

Wybrane Metody Inteligencji Obliczeniowej

Wojciech Jaśkowski

Instytut Informatyki,
Politechnika Poznańska

28 lutego 2013

- ▶ dr inż. Wojciech Jaśkowski,
wojciech.jaskowski@cs.put.poznan.pl,
p. 1.6.1 (tel. 3020)

Plan ramowy

- ▶ Daty: 28.02, 7.03, 14.03, 21.03, 28.03, 4.03, 11.04, 18.04, 25.04, (2.05), 9.05, 16.05, 23.05, 6.06, 13.06 (kolokwium)
 - ▶ Elementy ćwiczeń

Inteligencja Obliczeniowa vs. Sztuczna Inteligencja I

- ▶ Ang. Computational Intelligence (CI)
- ▶ Różne opinie na ten temat:
 1. CI=AI: study of the design of **intelligent agents**. Intelligent agent is a system that acts intelligently: what it does is appropriate for its circumstances and its **goal**, it is **flexible** to changing environments and changing goals, it **learns** from experience, and it makes appropriate choices given perceptual limitations and finite computation.
 2. AI: not natural, **engineered**; strongly oriented to **symbolic representations**, and reasoning in **top-down manner**
CI: **non-symbolic**, **bottom-up** fashion
 3. CI: only **low-level numerical data**; does not knowledge in the AI sense.

Inteligencja Obliczeniowa vs. Sztuczna

Inteligencja II

4. CI: (...) adaptation is arguably the most appropriate term for what computationally intelligent systems do. *Computation intelligence and **adaptation** are synonymous*
5. Central focus of AI: **emulating human behavior** by extracting rules and knowledge from **human experts**. Majority of AI has nothing with learning (ale $ML \in AI$).
6. CI: to adapt solutions to new problems and do not rely on **explicit human knowledge**
7. CI=**soft computing**, Operations Research=**hard computing**
8. **AI**: The science of creating a non-human intelligence with machines or computers
CI: Combined fields of: Neural Computation, Evolutionary Computation, Fuzzy Computation, DNA Computing, Quantum Computing

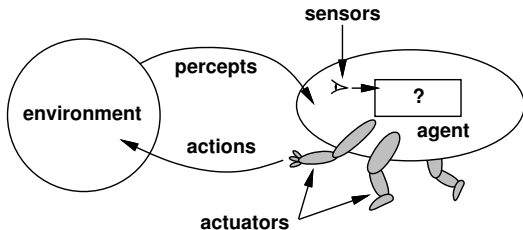
Inteligencja Obliczeniowa vs. Sztuczna Inteligencja III

- ▶ CI: Birth as a field: IEEE World Congress on CI in 1994, Orlando, Florida.
- ▶ Metody CI:
 1. Neural Computation and Neural Networks
 2. Evolutionary Computation and Genetic Algorithms
 3. Fuzzy Computation and Fuzzy Systems
 4. DNA Computing
 5. Quantum Computing
 6. Swarm Intelligence
 7. Reinforcement Learning [Mańdziuk]

Zagadnienia w ramach wykładu

1. Inteligentne agenty
2. Rachunek prawdopodobieństwa dla AI
 - ▶ Sieci bayesowskie i wnioskowanie probabilistyczne
 - ▶ Ukryty model Marcov'a
3. Planowanie w warunkach niepewności
 - ▶ Problem Decyzyjny Markov'a
4. Uczenie ze wzmocnieniem
 - ▶ Pasywne
 - ▶ Aktywne
5. Inteligencja w grach
 - ▶ Gry probabilistyczne i częściowo obserwowalne
 - ▶ Monte Carlo Tree Search
6. Problemy oparte na testach i algorytmy kooptymalizacyjne
7. Programowanie genetyczne

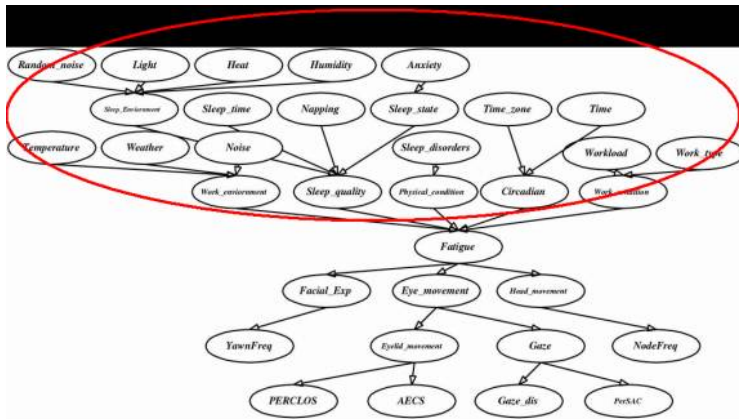
Inteligentne agenty



- ▶ Co to znaczy, że agent jest racjonalny?
- ▶ Agent żyje w środowisku — są różne środowiska

