Introduction to Soft Computing

Course overview

Pawel Herman

Department of Computational Biology

School of Computer Science and Communication

KTH Royal Institute of Technology

BEST summer school

Stockholm, June 9-13 2014

General remarks regarding the course

- Formula of the course
  - Introduction + 4 lectures (morning + partly afternoon sessions)
  - Labs + project work (afternoon sessions)
- Major constraints
  - TIME, TIME, TIME .....
  - breadth at the cost of depth
  - still, scope is limited
- Requirements and recommendations
  - your own laptop with Matlab installed + basic Matlab skills
  - attention, willingness to learn, readiness to ask questions
  - presense is strongly recommended

Intended learning outcomes

- To get general familiarity with SC methods, to understand their potential and recognize limitations
- To obtain fundamentaltheoretical understanding of computations involved
- To understand what problems SC can effectively address
- To obtain basic hands-on experience in applying SC methods, in particular FL, NN and GA in Matlab environment
- To learn at a basic level how to design SC-based solutions to stereotypical scheduling, optimisation, control, pattern recognition and diagnostic problems among others.

What is Soft Computing?

"Soft computing is not a single methodology. Rather, it is a consortium of computing methodologies which collectively provide a foundation for the conception, design and deployment of intelligent systems. At this juncture, the principal members of soft computing are fuzzy logic, neurocomputing, genetic computing, and probabilistic computing, with the last subsuming evidential reasoning, belief networks, chaotic systems, and parts of machine learning theory. In contrast to traditional hard computing, soft computing is tolerant of imprecision, uncertainty and partial truth. <u>The guiding principle of soft computing</u> is: exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness, low solution cost and better rapport with reality."

Zadeh

Hard vs Soft Computing

- Soft Computing is tolerant to
  - imprecision
  - uncertainty
  - approximation
  - noise
  - partial truth
- Unlike in hard computing
  - soft constraints
  - robustness more important than accuracy
  - finding satisfactory solutions in a reasonable amount of time

Hard vs Soft Computing

- Soft Computing is tolerant to
  - imprecision
  - uncertainty
  - approximation
  - noise
  - partial truth
- Unlike in hard computing
  - soft constraints
  - robustness more important than accuracy
  - finding satisfactory solutions in a reasonable amount of time

Human brain can store and process information, which is imprecise, uncertain, unlabeled

Hard vs Soft Computing

- Soft Computing is tolerant to
  - imprecision
  - uncertainty
  - approximation
  - noise
  - partial truth
- Unlike in hard computing
  - soft constraints
  - robustness more important than accuracy
  - finding satisfactory solutions in a reasonable amount of time

The real world problems are pervasively imprecise and uncertain.

