

RoboNewbie_1.0

How to start

Humboldt-University Berlin, Institute of Informatics, AI-Group
date of description: 14.6.2013, description by Monika Domanska

Programming with RoboNewbie will be easy, it assumes, that you have little experience with Java, and that you never have programmed robot controls before.

For the installation follow the instruction that can be downloaded from the RoboNewbie homepage www.naoteamhumboldt.de/projects/RoboNewbie/ .

1) Get familiar with the simulation program SimSpark.

First you need to know, what hardware your simulated Robot provides and how to use the simulator. Read the introduction to SimSpark, that can be downloaded from the RoboNewbie homepage (just below the download link for SimSpark).

2) Do your first steps with the “Quick start tutorial”

Now you know enough theory about the simulated Nao, let’s start programming. Download the “Quick start tutorial” from the homepage (just below the link for RoboNewbie) and try the exercises. The tutorial shows you some typical problems of intelligent robot controls. Apart from that it explains exactly how to run the example programs, what is specific for the usage of SimSpark together with RoboNewbie and NetBeans. This specific hints are also summarized below.

Hints for programming and running RoboNewbie agents

How to start the example agents:

The RoboNewbie package „examples“ contains a set of examples. Every one of them has a class containing the main(...) -method, the name of this class begins with „Agent_“. If a single example consist of more than just the runnable main() -class, it is organized as a subpackage of examples.

To start an agent, first start SimSpark.

Now the „Agent_“ -class of an example agent can be started (e.g. from NetBeans: right click on the file name → Run File). A robot marked blue or red appears on the soccer field and the example is executed.

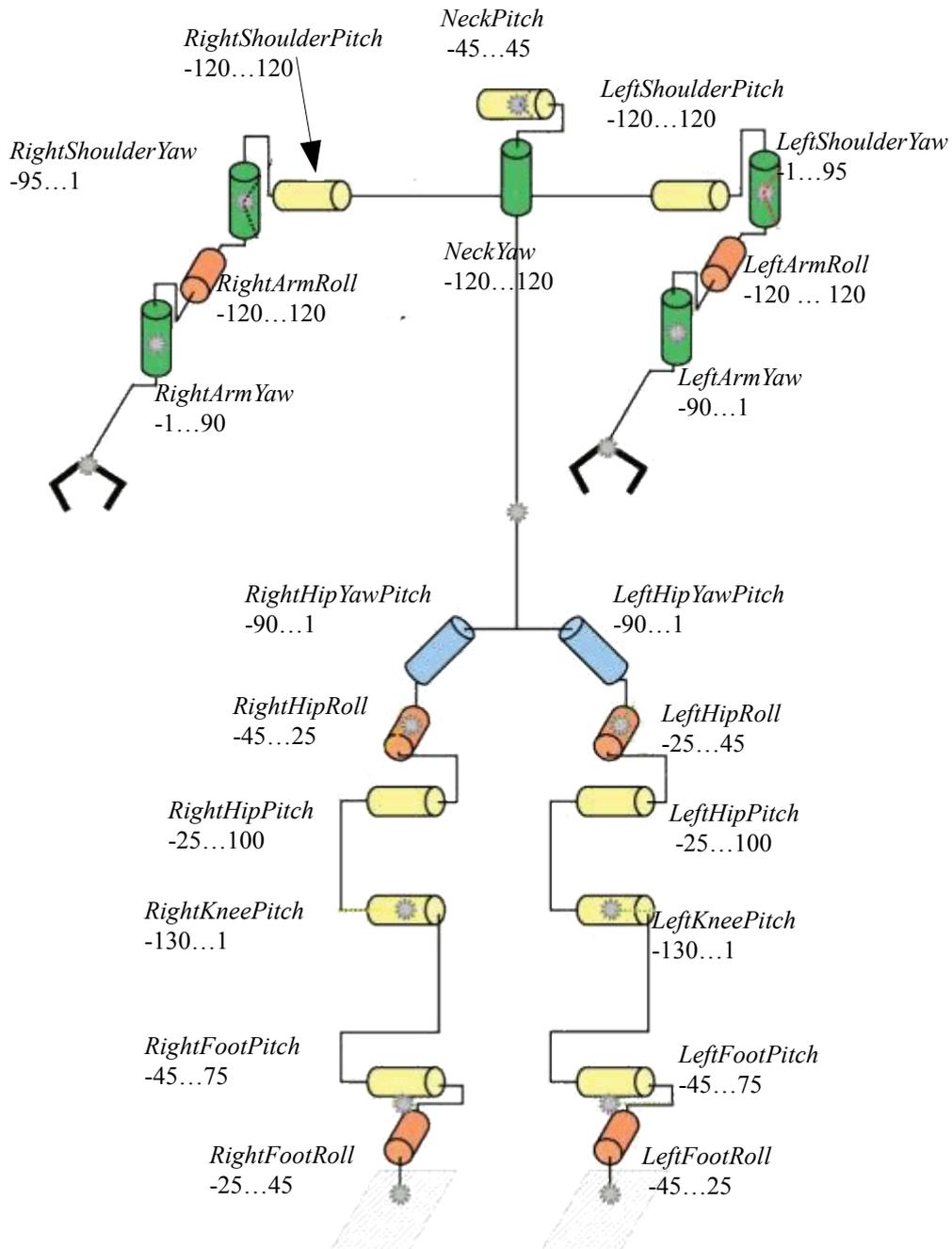
SimSpark has also a timer. To start it, change to the monitor window and hit “k” (kick-off) on your keyboard. (It cannot be stopped, so you have to restart SimSpark, if you want to start the timer again.) Regarding soccer rules, players must wait for the execution of the kick-off. You can skip this phase by hitting “b” (drop-ball).

