Research & Innovations

UvA Rescue

RoboCup IranOpen 2012 competition
Tehran, April 6, 2012

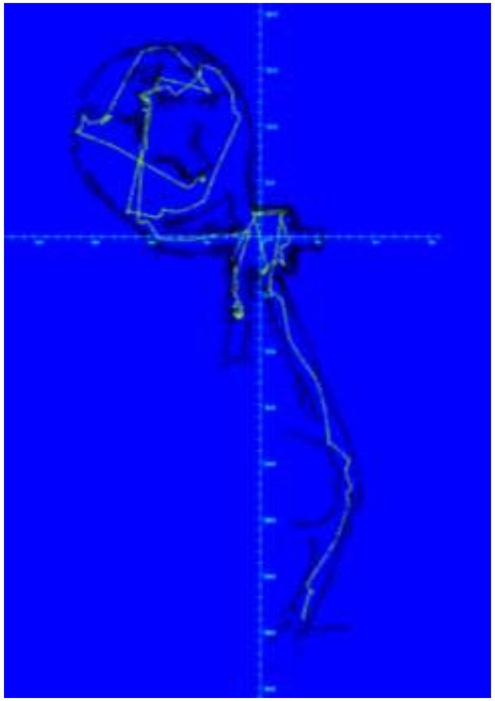
Universiteit van Amsterdam

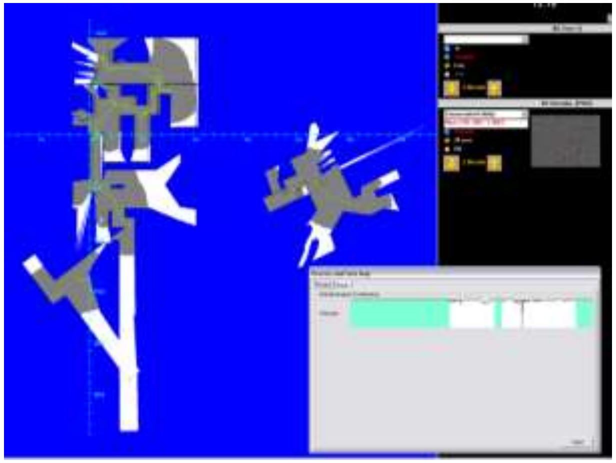
Intelligent Systems Laboratory

Amsterdam Oxford Joint Rescue Forces

Innovations for Iran Open 2010 (i.e.)

- Realistic Smoke
- Confidence selection in maps
- Autonomous AirRobots
- Local sonar maps





Other assets:

- Can control many robots (Matilda, Element, Talon, AirRobot, ATRVJr, Zerg. etc.)
- Graph based map, which can be easily shared and corrected
- Smooth transition from teleoperated to fully autonomous behavior

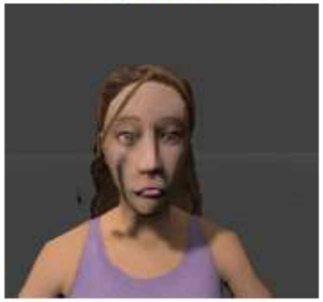
www.jointrescueforces.eu

Amsterdam Oxford Joint Rescue Forces

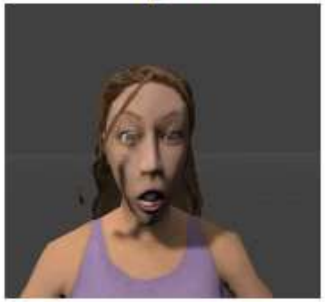
Innovations for Iran Open 2011 (i.e.)

- Realistic Victim behaviors
- Nao kinemetics model
- AR.Drone dynamics model
- Waypoint navigation

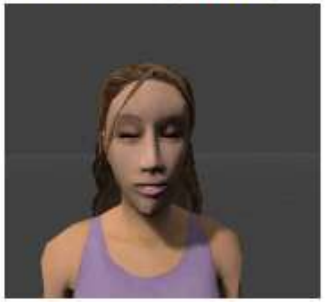
Sad (Sadness)



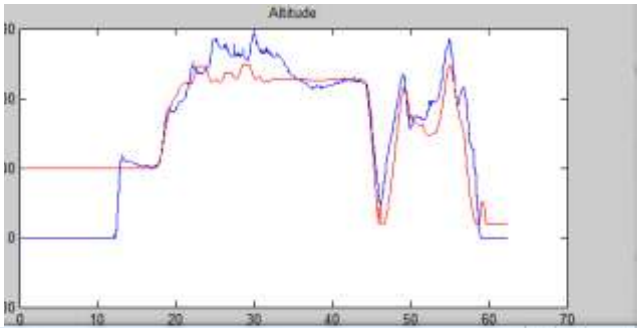
Surprise



bLink (Blink)



AR.Drone: simulated versus real



Other assets:

- Can interpret many sensors (laser scanners, sonar, radar, vision, etc.)
- Graph based map, which can be easily shared and corrected
- Smooth transition from teleoperated to fully autonomous behavior

www.jointrescueforces.eu

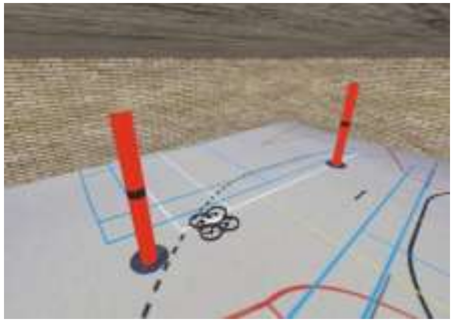
Universiteit van Amsterdam

UvA Rescue

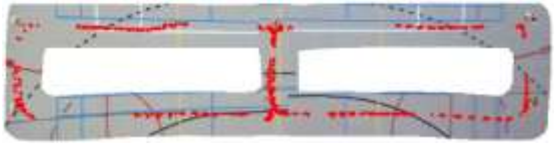
Intelligent Systems Laboratory

Innovations for Iran Open 2012 (i.e.)

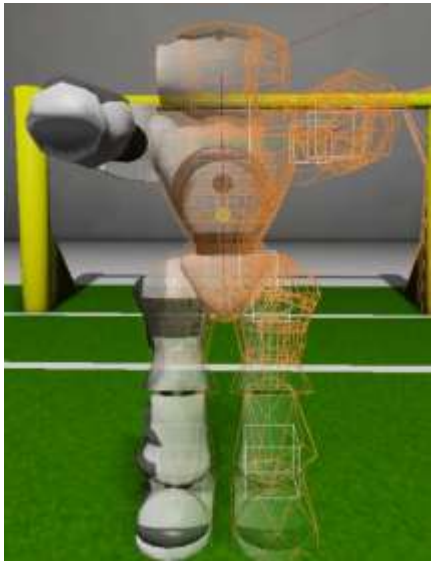
- Visual Localization And Mapping



AR.Drone localizing on visual map



- Nao humanoid robot



collision frame Nao

- Automatic map generator

map generated with high difficulty

Other assets:

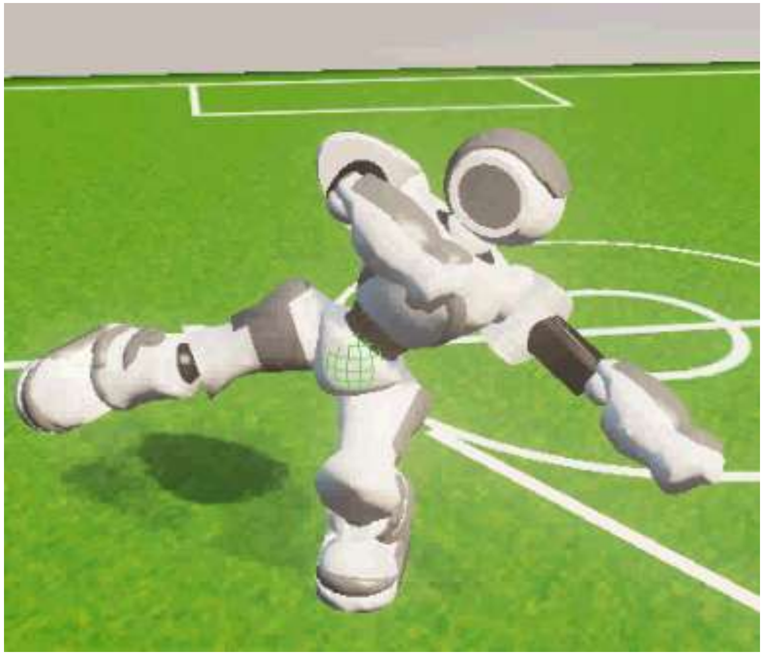
- Can read many logfile formats (Radish, Carmen, etc.)
- Graph based map, which can be easily shared and corrected
- Smooth transition from teleoperated to fully autonomous behavior

www.jointrescueforces.eu

In close cooperation with

University of Oxford
Computing Laboratory

Validation of the dynamics of an humanoid robot in USARSim

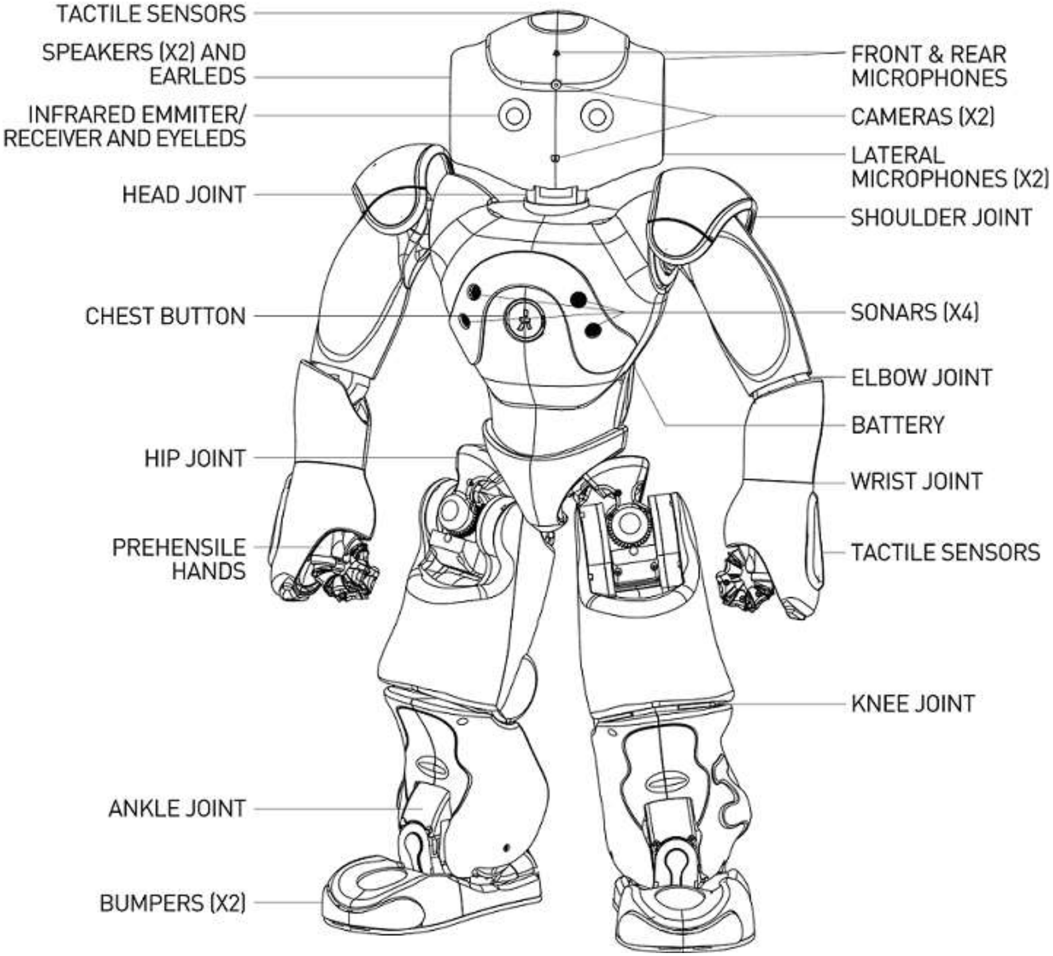


Sander van Noort & Arnoud Visser

Performance Metrics for Intelligent Systems workshop (PerMIS'12), College Park, MD, March 2012

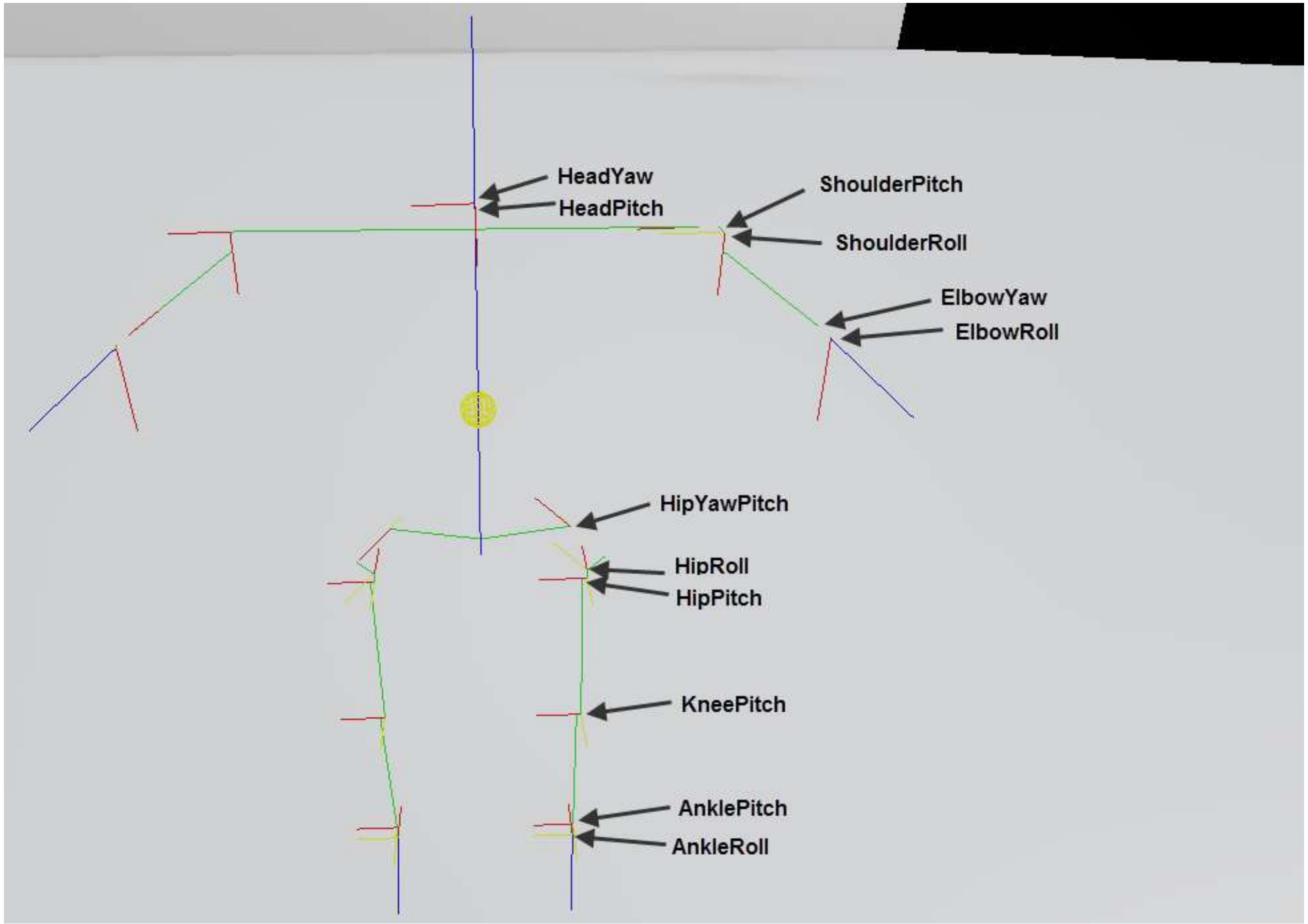
Universiteit van Amsterdam
Informatica Instituut
intelligent autonomous systems

