Robot-sumo

Bc. Jan Tomášek
Bc. Štěpán Havránek

Outline

Bc. Jan Tomášek

- Sumo robot rules
    - why mini sumo
- Hw design
    - AI approach
    - engineering approach
    - our robot

Bc. Štěpán Havránek

- Software architecture
    - Problem description
    - Simple/practical
    - Complex/theoretical
    - Problems
    - Out of the box thinking

Robot-sumo rules

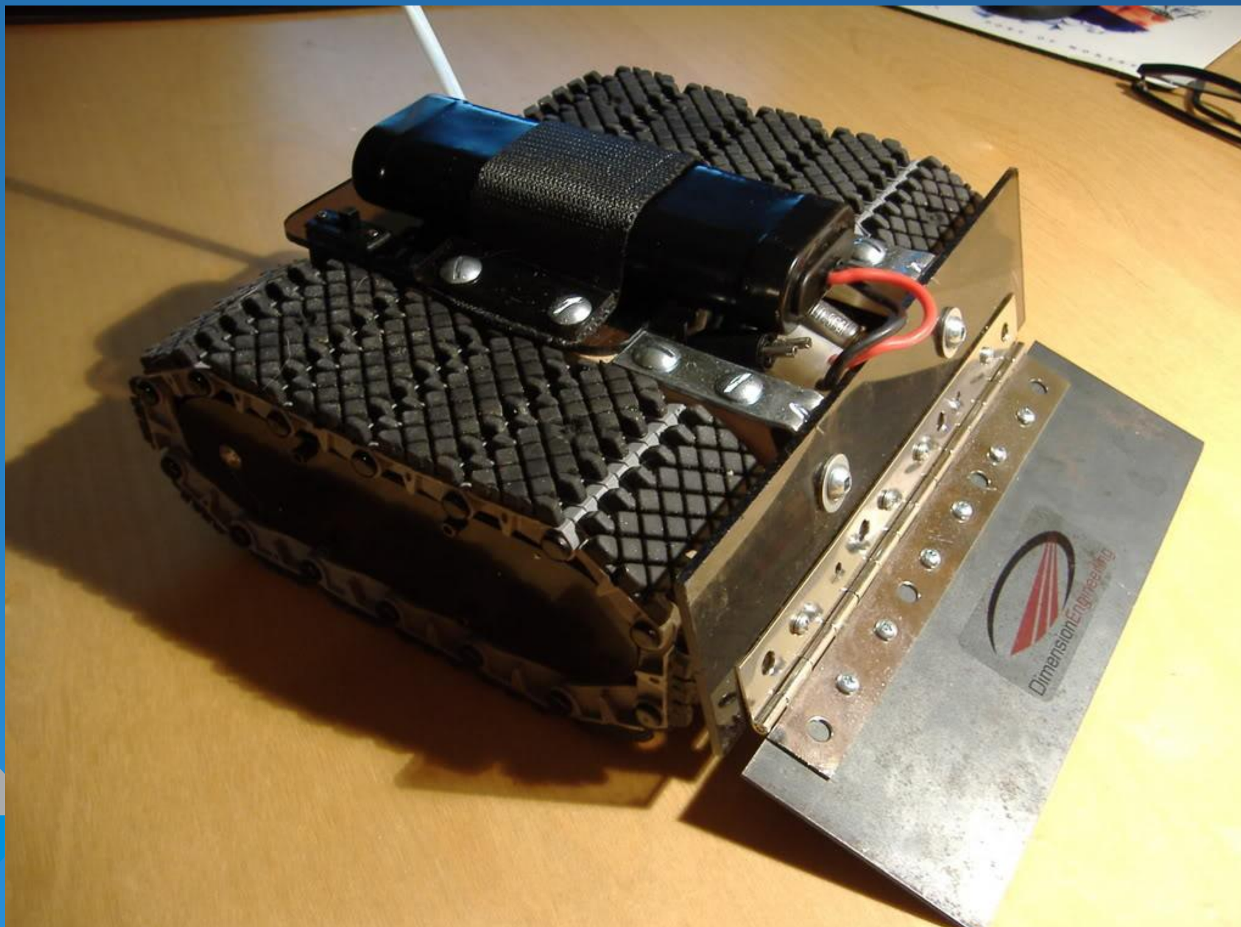
Robot-sumo is a sport in which two robots attempt to push each other out of a circle (in a similar fashion to the sport of sumo). The robots used in this competition are called sumobots.

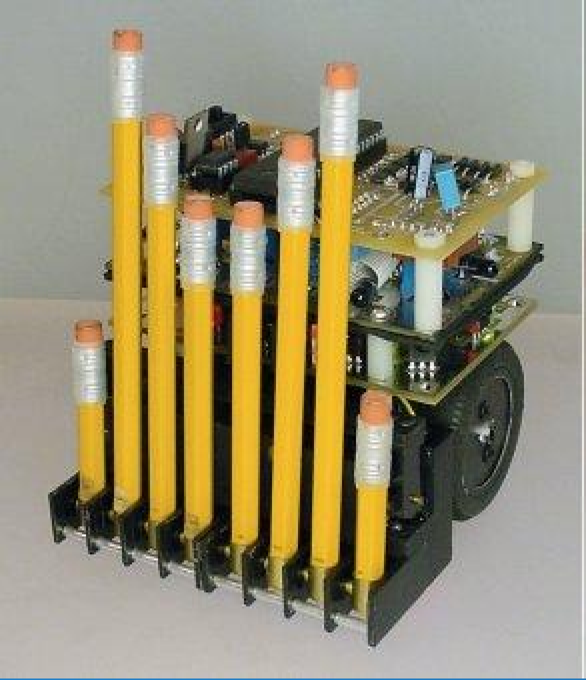
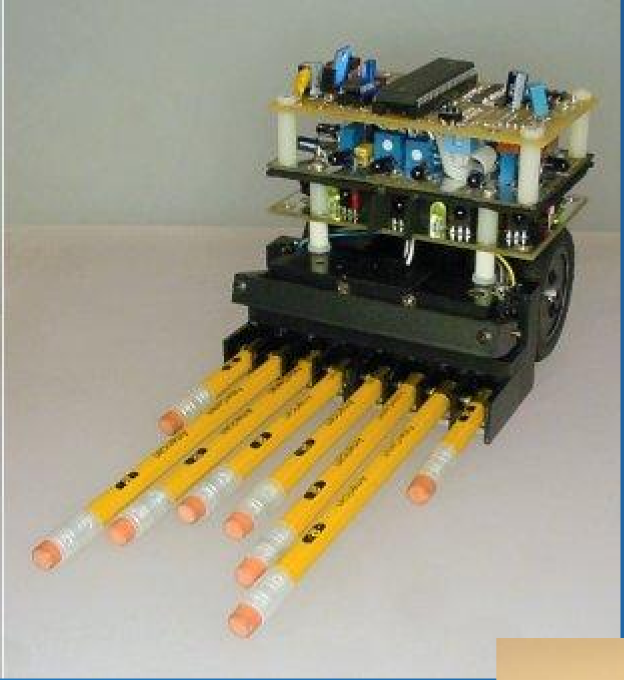
Robot-sumo categories

