificial neurons (i.e., ANNs) is stored within these sundry algebraic connection weights.

ANN



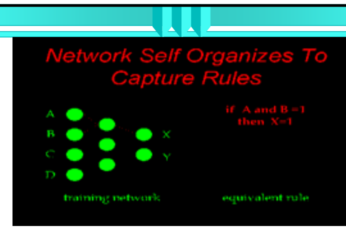
- **All of the information stored within an artificial neural network (i.e., its virtual computer programs) takes the form of connection strengths between neurons. These are values by which the signals from one artificial neuron to another are multiplied before being summed up within the receiving neuron. Important to note is that these weights are not 'hand wired' into these networks by computer nerds. Instead, special computer programs mathematically 'spank' the net until it consistently yields the correct outputs for any given set of inputs.**

ANN



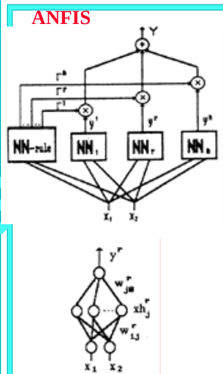
- **ANN Training -we successively apply all known inputs to the net (here the Exclusive Or data) propagating signals in the forward direction, observe network output, and then backwardly propagate corrections to the respective connections in the net. We continue this process until the net yields the correct output for all known test cases. At this point we say that we have a neural network model of some conceptual space. Applications of inputs unencountered during the network's training phase should yield reasonable estimates for network outputs (i.e., the model's predictions). The most important aspect of this process is that the network discovers on its own what the underlying rules actually are.**

ANN



- **How ANNs Capture Rules -** Artificial Neural Networks have taken a rap for being 'black boxes'. That is, they give the right results, but don't explain why they do so. In reality, they internally develop connection traces that embody the rules behind the conceptual space they are training on. Here we see a network learning three implicit rules hidden within a database of numbers.

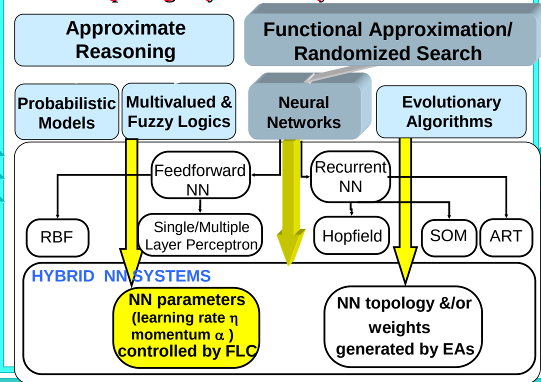
ANFIS



- **The ways to combine FL and ANN:**

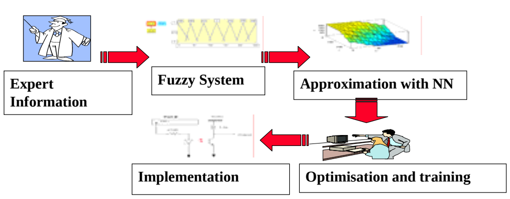
1) fuzzy systems where ANN learn the shape of the surface of membership functions, the rules and output membership values,
2) fuzzy systems that are expressed in the form of ANN and are designed using a learning capability of the ANN,
3) fuzzy systems with ANN which are used to tune the parameters of the fuzzy controller as a design tool but not as a component of the final fuzzy system.

Soft Computing: Hybrid NN Systems



Implementation of FS with NN

- **Advantages:**
- Knowledge acquisition ability of FS
- Learning ability of NN
- Optimisation and adjustment against any criteria (including multi-criteria)
- Simpler and cheaper implementation

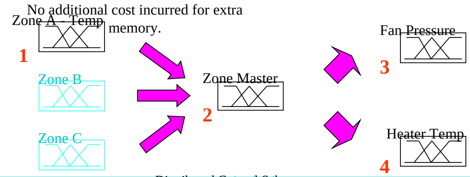


Multi-Zone Thermostat Controller

Project Benefited from 4 different Fuzzy Controllers.

Implemented with the same Neural Network Engine, using different neural weights for each controller.

No additional cost incurred for extra memory.