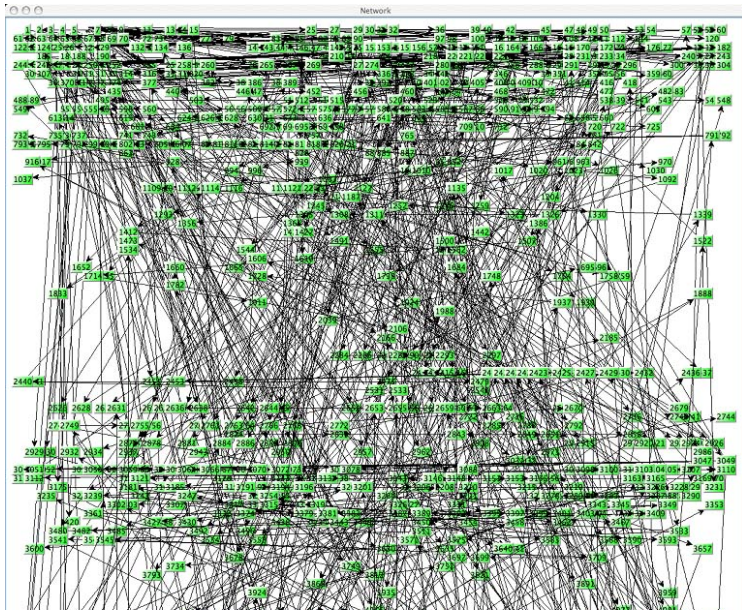
Sieci baysowskie

- ▶ Zastosowania: modelowanie wiedzy (ekspertckiej lub nauczanej z danych)
- ▶ Modelowanie zmęczenia

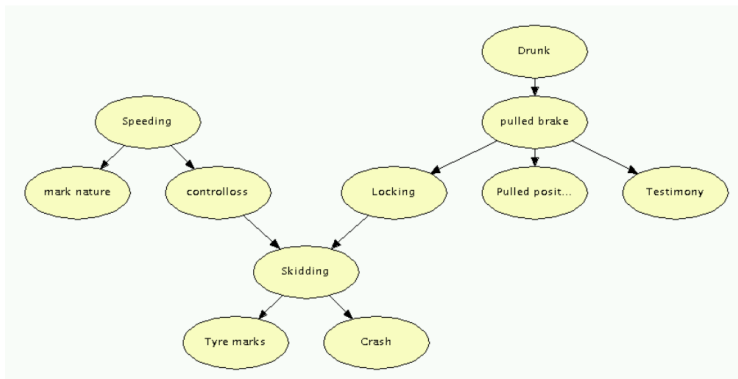


Przykład: system rekomendacyjny

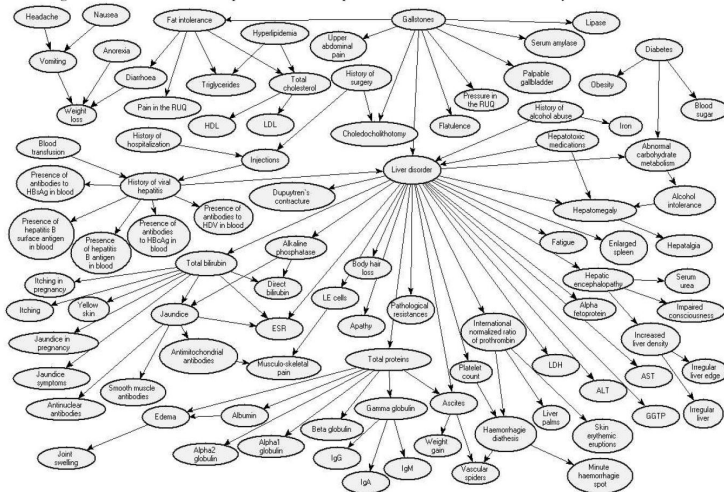
► Systemy rekomendacyjne (np. sklepy internetowe)



Przykład: Powód wypadku samochodowego

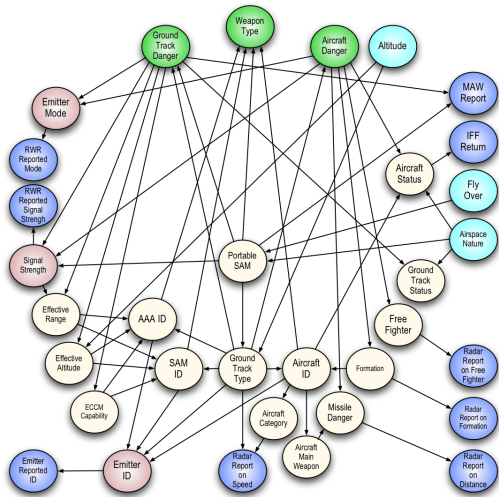


► Modelowanie chorób (np. chorób wątroby)