Hard vs Soft Computing

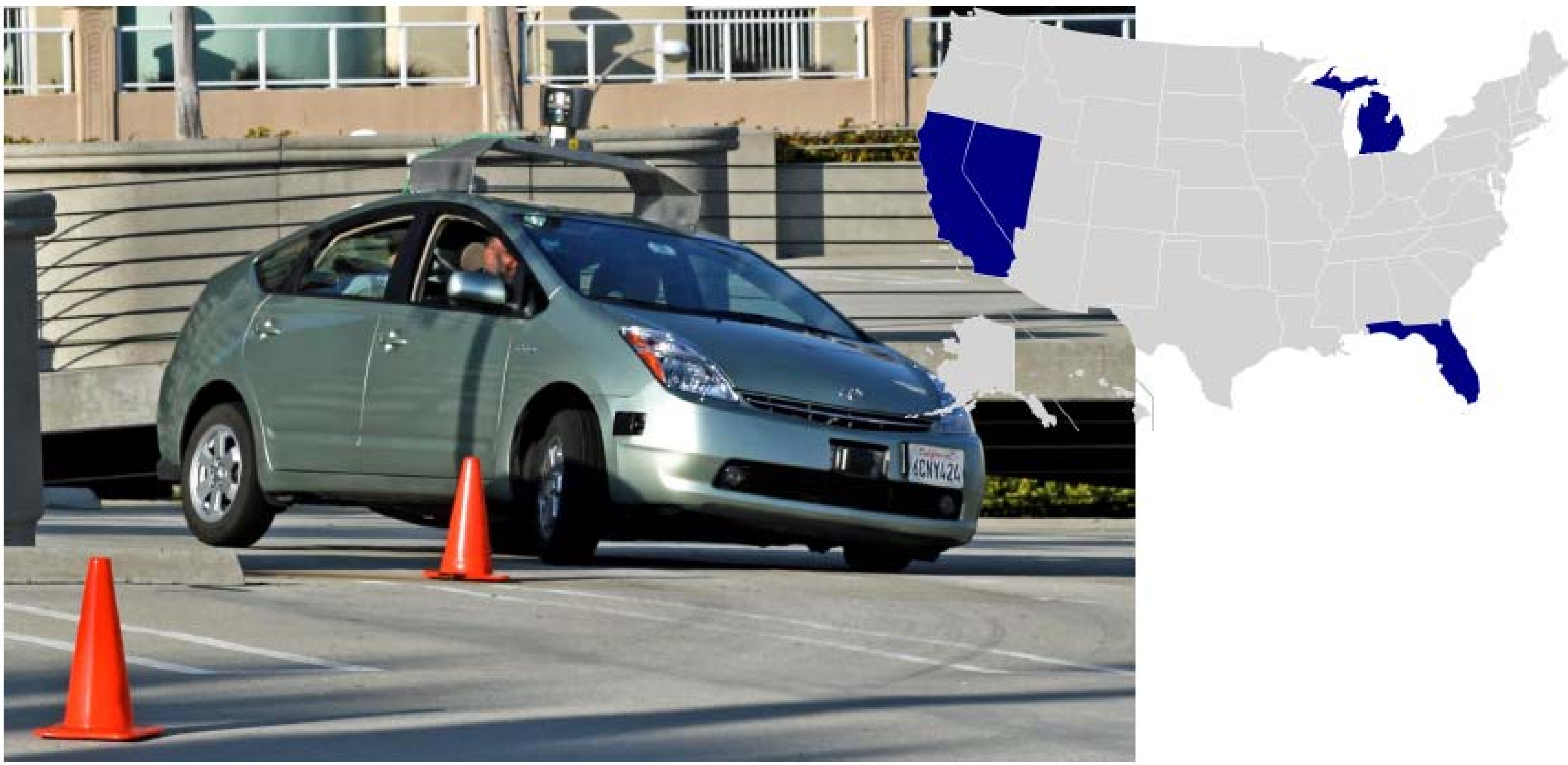
- Soft Computing is tolerant to
  - imprecision
  - uncertainty
  - approximation
  - noise
  - partial truth
- Unlike in hard computing
  - soft constraints
  - robustness more important than accuracy
  - finding satisfactory solutions in a reasonable amount of time

Use inexact solutions to computationally hard tasks

Problem domain for SC

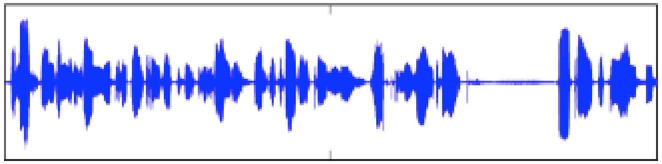
- Examples of problems that cannot be solved using traditional algorithmic approaches
  - understanding meaning of sentences, natural language analysis
  - perception
    - signal recognition (e.g. biomedical signals)
    - phoneme dicrimination
    - computer vision problems, e.g. face recognition
    - handwriting recognition
  - control of non-linear complex systems (robotics)
  - automotive systems, navigation
  - playing complex games (e.g., strategic)
  - diagnostic problems, e.g. in medicine