- Heavy-weight - ~56 kg, ~61cm
- Light-weight - ~23 kg, ~61cm
- Mini-sumo - 500g, 10cm
- Micro-sumo - 100g, 5cm
- Nano-sumo - 2.5 cm
- Femto-sumo - 1cm

Heavy-weight sumo

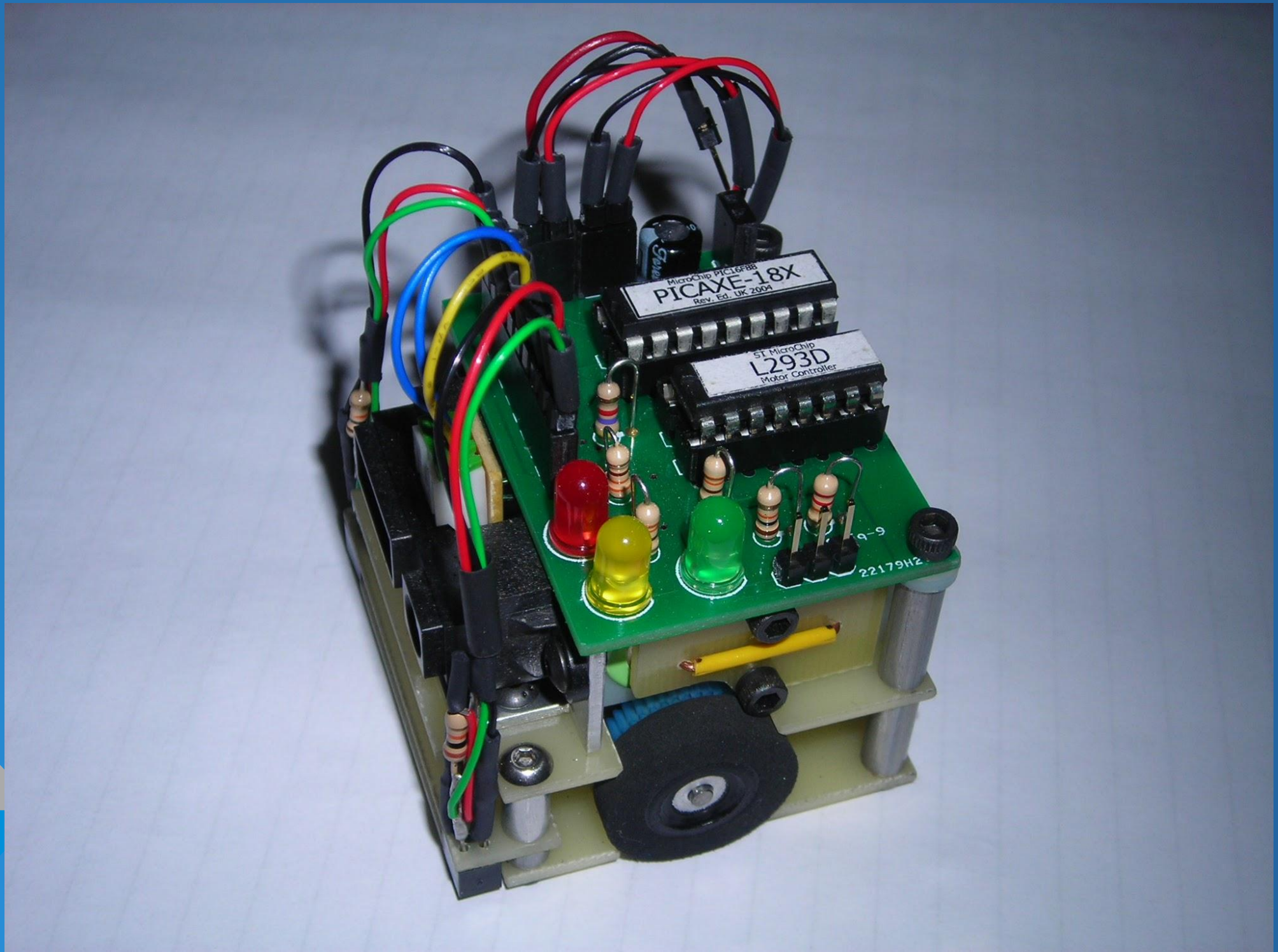