Przykład wojskowy

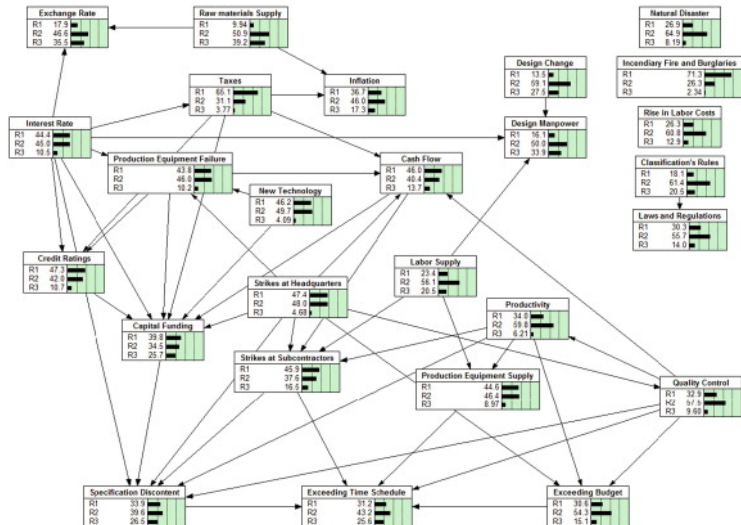
- ▶ Wsparcie dla decyzji pilota samolotu bojowego (Wise Pilot)



Modelowanie ryzyka

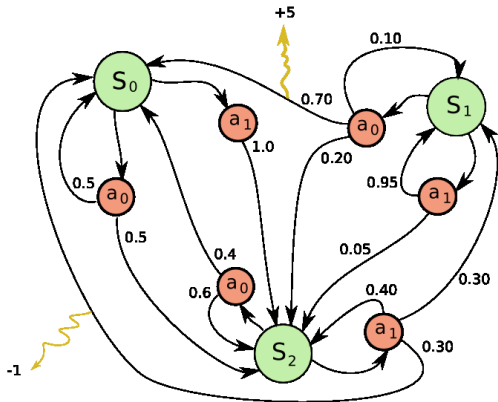
Wspomaganie decyzji:

- ▶ Ryzyko kredytowe
- ▶ Ryzyko projektu



Procesy decyzyjne Markov'a

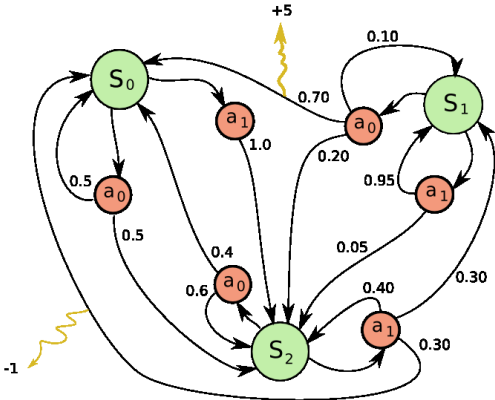
- ▶ Model matematyczny problemu: decyzje, element niedeterministyczny



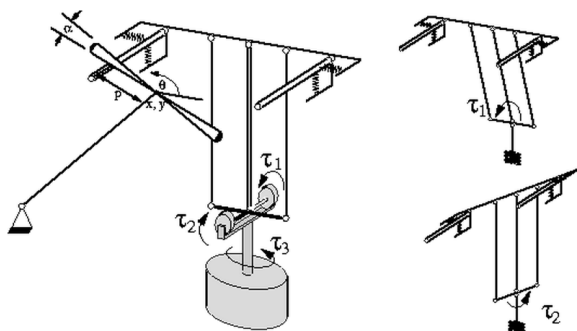
- ▶ Jeśli stan agenta nieznan, to POMDP

Uczenie ze wzmocnieniem

- ▶ Jeśli prawd. albo nagrody są nieznane



Przykład: Devil-sticking robot



- ▶ Devil Stick (pol. diabelski kijek)
- ▶ 6-wymiarowa przestrzeń stanów, tylko 40 prób uczących (lepiej niż człowiek)

Autonomous Helicopter via Reinforcement Learning [Ng, et al.]

