

Courses on Robotics by Guest Lecturing at Balkan Countries

Hans-Dieter Burkhard
Humboldt University Berlin

With Great Thanks to all participating
student teams and their institutes!

Courses on Balkan Countries (*2013-2016 founded by our program*)

- 2012 Ohrid (funded as DAAD INTENSIVE COURSE
"Robotics and Mathematics" together with Nevena Ackovska)
- 2013 *Novi Sad*
Rijeka
Sarajevo
- 2014 *Plovdiv*
Rijeka
- 2015 *Skopje*
Sarajevo
- 2016 *Rijeka*
Tirana
Plovdiv: knowledge transfer to teaching staff
- 2017 *Rijeka*
- 2018 planned: *Rijeka* (funded by Erasmus)

2017 Translation to Bulgarian Language

Courses in

- Plovdiv
- Burgas

Further courses:

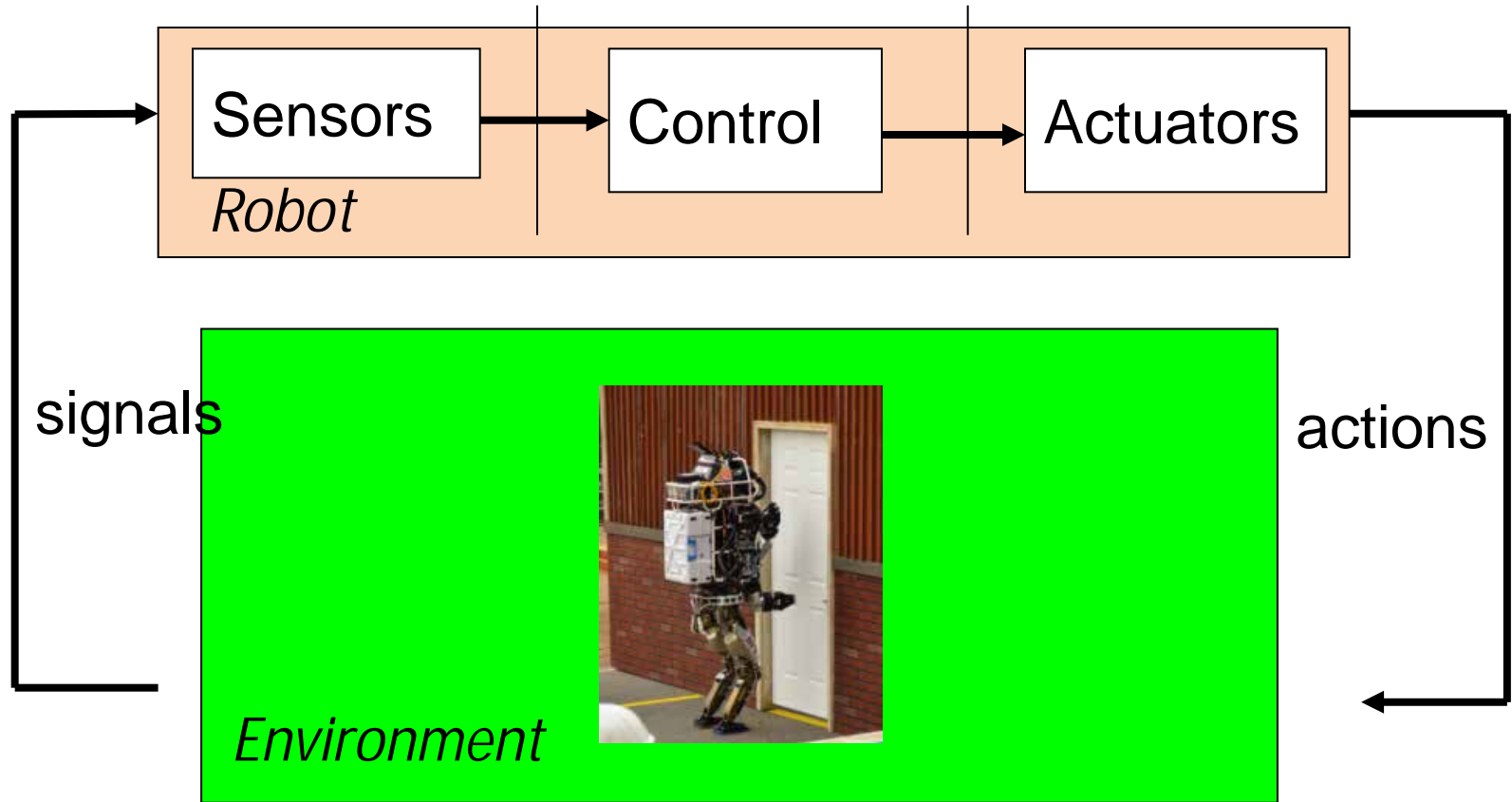
Poland:

- Vistula University Warsaw

Germany

- Humboldt University Berlin
- Anna-Seghers-Schule Berlin

Robot in the Real World



Robotics is an Integrative Task

- Software: Perception, Motion, Control, Communication, ...
- Hardware: Sensors, Actuators, Processors, Energy, ...
- Informatics, Artificial Intelligence, Physics, Mathematics, ...
- Electronics, Mechanics, Materials, Design, Engineering, ...
- Biology, Medicine, Sports, ...
- Psychology, Philosophy, Sociology, ...