Light-weight sumo



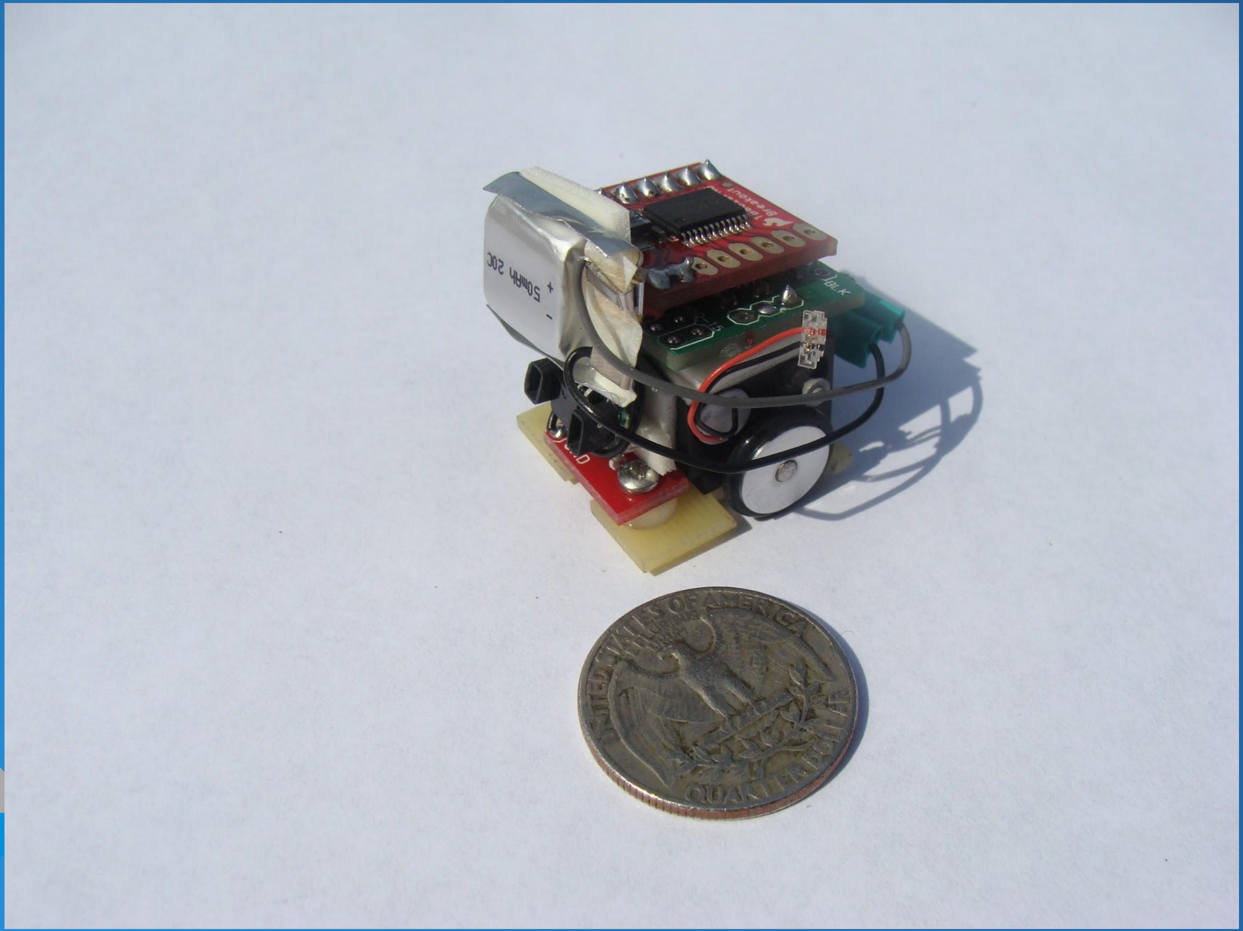



Mini sumo

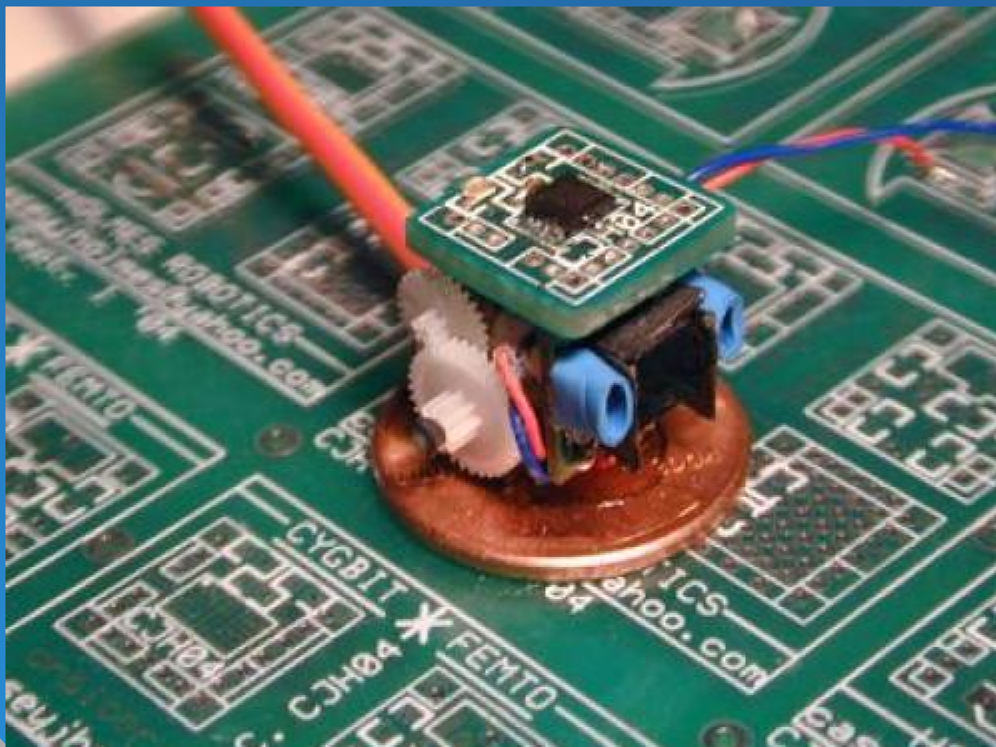
Micro sumo


MicroChip PIC16F88
PICAXE-18X
Rev. Ed. UK 2004
ST MicroChip
L293D
Motor Controller