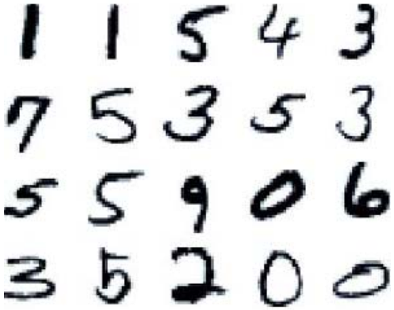
Google car



BigDog Google

Other real-world applications





Hard vs Soft Computing

- Soft Computing approach
  - learning from data
  - understanding, explaining, reasoning
  - qualitative methodology
  - linguistic models
  - moderate computational complexity and fast
  - partial truth
  - circumvents complexity by approximation, works when there is no mathematical model
  - good generalistion in approximation and interpolation tasks despite high-dimensionality
  - facilitates searching for hidden structure in data

Soft Computing as a basis for Intelligent Systems (grand goal)

- Intelligent Systems should
  - acquire (learn) and interpret information
  - understand relations
  - make inference, reason
  - adapt to new environments
  - apply the acquired knowledge in new conditions

Soft Computing as a basis for Intelligent Systems (grand goal)

- Intelligent Systems should
  - acquire (learn) and interpret information
  - understand relations
  - make inference, reason
  - adapt to new environments
  - apply the acquired knowledge in new conditions

van Gogh

Chagall

Soft Computing as a basis for Intelligent Systems

- Intelligent Systems should
  - acquire (learn) and interpret information
  - understand relations
  - make inference, reason
  - adapt to new environments
  - apply the acquired knowledge in new conditions

“Intelligence is the capability of a decision making system to adapt its behavior to meet its goals in a range of environments”

David B. Fogel

Soft Computing as a basis for Intelligent Systems

- Intelligent Systems should
  - acquire (learn) and interpret information
  - understand relations
  - make inference, reason
  - adapt to new environments
  - apply the acquired knowledge in new conditions

Intelligence as a general mental capability that involves an ability to reason, solve problems, conceptualise, process abstract objects, plan, understand complex ideas and learn from experience among others.

Broader context of Computational and Artificial Intelligence

- *"A system is computationally intelligent when it: deals only with numerical (low-level) data, has a pattern recognition component, does not use knowledge in the AI sense; and additionally, when it (begins to) exhibit (i) computational adaptivity; (ii) computational fault tolerance; (iii) speed approaching human-like turnaround, and (iv) error rates that approximate human performance"*

  Bezdek

Broader context of Computational and Artificial Intelligence

- *"Computational intelligence (CI) is a recently emerging area of fundamental and applied research exploiting a number of advanced information processing technologies. The main components of CI encompass neural networks, fuzzy set technology and evolutionary computation. In this triumvirate, each of them plays an important, well-defined, and unique role"*

  Pedrycz

Broader context of Computational and Artificial Intelligence

- *"Computational intelligence comprises practical adaptation and self-organization concepts, paradigms, algorithms and implementations that enable or facilitate appropriate actions (intelligent behavior) in complex and changing environments."*

  Pedrycz

  "A methodology involving computing that exhibits an ability to learn and/or to deal with new situations, such that the system is perceived to possess one or more attributes of reason, such as generalization, discovery, association and abstraction."

  Eberhart

Broader context of Computational and Artificial Intelligence

- *"Computational Intelligence is a branch of science dealing with problems that cannot be solved using effective computational algorithms"*

  Duch

  "Computational Intelligence provides success stories that are often hard to justify with formal mathematical models (which are but a subset of all computational models, some of which are based on mathematics, and some of which are not)."

