To appear in *Proceedings of the RoboCup International Symposium 2014* (RoboCup 2014), Joao Pessoa, Brazil, July 2014.

UT Austin Villa: RoboCup 2014 3D Simulation League Competition and Technical Challenge Champions

Patrick MacAlpine, Mike Depinet, Jason Liang, and Peter Stone

Department of Computer Science, The University of Texas at Austin
{patmac,msd775,jliang,pstone}@cs.utexas.edu

Abstract. The UT Austin Villa team, from the University of Texas at Austin, won the 2014 RoboCup 3D Simulation League, finishing with an undefeated record. During the course of the competition the team scored 52 goals and conceded none. Additionally the team won the RoboCup 3D Simulation League technical challenge by accumulating the most points over a series of three league challenges: drop-in player, running, and free challenge. This paper describes the changes and improvements made to the team between 2013 and 2014 that allowed it to win both the main competition and the technical challenge.

1 Introduction

UT Austin Villa won the 2014 RoboCup 3D Simulation League for the third time in the past four years, having also won the competition in 2011 [1] and 2012 [2] while finishing second in 2013. During the course of the competition the team scored 52 goals and conceded none along the way to finishing with an undefeated record. Many of the components of the 2014 UT Austin Villa agent were reused from the team's successful previous years' entries in the competition. This paper is not an attempt at a complete description of the 2014 UT Austin Villa agent, the base foundation of which is the team's 2011 championship agent fully described in a team technical report [3], but instead focuses on changes made in 2014 that helped the team reclaim the championship.

In addition to winning the main RoboCup 3D Simulation League competition, UT Austin Villa also won the inaugural RoboCup 3D Simulation League technical challenge consisting of three league challenges: drop-in player, running, and free challenge. This paper also serves to document these challenges and the approaches used by the UT Austin Villa team when competing in the challenges.

The remainder of the paper is organized as follows. In Section 2 a description of the 3D simulation domain is given. Section 3 details changes and improvements to the 2014 UT Austin Villa team including those for kicking and passing, localization, and working with a new robot model having a toe, while Section 4 analyzes the contributions of these changes, and the use of heterogeneous robot types, in addition to the overall performance of the team at the competition. Section 5 describes and analyzes the league challenges that were used to determine the winner of the technical challenge, and Section 6 concludes.

2 Domain Description

The RoboCup 3D simulation environment is based on SimSpark,[1] a generic physical multiagent system simulator. SimSpark uses the Open Dynamics Engine[2] (ODE) library for its realistic simulation of rigid body dynamics with collision detection and friction. ODE also provides support for the modeling of advanced motorized hinge joints used in the humanoid agents.

Games consist of 11 versus 11 agents playing on a 30 meters in length by 20 meters in width field. The robot agents in the simulation are modeled after the Aldebaran Nao robot,[3] which has a height of about 57 cm, and a mass of 4.5 kg. Each robot has 22 degrees of freedom: six in each leg, four in each arm, and two in the neck. In order to monitor and control its hinge joints, an agent is equipped with joint perceptors and effectors. Joint perceptors provide the agent with noise-free angular measurements every simulation cycle (20 ms), while joint effectors allow the agent to specify the torque and direction in which to move a joint.

Visual information about the environment is given to an agent every third simulation cycle (60 ms) through noisy measurements of the distance and angle to objects within a restricted vision cone (120°). Agents are also outfitted with noisy accelerometer and gyroscope perceptors, as well as force resistance perceptors on the sole of each foot. Additionally, agents can communicate with each other every other simulation cycle (40 ms) by sending 20 byte messages.

In addition to the standard Nao robot model, four additional variations of the standard model, known as heterogeneous types, are available for use. These variations, and rules regarding how they may be used, are described in Section 4.2.

3 Changes for 2014

While many components contributed to the success of the UT Austin Villa team, including an optimization framework used to learn low level behaviors for getting up, walking, and kicking via an overlapping layered learning approach [4], the following subsections focus only on those that are new for 2014. Analysis of the performance of these components is provided in Section 4.1.