Nano sumo



Femto sumo



Mini sumo

- up to 500g
- dimensions
  - 10 x 10 cm, unlimited height
  - on start
- can not destroy another robot
- ring
  - diametral 77 cm
- no adhesion boosting allowed
- splitting allowed

Hw construction

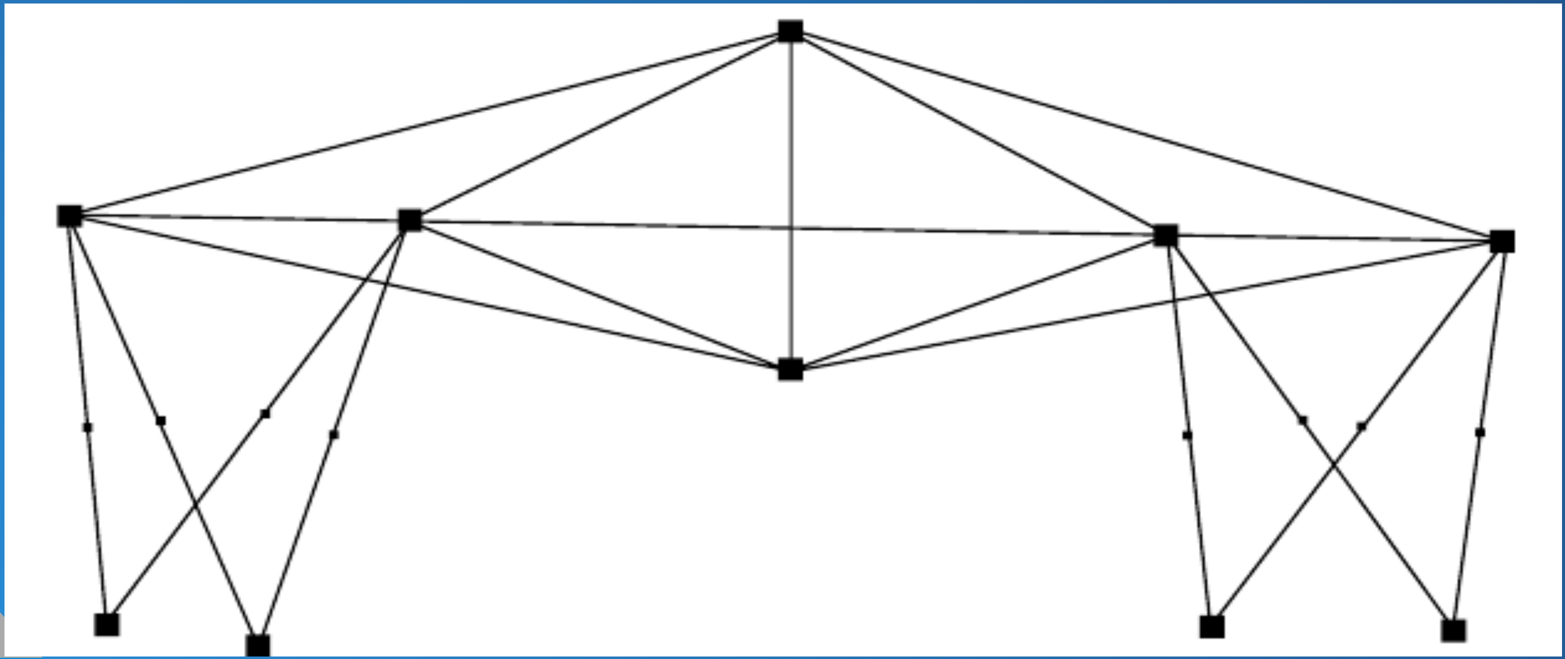
- Artificial intelligence
  - evolution algorithms
    - spring models
    - box models
  - why we can’t use AI in construction
  - our robot

Spring model

- inspired by biological muscles
- organism consists of linked set of node masses
- to provide movement some sprigs are representing a muscles
  - drives in simple harmonic motion