  Bezdek

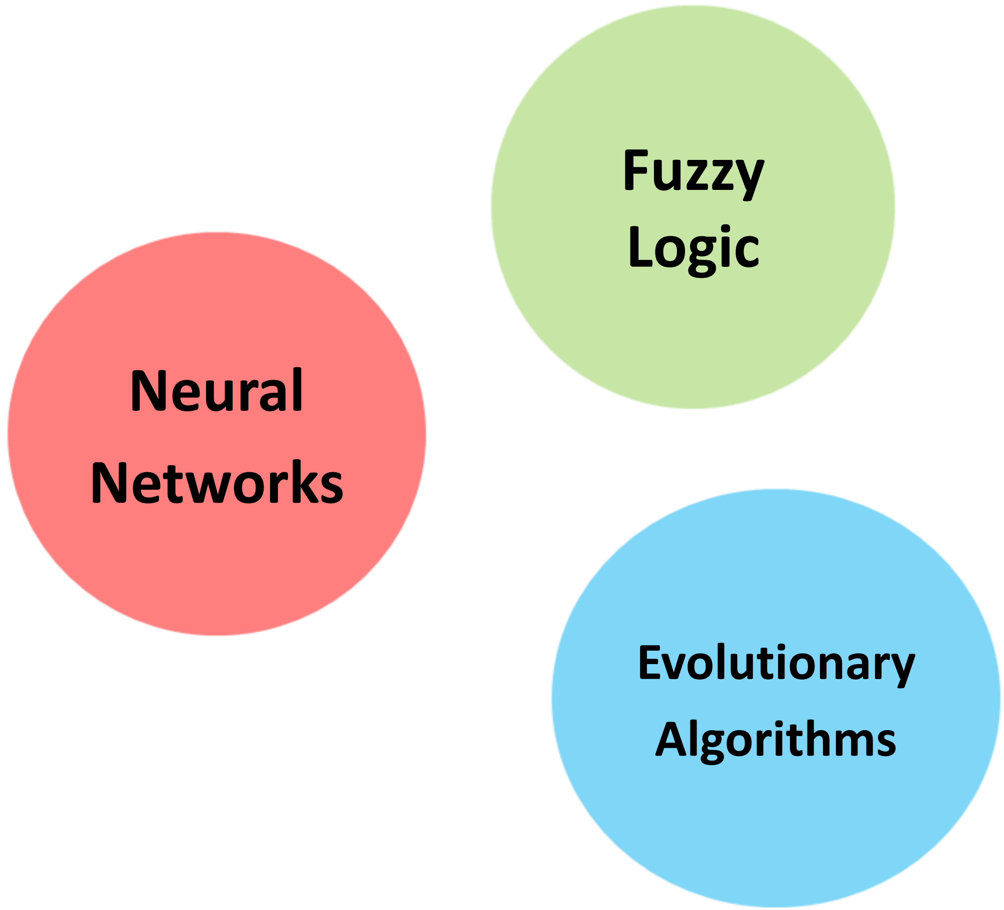
Broader context of Computational and Artificial Intelligence

- Artificial Intelligence
  - symbolic representation of knowledge
  - relies on expert systems
  - focused on higher cognitive processes like language or reasoning, whereas CI discovers knowledge hidden in data
  - nowadays, a very little overlap between AI and CI

Broader context of Computational and Artificial Intelligence

- CI and SC are sometimes considered as synonyms, but
  - increasingly CI is attributed broader meaning
  - it also encompasses a broader family of methods in the context of machine learning and pattern recognition
  - long-term vision for CI is to follow and mimic brain's problem solving capability

Fusion of methodologies



Fusion of methodologies

Neural Networks

- inspired by connectionist structure of the brain
- networks of neurons
- learning structures with potential for good generalisation (still, there are many pitfalls)
- wealth of network types
- application dependent choice of network type
- most common feed-forward multi-layer perceptron has impressive approximation power
- unsupervised learning is also possible with NN