3.1 Kicking

The 2014 UT Austin Villa agent includes sweeping changes to kicking from the 2013 agent. In addition to learning to kick further from a known starting point by mimicking another agent's existing kick as described in [5], the agent is also now able to reliably kick the ball after taking necessary steps to approach it. This latter improvement is achieved by learning a new kick approach walking parameter set for the team's omnidirectional walk engine, the purpose of which

[1] http://simspark.sourceforge.net/
[2] http://www.ode.org/
[3] http://www.aldebaran-robotics.com/eng/

is to stop within a small bounding box of a target point while guaranteeing that the agent does not overshoot that target. This new parameter set is added to three existing walk engine parameter sets as described in [6]. With this new walk, the agent is able to successfully approach and kick a ball without thrashing around or running into the ball.

The kick approach parameter set updates target walk velocities in the X and Y directions based on the following equation:

$$\begin{aligned} \mathit{desired[X,Y]Vel} = \;& \mathrm{sqrt}(2 * \textsc{maxDecel}[\mathrm{X,Y}] \\ & *(\mathit{distToBall[X,Y]} > 2 * \textsc{buffer}\ ? \\ & \mathit{distToBall[X,Y]} : \mathit{distToBall[X,Y]} - \textsc{buffer})) \end{aligned}$$

The values for $\textsc{maxDecel}[\mathrm{X,Y}]$ and $\textsc{buffer}$ are optimized using the CMA-ES [7] algorithm over a task where the robot walks up to the ball to a position which it can kick the ball from. The robot is given 12 seconds to reach a position where it can kick the ball and is given the following reward during optimization:

$$\begin{aligned} \mathit{reward} = \;& -\mathit{timeTaken}\ \text{(in seconds)} \\ & +\mathit{fFellOver}\ ?\ -1 : 0 \\ & +\mathit{timeTaken} > 12\ \text{seconds}\ ?\ -0.7 : 0 \\ & +\mathit{fRanIntoBall}\ ?\ -0.5 : 0 \\ & +\mathit{velocityWhenInPositionToKick} > 0.005\ \text{m/s}\ ?\ -0.5 : 0 \end{aligned}$$

The 2014 team also adds the ability to legally score on indirect kickoffs. This is the result of a multiagent optimization where one agent uses a static, accurate, long-distance kick and the other attempts to touch the ball while moving it as little as possible [5].

3.2 Passing with Kick Anticipation

When deciding where to kick the ball, the UT Austin Villa agent first checks to see if it can kick the ball and score from the ball's current location. If the agent thinks it can score then it tries to do so. If not, the agent then samples kicking the ball at targets in 10 degree direction increments and, for all viable kicking direction targets (those which don't kick the ball out of bounds or too far backwards), the agent assigns each a score based on Equation 1. Equation 1 rewards kicks for moving the ball toward the opponent's goal, penalizes kicks that have the ball end up near opponents, and also rewards kicks for landing near a teammate. All distances in Equation 1 are measured in meters. The kick direction chosen is the kick whose target ball location receives the highest score. When the agent is close (within 0.8 meters) to the ball its chosen kick direction is fixed and held for 5 seconds to prevent thrashing between kick directions.

$$\begin{aligned} \texttt{score}(\mathit{target}) = \;& -\|\mathit{opponentGoal} - \mathit{target}\| \\ & \forall \mathit{opp} \in \mathit{Opponents}, -\max(25 - \|\mathit{opp} - \mathit{target}\|^2, 0) \\ & + \max(10 - \|\mathit{closestTeammateToTarget} - \mathit{target}\|, 0) \end{aligned} \tag{1}$$

Once an agent has decided on a target to kick the ball at it then broadcasts this target to its teammates. A couple agents then use "kick anticipation" where they run toward locations on the field that are good for receiving the ball based on the ball's anticipated location after it is kicked. The agents assigned to run to these anticipated positions are chosen by a dynamic role assignment system [8].

Kick anticipation[4] was first used by the team in 2012 [2] only when an agent was close to the ball and trying to position around the ball to kick it. New for 2014 kick anticipation has been extended such that an agent going to the ball will broadcast where it intends to kick the ball at any time, not just when close to the ball, as long as requirements for having time to kick the ball instead of dribbling it are met (no opponents are within two meters of the ball and no opponent is closer to the ball than the agent considering kicking it). By extending the amount of time that agents broadcast where they are going to kick the ball before doing so, teammates get more time to run to the anticipated location that the ball is to be kicked in order to receive a pass from the agent kicking the ball. Also new for 2014 teammates avoid getting in the way of the projected trajectory of the ball before it is kicked to prevent them from accidentally blocking the kick.