Spring model examples

Daintywalker the SodaRace mascot



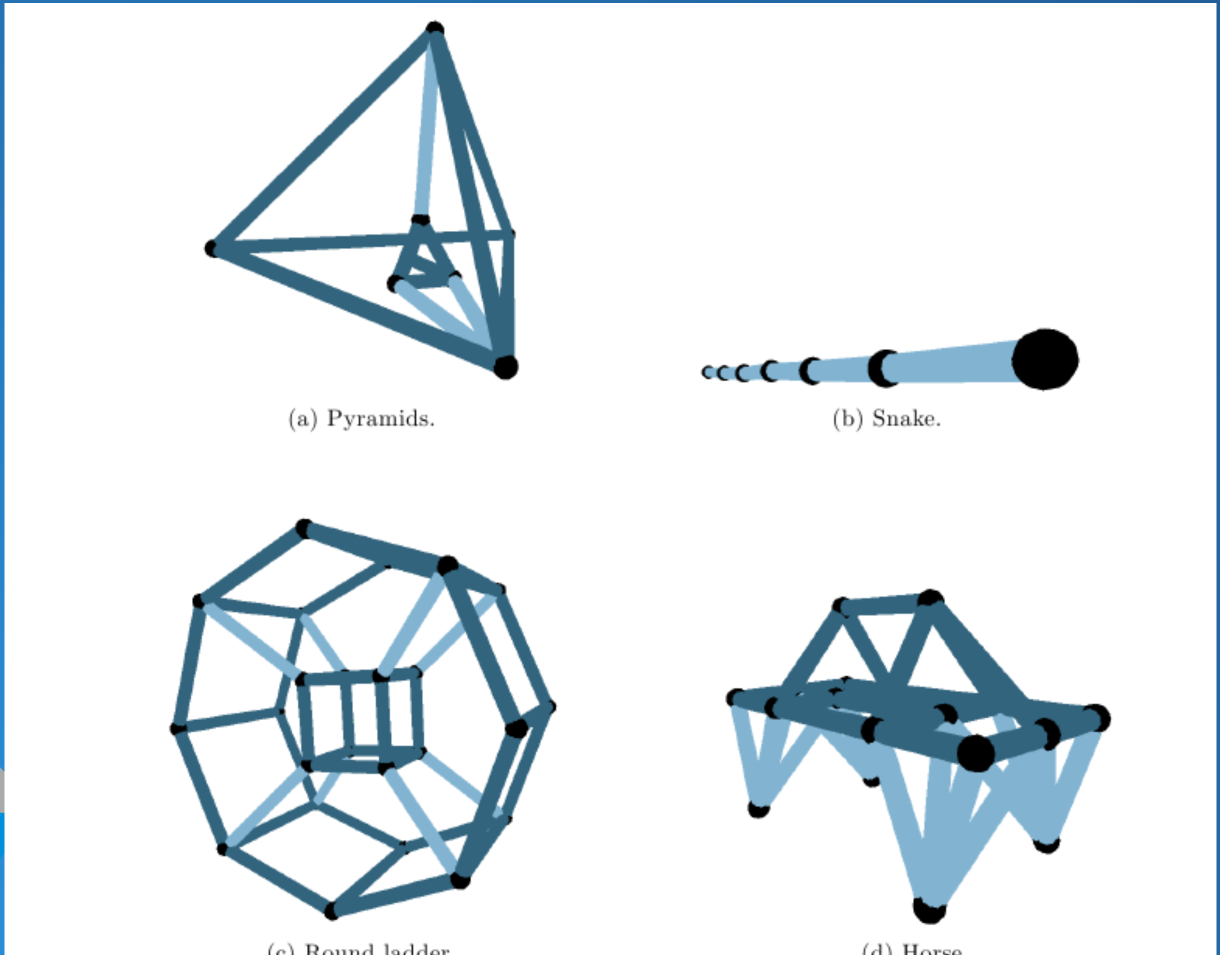
Sodaplay

- game engine
  - human designed vs. genetic designed spring robots
- manual robot/races creation
  - easy sharing between participants
- problem of this method
  - extreme overfit
  - need lot of different races

3D springs

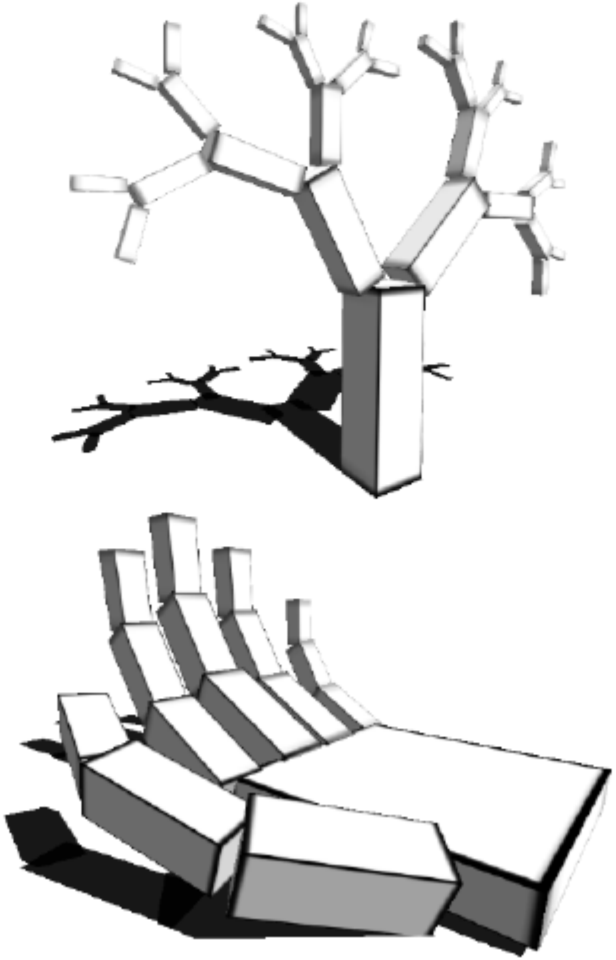
- this model can be generalized to 3D
- Marcel Krčah in his thesis “Evolution of Springy Organisms”

3D models examples



another genetic model

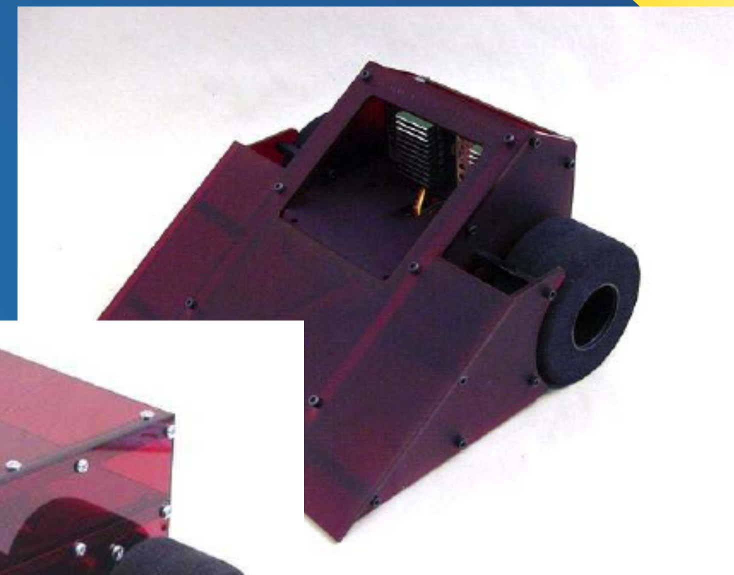
- presented on seminar last year
- boxes and joints