Humanoid robot NAO



Aldebaran Robotics, France

Constrained Kinematic Chains



5 Kinematic chains; 21 Degrees of Freedom.

Denavit Hartenberg representation

- Offset and range of each joint

$$LShoulderPitch = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0.0900\cos\theta_1 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0.0900\cos\theta_1 \\ 0 & 1 & 0 & 0.08 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LShoulderRoll = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & 0.0100\cos\theta_2 \\ \sin\theta_2 & 0 & -\cos\theta_2 & 0.0100\cos\theta_2 \\ 0 & 1 & 0 & 0.01 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LElbowYaw = \begin{bmatrix} \cos\theta_3 & 0 & \sin\theta_3 & 0.1097\cos\theta_3 \\ \sin\theta_3 & 0 & -\cos\theta_3 & 0.1097\cos\theta_3 \\ 0 & 1 & 0 & 0.01 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LElbowRoll = \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 & 0 \\ \sin\theta_4 & 0 & -\cos\theta_4 & 0 \\ 0 & 1 & 0 & 0.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LHipYawPitch = \begin{bmatrix} \cos\tau_1 & -\frac{1}{4}\pi\sin\tau_1 & \frac{1}{4}\pi\sin\tau_1 & 0.0461\cos\tau_1 \\ \sin\tau_1 & \frac{1}{4}\pi\cos\tau_1 & -\frac{1}{4}\pi\cos\tau_1 & 0.0461\cos\tau_1 \\ 0 & \frac{1}{4}\pi & \frac{1}{4}\pi & 0.07 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LHipRoll = \begin{bmatrix} \cos\tau_2 & 0 & \sin\tau_2 & 0.0134\cos\tau_2 \\ \sin\tau_2 & 0 & -\cos\tau_2 & 0.0134\cos\tau_2 \\ 0 & 1 & 0 & 0.03 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

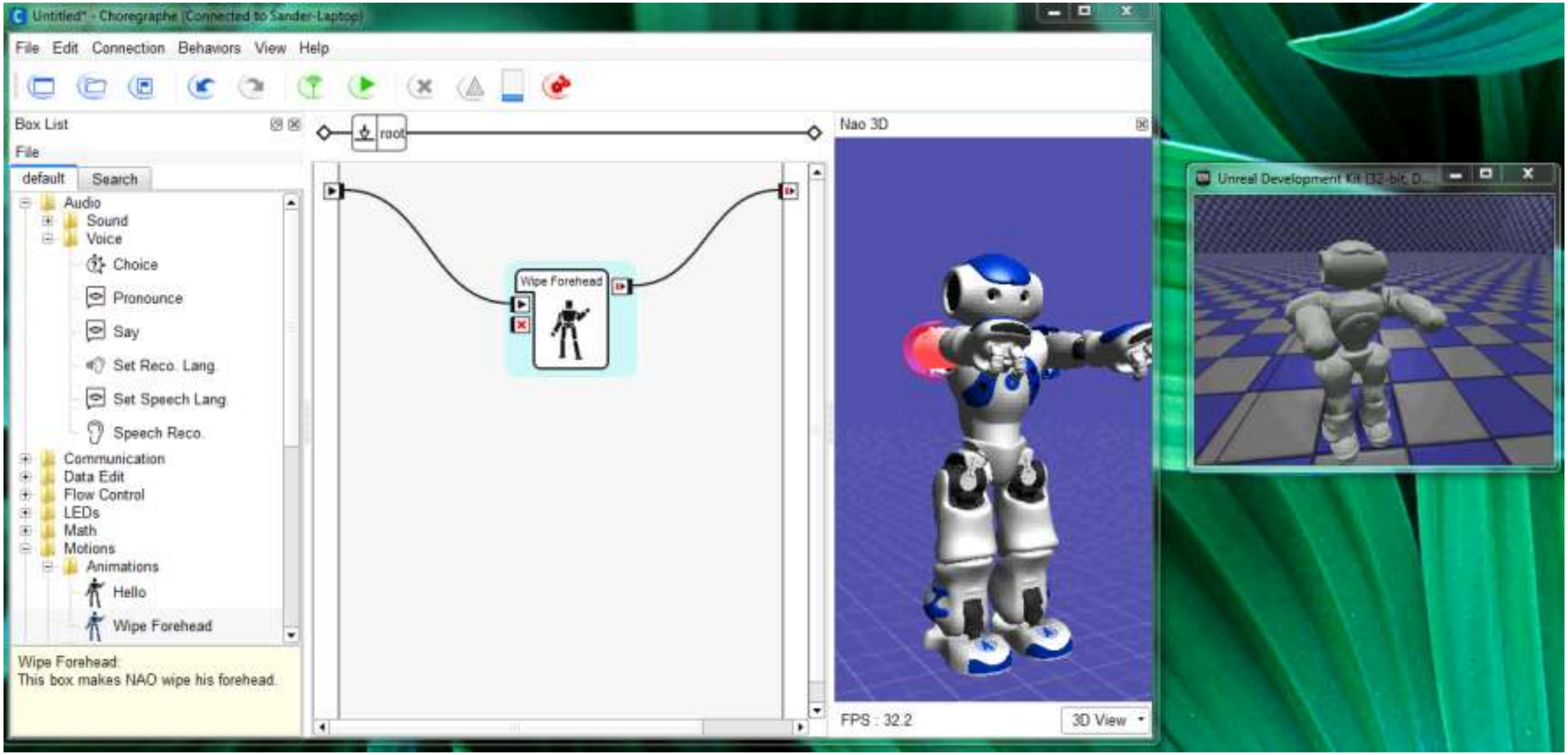
$$LHipPitch = \begin{bmatrix} \cos\tau_3 & 0 & \sin\tau_3 & 0.0050\cos\tau_3 \\ \sin\tau_3 & 0 & -\cos\tau_3 & 0.0050\cos\tau_3 \\ 0 & 1 & 0 & 0.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LKneePitch = \begin{bmatrix} \cos\tau_4 & -\sin\tau_4 & 0 & 0.0880\cos\tau_4 \\ \sin\tau_4 & \cos\tau_4 & 0 & 0.0880\cos\tau_4 \\ 0 & 0 & 1 & 0.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$LAnklePitch = \begin{bmatrix} \cos\tau_5 & -\sin\tau_5 & 0 & 0.1001\cos\tau_5 \\ \sin\tau_5 & \cos\tau_5 & 0 & 0.1001\cos\tau_5 \\ 0 & 0 & 1 & 0.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

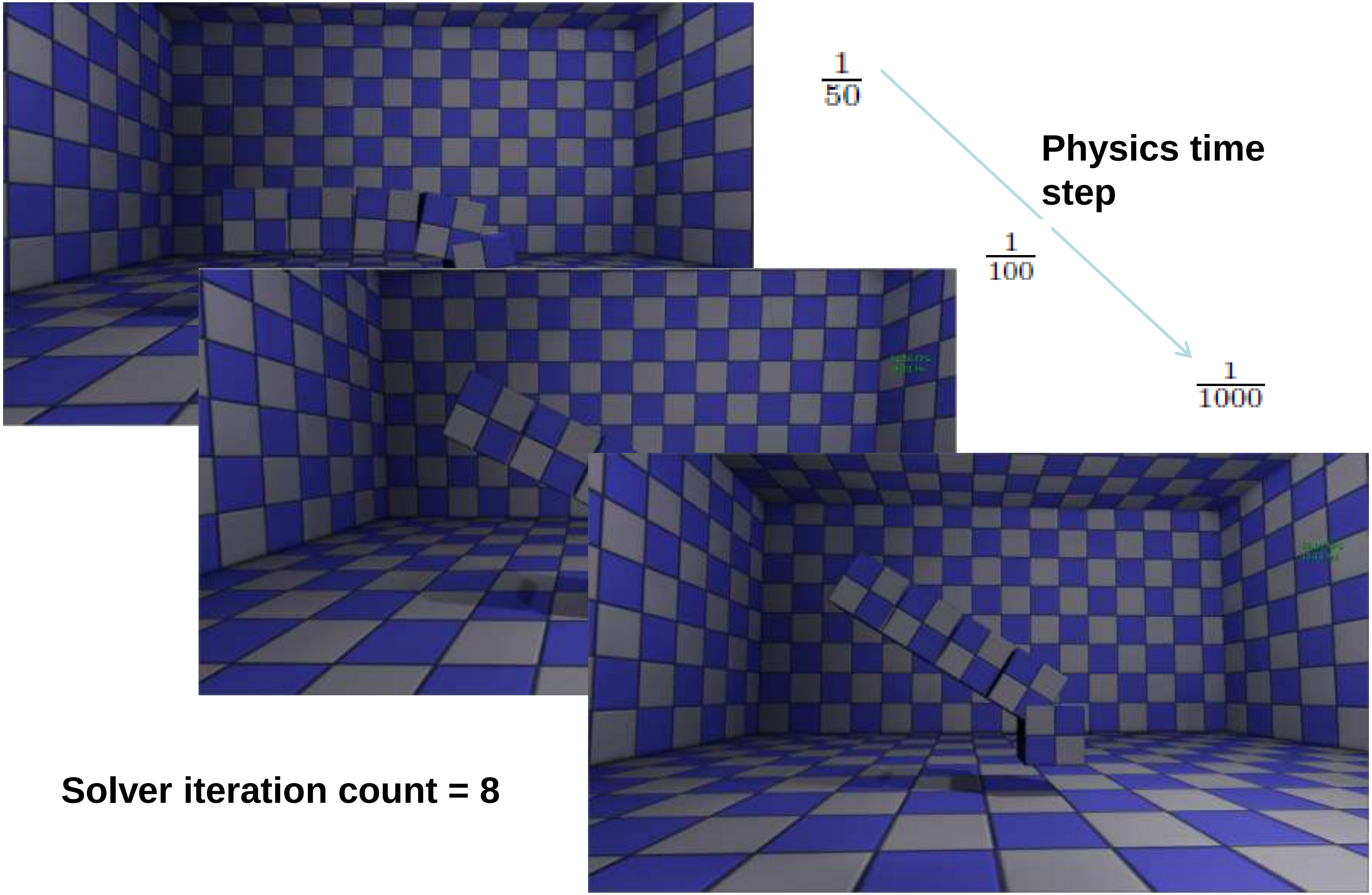
$$LAnkleRoll = \begin{bmatrix} \cos\tau_6 & 0 & \sin\tau_6 & 0.0100\cos\tau_6 \\ \sin\tau_6 & 0 & -\cos\tau_6 & 0.0100\cos\tau_6 \\ 0 & 1 & 0 & 0.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Constrained movement of joints



Wipe Forehead:
This box makes NAO wipe his forehead.

Constraint movement



$\frac{1}{50}$

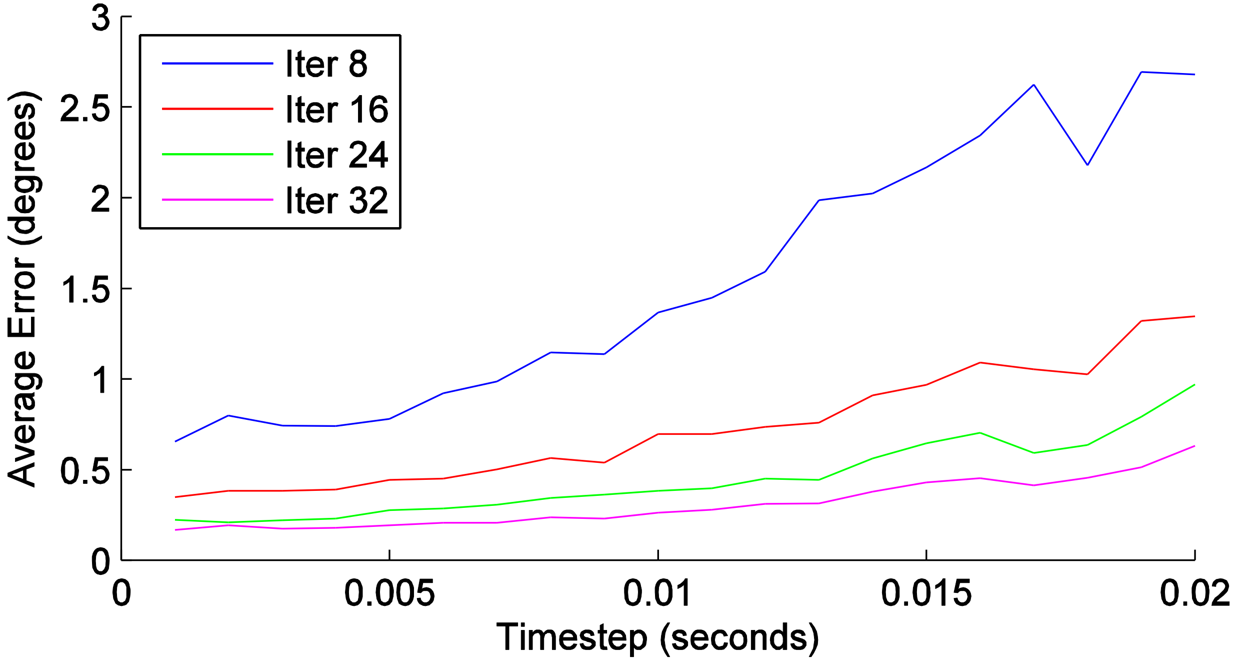
Physics time step

$\frac{1}{100}$

$\frac{1}{1000}$

Solver iteration count = 8

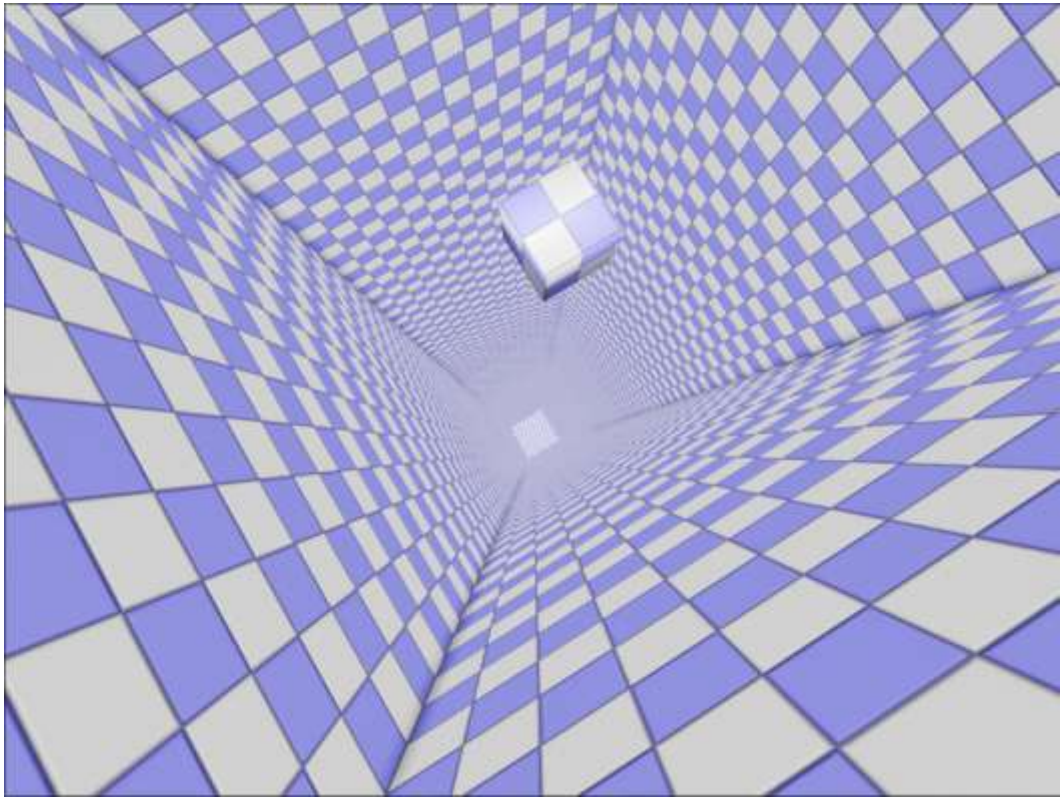
Parameters constrained movement



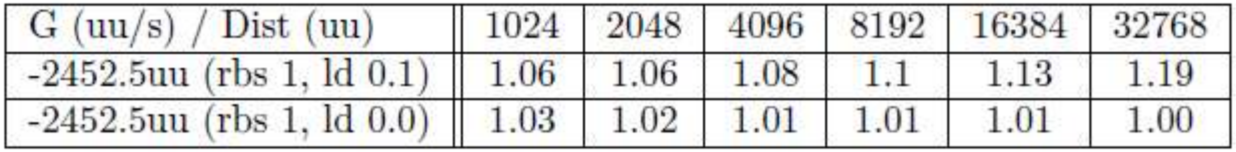
Default values for the Unreal Engine:

- Solver iteration count = 8
- Time step = $\frac{1}{50}$ = 0.02 seconds

Gravity

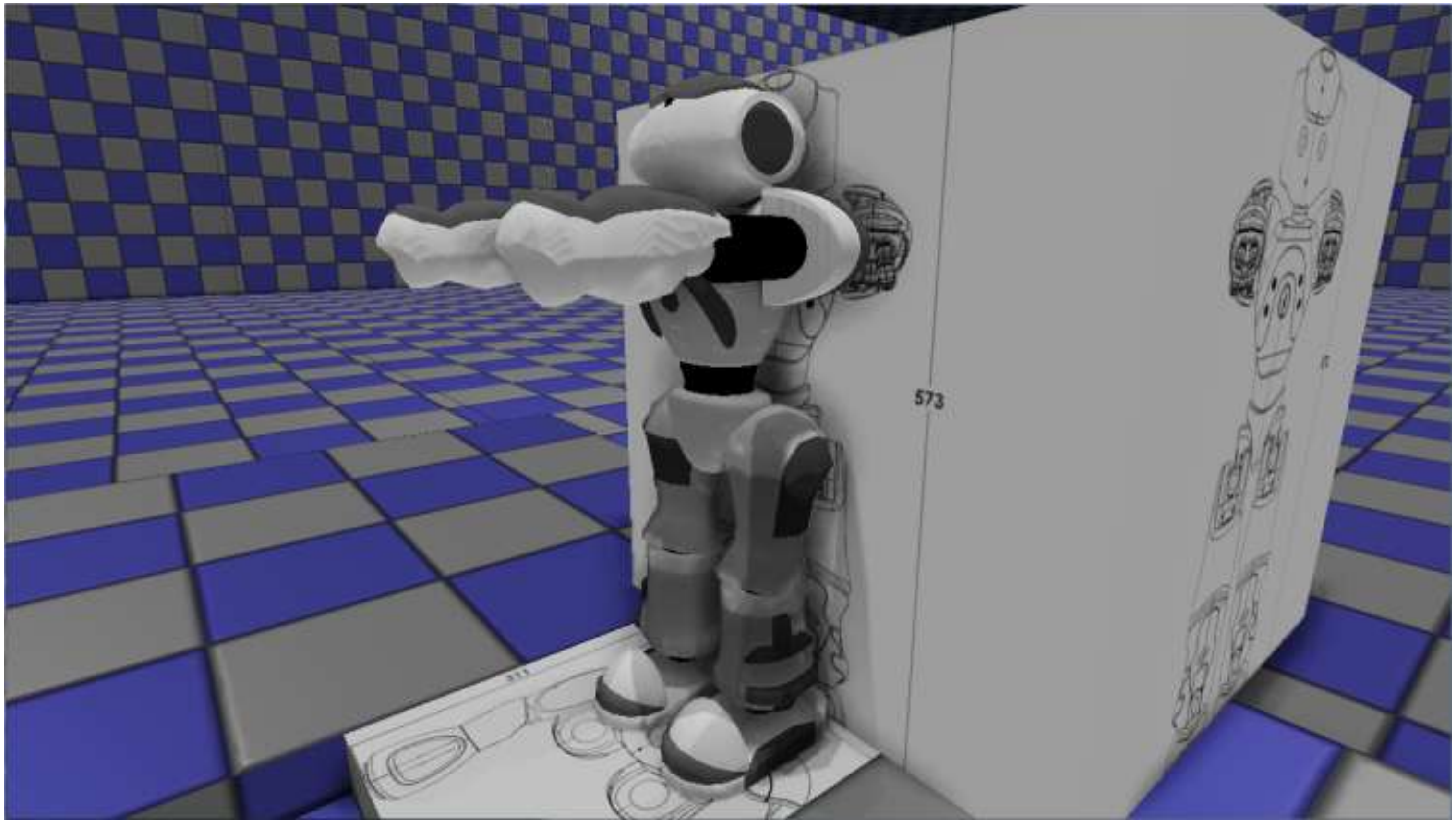


Default values for the Unreal Engine had to be corrected with a factor



G (uu/s) / Dist (uu)	1024	2048	4096	8192	16384	32768
-2452.5uu (rbs 1, ld 0.1)	1.06	1.06	1.08	1.1	1.13	1.19
-2452.5uu (rbs 1, ld 0.0)	1.03	1.02	1.01	1.01	1.01	1.00

Advanced experiments



Three full body movements:

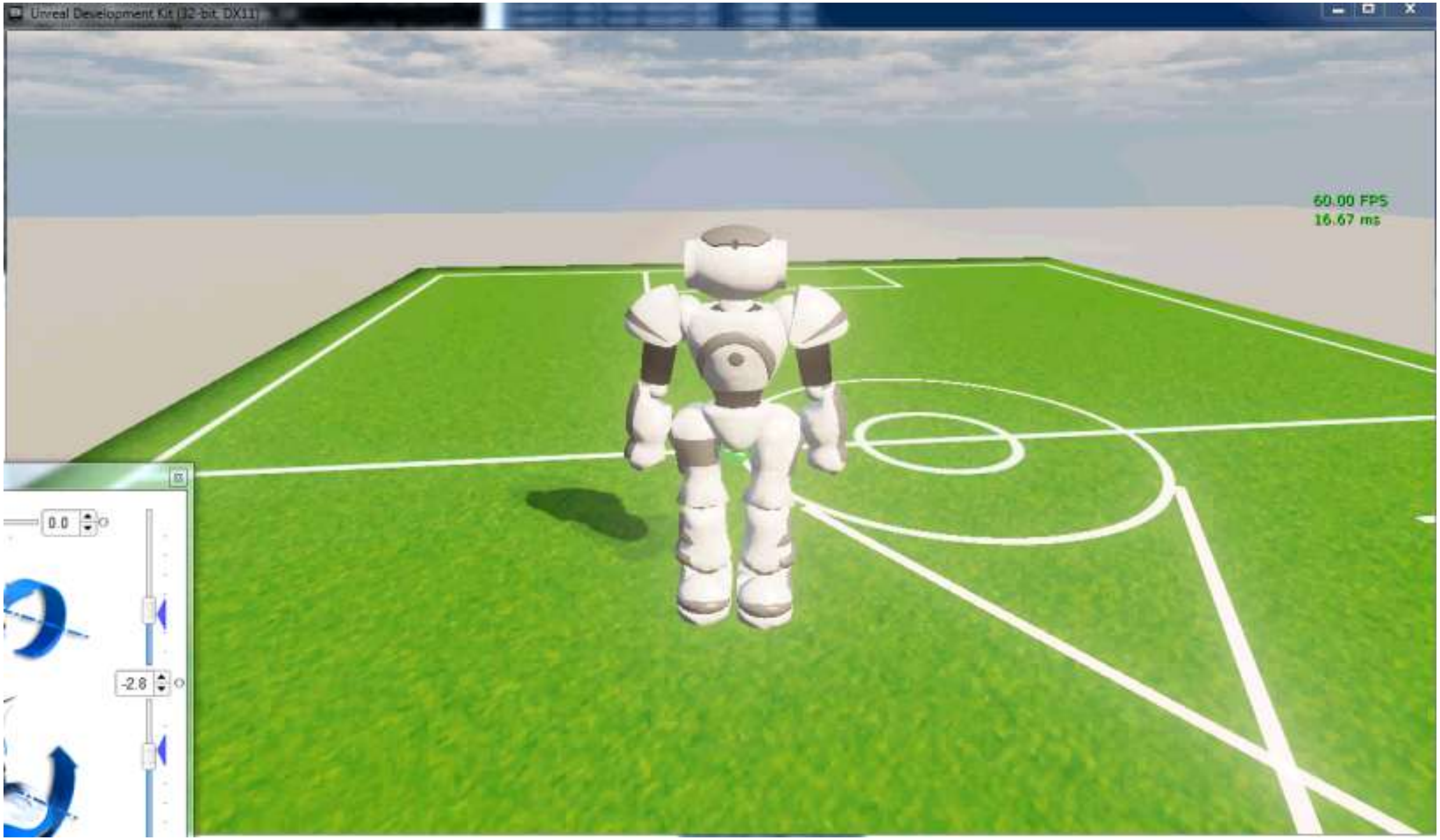
- **A kick**
- **Balance act (Tai Chi Chuan)**
- **Single step**

Balance act

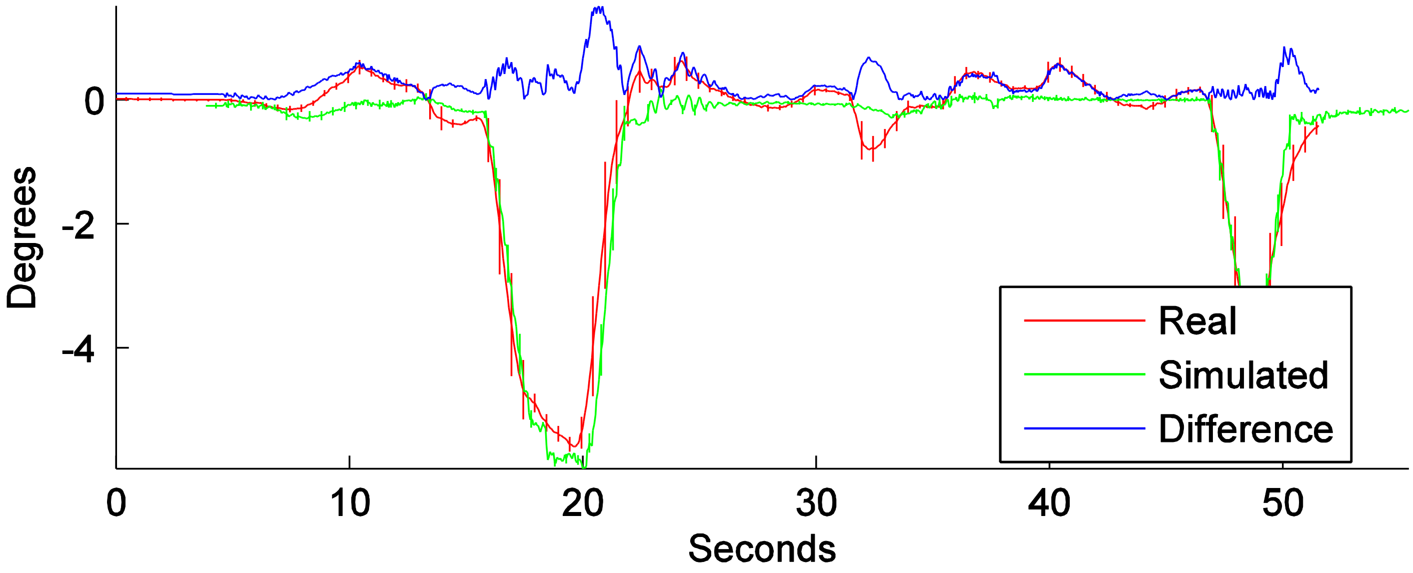
Diagnostic movement: Tai Chi Chuan

- **Real robot: all motors and joints still functional**
- **Simulated robot: weight correctly distributed over body**

Tai Chi Chuan



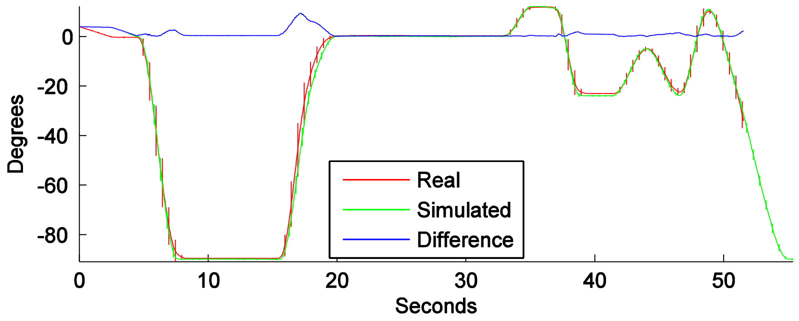
Tai Chi Chuan

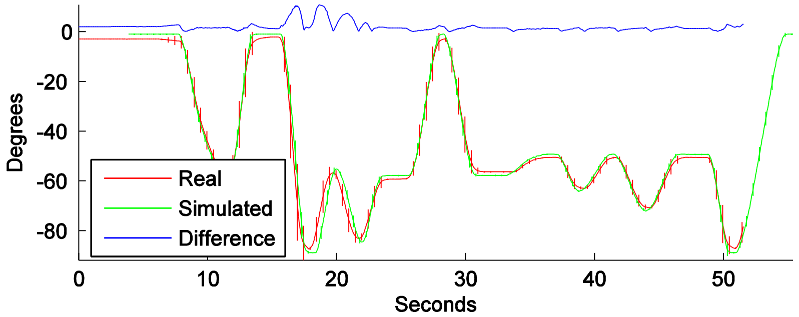


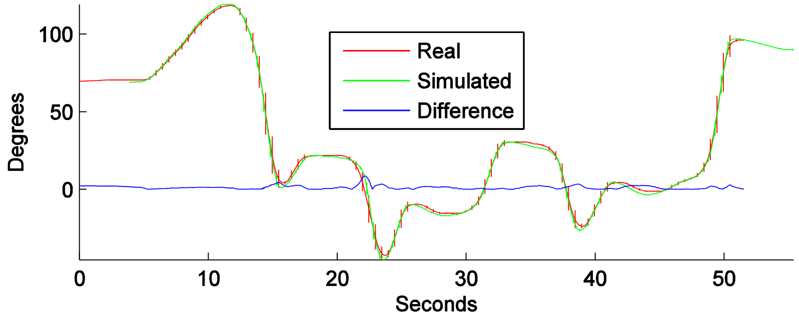
Movement of the Right Hip (yaw / pitch):

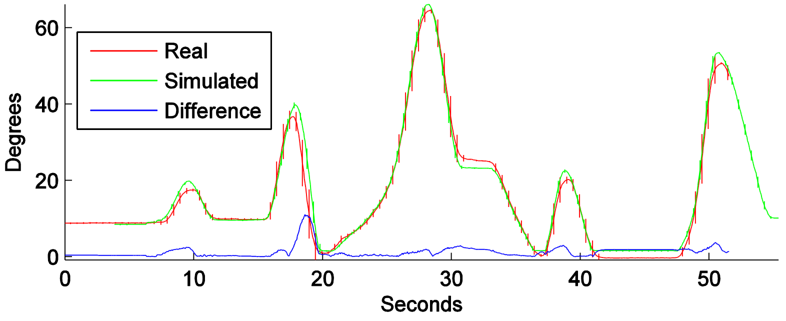
- **Good correspondence, except for deceleration**
- **Differences in the order of natural variance**