Typical Duration of a course: 30 hours.

Up to 30 participants.

Lectures and exercises are mixed.

Topics of lectures:

- International Competitions (DARPA, RoboCup)
- Motion
- Sensors/Perception/World Models
- Behavior Control

International Competitions: Illustration and exercises

DARPA Challenges

Autonomous Cars (2004-2007)



Robot Challenge

Desaster scenarios (2012-2015)



DARPA Robotocs Challenge

The robot

1. drives down an obstacle course,
2. dismounts the vehicle,
3. opens and goes through the door,
4. finds and closes a valve,
5. chooses a tool and carves out a hole,
6. solves the "surprise task" (e.g. plug a switch),
7. walks or climbs over some rubble,
8. climbs the outside stairs.



RoboCup: Soccer playing Robots as Testbed for Robotics and AI

Perception:

- Where are the ball, the goals, other players
- Where am I
- What are other players doing

Control:

- What should I do?
- Attacking, Defending, Supporting, ...
- Go to ball, Kick the ball (to which direction?)

Motion:

- Walk forward/sideward/backward, Turn, StandUp
- Kick, Catch, ...



Exercises with Simulated Soccer Robots

Support for Programming by **RoboNewbie**

Diploma Thesis by Monika Domanska
at Humboldt University 2012

- Framework based on Java and Netbeans.
- Hides non-robotics aspects
(e.g. communication with soccer simulation).
- Basic motion and perception.



Download of all programs and materials from our website
<http://www.naoteamhumboldt.de/projects/robonewbie/>



Berlin United - Nao Team Humboldt

Artificial Intelligence - Humboldt University Berlin

[Team](#) [Projects](#) [Publications](#) [RoboCup](#) [Events](#) [Videos](#) [Images](#) [Sponsors](#) [Contact](#)

RoboNewbie

RoboNewbie is a basic framework for experiments with simulated humanoid robots. It provides interfaces to the simulated sensors and effectors of the robot, and a simple control structure. The framework and the examples are implemented in JAVA with detailed documentations and explanations. That makes it useful even for beginners in Robotics.

RoboNewbie runs in the soccer simulation environment of the official RoboCup 3D simulator. The simulated soccer players are models of the Humanoid Robot NAO of the French Company Aldebaran. Besides other examples, a simple soccer playing robot demonstrates the architecture and the features. Users are encouraged to extend it by different means.

Thanks are due to the RoboCup community for continuous help and inspiration.

Last updated of project files: June, 14th, 2013.

Resources for Download

- [Installation](#)
- [How to start](#)
- [RoboNewbie_1.0](#) – the framework and example programs. It is programmed in JAVA 7 and prepared for use under Netbeans. The "QuickStartTutorial" gives an introduction to the features and the usage of RoboNewbie.
- The [SimSpark RoboCup 3D Soccer Simulation \(SimSpark RCSS\)](#)-Version r300 for Windows is configured for RoboNewbie. SimSpark RCSS was developed by the RoboCup Soccer Server Maintenance Group. A short overview is given by "[SimSpark/SoccerServer RCSS as used for RoboNewbie](#)", the detailed information can be found on the [SimSpark Wiki](#).
- The [MotionEditor](#) can be used for the design of motions. Installation and usage are described by the "[MotionEditor Tutorial](#)". To use the motion editor you need [JAVA 3D Version 1.5.1](#) on your computer.



English
Deutsch

Lab: RUD 25, 3.110

+49 (30) 2093 3811
nao-team (at) informatik.hu-berlin.de

Lab-meeting: Monday 13:00 – 15:00

Search for:

RECENT POSTS

- [RoboCup 2013 Links and Livestream](#)
- [SPL Qualifikations Video 2013](#)

NAO'S CALENDAR

Termine werden eingestellt ab
21.8. [Frühere Termine suchen](#)
Termine werden eingestellt bis
30.9. [Weitere Termine suchen](#)