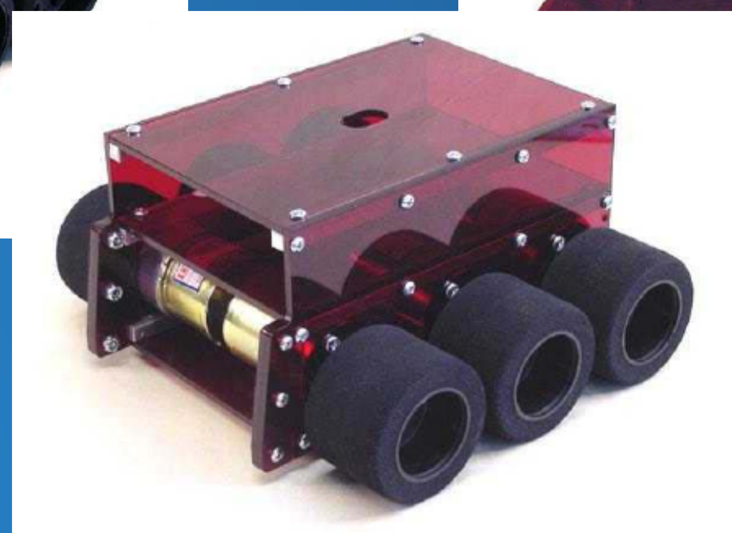


Genetic algorithms for sumo

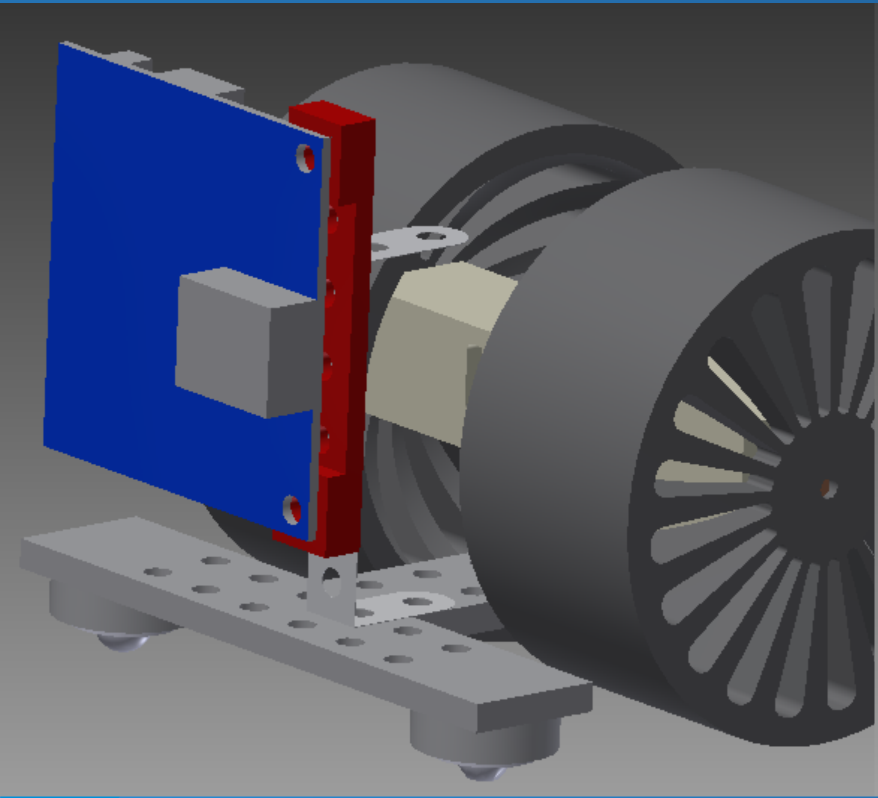
- none of previous models consider what can be made (by 3D printer)
- simulation
  - very hard physical model
    - friction, adhesion
  - opponent simulation
  - “crash test”
    - best friction, best adhesion, lowest battery consumption
    - which is important ?!

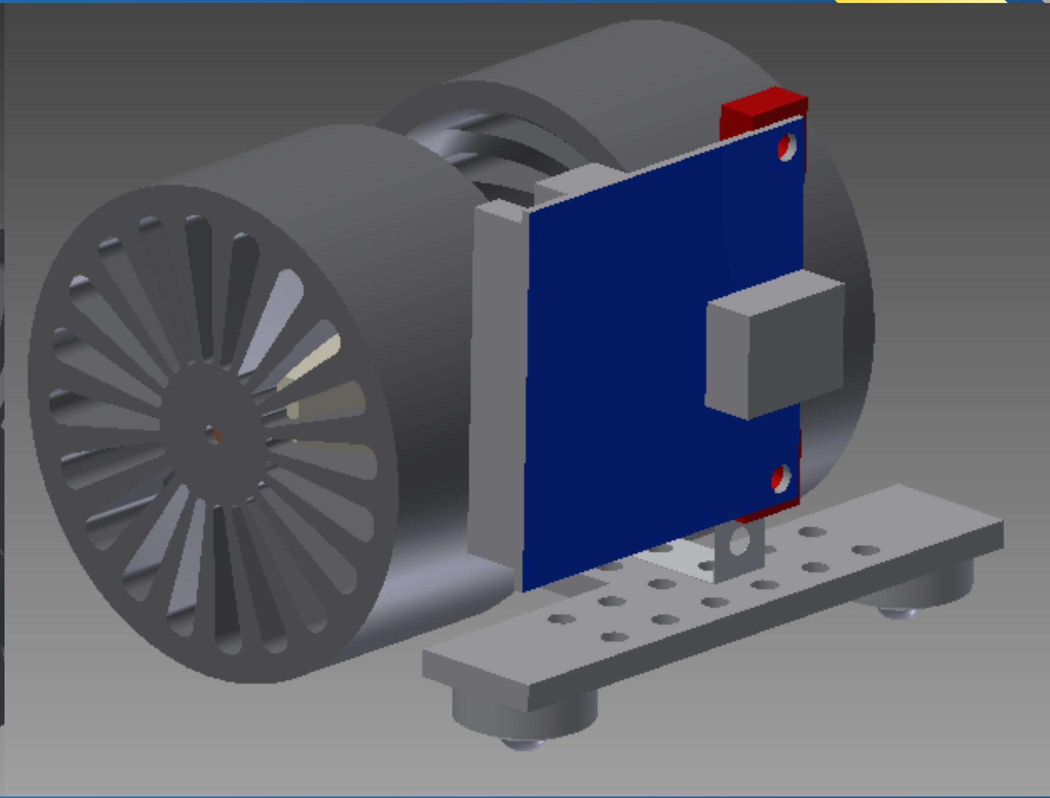
Engineering solution





Our robot





Robot’s software

- Inputs
  - Sensors
    - Numeric values
- Outputs
  - Motors
  - Other devices
- Autonomous embedded computer