Distributed Control Scheme

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)**

Evolutionary Algorithms Characteristics

- Most Evolutionary Algorithms can be described by

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

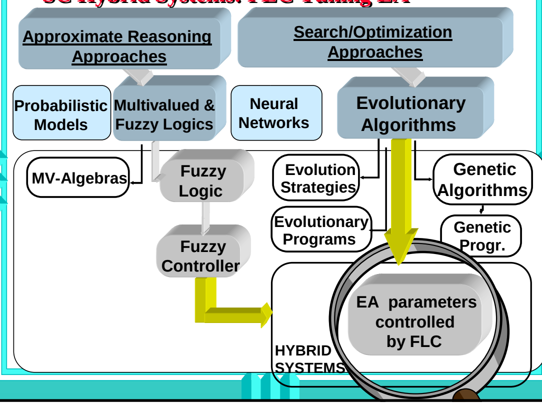
- $x[t]$: the population at time t under representation x
- v : is the variation operator(s)
- s : is the selection operator

Evolutionary Algorithms Characteristics

- **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.**

SC Hybrid Systems: FLC Tuning EA



Evolutionary Algorithms: ES

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

Evolutionary Algorithms: EP

- Evolutionary Programming (EP)
  - Originally proposed for sequence prediction 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)***

Where to next ?

Why Study Artificial Intelligence?

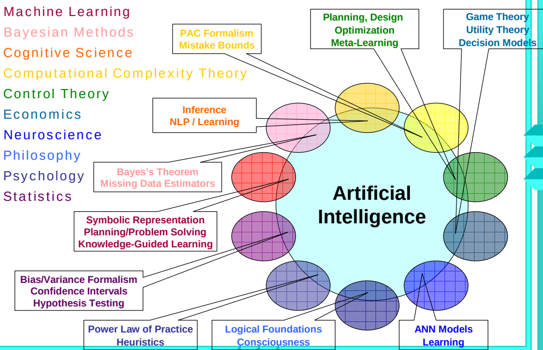
- New Computational Capabilities
  - Advances in uncertain reasoning, knowledge representations
  - Learning to act: robot planning, control optimization, decision support
  - Database mining: converting (technical) records into knowledge
  - Self-customizing programs: learning news filters, adaptive monitors
  - Applications that are hard to program: automated driving, speech recognition

Why Study Artificial Intelligence?

- Better Understanding of Human Cognition
  - Cognitive science: theories of knowledge acquisition (e.g., through practice)
  - Performance elements: reasoning (inference) and *recommender* systems
- Time is Right
  - Recent progress in algorithms and theory
  - Rapidly growing volume of online data from various sources
  - Available computational power
  - Growth and interest of AI-based industries (e.g., data mining/KDD, planning)

Relevant Disciplines

- Machine Learning
- Bayesian Methods
- Cognitive Science
- Computational Complexity Theory
- Control Theory
- Economics
- Neuroscience
- Philosophy
- Psychology
- Statistics



Synergy in SC: Reasons & Approaches

- Hybrid Soft Computing
  - **Leverages** *tolerance for imprecision*, **uncertainty, and incompleteness - intrinsic to the problems to be solved**
  - **Generates *tractable, low-cost,* *robust* solutions to such problems** *by*
- Tight Hybridization
  - Data-driven Tuning of Knowledge-derived Models
    - » Translate domain knowledge into initial structure and parameters
    - » Use Global or local data search to tune parameters
  - Knowledge-driven Search Control
    - » Use Global or local data search to derive models (Structure + Parameters)
    - » Translate domain knowledge into an algorithm's controller to improve/manage solution convergence and quality

Synergy in SC: Reasons & Approaches

- Loose Hybridization (Model Fusion)
  - **Does not combine** *features* **of methodologies - only their results**
  - **Their** *outputs* **are compared, contrasted, and aggregated, to increase reliability**
- Hybrid Search Methods
  - Intertwining ***local*** search within ***global*** search
  - Embedding knowledge in operators for global search
- Future:
  - Circle of SC's related technologies will probably widen beyond its current constituents.
  - Push for low-cost solutions and intelligent tools will result in deployment of hybrid SC systems that ***integrate reasoning and search techniques***.