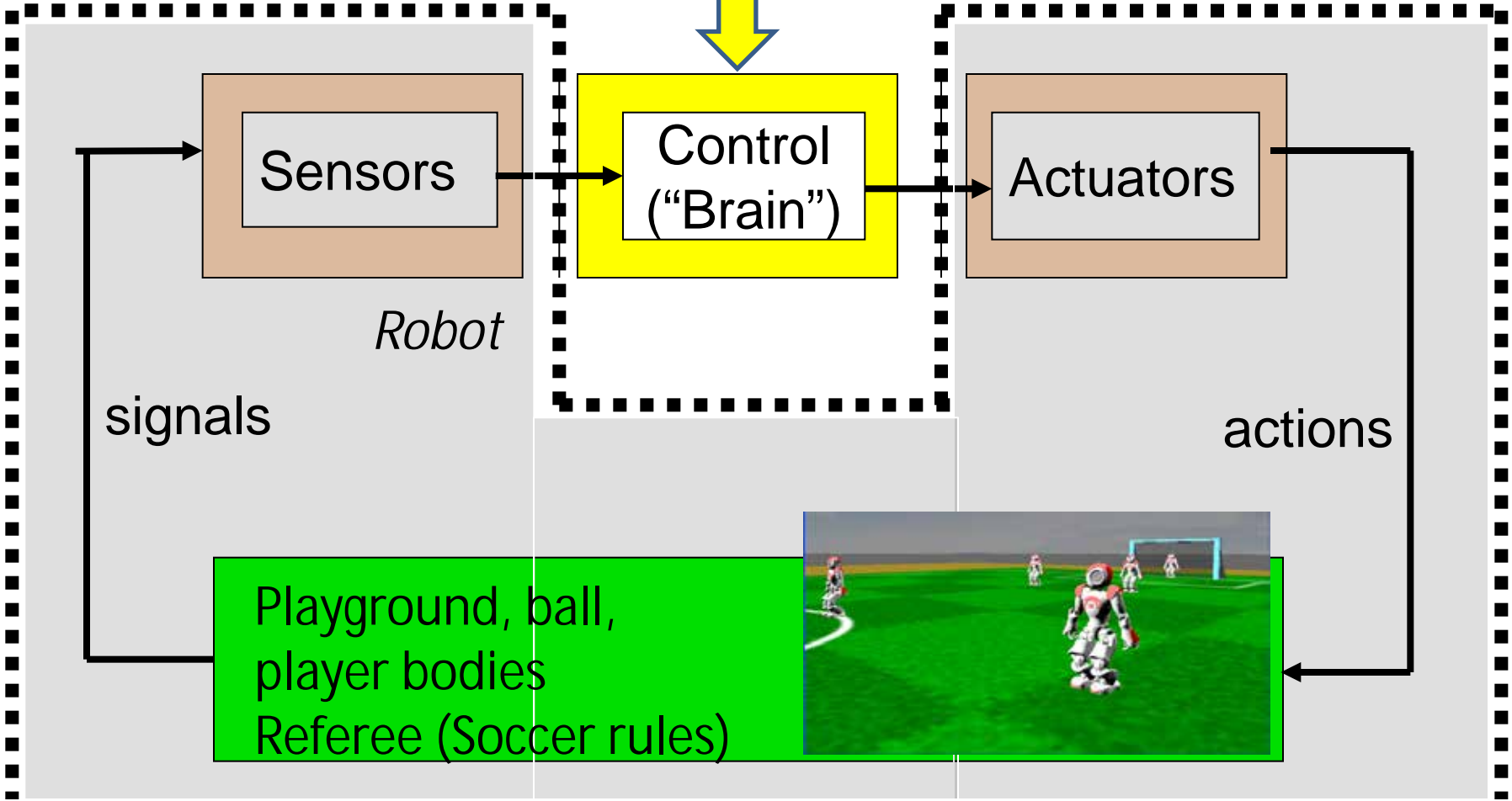
[Login](#) [Logout](#)

[Übersicht](#)

RECENT LINKS

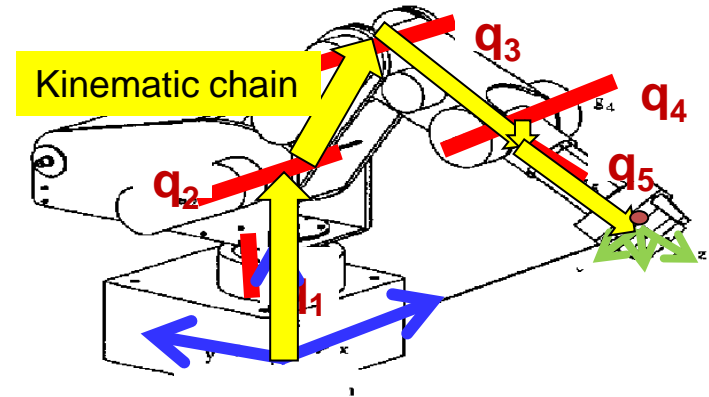
- [German National RoboCup Committee](#)

