ROS-interface to USARSL
Arnoud Viss

Based on work by Zeid Kootbally and Stephen Balakin
What is USARSim?

• High-fidelity multi-robot simulator developed on top of an existing game engine
  – High performance physics and 3D rendering

• Originally conceived as tool for Urban Search and Rescue (USAR), it has a much broader scope [1]

Basic Premise

- Would like to be able to develop, debug, and evaluate cognitive systems
  - Repeatable trials
  - Known ground truth, noise, detections, false detections
- Evaluation environment should provide realism
  - Realistic complexity
  - Tailored data output
  - Environmental interaction
  - Obey basic laws of physics in sensing and mobility

Images from USARSim / MOAST Tutorial
USARSim variety of platforms
A wide variety of simulated worlds

DM-Arda Map

DM-spqrSoccer2006 Map

DM-TallTestWorld Map

DM-RoboCup06Worlds Maps
UDK based simulated worlds

Dutch Open Final

VMAC factory

NIST campus

RoboCup 2012 Preliminary
GameBots is origin USARSim

Interface defined in 2003

Interface stable since 2013

e.g.: INIT {ClassName \textit{robot\_class} } {Name \textit{robot\_name} } 
{Location \textit{x},\textit{y},\textit{z} } {Rotation \textit{r}, \textit{p}, \textit{y} } 

DRIVE {Left \textit{float} } {Right \textit{float} } {Normalized \textit{bool} } 
{Light \textit{bool} } {Flip \textit{bool} }

GEO {Type \textit{GroundVehicle} } {Name \textit{string} } 
{Dimensions \textit{x}, \textit{y}, \textit{z} } {COG \textit{x}, \textit{y}, \textit{z} } 
{WheelRadius \textit{float} } {WheelSeparation \textit{float} } 
{WheelBase \textit{float} }

CONF {Type \textit{AerialVehicle} } {Name \textit{string} } 
{SteeringType \textit{string} } {Mass \textit{float} }

CONF {Type \textit{Camera} } {CameraDefFov 0.8727} 
{CameraMinFov 0.3491} {CameraMaxFov 2.0943} 
{CameraFov 0.8726}

http://sourceforge.net/apps/mediawiki/usarsim/
The configuration of a robot is converted into the Transform Trees of ROS
Coupling to ROS navigation stack

```
move_base_simple/goal
geometry_msgs/PoseStamped
```

```
move_base
```

```
global_planner
```

```
global_costmap
```

```
recovery_behaviors
```

```
local_planner
```

```
local_costmap
```

```
/cmd_vel
geometry_msgs/Twist
```

```
base_controller
```

```
map_server
```

```
"/map"
nav_msgs/GetMap
```

```
sensor_topics
sensor_msgs/LaserScan
sensor_msgs/PointCloud
```

```
sensor sources
```

```
Provided node
```

```
Optional provided node
```

```
Platform specific node
```
1. Bring up an environment in USARSim.
2. $roscore
3. $roslaunch usarsim usarsim.launch
4. $rosrun teleop_twist_keyboard teleop_twist_keyboard.py
5. $rosrun gmapping slam_gmapping scan:=lms200 _odom_frame:=odom
TeleOperation Example
Autonomous Example
Adaption by the RoboCup teams

USARSim can now be used as simulator for ROS modules.

USARSim is based on a state-of-the-art Game Engine, which allows the creation of detailed worlds, realistic lighting conditions and reliable physics.
Amsterdam Oxford Joint Rescue Forces
RoboCup Rescue Simulation - Virtual Robots Competition

Publications

Publications listed below are relevant to research conducted by UVARescue and Amsterdam Oxford Joint Rescue Forces in the USARSim simulator. For a more extensive list of publications related to this competition see the RoboCup Rescue wiki and the Success Stories on Sourceforge.

2013


- Sander van Noort and Arnoud Visser, "Extending Virtual Robots towards RoboCup Soccer Simulation and @Home", To be published in the Springer Lecture Notes on Artificial Intelligence series, volume 7500, pp. 332-343. (PDF).