3.3 Localization with Line Data

In 2013 the UT Austin Villa team employed a particle filter for robot self localization that used only the observations of landmarks (four corner flags and two goal posts at each end of the field) along with odometry updates. It was noticed that sometimes robots would walk out of bounds near the middle of the field, where no landmarks are present, get lost and never return to the field of play. To try and prevent robots getting lost in this way, line information was added to the particle filter, based on previous work by Hester and Stone [9], to improve localization when landmarks are not present. In particular, the longest K observed lines were each compared to known positions of all the lines that exist on the field. Metrics such as the distance between endpoints, acute angle between the lines, and line length ratio were used to determine the similarity of an observed line with each actual line. For each observed line, the highest similarity value was expressed as a probability and used to update particles. Figure 1 shows how line information improves localization accuracy for various values of K.

As there are lines completely surrounding the field, assuming a robot is standing up it should always be able to see at least one line if it is currently on the field. if a robot doesn't see a line for a prolonged period of time (four seconds), the robot automatically assumes that it is now lost and off the field such that the robot just stops and turns in place until it sees a line to relocalize itself. Additionally, if a robot doesn't see any lines, it broadcasts to its teammates that it is not localized. If any teammates see a robot that reports itself as not being localized they will broadcast the current x, y position and (new for 2014) angle of orientation of the unlocalized robot so that it may use other robots' observations to localize itself. Empirically we have found that after incorporating line data into localization our agents no longer get lost when leaving the field.

[4] Videos of kick anticipation being used for passing can be found at `http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/AustinVilla3DSimulationFiles/2014/html/kickanticipation.html`

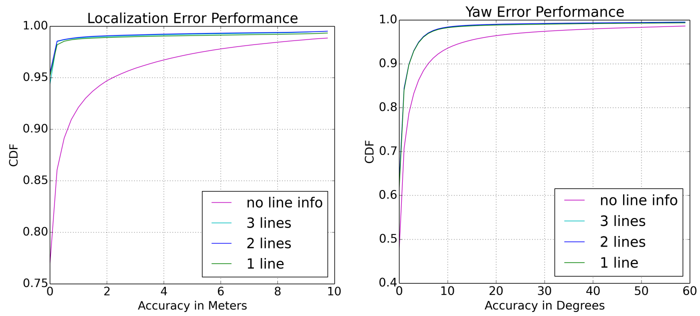

Fig. 1: CDF of localization error (left) and yaw error (right) for using K=1,2,3 when incorporating line information. For comparison, not using line information (purple line) is shown as well.

3.4 Integration of Robot Toe Model

For the 2014 competition two new heterogeneous robot models were introduced including a robot model with a toe joint (known as a Type 4 model). Therefore, we modified the walk engine to use this added joint. The walk engine is described in depth in previous work [6]. The only modifications made to take advantage of the new joint was to add an offset to both the ankle pitch and to the new toe joint. We decided to alter the ankle pitch in addition to the toe joint as the ankle pitch can counteract the toe joint's effect on the robot's center of mass. This correction allows the remainder of the walk engine to perform as designed, resulting in a well-tuned walk. The offset to both joints takes the form of

$$\text{offset} = a\cos(t\pi + p) + c$$

where a is the amplitude of the movement, p controls the phase, and c is a constant offset. We chose to use a sinusoidal curve to maintain smooth movement that repeats once per step. The parameters for the ankle pitch and toe joint are not linked, resulting in an additional 6 parameters in the walk engine. These parameters were then optimized using CMA-ES as described in [6].

4 Main Competition Results and Analysis

In winning the 2014 RoboCup competition UT Austin Villa finished with an undefeated record of 13 wins and 2 ties.[5] During the competition the team scored 52 goals without conceding any. Despite finishing with an undefeated record, the relatively few number of games played at the competition, coupled with the complex and stochastic environment of the RoboCup 3D simulator, make it difficult to determine UT Austin Villa being better than other teams by a statistically significant margin. At the end of the competition, however, all teams were required to release their binaries used during the competition. Results of UT Austin Villa playing 1000 games against each of the other 11 teams' released binaries from the competition are shown in Table 1.