Upper body during Tai Chi Chuan



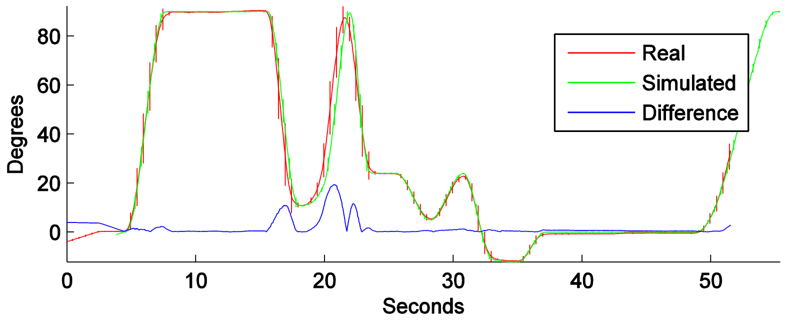


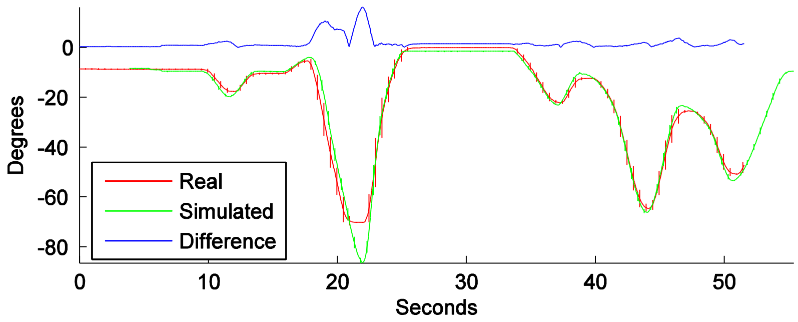


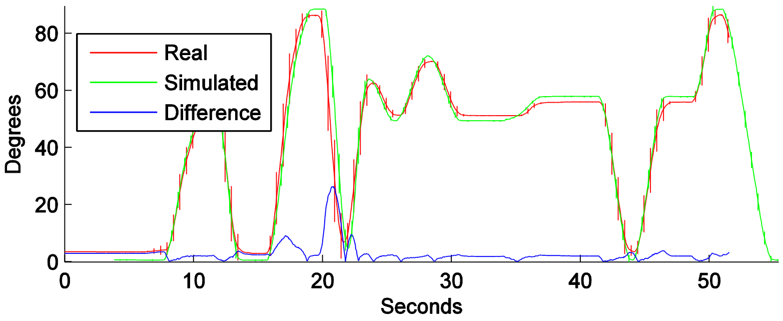


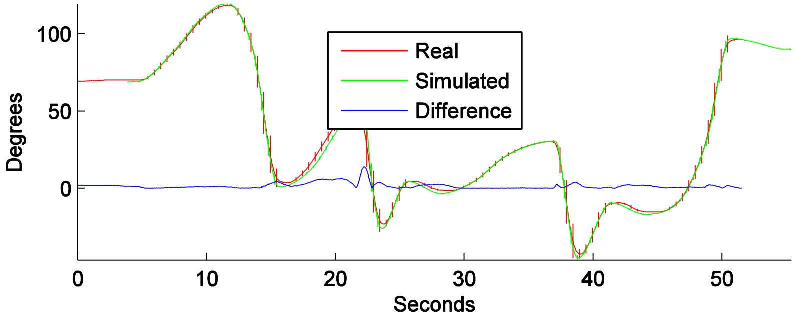
Left side

Upper body during Tai Chi Chuan



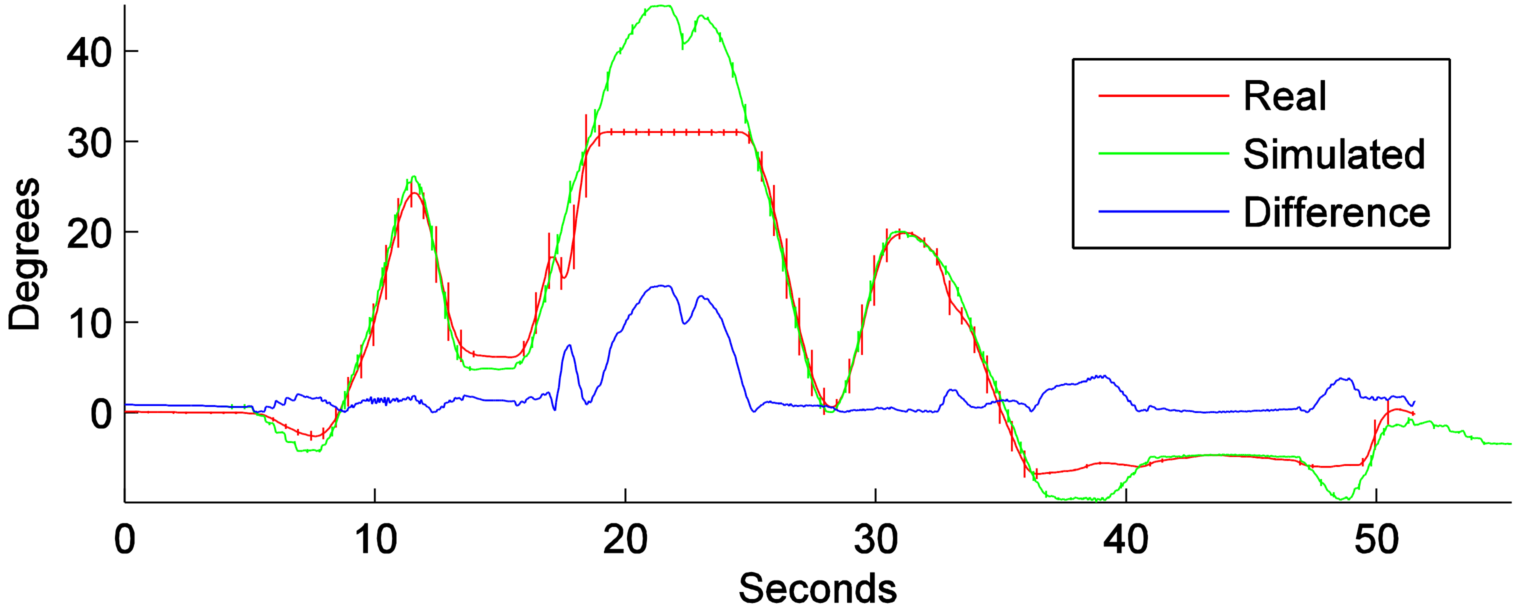






Right side

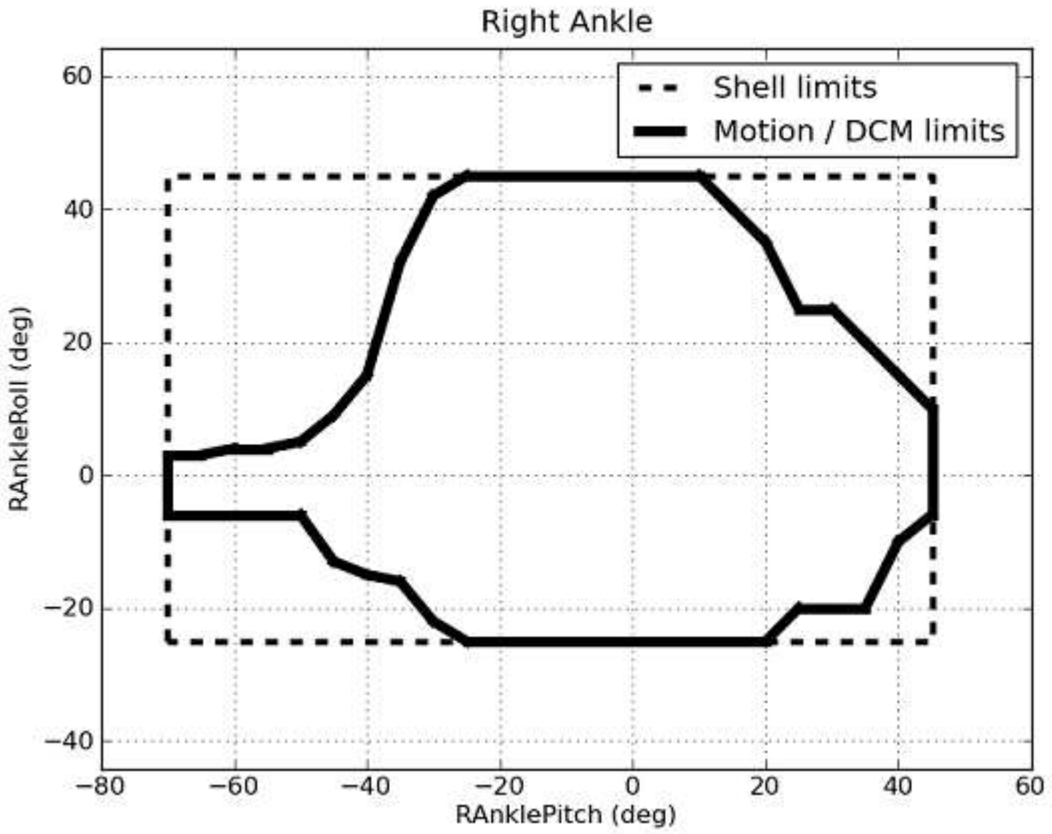
Tai Chi Chuan



Movement of the Right Ankle (roll):

- **Good correspondence, except halfway experiment**
- **Again hardware limits for combination roll / pitch encountered**

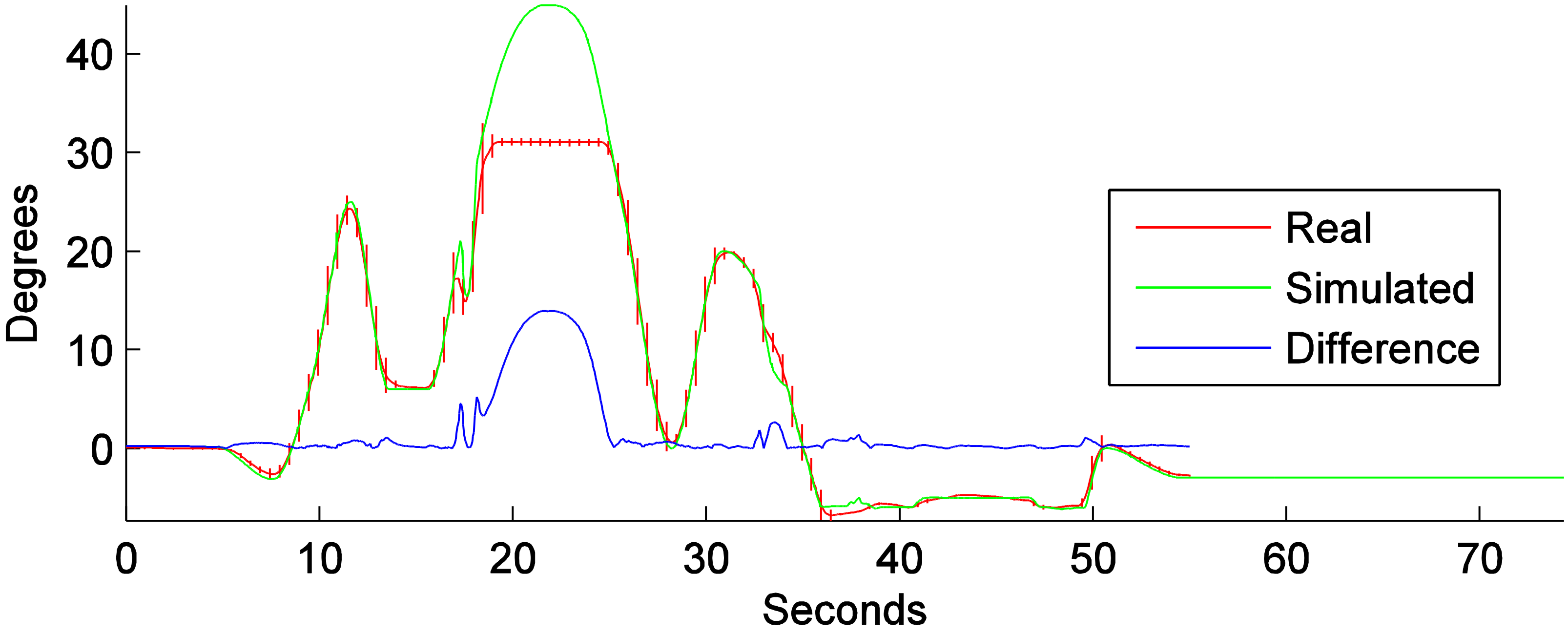
Shell limits



Reason for discrepancy Right Ankle roll during kick:

- **Hardware limits, depended on Right Angle pitch**

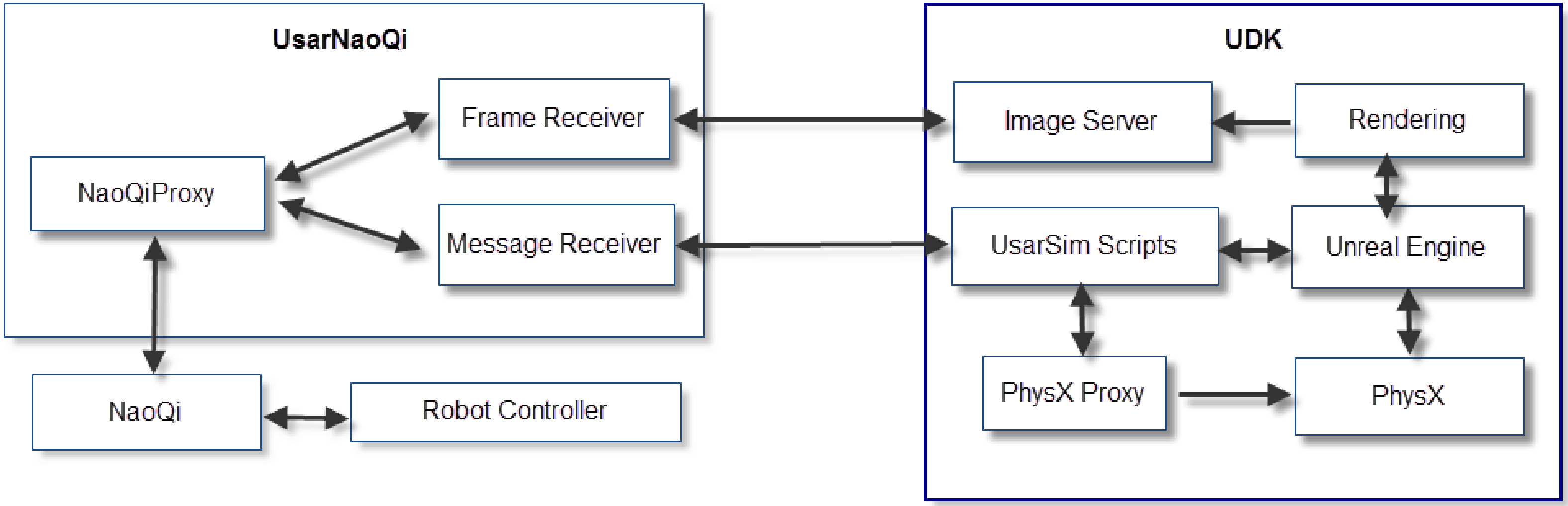
Tai Chi Chuan



Movement of the Right Ankle (roll) for NaoSim:

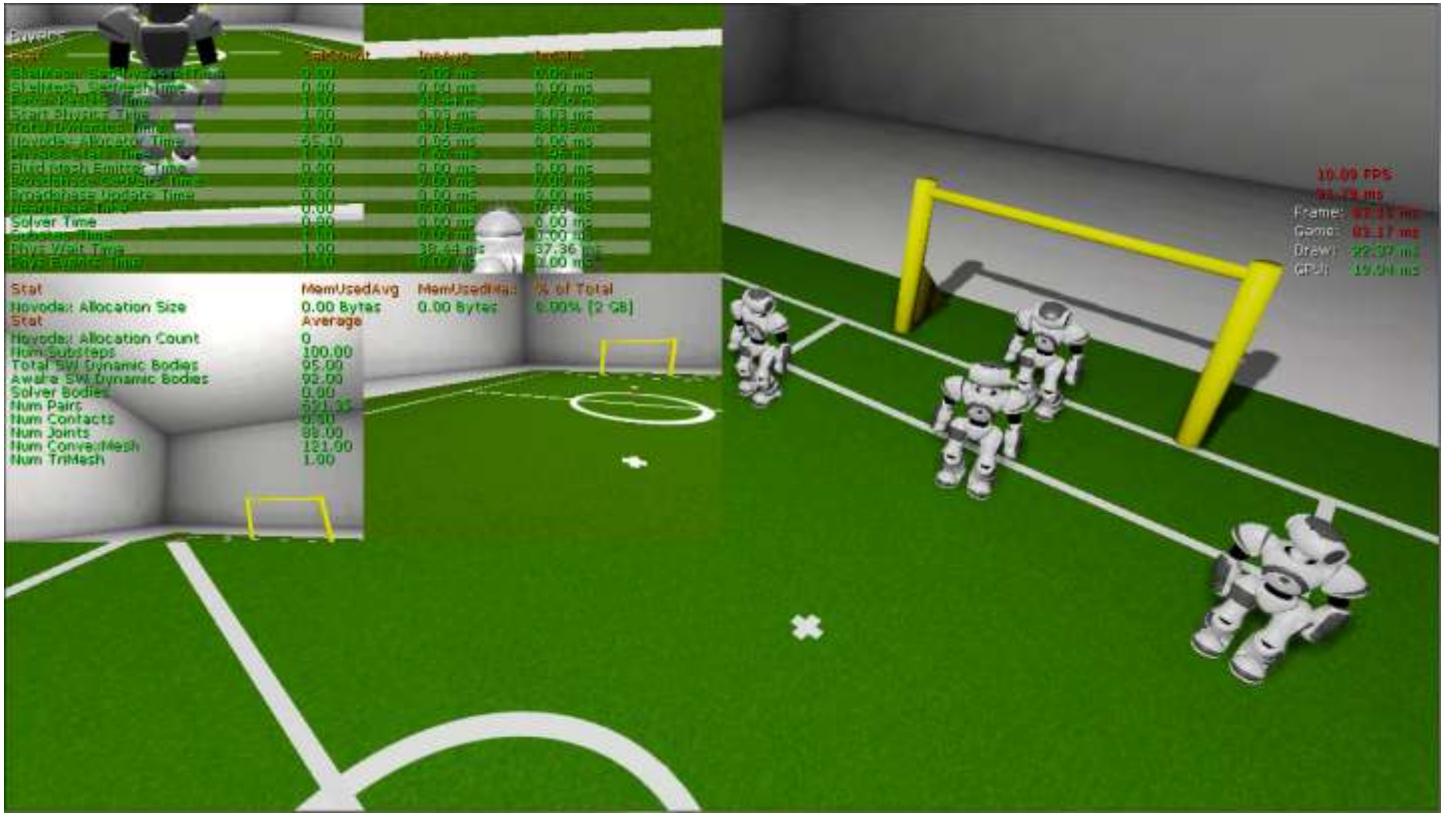
- **Also for the official simulator the hardware limits are not modeled**

Full application



A proxy server was built which allows to command the Nao via its natural interface (NaoQi). NaoQi has e.g. a C++ and Python interface.