Right place for your own code

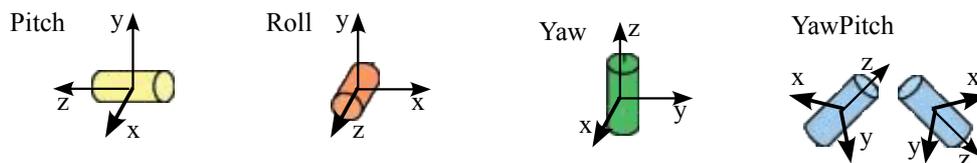
If you implement own agents, they must be stored inside the RoboNewbie NetBeans project (e.g. make a package “myAgents”), otherwise your code might pass the compilation but get problems at runtime due to missing resources (data and library).

SimSpark constants: the joints of the simulated Nao

This picture shows the joint names and the minimal and maximal angles they can achieve.

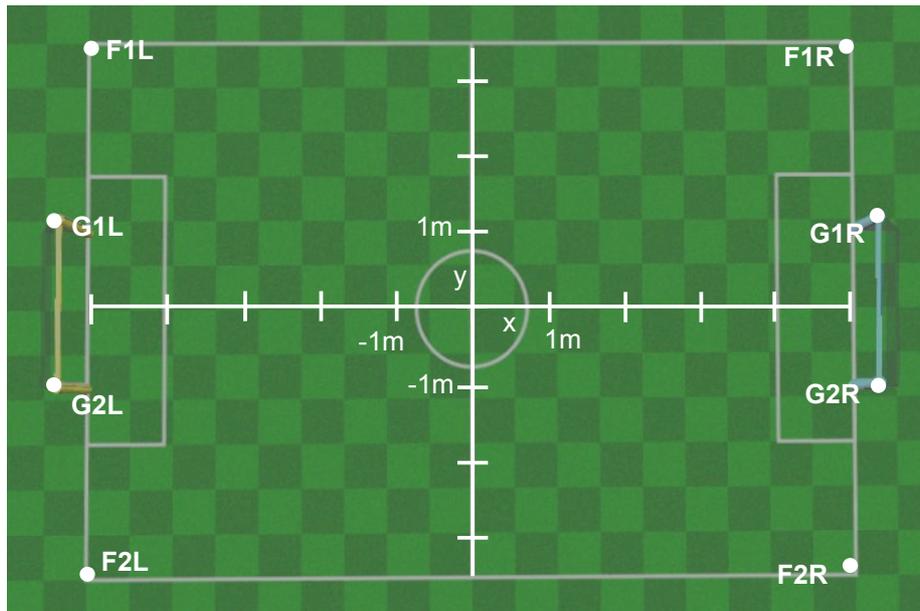


Positive turn direction is from x- to y-axis.



SimSpark constants: the soccer field

The field is 7m*10m big. The goal posts are 0,8m high.



Overview of the most important classes in RoboNewbie:

Class	Package	Description	Examples
ServerCommunication	agentIO	Connection to the SimSpark server, receiving and sending of messages via TCP.	BasicStructure and all other agents
PerceptorInput	agentIO	Read perceptor values out of server messages.	BasicStructure, TestPerceptorInput, KeyframeDeveloper, SimpleSoccer
EffectorOutput	agentIO	Send effector commands in agent messages.	BasicStructure, SimpleSoccer
KeyframeMotion	keyframeMotion	Movements for the robot by processing of keyframe sequences.	SimpleWalkToBall, KeyframeDeveloper, SimpleSoccer
LookAroundMotion	directMotion	Very simply implemented movement for constantly turning the head of the robot.	SimpleSoccer
LocalFieldView	localFieldView	This class processes the vision perceptor values and models information about the situation on the soccer field.	TestLocalFieldView, SimpleWalkToBall, SimpleSoccer
RobotConsts FieldConsts GameStateConsts	util	Classes with SimSpark constants.	SimpleSoccer
Logger	util	Logger class, recommended because test outputs on standard output could take too much time.	TestPerceptorInput, TestLocalFieldView, and others