[5] Full tournament results can be found at `http://wiki.robocup.org/wiki/Soccer_Simulation_League/RoboCup2014#3D`

Opponent	Place	Avg. Goal Diff.	Record (W-L-T)	Goals (F/A)	KO Score %
BahiaRT	5-8	2.075 (0.030)	990-0-10	2092/17	96.2
FCPortugal	4	2.642 (0.034)	986-0-14	2748/106	83.4
magmaOffenburg	3	2.855 (0.035)	990-0-10	2864/9	88.3
RoboCanes	2	3.081 (0.046)	974-0-26	3155/74	69.4
FUT-K	5-8	3.236 (0.039)	998-0-2	3240/4	96.3
SEU_Jolly	5-8	4.031 (0.062)	995-0-5	4034/3	87.6
KarachiKoalas	9-12	5.681 (0.046)	1000-0-0	5682/1	87.5
ODENS	9-12	7.933 (0.041)	1000-0-0	7933/0	92.1
HfutEngine	5-8	8.510 (0.050)	1000-0-0	8510/0	94.7
Mithras3D	9-12	8.897 (0.041)	1000-0-0	8897/0	90.4
L3M-SIM	9-12	9.304 (0.043)	1000-0-0	9304/0	93.7

Table 1: UT Austin Villa's released binary's performance when playing 1000 games against the released binaries of all other teams at RoboCup 2014. This includes place (the rank a team achieved at the competition), average goal difference (values in parentheses are the standard error), win-loss-tie record, goals for/against, and the percentage of own kickoffs which the team scored from.

UT Austin Villa finished with at least an average goal difference greater than two goals against every opponent. Additionally UT Austin Villa did not lose a single game out of the 11,000 that were played in Table 1. This shows that UT Austin Villa winning the 2014 competition was far from a chance occurrence. The following subsection analyzes some of the components described in Section 3 that contributed to the team's dominant performance.

4.1 Analysis of Components

		Not Scoring on Kickoff			
Opponent	UTAustinVilla	NoScoreKO	NoKickAnt	Dribble	NoLines
NoScoreKO	0.82 (0.03)	—	-0.20 (0.03)	-0.12 (0.03)	-0.32 (0.03)
BahiaRT	2.08 (0.03)	1.35 (0.03)	1.36 (0.03)	1.36 (0.03)	1.02 (0.03)
FCPortugal	2.64 (0.03)	2.13 (0.04)	1.98 (0.03)	1.69 (0.03)	1.76 (0.04)
magmaOffenburg	2.85 (0.03)	2.20 (0.04)	2.10 (0.03)	2.29 (0.04)	1.91 (0.03)
RoboCanes	3.08 (0.05)	2.54 (0.05)	1.77 (0.04)	1.53 (0.03)	2.09 (0.04)

Table 2: Average goal difference (standard error shown in parentheses) achieved by different versions of the UT Austin Villa team (rows) when playing 1000 games against both top opponents at the RoboCup 2014 competition and a version of the UT Austin Villa team which does not try and score on kickoffs (NoScoreKO).

Table 2 shows the average goal difference achieved by the following different versions of the UT Austin Villa team when playing 1000 games against top opponents at RoboCup 2014 as well as a version of itself that does not try and score on kickoffs (NoScoreKO).

UTAustinVilla Released binary that does attempt to score on kickoffs.
NoScoreKo Does not try and score on kickoffs but instead attempts to kick the ball as far as possible toward the opponent's goal posts without scoring.
NoKickAnt Same as NoScoreKO but does not use kick anticipation.
Dribble Same as NoScoreKO but always dribbles the ball and never kicks except for free kicks (e.g. goal kicks, corner kicks, kick-ins).
NoLines Same as NoScoreKO but does not use any line observation data for localization.