Software architecture

- Straight techniques
  vs.
- "Clever" / theoretical approaches

Software architecture

- Hardcoding
    - Problem-specific
    - Fast
        - => Lower HW requirements
            - Often the only possible solution
    - System If-Then rules
        - Infinite loop
        - Locks

Software architecture

- Subproblems organization
  - => Abstraction layers
    - Input
    - Output
    - Logic
      - Model
        - States
        - Process workflow

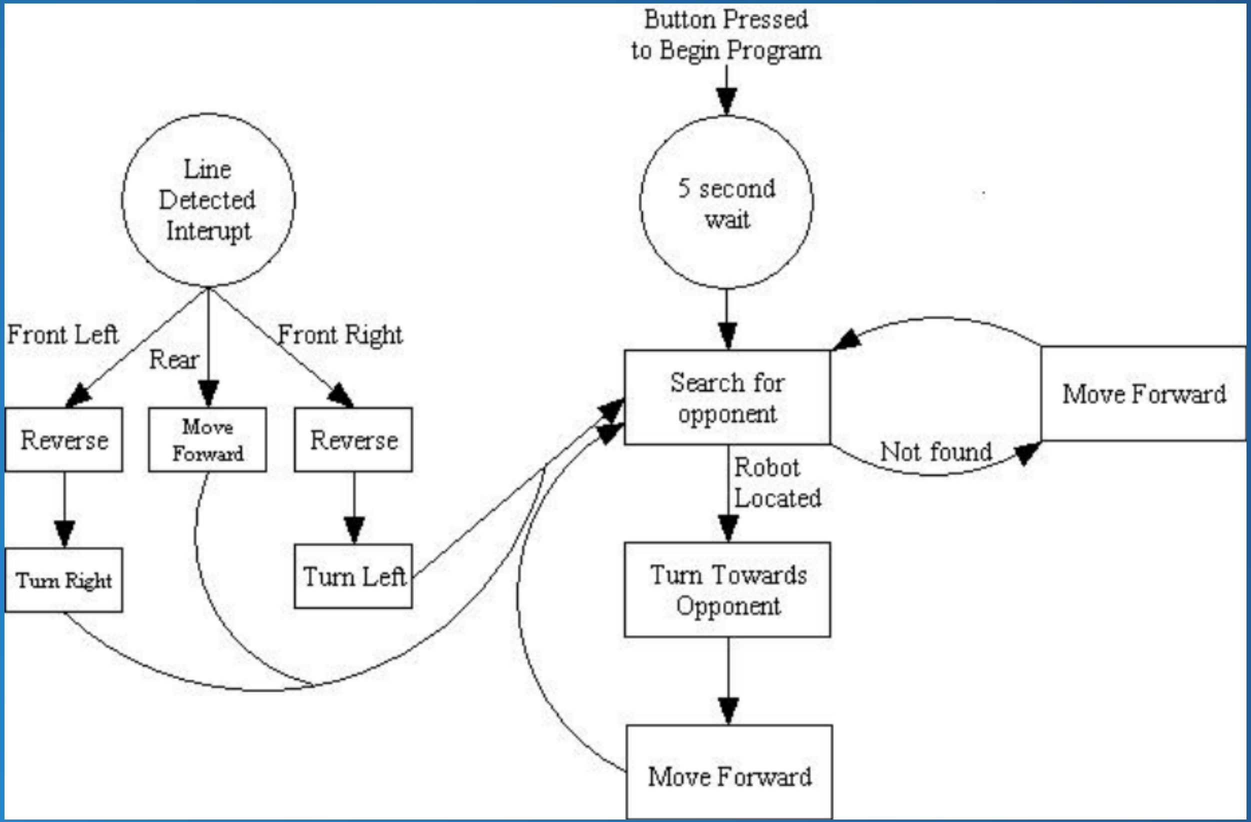
Layered model

- System Action-Interruption
  - On all the layers
  - HFSM
  - Extensions
    - Priority
    - Dependences
    - Probability