Fusion of methodologies

Fuzzy Logic

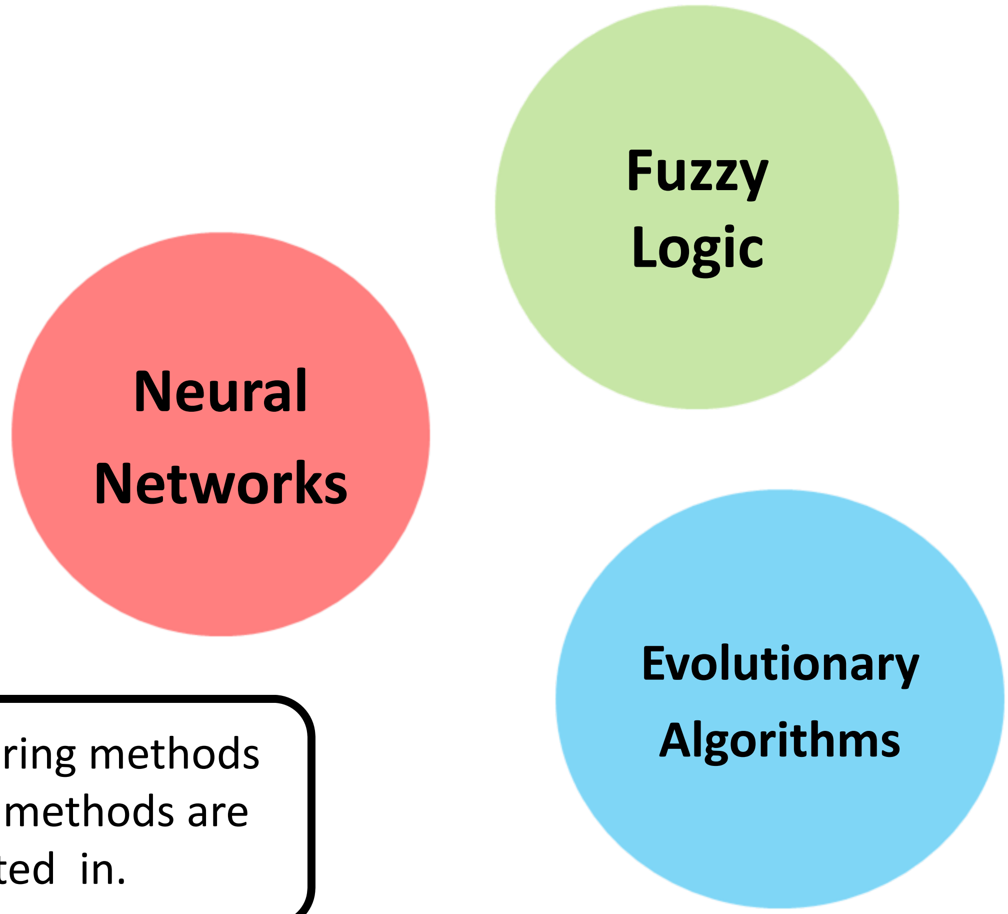
- rule-based systems
- systems that deal with reasoning
- fuzzy (non-crisp, many-valued logic) sets allow for modelling uncertain information
- rules are defined linguistically based on available expert knowledge
- optimisation can be difficult
- interpretable and intuitive systems

Fusion of methodologies

Evolutionary Algorithms

- population based method
- based on mechanisms of natural selection
- fitness driven optimisation – more fit individuals produce more offspring
- functions as a search, optimisation scheme
- important role of coding and cost definition
- most common are genetic algorithms (GAs) with genetic operators of mutation and crossover

Fusion of methodologies



Sometimes SVM, clustering methods or other fuzzy relatted methods are additionall counted in.

Synergistic effects at the core of SC

- It distinguishes SC from other data-oriented computational branches of computer science
- Complementary effects of different methods and true synergies
- Most common hybrids are:
  - Neuro-Fuzzy Systems
  - Genetic Fuzzy Systems
  - Evolutionary Neural Networks

Additional issues

- trade-off between precision and robustness, as well as between precision and simplicity
- evaluation criteria for SC systems