RoboNewbie + Student programs

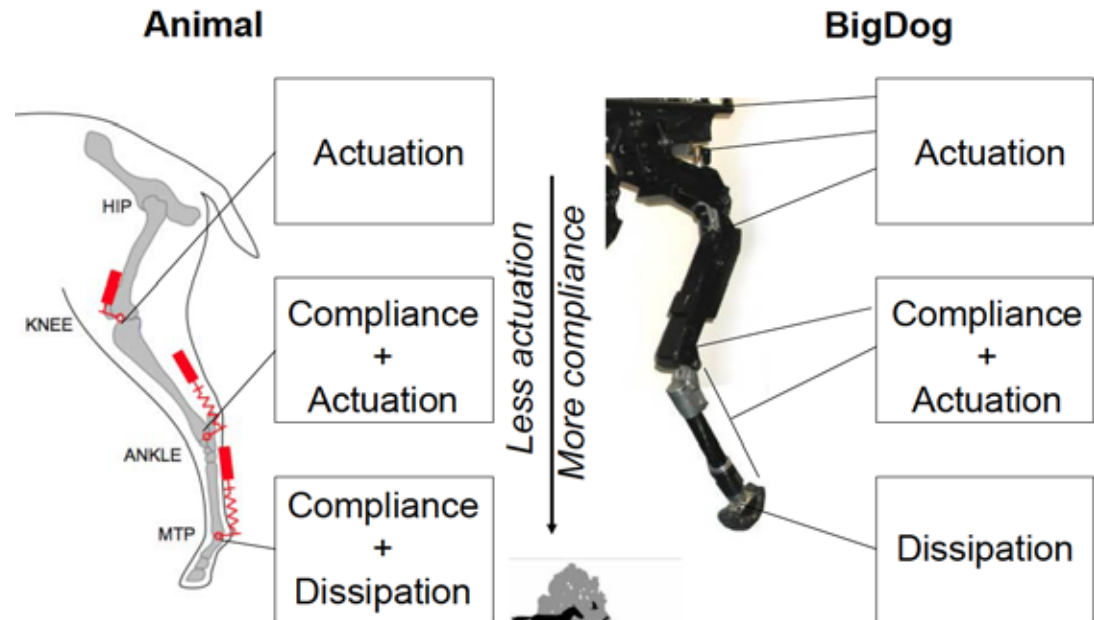


Motion

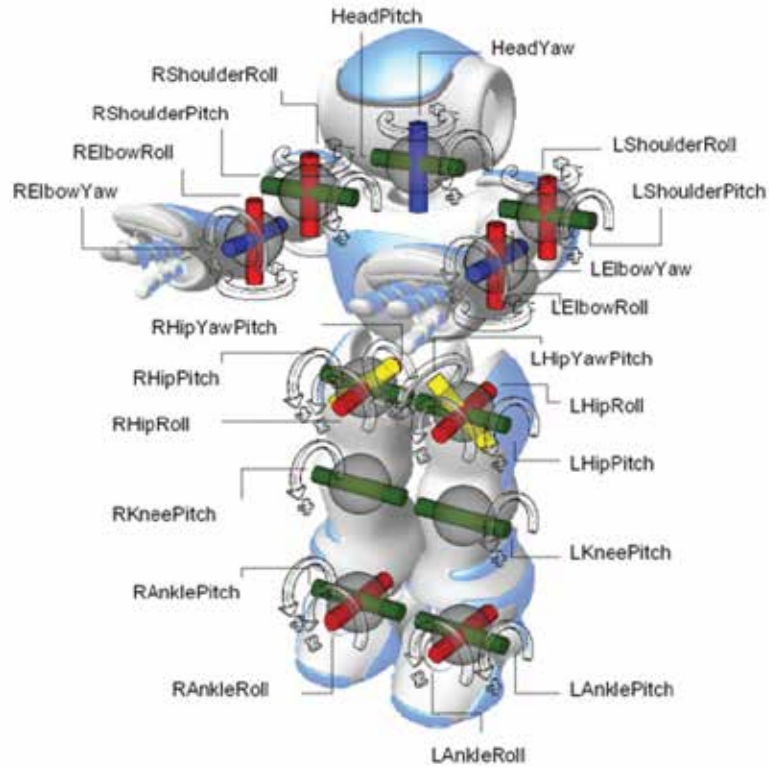
- Kinematics, Drive Systems,
- Legged Robots,
- Motion Planning and Control,
- Learning,
- Biologically Inspired Motions