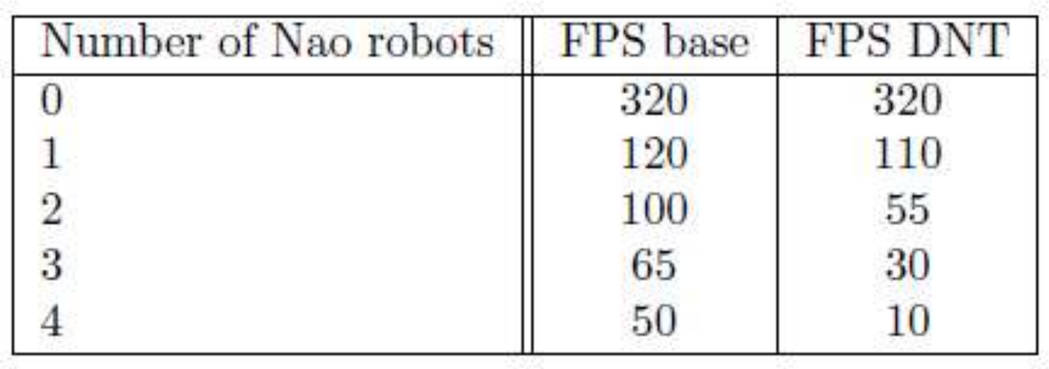
RoboCup Soccer



```
Stat                          MemUsedAvg    MemUsedMax    % of Total
Novodex: Allocation Size      0.00 Bytes    0.00 Bytes    0.00% (2 GB)
Stat                          Average
Novodex: Allocation Count     0
Num Substeps                  100.00
Total SW Dynamic Bodies       95.00
Aware SW Dynamic Bodies       92.00
Solver Bodies                 0.00
Num Pairs                     521.35
Num Contacts                  0.00
Num Joints                    88.00
Num ConvexMesh                121.00
Num TriMesh                   1.00
```

The Python code of an actual RoboCup team (Dutch Nao Team) was used to play a game of soccer.

Solution Scale

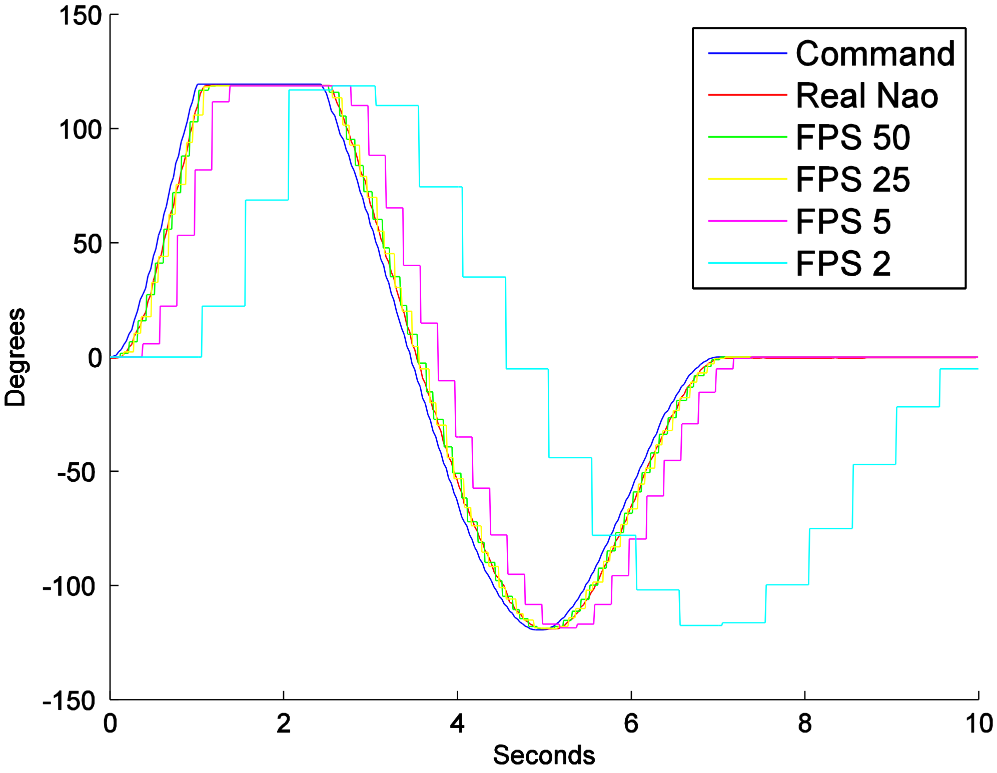


Number of Nao robots	FPS base	FPS DNT
0	320	320
1	120	110
2	100	55
3	65	30
4	50	10

A single computer could simulate up to 4 robots, as long as the robots didn't run around.

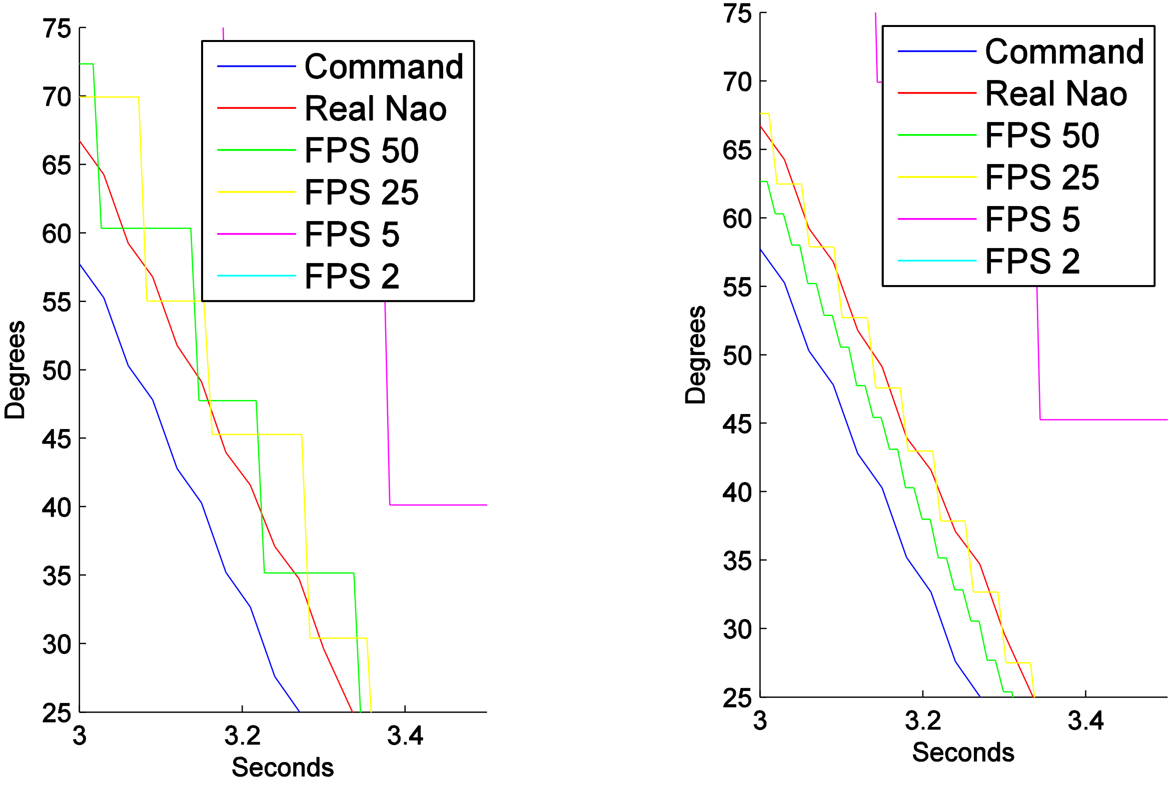
The profiler showed that in the latter case 50% was spent inside the physics engine. The other 50% could be attributed to sensing and message handling.

Acceptable frame rate



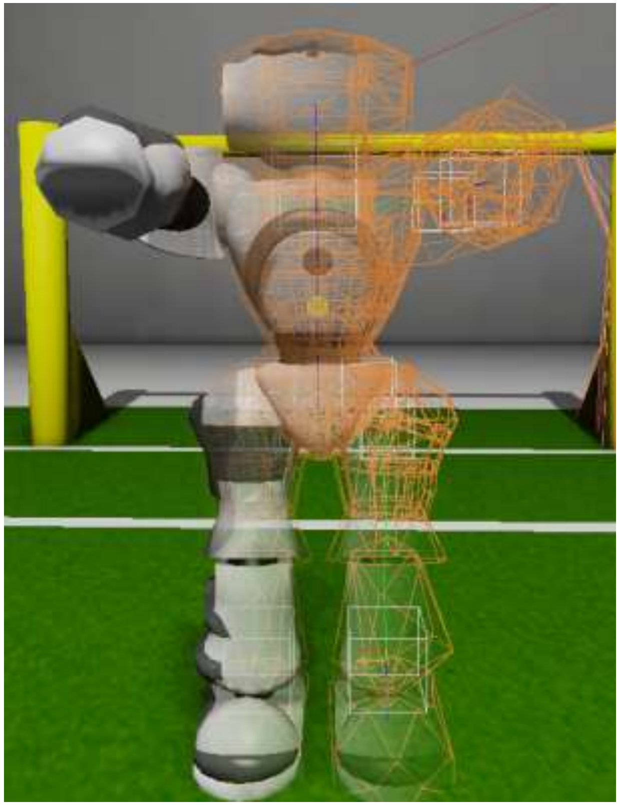
Moving the head with clipped frame rates.

Acceptable frame rate



Details, when moving the head with clipped frame rates (varying status message updates from 100ms to 10ms).

Conclusion

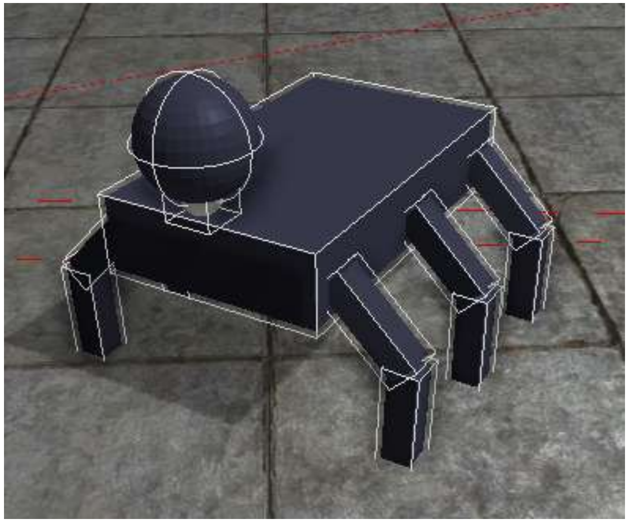


Presented a validated humanoid robot in USARSim UDK

Conclusion

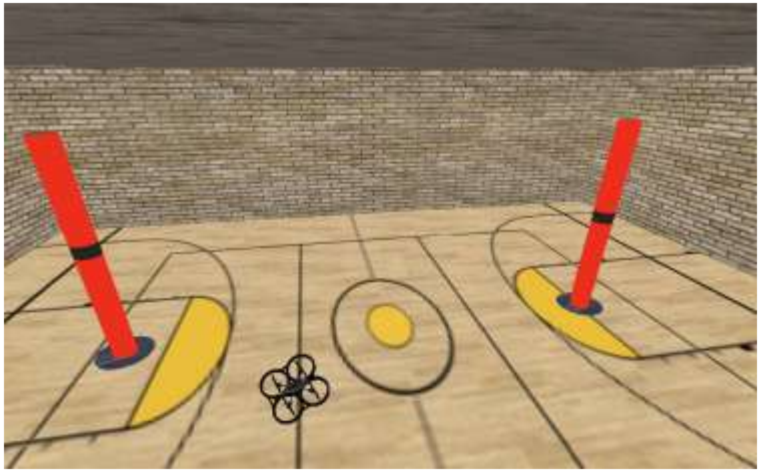
Demonstrated a methodology to validate such robot with a sequence of experiments

Conclusion



Validated the dynamics of multiple kinetic chains in contact with the ground

Closing the gap between simulation and reality in the sensor and motion models of an autonomous AR.Drone



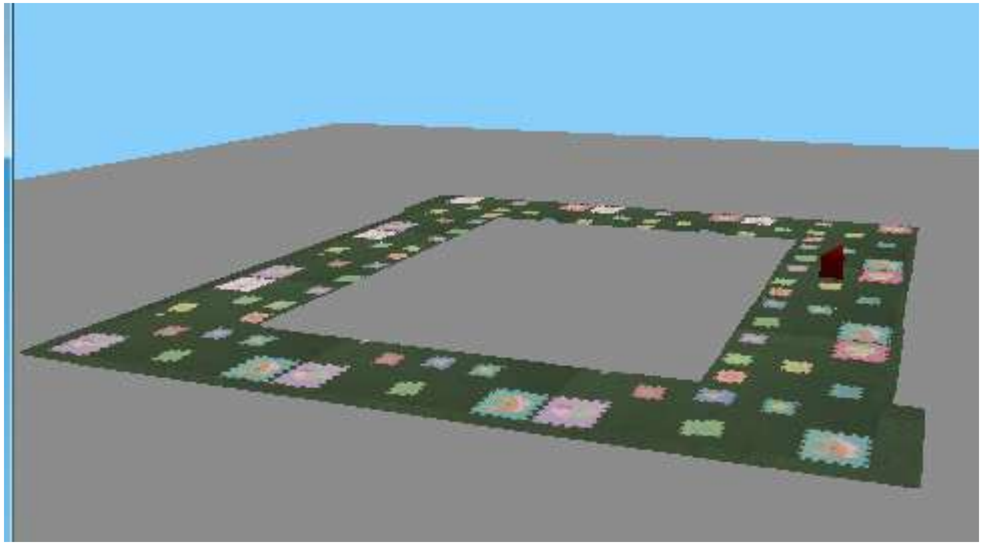
Arnoud Visser, Nick Dijkshoorn,
Martijn van der Veen and Robrecht Jurriaans

Universiteit van Amsterdam
Intelligent Systems Laboratory

International Micro Air Vehicle Conference and Flight Competition
(IMAV11), September 12, 2011

Localization and Mapping challenge:

Visual SLAM combined with sonar and inertia measurements

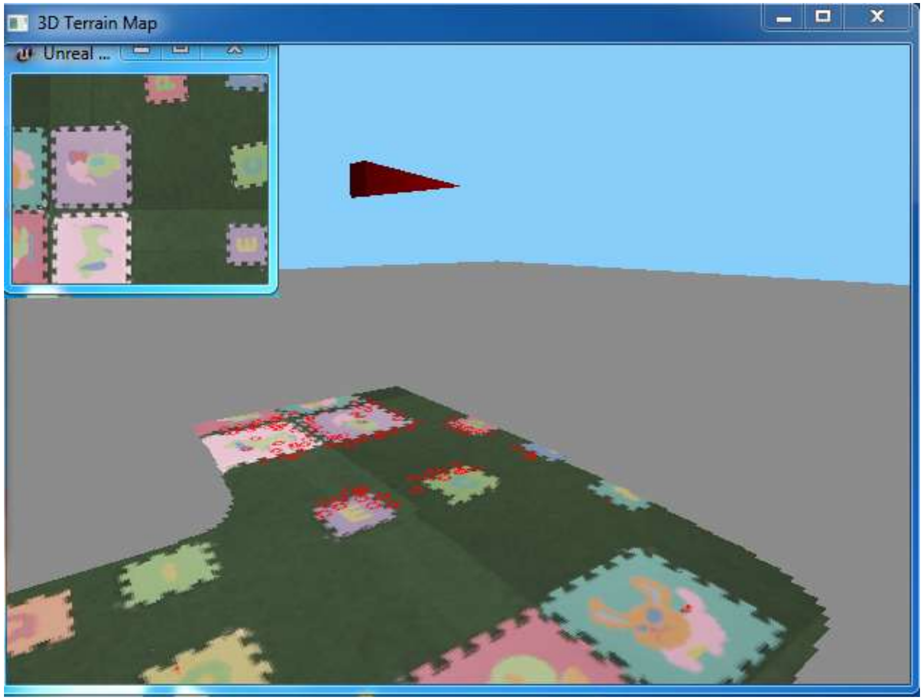


Nick Dijkshoorn

intelligent autonomous systems

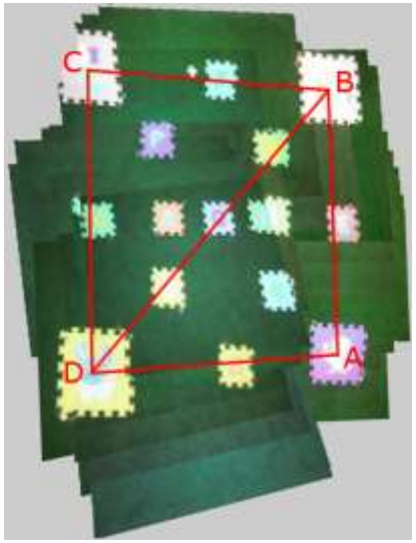
Universiteit van Amsterdam
Informatica Instituut