When comparing the performance of UTAustinVilla, which tries to score on kickoffs, to that of NoScoreKO, which does not attempt to score on kickoffs, we see at least a half goal advantage for UTAustinVilla against all opponents. This is not surprising as Table 1 shows that the kickoff was able to score against almost all opponents over 80% of the time. The only two opponents that the team had lower kickoff scoring percentages against were RoboCanes (69.4%) and NoScoreKO (73.7%). The ability to reliably score directly off the kickoff is a huge advantage. If it were possible to score 100% of the time on kickoffs it would be almost impossible to lose — a team would quickly score on ensuing kickoffs right after their opponent scored in addition to scoring right at the beginning of a half when given the initial kickoff. Being able to score on kickoffs was a critical factor for the UT Austin Villa not to lose any games in Table 1 as the NoScoreKO team lost 10 games when playing 1000 games against all opponents (5 to RoboCanes, 4 to BahiaRT, and 1 to SEU_Jolly).

Advantages of using kick anticipation for passing described in Section 3.2 can be seen in Table 2 when evaluating the drop in performance from NoScoreKo to NoKickAnt, and gains in the ability to kick during game play by using the new walk approach detailed in Section 3.1 are evident in the performance drop between NoScoreKo and Dribble. Noticeable gains for using kicking and passing are seen against NoScoreKO, FCPortugal, and RoboCanes. This contrasts with the performance of previous year's teams when kicking was found to be detrimental [1] or negligible [2] to the team's performance. We believe the reason we do not see the same gains in performance against BahiaRT and magmaOffenburg is because they do not have as long kicks and kickoffs as the other opponents in Table 2, and thus do not kick the ball as much into open space where our kicking and passing components can best be utilized.

Using line data for localization as discussed in Section 3.3 is shown in the performance drop from NoScoreKO to NoLines. The incorporation of line data improved performance against all opponents as it helps to prevent agents from getting lost and wandering off the field.

4.2 Heterogeneous Types

At the RoboCup competition teams were given the option of using five different robot types with the requirement that at least three different types of robots must be used on a team, no more than seven of any one type, and no more than nine of any two types. The five types of robots available were the following:

Type 0: Standard Nao model
Type 1: Longer legs and arms
Type 2: Quicker moving feet
Type 3: Wider hips and longest legs and arms
Type 4: Added toes to foot

	Type 0	Type 1	Type 2	Type 3	Type 4
Maximum General Walk Speed (m/s)	0.74	0.79	0.75	0.83	0.84
Maximum Sprint Walk Speed (m/s)	0.78	0.94	0.78	0.97	0.89
Median Kick Length (m)	16	17	16	15	18
Maximum Kick Length (m)	19	20	19	17	20

Table 3: Maximum speeds for general and sprint parameter sets, as well as median and maximum kick distances, for each of the different heterogeneous robot types used at the RoboCup 2014 competition.

Table 3 shows performance metrics after optimizing both walks [6] and kicks [5] for all robot types. The general walk speed is for moving to different target positions on the field while the sprint speed is for walking forward to targets within 15° of a robot's current heading. The median and maximum kick lengths are approximate values used by the robots to estimate where the ball will travel after it is kicked. Types 1 and 3 with longer legs have the fastest walking speeds. Type 4 with a toe is also relatively quick and has longer and more robust kicking than the other robot types.

	Not Scoring on Kickoff					No Kicking	
Opponent	Type 0	Type 1	Type 2	Type 3	Type 4	Type 4	Type 4 NoToe
NoScoreKO	-0.51	-0.57	-0.35	-0.56	0.32	-0.19	-0.53
BahiaRT	0.78	1.18	1.26	1.16	1.62	1.10	0.88
FCPortugal	1.30	1.67	1.89	1.58	2.50	1.37	1.06
magmaOffenburg	1.40	1.47	1.73	1.48	2.81	2.21	1.65
RoboCanes	1.13	2.01	1.90	1.98	3.13	1.38	1.11

Table 4: Average goal difference achieved by teams using different heterogeneous types (rows) when playing 1000 games against both top opponents at the RoboCup 2014 competition and a version of the UT Austin Villa team which does not try and score on kickoffs (NoScoreKO). All standard error values are in the range 0.03 to 0.05.

Table 4 shows the results of playing teams consisting of all the same robot type against different opponents in order to isolate the performance of the different robot types. All the "not scoring on kickoff" teams do not try and score on the kickoff, but instead have the same behavior as NoScoreKO, so as to not have a robot type's ability to score on kickoffs overshadow the rest of its performance. From the data we see that Type 4 performs much better than the other types. For this reason we used seven Type 4 agents in the final round of the competition (the maximum number possible). We also used two Type 0 agents as they were the best at scoring on kickoffs. Our final two robots types used were a Type 3 because it could run the fastest, and a Type 1 robot as our goalie due to its larger body useful for blocking shots and good long kicks for goal kicks. The above number of robot types used are the same as those used in NoScoreKO.

