Practical Session - RoboCup MSL Workshop 2014

Location: Eindhoven, The Netherlands

Date: 10 - 12 November, 2014

Organizers: Robin Soetens, Lotte de Koning, René van de Molengraft

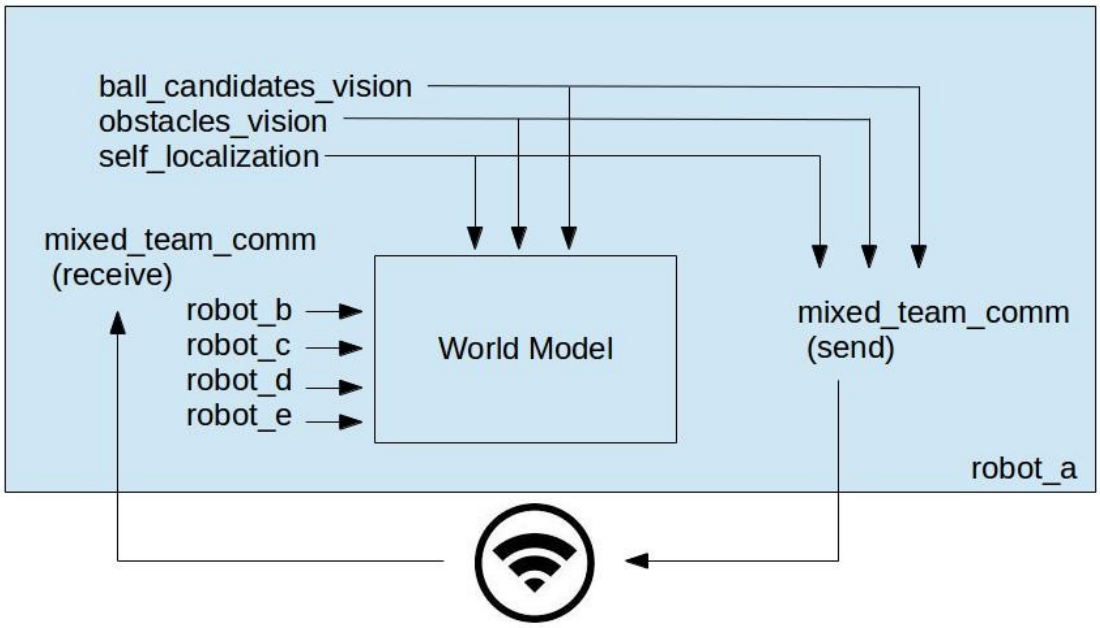
Contact: robinsoetens@gmail.com, c.k.m.d.koning@student.tue.nl, m.j.g.v.d.molengraft@tue.nl

Website: http://wiki.robocup.org/wiki/Middle_Size_League/RoboCup_MSL_Workshop

Goal of the practical session is to give a demo on a full-size MSL field with robots of different teams running their own worldmodel software. The demo shows we are able to share knowledge among these robots, who have a different software architecture and different worldmodel implementations. By doing this we work towards standardization of robot-robot communication and of software component interfaces, which eases mixed-team challenges and sharing of code.

Worldmodel

By worldmodel we mean the piece of software where perceived obstacles, ball, and self-localisation of the robot and its peers are merged into a single, stable, representation of the state of the world. So schematically, for robot_a working with b, c, d and e in a mixed team:



Robot Availability

Carpe Noctem, ASML Falcons, VDL Robot Sports and Tech United bring a robot.

NuBot works with a separate omnivision unit, connected to a laptop.

CAMBADA communicates the mixed team package from their base station laptop, based on a simulation of the situation on the field.

Demo Setup

Robots and robot stand-ins of five different teams are randomly placed somewhere on the field, they remain static. Occasionally the sixth robot gets substituted for one of the robots on the field. A number of obstacles is put on the field as well, also randomly placed and static. The ball is the only non-static element. It is kicked by human team members on the side of the field and passes the robots in different directions

Next to the field five monitors visualize the worldmodel of each of the robots on the field individually.

An MSL_FIELD_A router provides the network, configured as during normal competition, see rulebook for specifications.

Mixed team comm package

As far as we know, all teams use UDP Multicast packages to communicate among their robots. For the workshop mixed-team demo we propose to simply prepend the data structure you are already using with a standardized piece of data 'mixed-team-comm', which looks like this:

| mixed_team_flag (1 byte) | robot_id (1 byte) | ball_cand_vision (39 bytes) | obstacles (50 bytes) | self-loc (5 bytes) |

Further specified:

- **mixed_team_flag** is 123 in UINT8
- **robot_id** corresponds to the jersey number of the robot (1 – 6), in UINT8
- Distances and speeds are in millimeters or millimeter per second, to prevent the need of floats instead of ints.
- All coordinates are in a global frame, specified in the standardized logging document: http://wiki.robocup.org/wiki/Middle_Size_League#Standardized_Logging
- **ball_cand_vision** contains 3 ball candidates ordered from most probably to least probable. Each ball candidate is stored in 6 INT16 fields followed by one UINT8. These fields represent a sequence of $x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}$, followed by a confidence score on a scale of 1 to UINT8 MAX

- **obstacles** contains a list of 10 obstacles, ordered from highest confidence to lowest confidence. Each obstacle is stored in 2 INT16 fields followed by one UINT8. These fields represent a sequence of $x\ y$, followed by a confidence score on a scale of 1 to UINT8 MAX
- Use INT16 MIN to indicate a coordinate or velocity field is not used, that is -32768 or 0x8000 in hexadecimal. This is relevant for instance if less than three ball candidates are known.
- Use UINT8 MIN to indicate a confidence field is not used.
- **self-loc** is stored in two INT16 fields followed by one UINT8. These fields represent a sequence of $x\ y$ followed by a confidence score.
- Where relevant we use a little-endian sequence of bytes