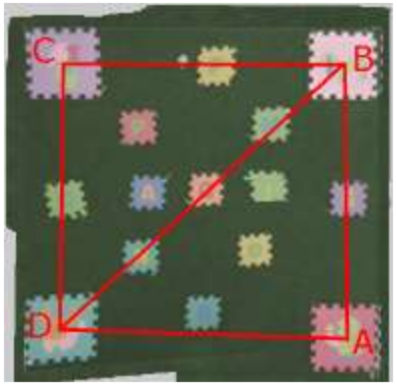
Map Stitching algorithm



- Tracking of SURF features
- RANSAC to classify inliers / outliers
- Back-projection with least-square optimization to estimate the perspective transformation (replaced by an estimate of the camera’s transformation in OpenCV’s SolvePNP)

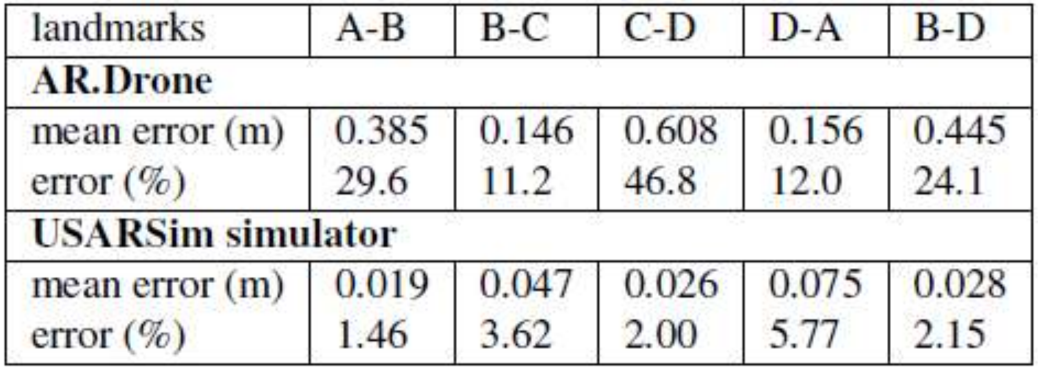
Map Stitching results





Care has been taken to reproduce the real circumstances:

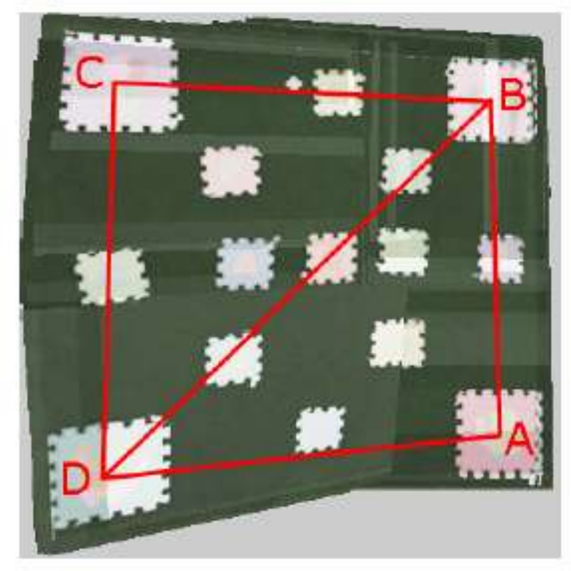
- decreased saturation,
- increased brightness,
- downsampled resolution.

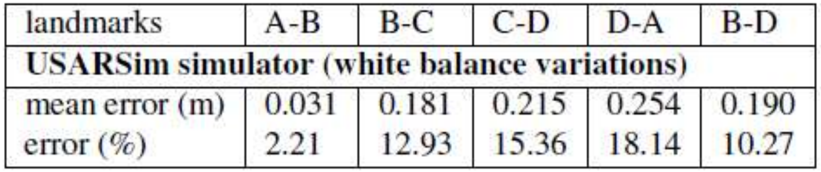


landmarks	A-B	B-C	C-D	D-A	B-D
AR.Drone					
mean error (m) error (%)	0.385 29.6	0.146 11.2	0.608 46.8	0.156 12.0	0.445 24.1
USARSim simulator					
mean error (m) error (%)	0.019 1.46	0.047 3.62	0.026 2.00	0.075 5.77	0.028 2.15

Map Stitching results

In addition, white balance variations are added. Now the average feature distance increases from *22.1px* to *32.9px* (real AR,Drone *32.7px)*

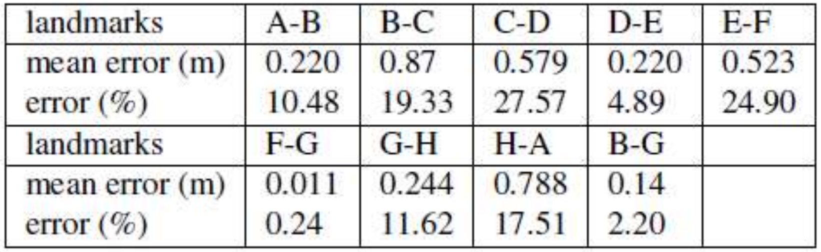




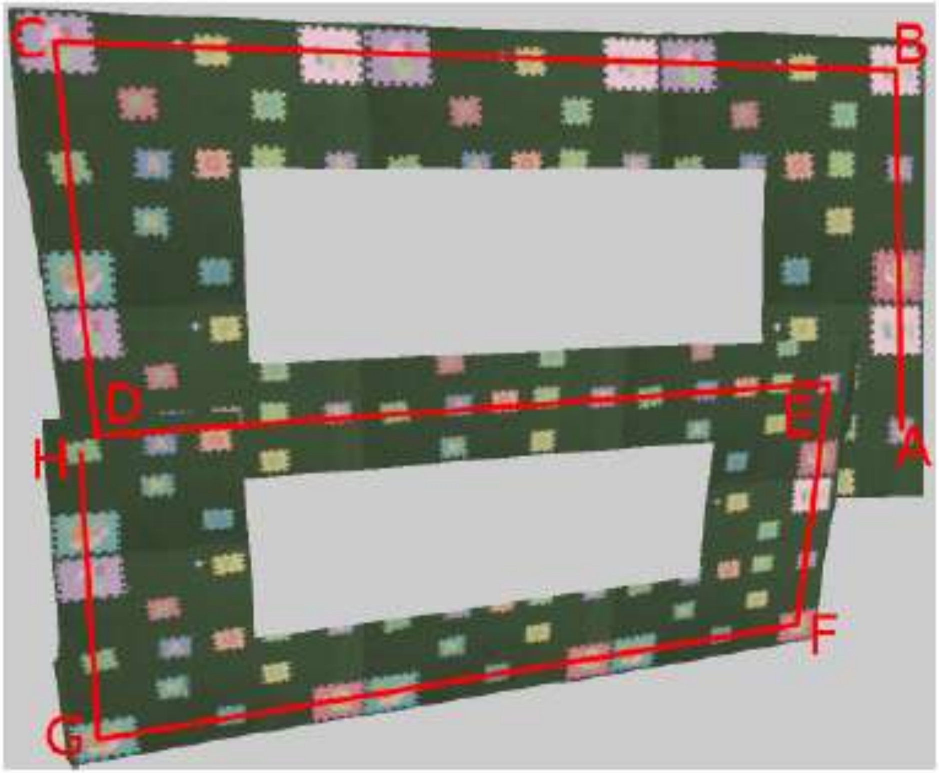
landmarks	A-B	B-C	C-D	D-A	B-D
USARSim simulator (white balance variations)					
mean error (m)	0.031	0.181	0.215	0.254	0.190
error (%)	2.21	12.93	15.36	18.14	10.27

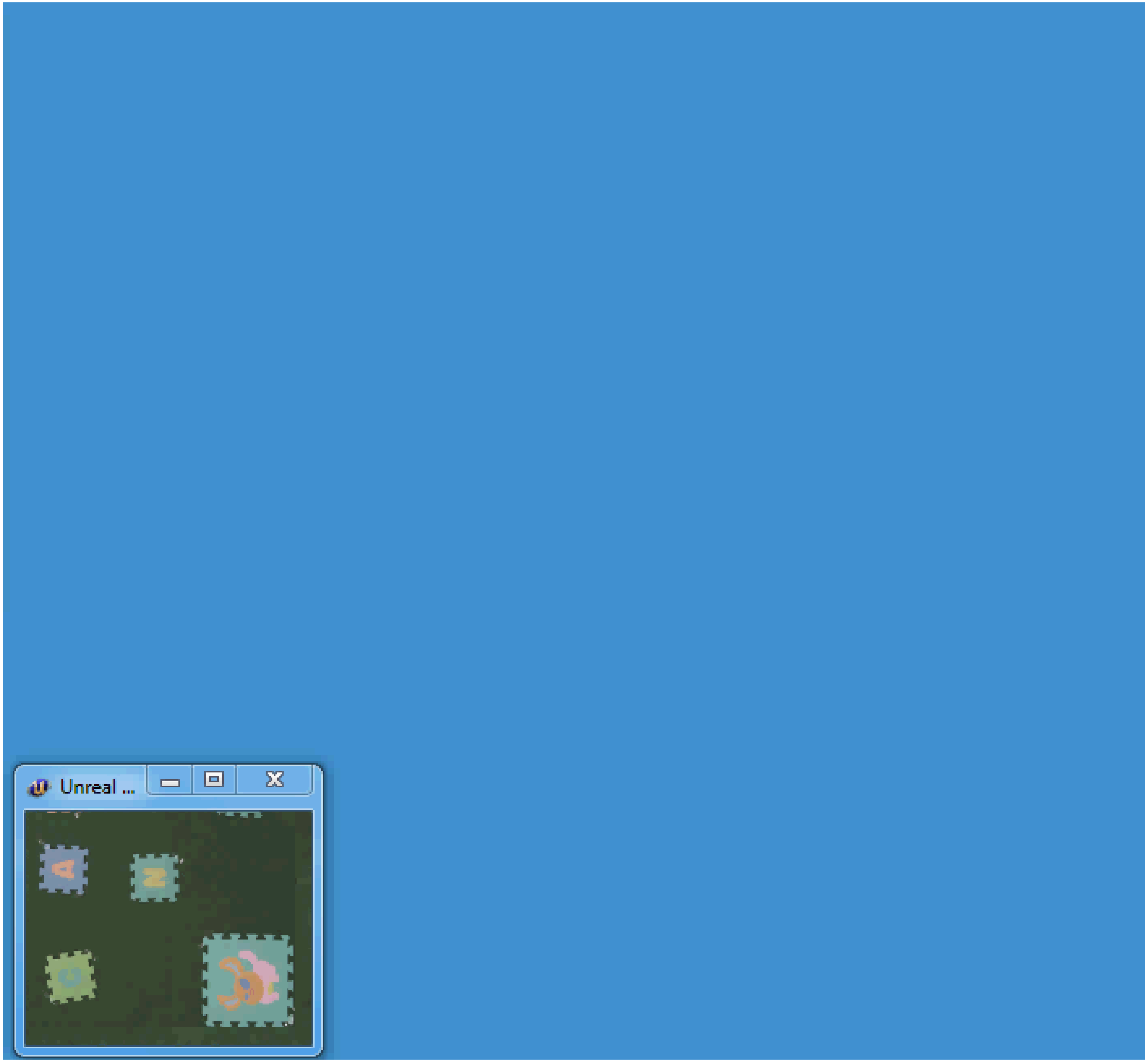
Scaling up results

When the map is scaled up 3x, the lack of global optimization becomes visible.



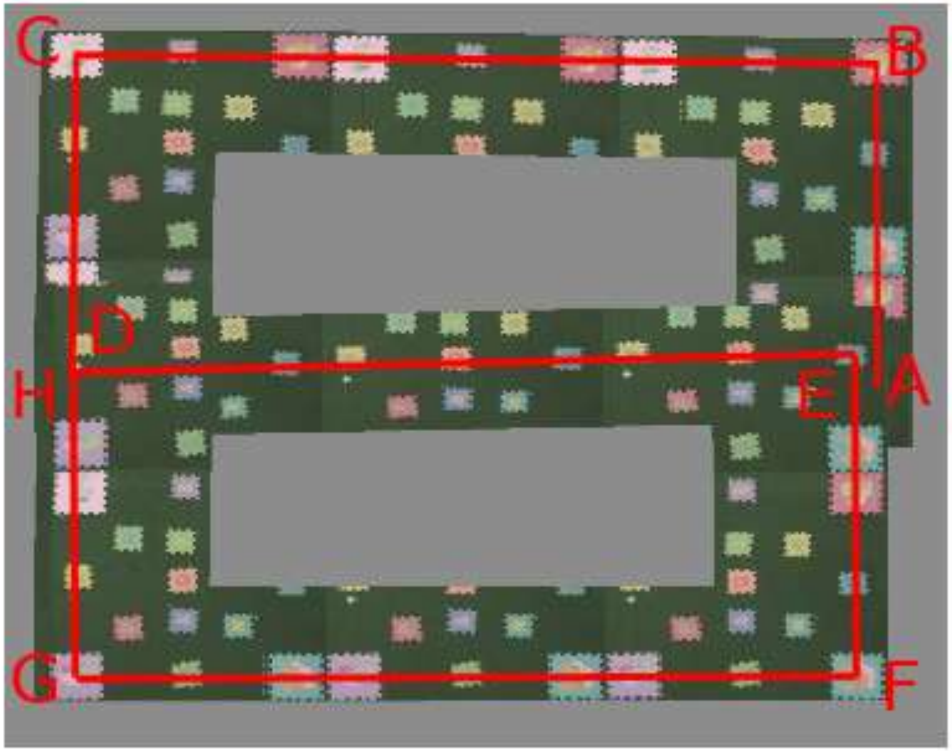
landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m) error (%)	0.220 10.48	0.87 19.33	0.579 27.57	0.220 4.89	0.523 24.90
landmarks	F-G	G-H	H-A	B-G	
mean error (m) error (%)	0.011 0.24	0.244 11.62	0.788 17.51	0.14 2.20	

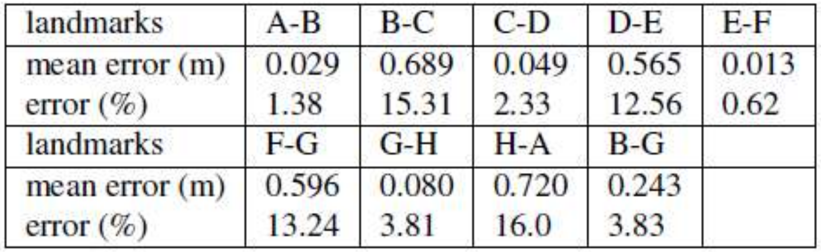



Unreal ...

Additional information results

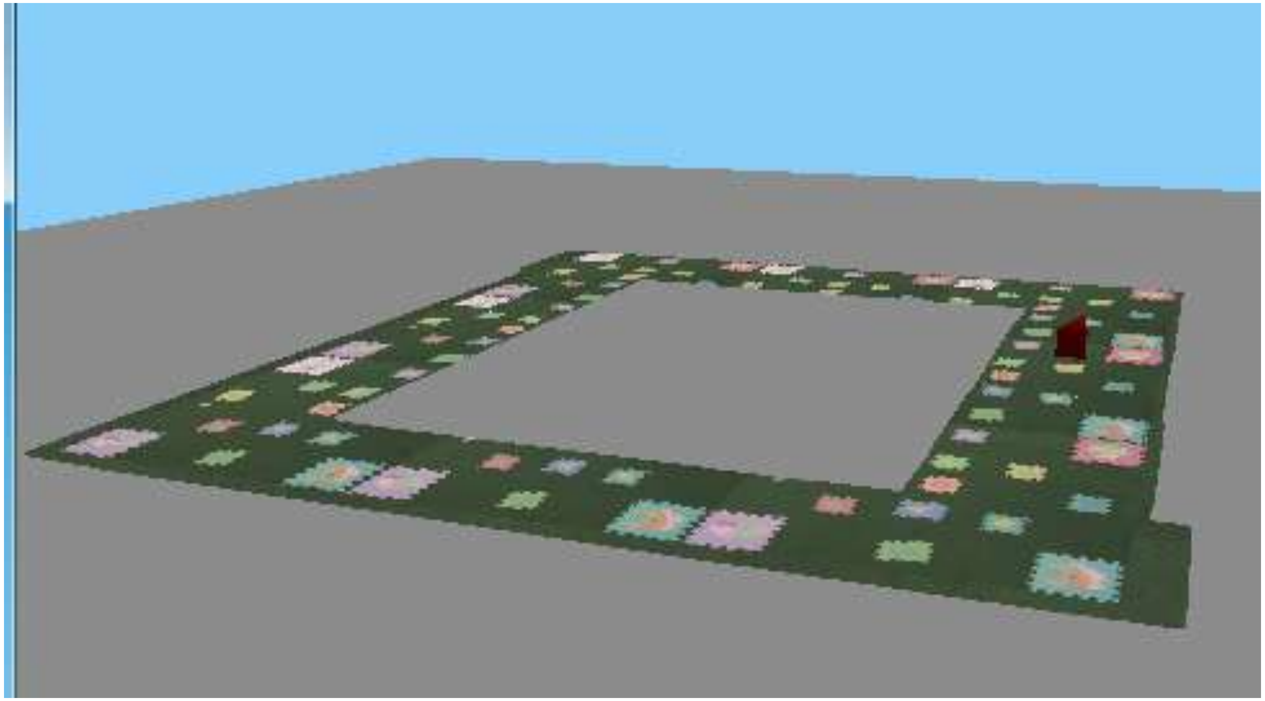
Including information from inertia sensors (using a extended Kalman filter) solves part of the problem.





landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m) error (%)	0.029 1.38	0.689 15.31	0.049 2.33	0.565 12.56	0.013 0.62
landmarks	F-G	G-H	H-A	B-G	
mean error (m) error (%)	0.596 13.24	0.080 3.81	0.720 16.0	0.243 3.83	

Resumé



A visual map can be created with the low resolution bottom camera, which could be used for localization.