To evaluate the incorporation of the Type 4 robot body's toe into our omnidirectional walk engine (detailed in Section 3.4), we optimized a walk for the type 4 robot type that kept the toe at a fixed default flat position as if the toe joint did not exist (Type 4 NoToe). Table 4 compares the results of Type 4 NoToe to that of Type 4 with the toe integrated into our omnidirectional walk engine. The "no kicking" teams using these walks for comparison were not allowed to kick, except for kicking the ball on kickoffs but not trying to score when doing so, so as to isolate the utility of walking and dribbling performed by the walk engine. Type 4 had better performance than Type 4 NoToe against all opponents revealing that integrating the toe joint into the omnidirectional walk engine was useful. Additionally Type 4 NoToe, with a maximum general walk speed of 0.80 m/s and a maximum sprint walk speed of 0.88 m/s, was slightly slower than Type 4.

5 Technical Challenges

New at RoboCup this year was an overall technical challenge consisting of three different league challenges: drop-in player, running, and free challenge. For each league challenge a team participated in points were awarded toward the overall technical challenge based on the following equation:

$$\texttt{points}(rank) = 25 - 20 * (rank - 1)/(numberOfParticipants - 1)$$

	Overall		Drop-in Player		Running		Free	
Team	Rank	Points	Rank	Points	Rank	Points	Rank	Points
UTAustinVilla	**1**	**71.67**	**1**	**25**	**1**	**25**	**2**	**21.67**
FCPortugal	2	60.5	4-5	18	4	17.5	1	25
magmaOffenburg	3	53.33	6	15	3	20	3	18.33
FUT-K	4	43.83	7	13	2	22.5	6	8.33
BahiaRT	5	34.5	10	7	6	12.5	4	15
L3M-SIM	6	30.67	9	9	7	10	5	11.67
SEU_Jolly	7	28.25	2-3	22	8-9	6.25	—	—
RoboCanes	8	22	2-3	22	—	—	—	—
KarachiKoalas	9	18	4-5	18	—	—	—	—
Mithras3D	10	16.25	11	5	8-9	6.25	7	5
ODENS	11	15	—	—	5	15	–	–
HfutEngine	12	11	8	11	—	—	—	—

Table 5: Overall ranking and points totals for each team participating in the RoboCup 2014 3D Simulation League technical challenge as well as ranks and points awarded for each of the individual league challenges that make up the technical challenge.

Table 5 shows the ranking and cumulative team point totals for the technical challenge as well as for each individual league challenge. UT Austin Villa earned the most points and won the technical challenge by taking first in the drop-in player and running challenges and second in the free challenge. The following subsections detail UT Austin Villa's participation in each league challenge.

5.1 Drop-in Player Challenge

The drop-in player challenge,[6] also known as an ad hoc teams challenge, is where agent teams consisting of different players randomly chosen from participants in the competition play against each other. Each participating team contributes two agents to one drop-in player team where drop-in player games are 10 vs 10 with no goalies. An important aspect of the challenge is for an agent to be able to adapt to the behaviors of its teammate. During the challenge agents are scored on their average goal differential across all games played.

		At RoboCup 2014	
Team	Avg. Goal Diff.	Rank	Avg. Goal Diff.
UTAustinVilla	**1.782 (0.050)**	**1**	**1.75**
FCPortugal	0.574 (0.068)	4-5	0.80
SEU_Jolly	0.430 (0.068)	2-3	1.20
magmaOffenburg	0.170 (0.070)	6	-0.25
RoboCanes	0.129 (0.070)	2-3	1.20
BahiaRT	-0.178 (0.071)	10	-1.60
HfutEngine	-0.383 (0.070)	8	-0.75
KarachiKoalas	-0.525 (0.069)	4-5	0.80
L3M-SIM	-0.610 (0.069)	9	-1.50
FUT-K	-0.687 (0.068)	7	-0.40
Mithras3D	-0.703 (0.068)	11	-1.75

Table 6: Average goal differences for each team in the drop-in player challenge when playing all possible parings of drop-in player games (1386 games in total with each team playing 1260 games).