- ▶ <http://www.youtube.com/watch?v=kN6ifrqwIMY>
- ▶ <http://www.youtube.com/watch?v=VCdxqn0fcnE>

TD-Gammon [Tesauro]

- ▶ Tryktrak (Backgammon)
- ▶ Uczenie poprzez grę z samym sobą (300tys-1.5mln rozgrywek)
- ▶ TD-Gammon gra na poziomie mistrzowskim = ζ Rozwój uczenia ze wzmocnieniem
- ▶ Metoda różnic czasowych (Temporal Difference (TD))
- ▶ Sieć neuronowa



Figure 3. A complex situation where TD-Gammon's positional judgment is apparently superior to traditional expert thinking. White is to play 4-4. The obvious human play is 8-4*, 8-4, 11-7, 11-7. (The asterisk denotes that an opponent checker has been hit.) However, TD-Gammon's choice is the surprising 8-4*, 8-4, 21-17, 21-17! TD-Gammon's analysis of the two plays is given in Table 3.

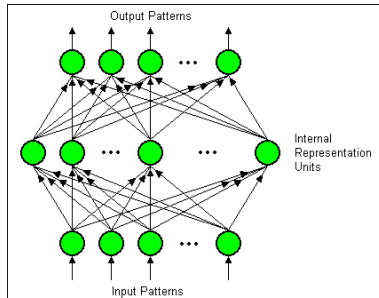
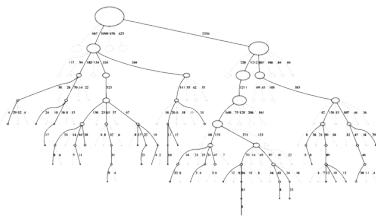
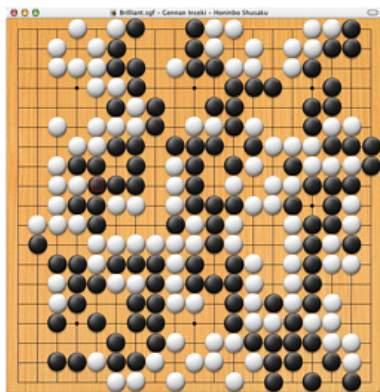


Figure 1. An illustration of the multilayer perceptron architecture used in TD-Gammon's neural network. This architecture is also used in the popular backpropagation learning procedure. Figure reproduced from [9].

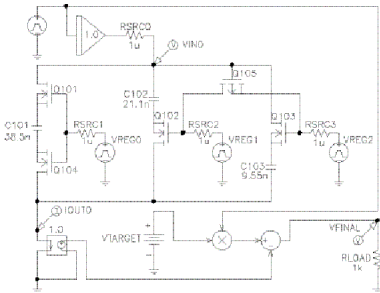
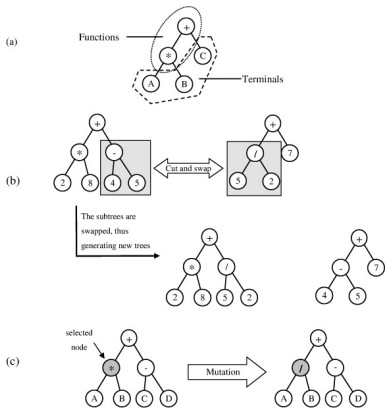
- ▶ Ukryty model Markov'a (prosta dynamiczna sieć bayesowska)
- ▶ <http://www.youtube.com/watch?v=ABzzFMzFE3Y>
- ▶ <http://robots.stanford.edu/movies/sca80a0.avi>

Monte Carlo Tree Search (MCTS)

- ▶ Dla problemów planowania
- ▶ Gra w Go
- ▶ Algorytmy bazujące na MCTS osiągają siłę kilku danów.



► Ewolucja układów bramek logicznych [Koza]



Naprawianie błędów w kodzie

► Automatyczne naprawianie błędów w programach w C [Weimer et al.]

```
1 void zunebug(int days) {
2     int year = 1980;
3     while (days > 365) {
4         if (isLeapYear(year)){
5             if (days > 366) {
6                 days -= 366;
7                 year += 1;
8             }
9             else {
10                }
11        }
12        else {
13            days -= 365;
14            year += 1;
15        }
16    }
17    printf("current year is %d\n", year);
18 }
```

```
1 void zunebug_repair(int days) {
2     int year = 1980;
3     while (days > 365) {
4         if (isLeapYear(year)){
5             if (days > 366) {
6                 // days -= 366; // repair deletes
7                 year += 1;
8             }
9             else {
10                }
11            days -= 366; // repair inserts
12        } else {
13            days -= 365;
14            year += 1;
15        }
16    }
17    printf("current year is %d\n", year);
18 }
```


Ewolucja Anteny dla NASA - programowanie Genetyczne [Lohn et al.]

- ▶ Wyewoluowana za pomocą programowania genetycznego.
- ▶ Użyta w statku Space Technology 5 (ST5)
- ▶ Jakość porównywalna z anteną zaprojektowaną przez ludzi.