Adapting the mapping difficulty for the automatic generation of rescue challenges

Olaf Zwennes, Astrid Weiss, Arnoud Visser

RoboCup IranOpen 2012 Symposium
Tehran, April 4, 2012

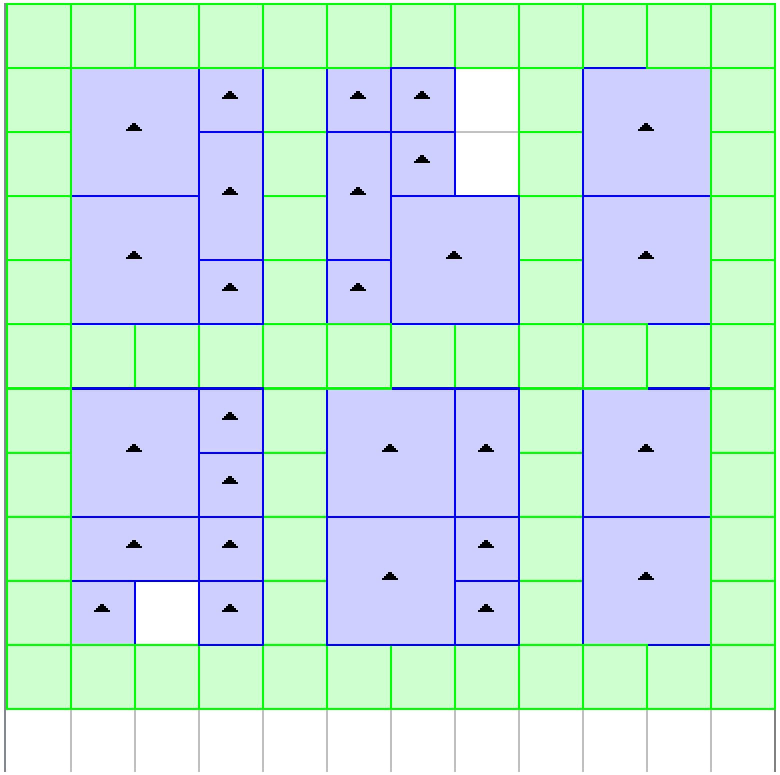
Universiteit van Amsterdam
Intelligent Systems Laboratory

Simulation League

World Generator Tool

Three main rules:

- Hallway rule
- Room rule
- Doorway rule



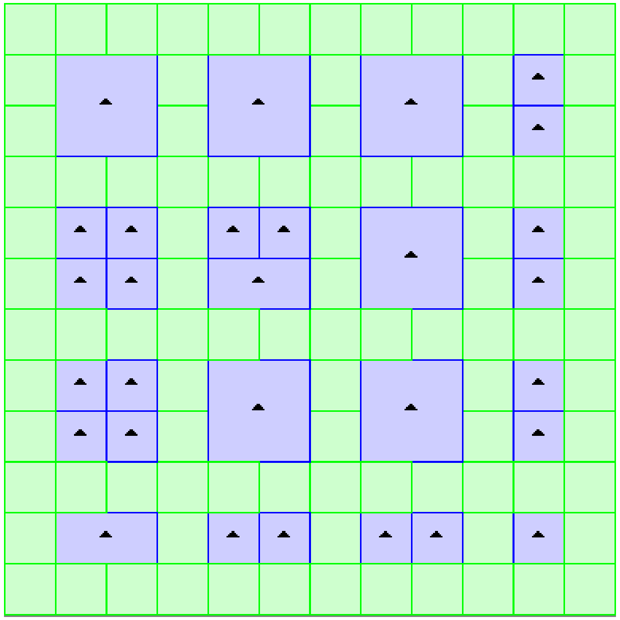
Adaptive Map Generator

Four new rules:

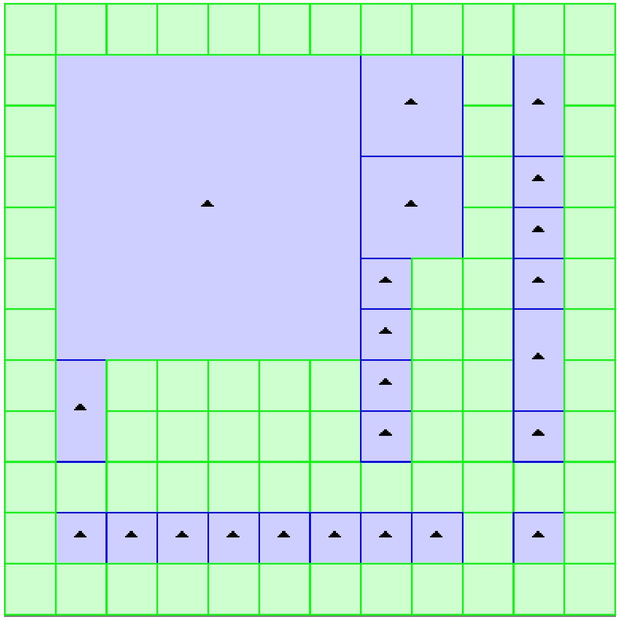
- Variable hallway distance
- Room selection based on difficulty
- Variable doorway distance
- Prevent cycles

Variable hallway distance

$$h = (0.06 \times d + 0.2) \times s + r \times 0.1 \times s$$



Low difficulty: small regular distance



High difficulty: large variable distance

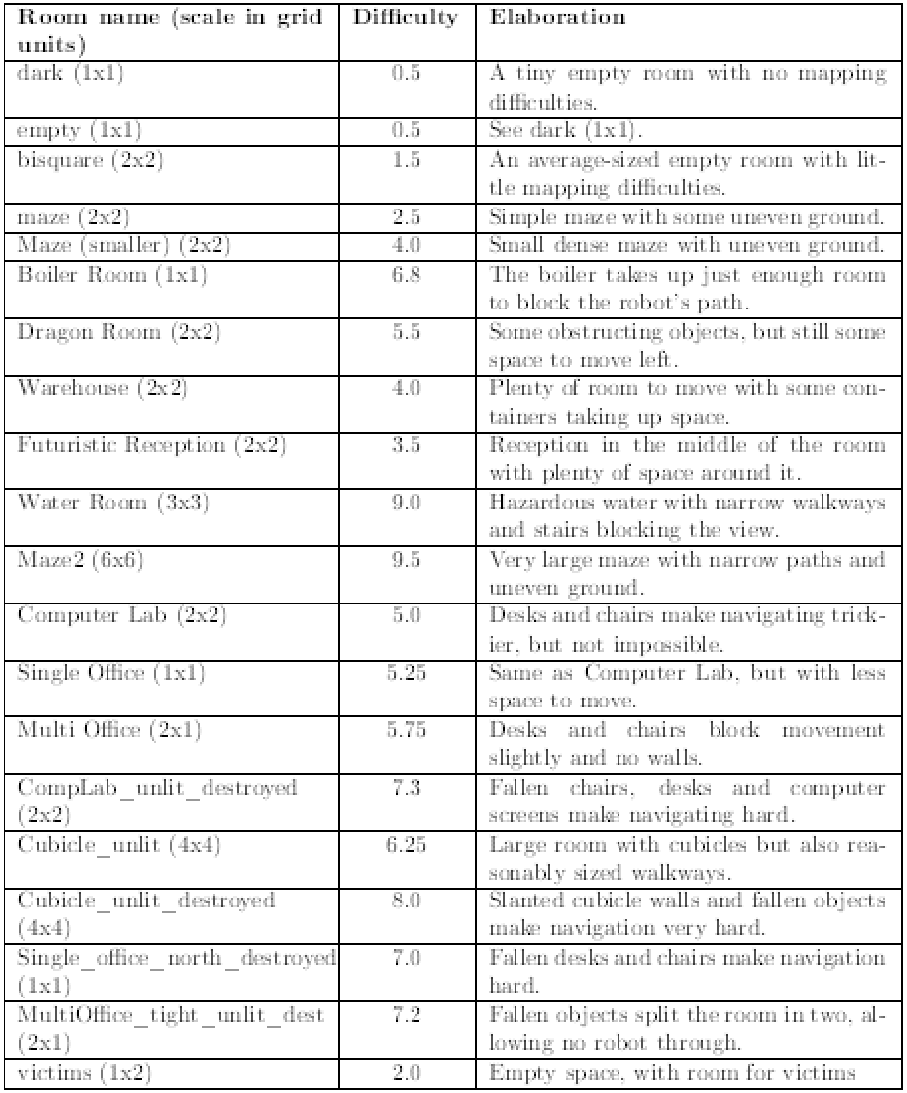
Room selection based on difficulty

$$ISE(room) = \frac{1}{(d - d(room))^2 + 1}$$

$$P(room) = \frac{ISE(room)}{\sum_r^{rooms} ISE(room)}$$

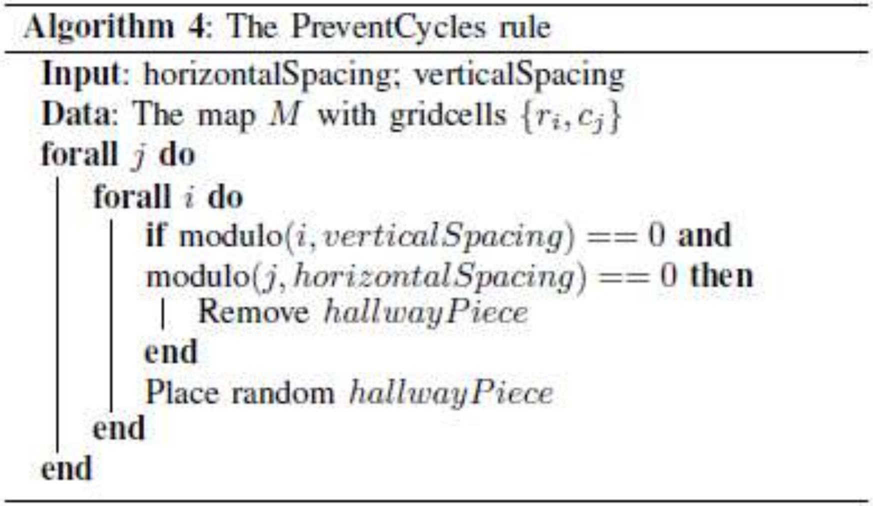
Large room with maze inside; *d(room)=9.5*

Room difficulty



Room name (scale in grid units)	Difficulty	Elaboration
dark (1x1)	0.5	A tiny empty room with no mapping difficulties.
empty (1x1)	0.5	See dark (1x1).
bisquare (2x2)	1.5	An average-sized empty room with little mapping difficulties.
maze (2x2)	2.5	Simple maze with some uneven ground.
Maze (smaller) (2x2)	4.0	Small dense maze with uneven ground.
Boiler Room (1x1)	6.8	The boiler takes up just enough room to block the robot's path.
Dragon Room (2x2)	5.5	Some obstructing objects, but still some space to move left.
Warehouse (2x2)	4.0	Plenty of room to move with some containers taking up space.
Futuristic Reception (2x2)	3.5	Reception in the middle of the room with plenty of space around it.
Water Room (3x3)	9.0	Hazardous water with narrow walkways and stairs blocking the view.
Maze2 (6x6)	9.5	Very large maze with narrow paths and uneven ground.
Computer Lab (2x2)	5.0	Desks and chairs make navigating trickier, but not impossible.
Single Office (1x1)	5.25	Same as Computer Lab, but with less space to move.
Multi Office (2x1)	5.75	Desks and chairs block movement slightly and no walls.
CompLab_unlit_destroyed (2x2)	7.3	Fallen chairs, desks and computer screens make navigating hard.
Cubicle_unlit (4x4)	6.25	Large room with cubicles but also reasonably sized walkways.
Cubicle_unlit_destroyed (4x4)	8.0	Slanted cubicle walls and fallen objects make navigation very hard.
Single_office_north_destroyed (1x1)	7.0	Fallen desks and chairs make navigation hard.
MultiOffice_tight_unlit_dest (2x1)	7.2	Fallen objects split the room in two, allowing no robot through.
victims (1x2)	2.0	Empty space, with room for victims

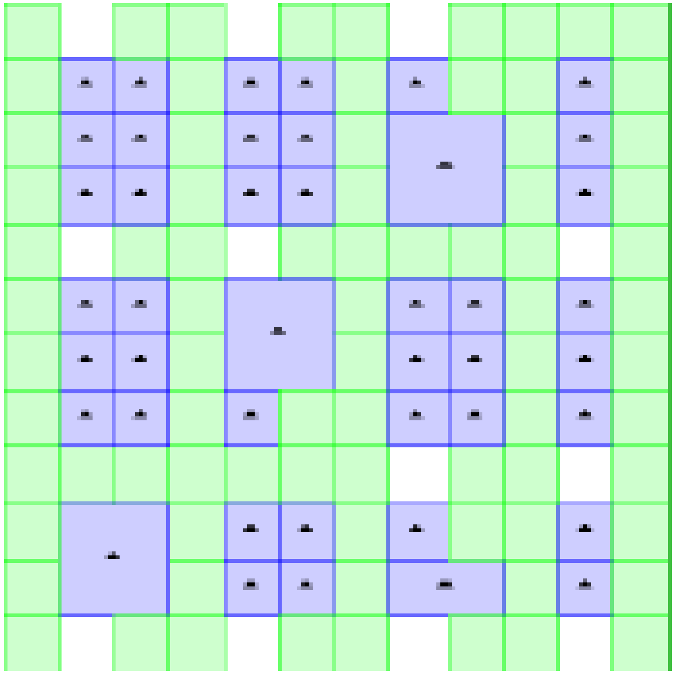
Prevent Cycle rule



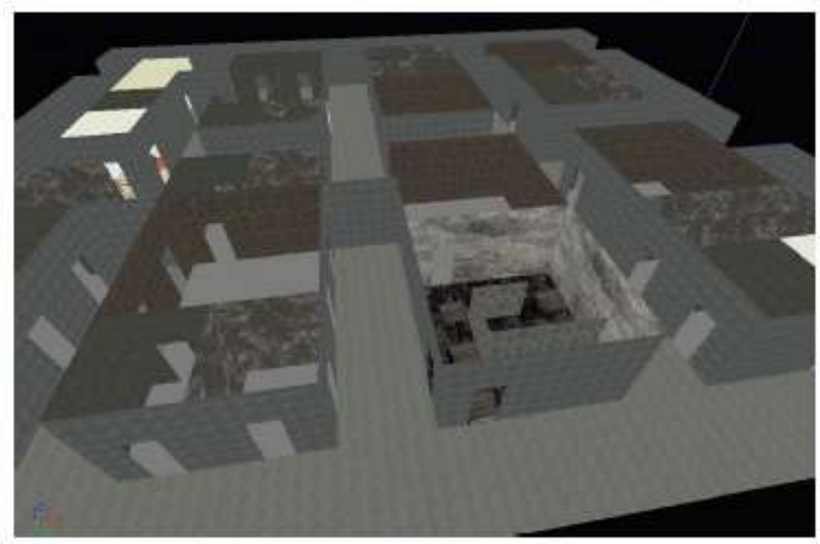
Algorithm 4: The PreventCycles rule

```
Input: horizontalSpacing; verticalSpacing
Data: The map $M$ with gridcells $\{r_i, c_j\}$
forall $j$ do
    forall $i$ do
        if modulo$(i, verticalSpacing) == 0$ and
        modulo$(j, horizontalSpacing) == 0$ then
            Remove $hallwayPiece$
        end
        Place random $hallwayPiece$
    end
end
```