Table 6 shows the results of the drop-in player challenge at RoboCup under the heading "At RoboCup 2014". The challenge was played across 5 games such that every agent played at least one game against every other agent participating in the challenge. UT Austin Villa used the same strategy employed in the 2013

[6] Details of the drop-in player challenge at `http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/2014_dropin_challenge/3D_DropInPlayerChallenge.pdf`

drop-in player challenge [10], and in doing so was able to win this year's drop-in player challenge. The agent's performance was bolstered by longer kicks as discussed in Section 3.1, and also by using a Type 4 agent which was found to be the best performing type in Section 4.2.

Drop-in player games are inherently very noisy and it is hard to get statistically significant results when only playing 5 games. In order to get a better idea of each agents' true drop-in player performance we replayed the challenge with released binaries across all $(\binom{11}{5} * \binom{6}{5})/2 = 1368$ possible team combinations of drop-in player games. Results in Table 6 of replaying the competition over many games show that UT Austin Villa has an average goal difference more than three times higher than any other team, thus validating UT Austin Villa winning the drop-in player challenge.

5.2 Running Challenge

For the running challenge[7] robots were given 10 seconds to run forward as far as possible and then were given a score based on a combination of their average speed and the percentage of time both feet were off the ground. Teams were allowed to use any of the five robot types during the challenge. Teams were also allowed to submit their own custom robot types where the vertical offsets between the robot's hip, knee, and ankle joints could be changed within certain constraints as long as the the overall height of the robot remained the same.

Robot Model	Score	Speed	Off Ground
Type 0	1.491	0.993	0.498
Type 1	1.741	1.241	0.500
Type 2	1.299	0.782	0.517
Type 3	1.839	1.339	0.500
Type 4	1.572	0.972	0.600
Type X	∼1.75	∼1.25	∼0.5

Table 7: Running challenge scores as well as speed and off ground values of optimized walks for each of the different robot types. Type X is a body type optimized for the challenge.

Table 7 shows the performance of various robot types where the walk engine parameters of the robot were optimized with CMA-ES [7] to maximize the running score. During optimization the average feet pressure of the robot was constrained within reasonable values to ensure that the robot did not learn strange running gaits (like running on its knees). For Type X, the body morphology and walk engine parameters were optimized simultaneously within the allowed constraints. Unfortunately, these constraints limited the top speed of the robot and its top running speed was not the fastest out of all the body types. The robots with longer legs (types 1 and 3) were able to achieve faster speeds and higher scores. UT Austin Villa used a Type 3 robot during the challenge.

Results of the running challenge are shown in Table 8. UT Austin Villa had the highest values for speed and off ground percentage, and won the challenge with a score almost 50% higher than the next competitor.

[7] Details of the running challenge at http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/AustinVilla3DSimulationFiles/2014/files/RunChallenge.pdf

Team	Score	Speed	Off Ground
UTAustinVilla	**1.845**	**1.343**	**0.502**
FUT-K	1.251	0.916	0.334
magmaOffenburg	1.152	0.986	0.166
FCPortugal	1.124	0.776	0.348
ODENS	0.632	0.632	0
BahiaRT	0.577	0.483	0.095
L3M-SIM	0.517	0.515	0.001
Mithras3D	0	0	0
SEU_Jolly	0	0	0

Table 8: Scores as well as speed and off ground values for each of the participating teams in the running challenge.

5.3 Free Challenge

During the free challenge teams give a five minute presentation on a research topic related to their team. Each team in the league then ranks the top five presentations with the best receiving 5 votes and the 5th best receiving 1 vote. Additionally several respected research members of the RoboCup community outside the league vote as well with their votes being counted double. The winner of the free challenge is the team that receives the most votes. The top three teams were FCPortugal with 57 votes, UTAustinVilla with 53 votes, and magmaOffenburg with 44 votes.