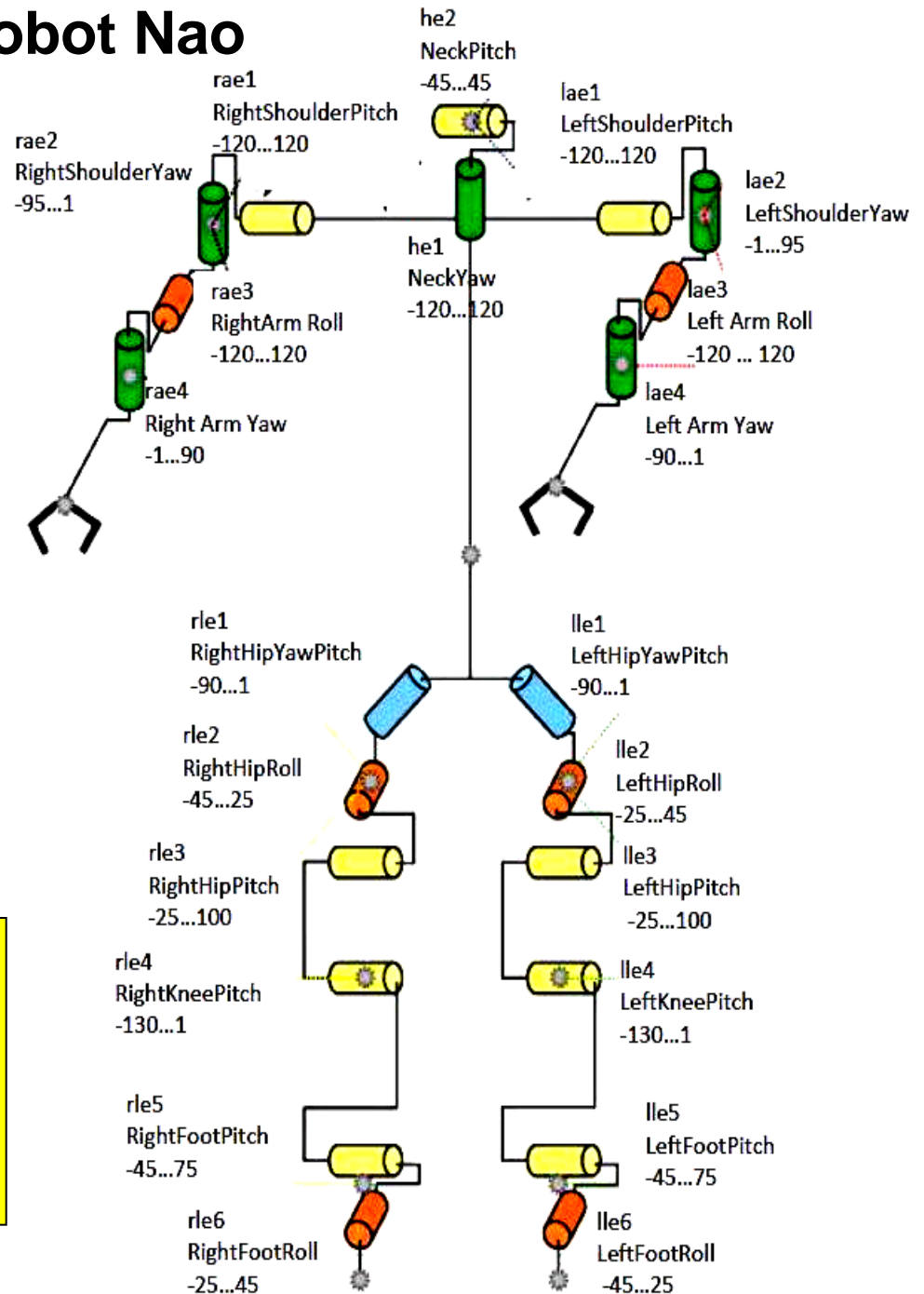
Multi-jointed Legs



Exercises with simulated robot Nao



22 joints
commands every 20 msec
1100 commands per second



First exercises: Implement knee bend, dancing, ...

Using setter methods from RoboNewbie, e.g.

```
effOut.setJointCommand(RobotConsts.LeftArmRoll, 2.3);
```

```
effOut.setJointCommand(RobotConsts.RightShoulderPitch, -2.0);
```

```
effOut.setJointCommand(RobotConsts.NeckYaw, 0.0);
```

for direct control of motor speed.

Problem: up to 22 commands every 20 msec

Keyframe techniques:

Define characteristic postures and let the robot interpolate between them

Motion exercises continued: Keyframes for Walk, Turn, Kick, Catch, ... supported by Motion Editor

MotionNet Editor

File Help

MotionNet Joints

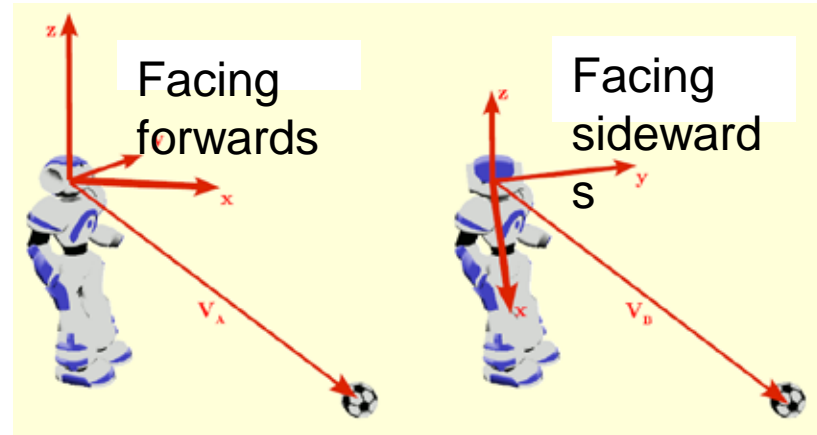
3D

nothing selected

- Head
 - Yaw -28 -120.0
 - Pitch 23 -45.0
- Shoulder
 - Pitch left -97 -120.0
 - Pitch right -120 -120.0
 - Yaw left 37 -1.0
 - Yaw right -34 -95.0
- Arm
 - Roll left -82 -120.0
 - Roll right 119 -120.0
 - Yaw left -80 -90.0
 - Yaw right 89 -1.0
- Hip
 - YawPitch left -58 -90.0
 - YawPitch right -58 -90.0
 - Pitch left -21 -25.0
 - Pitch right 51 -25.0
 - Roll left 45 -25.0
 - Roll right 18 -45.0
- Knee
 - Pitch left -63 -130.0
 - Pitch right -63 -130.0
- Ankle
 - Pitch left -9 -45.0
 - Pitch right 17 -45.0
 - Roll left 4 -45.0
 - Roll right -25 -25.0