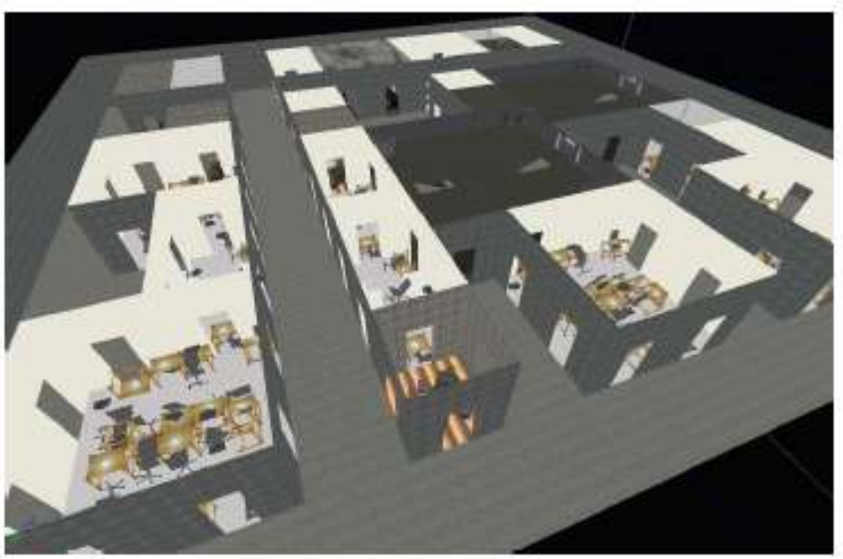
$$P_{PreventCycles} = 1 - \frac{d}{10}$$



Adaptive generated maps



Map generated with difficulty 1



Map generated with difficulty 9

Validation

- Several USARSim operators were asked to explore three maps and estimate the relative and absolute difficulty

- A questionary was designed and filled in by each operator

Questionnaire

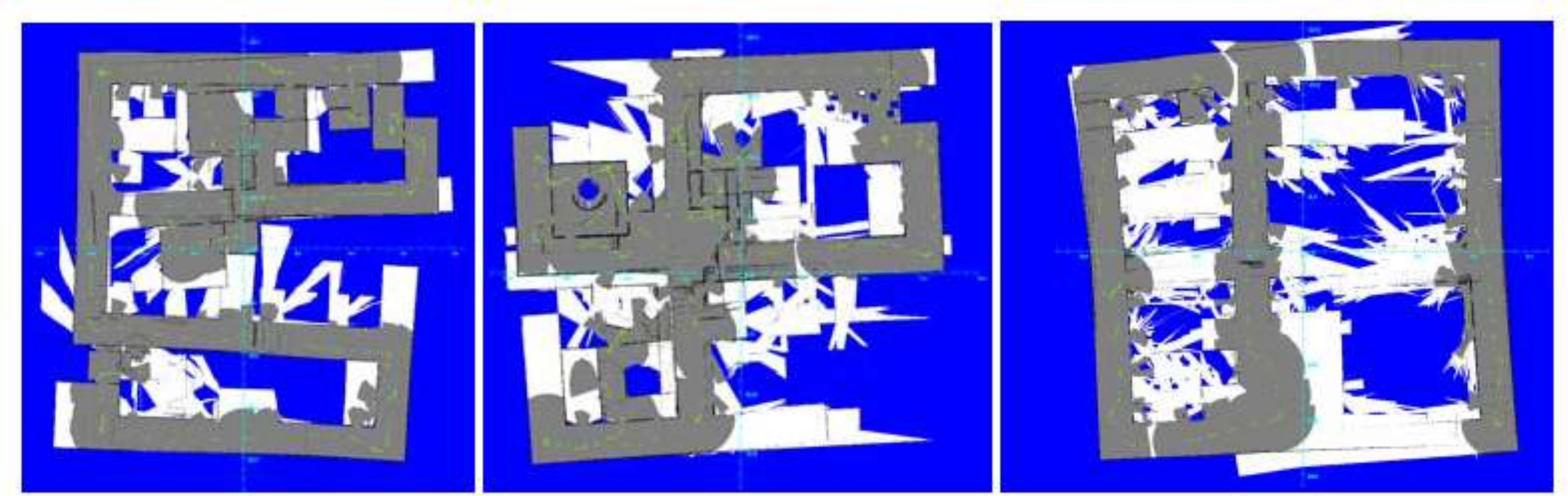
USARSim Robot Mapping Feedback Form

Overall, how would you rate the difficulty of the environment in terms of mapping (scale 0-10):

- *NOTE: 0 difficulty means the environment does not include any aspects that 'force' a mapping error on the robot. 10 difficulty means the map resulting from the mapping process can not be properly navigated, due to excessive mapping errors.*

What aspect(s) of the map contributed most to the difficulty rating given above:

Explorations performed



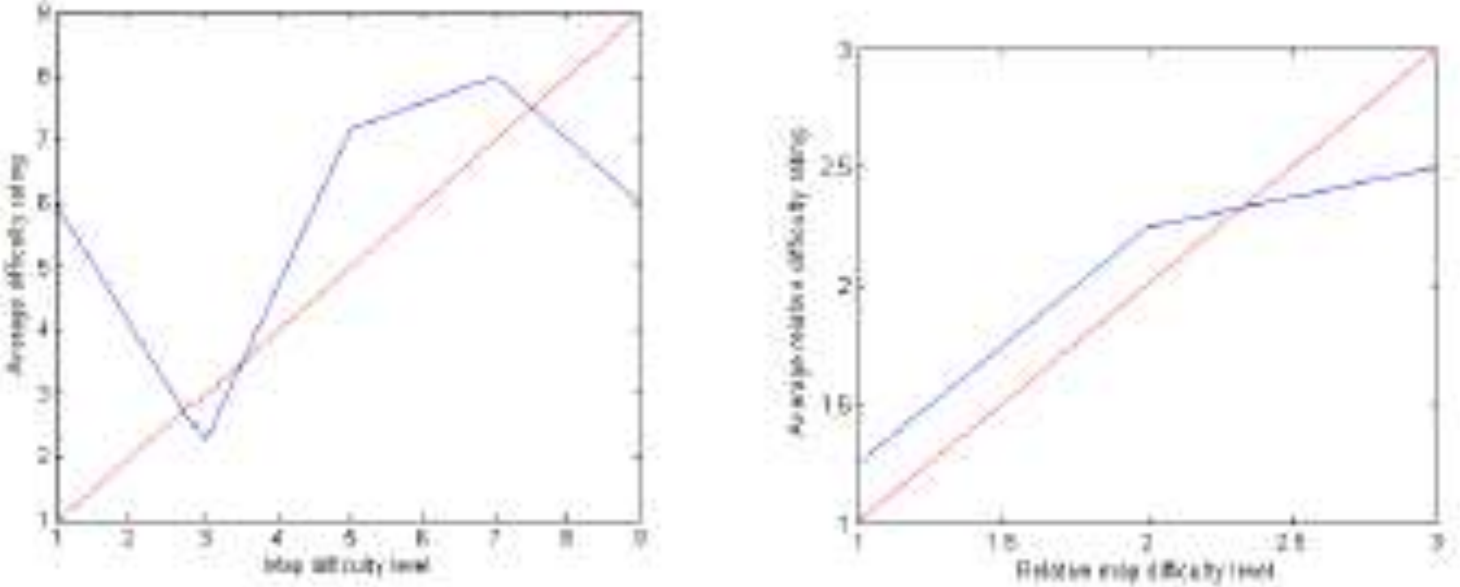
(a) $d = 1$ (b) $d = 3$ (c) $d = 7$

“Dead ends at the corners made the mapping difficult”

“Only one central crossing, which forces you to come repeatable come back at the same location”

“Long straight corridors and square rooms with openings at several sides made it easier to map”

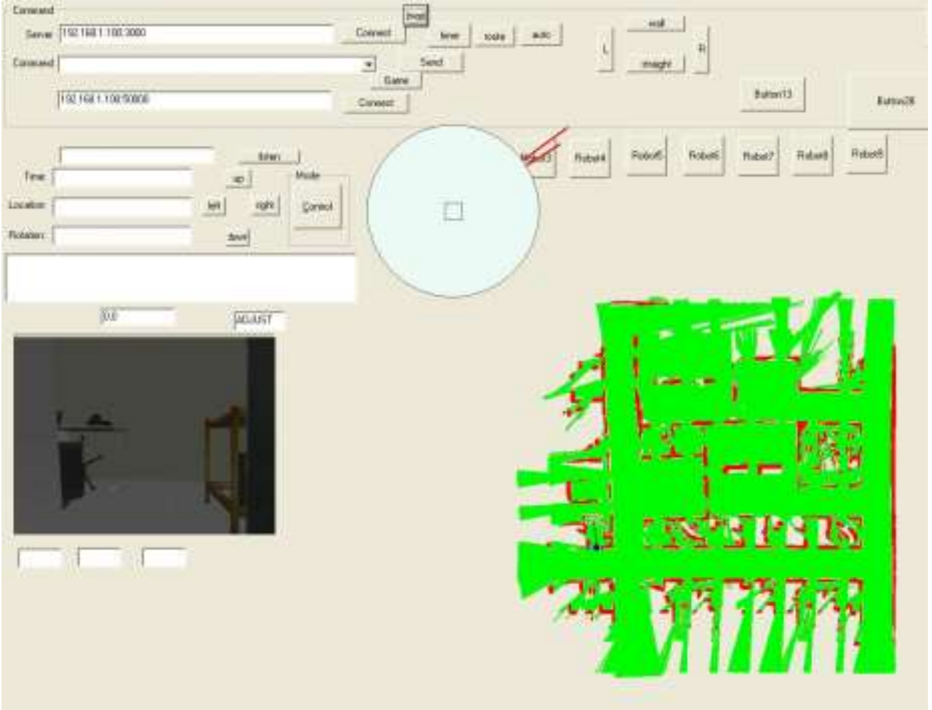
Perceived difficulty



(a) Absolute difficulty level against difficulty rating.

(b) Relative difficulty level against difficulty rating.

Another team



Server: 192.168.1.100:3000

192.168.1.100:50800

0.0

ADJUST

“A long corridor with few rooms. It's hard to locate with the laser data with too few reference object.”

“Ordinary office scene. The slam method is almost designed for this situation...”

Discussion

Four new rules:

± Variable hallway distance

+ Room selection based on difficulty

− Variable doorway distance

− Prevent cycles

Conclusion

Adaptive Map generation can make benchmarks against a certain measure.

This allows:

- Reduces the number experiments to be formed.
- Fair distribution of variance
- The mapping difficulty has become explicit

RoboCup 2006 Bremen·Germany

3rd place

RoboCup Atlanta 2007

4th place

RoboCup German Open

2nd place

Brazil Open

1st place

RoboCup 2009 Graz · Austria

3rd place

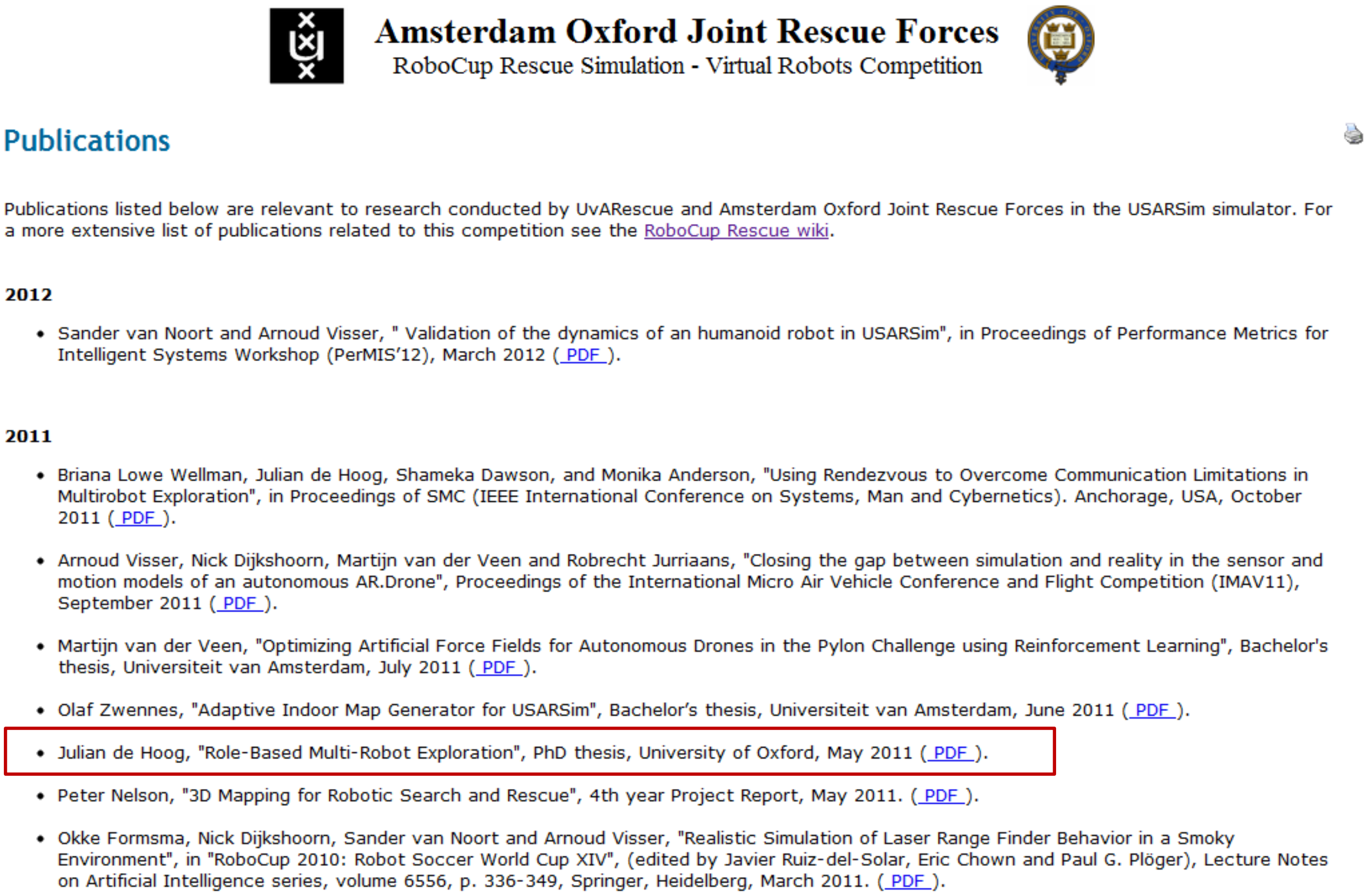
Iran Open 2010

Development price

Iran Open 2011

3rd place

www.jointrescueforces.eu



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.

2012

- Sander van Noort and Arnoud Visser, " Validation of the dynamics of an humanoid robot in USARSim", in Proceedings of Performance Metrics for Intelligent Systems Workshop (PerMIS'12), March 2012 (PDF).

2011

- Briana Lowe Wellman, Julian de Hoog, Shameka Dawson, and Monika Anderson, "Using Rendezvous to Overcome Communication Limitations in Multirobot Exploration", in Proceedings of SMC (IEEE International Conference on Systems, Man and Cybernetics). Anchorage, USA, October 2011 (PDF).
- Arnoud Visser, Nick Dijkshoorn, Martijn van der Veen and Robrecht Jurriaans, "Closing the gap between simulation and reality in the sensor and motion models of an autonomous AR.Drone", Proceedings of the International Micro Air Vehicle Conference and Flight Competition (IMAV11), September 2011 (PDF).
- Martijn van der Veen, "Optimizing Artificial Force Fields for Autonomous Drones in the Pylon Challenge using Reinforcement Learning", Bachelor's thesis, Universiteit van Amsterdam, July 2011 (PDF).
- Olaf Zwennes, "Adaptive Indoor Map Generator for USARSim", Bachelor's thesis, Universiteit van Amsterdam, June 2011 (PDF).
- Julian de Hoog, "Role-Based Multi-Robot Exploration", PhD thesis, University of Oxford, May 2011 (PDF).
- Peter Nelson, "3D Mapping for Robotic Search and Rescue", 4th year Project Report, May 2011. (PDF).
- Okke Formsma, Nick Dijkshoorn, Sander van Noort and Arnoud Visser, "Realistic Simulation of Laser Range Finder Behavior in a Smoky Environment", in "RoboCup 2010: Robot Soccer World Cup XIV", (edited by Javier Ruiz-del-Solar, Eric Chown and Paul G. Plöger), Lecture Notes on Artificial Intelligence series, volume 6556, p. 336-349, Springer, Heidelberg, March 2011. (PDF).