UT Austin Villa's free challenge submission[8] focused on optimizing robot body types for the tasks of running and kicking. Running performance was evaluated on the same task as the running challenge, but additional body morphology parameters were optimized outside of the constraints of the running challenge. The final optimized body morphology allowed the robot to run at approximately 2.8 m/s running speed, and have its feet off the ground 55% of the time, giving it a running challenge score of 3.35. A robot body type optimized for long distance kicking was able to kick the ball almost 27 meters with the ball traveling over 17 meters in the air (previous optimized kicks with fixed body morphologies only were not able to travel much farther than 22 meters).[9]

6 Conclusion

UT Austin Villa won both the 2014 RoboCup 3D Simulation League main competition and technical challenge.[10] Data taken using released binaries from the competition show that UT Austin Villa winning the competition was statistically significant. The 2014 UT Austin Villa team improved dramatically from 2013 as it was able to beat the team's 2013 second place binary by an average of 1.525 (+/-0.034) goals and also beat the 2013 first place team (Apollo3D) by an average of 2.726 (+/-0.041) goals across 1000 games.

[8] Free challenge entry description at `http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/AustinVilla3DSimulationFiles/2014/files/FreeChallenge_Sim3D_UTAustinVilla.pdf`

[9] Videos of optimized robot body types running and kicking can be found at `http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/#2014challenges`

[10] More information about the UT Austin Villa team, as well as video highlights from the competition, can be found at the team's website: `http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/#2014`

A large factor in UT Austin Villa's success in 2014 was due to improvements in kicking and passing where in previous years the team focused more on dribbling. This paradigm shift within the team is also reflected by the league as the other teams in the semifinals (RoboCanes, magmaOffenburg, and FCPortugal) all possess above average kicking and passing behaviors. In order to remain competitive, and challenge for the 2015 RoboCup championship, teams will likely need to improve multiagent team behaviors such as passing and marking.

Acknowledgments

Thanks to Samuel Barrett for implementing changes to UT Austin Villa's walk engine to support the toe joint of the Type 4 robot model. Also thanks to Klaus Dorer for putting together the running challenge. This work has taken place in the Learning Agents Research Group (LARG) at UT Austin. LARG research is supported in part by NSF (CNS-1330072, CNS-1305287) and ONR (21C184-01).

References

1. MacAlpine, P., Urieli, D., Barrett, S., Kalyanakrishnan, S., Barrera, F., Lopez-Mobilia, A., Ştiurcă, N., Vu, V., Stone, P.: UT Austin Villa 2011: A champion agent in the RoboCup 3D soccer simulation competition. In: Proc. of 11th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2012). (2012)
2. MacAlpine, P., Collins, N., Lopez-Mobilia, A., Stone, P.: UT Austin Villa: RoboCup 2012 3D simulation league champion. In: RoboCup-2012: Robot Soccer World Cup XVI. Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin (2013)
3. MacAlpine, P., Urieli, D., Barrett, S., Kalyanakrishnan, S., Barrera, F., Lopez-Mobilia, A., Ştiurcă, N., Vu, V., Stone, P.: UT Austin Villa 2011 3D Simulation Team report. Technical Report AI11-10, The Univ. of Texas at Austin, Dept. of Computer Science, AI Laboratory (2011)
4. MacAlpine, P., Depinet, M., Stone, P.: UT Austin Villa 2014: RoboCup 3D simulation league champion via overlapping layered learning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15). (2015)
5. Depinet, M., MacAlpine, P., Stone, P.: Keyframe sampling, optimization, and behavior integration: Towards long-distance kicking in the robocup 3d simulation league. In: RoboCup-2014: Robot Soccer World Cup XVIII. Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin (2015)
6. MacAlpine, P., Barrett, S., Urieli, D., Vu, V., Stone, P.: Design and optimization of an omnidirectional humanoid walk: A winning approach at the RoboCup 2011 3D simulation competition. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12). (2012)
7. Hansen, N.: The CMA Evolution Strategy: A Tutorial. (2009) `http://www.lri.fr/~hansen/cmatutorial.pdf`.
8. MacAlpine, P., Price, E., Stone, P.: SCRAM: Scalable collision-avoiding role assignment with minimal-makespan for formational positioning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15). (2015)
9. Hester, T., Stone, P.: Negative information and line observations for monte carlo localization. In: IEEE Int. Conf. on Robotics and Automation (ICRA). (2008)
10. MacAlpine, P., Genter, K., Barrett, S., Stone, P.: The RoboCup 2013 drop-in player challenges: Experiments in ad hoc teamwork. In: Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS). (2014)