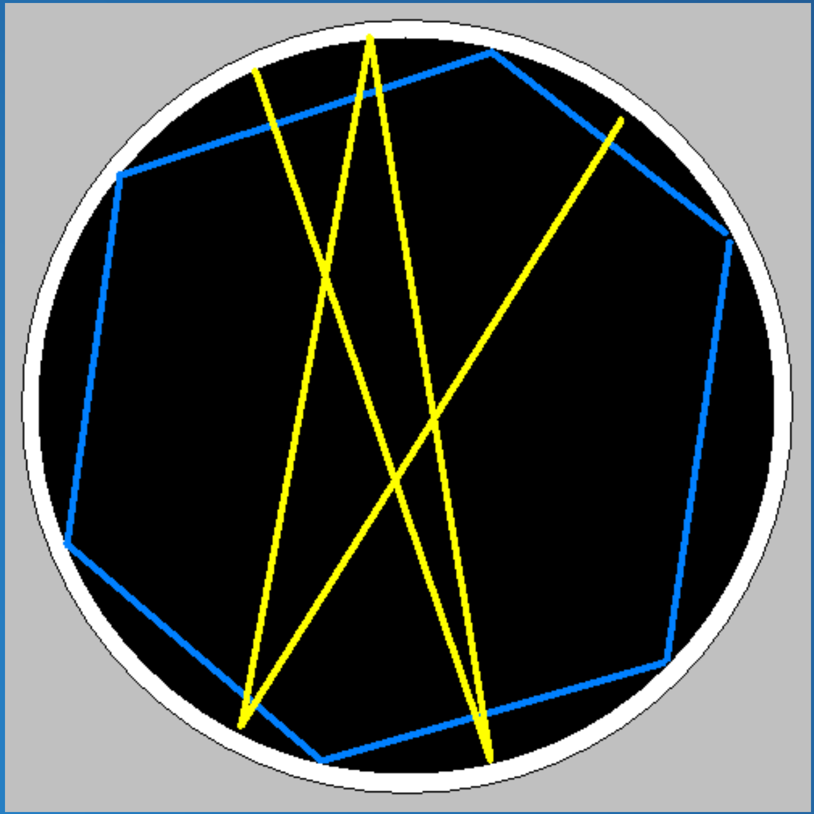
Strategies

- Simple automata
  - Offensive
    - Search and attack
  - Defensive
    - Search and run away
  - Random movement
    - Without fight

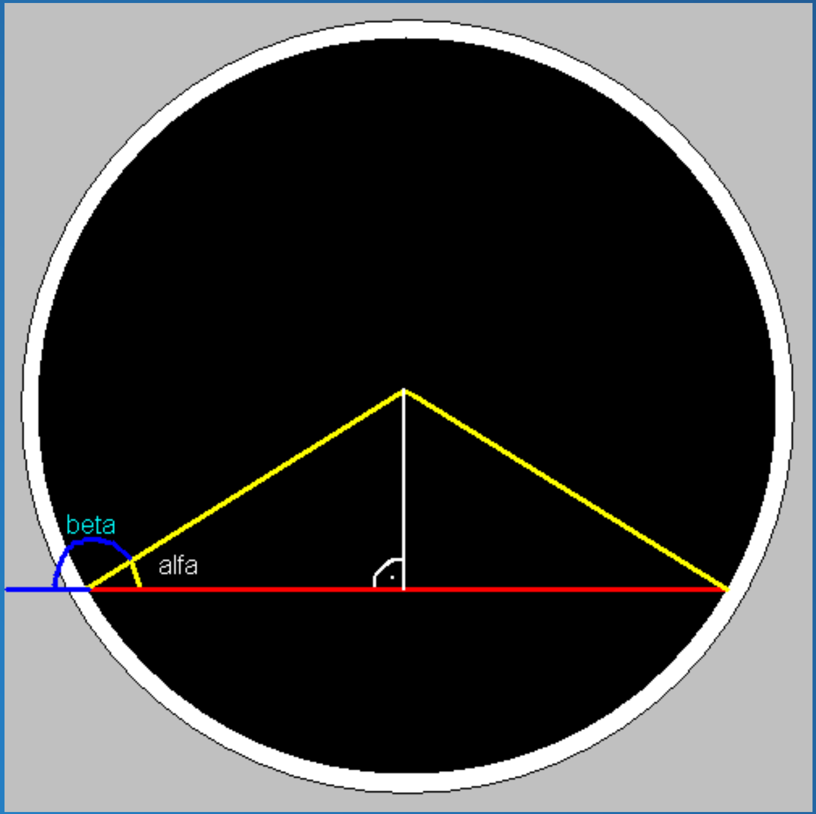
Search and attack strategy



Simple strategies - keep inside



Simple strategies - keep inside



Strategies

- Combination of different strategies
  - Superior decision automata
- Artificial intelligence techniques
  - High level of abstraction

AI approach

- Environment
  - Partially observable
  - Stochastic behaviour
  - Sequential process
  - Dynamic surroundings
  - Continuous
  - Multi-agent space

AI approach

- Searching
  - What?
  - Where?
  - How to use the results?

AI approach

- Classical way
  - Searched space = possible actions
    - According to input
  - Searching for highest reward
    - Action sequence ending with victory
  - Do the result action sequence
  - => Planning
    - Problem solved?

AI approach

- Planning
  - Ignores future environment states
- Another AI/TI techniques
  - Evolution algorithms
  - Neural networks

AI Evolution approach

- Cultivating the strategy
- Base population
  - Presented simple strategies
- “The genes”
  - Quantitative values
- Fit function
  - Selection naturally made by the fights

AI Evolution approach - concrete

- Population
  - Strategy programs in If-Then form
    - Expressions converted to numeric values (quantitative)
    - Added the rest unused combinations as multiplied by zero
    - Ex.
      - IF 1*FrontOptSensor THEN MoveFront(0)
      - IF 1*FrontOptSensor THEN MoveBack(1)

AI Evolution approach - concrete

- Mutation
  - Small value changes

AI NN approach

- We don’t know the right results of the neural network
- We want the right synaptic weights
- Basically the same as the evolution approach

Q & A

Sources

- Genetic robot design
  - http://artax.karlin.mff.cuni.cz/~krcap1am/ero/doc/krcah-ices08.pdf
  - Evolution of Springy Organisms by Marcel Krčah
- Sumo Robot wiki
  - http://en.wikipedia.org/wiki/Robot-sumo
- Soda race
  - http://sodaplay.com/creators/soda/items/race
- femo sumo on youtube
  - http://www.youtube.com/watch?v=n2Ip7rOKB0c
- Human-like and Animal-like Agents
  - http://artemis.ms.mff.cuni.cz/main/tiki-index.php?page=A%20lecture%20on%20Human-like%20and%20Animal-like%20Agents
- JSumo
  - http://jsumo.com
- Robodoupě
  - http://robodoupe.cz
- Jizhong Xiao
  - http://www-ee.ccny.cuny.edu/www/web/jxiao/

Bonus

- http://www.youtube.com/watch?v=n2lp7rOKB0c
- http://www.youtube.com/watch?v=iL8IRF4wQmU