Sensors/Perception

- Sensor types
- Vision/Camera Model, Interpretation, no image processing



- World Models
- Representations, Probabilistic Methods, Monte Carlo Methods



Perceptors in Simulation

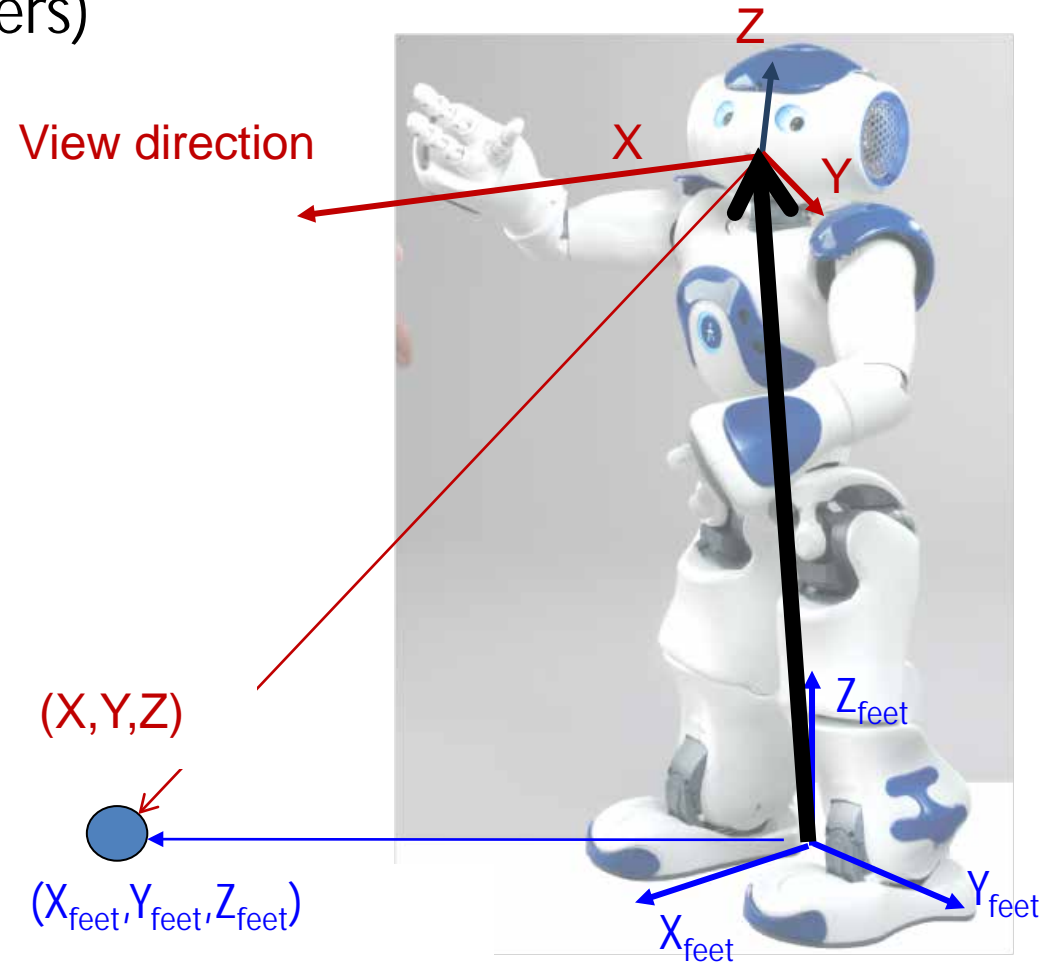
- Joint perceptors
- Camera at the head
- ...

RoboNewbie provides getter methods for perceptor data

```
percln.getJoint(RobotConsts.LeftShoulderPitch);  
percln.getGoalPost(FieldConsts.GoalPostID.G2L);  
percln.getBodyPart(PlayerVisionPerceptor.BodyPart.llowerarm);
```

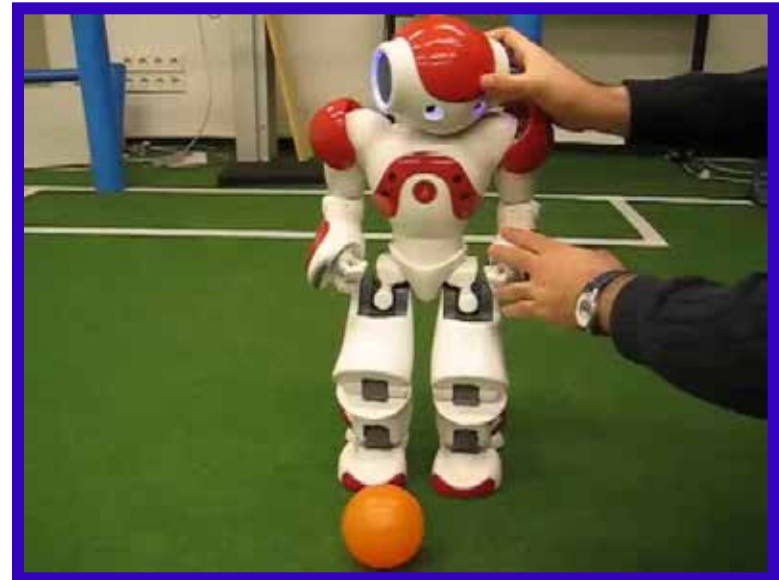
Exercises for perception:

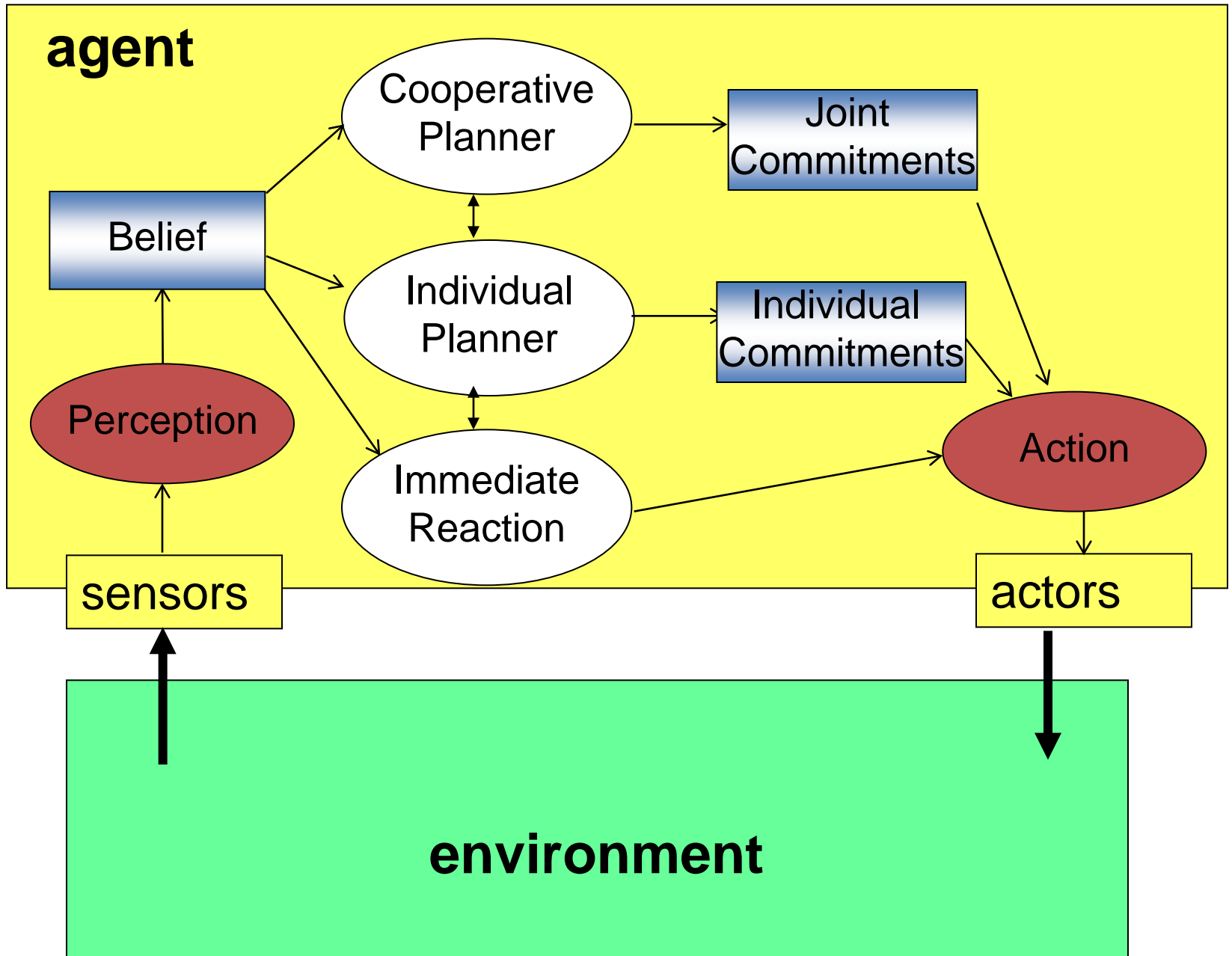
- Where is the ball
- Where are the goals
- (Where are the other players)
- (Where am I on the field)



Behavior Control

- Agent Architectures, e.g. BDI-approach
- Rationality,
- Behavior Based Robotics
- Sensor actor coupling

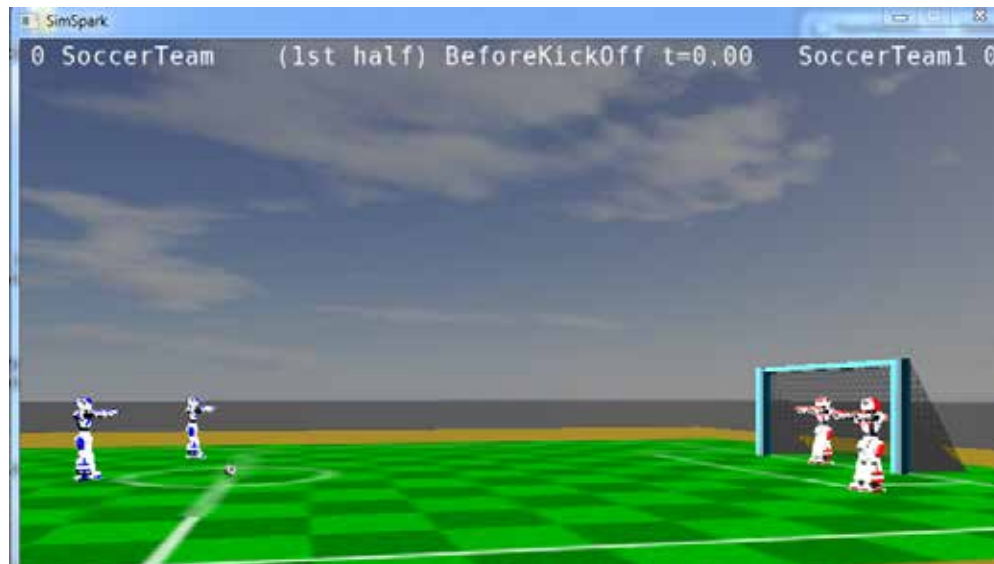




Exercises for integrating perception, motion and control:

Implement a soccer team by joint work in groups of 4-5 students

Competition at the end of the course



Very Simple program:

Repeat:

If robot has fallen down: Stand up

If position of ball is not known:

 Search for ball by turning head (and body)

else if ball is far away: turn to ball, walk to ball

else if ball not between player and goal: turn around ball

else walk forward („dribbling“)

Improvements;

- Better skills: walk, turn, kick, catch, ...
- Different roles: goalkeeper, attacker, supporter, defender

Different kinds of competitions over the years

- Fastest scoring (2012-2013)
- Matches 4 by 4 (2014)
- Matches 1 by 1 (2015)
- Matches 2 by 2 (since 2015)

Problem with too many players:
Crowding around the ball



Actual rules for games 2 by 2

Offending team (left team with kick-off):

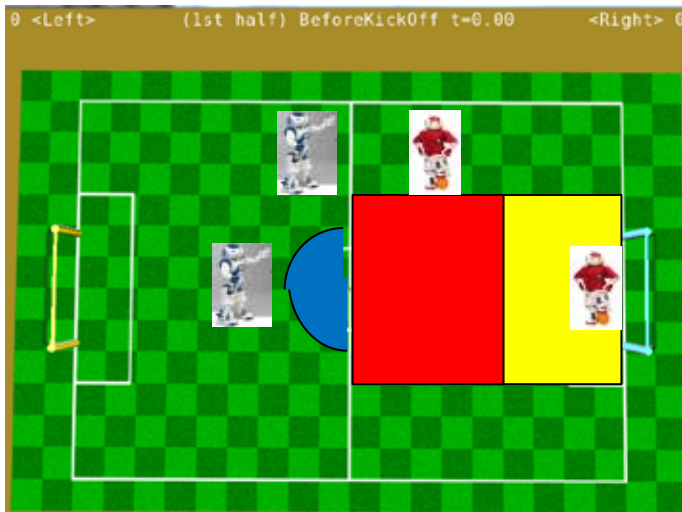
Both players outside of the blue area

Defending team (right team)

Player 1: outside of red area (goalkeeper)

Player 2: outside of red and yellow areas

2 times 2 minutes, with change at half time



Last competitions in our DAAD program: Rijeka 2016



Tirana 2016

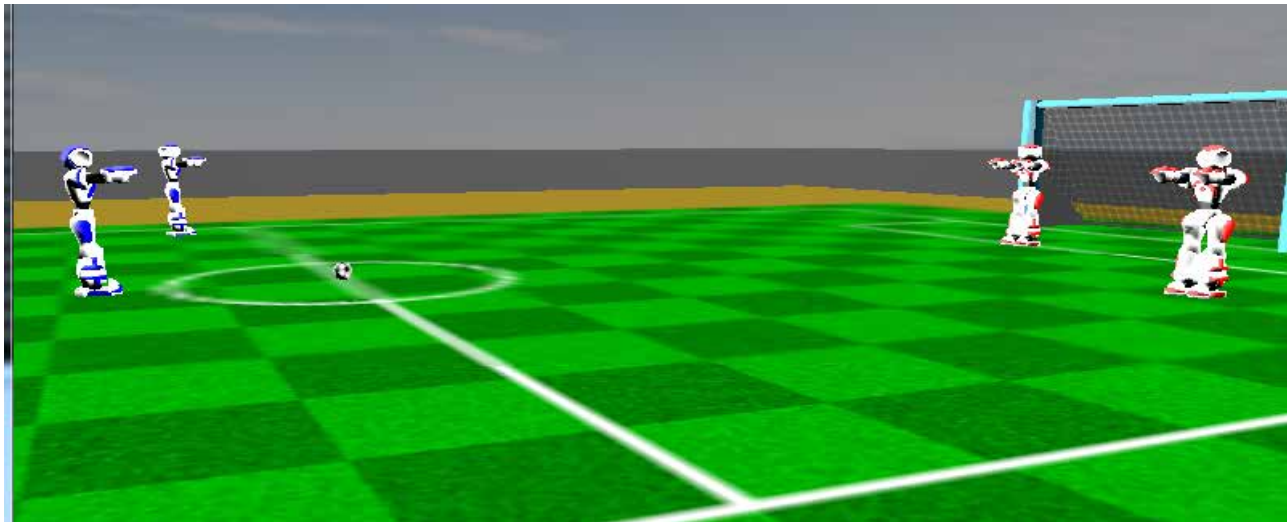


Example of a match (only one half)

Champions from Tirana 2016: RockinRobos
as offending team (blue)

VS.

Champions from Rijeka 2016: Hertha Berlin
as defending team (red)



Opinions of students:

Like the work with RoboNewbie. Want more time for exercises.

Need for better cooperative play:

Better skills for motions (omnidirectional walking).

Need for methods from Machine Learning.

Problem: 30 hours on 8 days are not sufficient.

Project in Plovdiv

Forthcoming paper by *A.Toskova*:

A Java Module for Humanoid Robot Self-Learning

Thank you!

You are invited to the next RoboCup Competition:

