To appear in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS), Chicago, Illinois, USA, September 2014.

The RoboCup 2013 Drop-In Player Challenges: Experiments in Ad Hoc Teamwork

Patrick MacAlpine, Katie Genter, Samuel Barrett, and Peter Stone
Department of Computer Science
The University of Texas at Austin
Austin, TX 78701, USA
{patmac,katie,sbarrett,pstone}@cs.utexas.edu

***Abstract*— As the prevalence of autonomous agents grows, so does the number of interactions between these agents. Therefore, it is desirable for these agents to be capable of banding together with previously unknown teammates towards a common goal: to collaborate without pre-coordination. While past research on ad hoc teamwork has focused mainly on theoretical treatments and empirical studies in relatively simple domains, the long-term vision has been to enable robots and other autonomous agents to exhibit the sort of flexibility and adaptability on complex tasks that people do, for example when they play games of "pick-up" basketball or soccer. This paper introduces a series of pick-up robot soccer experiments that were carried out in three different leagues at the international RoboCup competition in 2013. In all cases, agents from different labs were put on teams with no pre-coordination. This paper introduces the structure of these experiments, describes the strategies used by UT Austin Villa in each challenge, and analyzes the results. The paper's main contribution is the introduction of a new large-scale ad hoc teamwork testbed that can serve as a starting point for future experimental ad hoc teamwork research.**

I. Introduction

The increasing capabilities and decreasing costs of robots makes it increasingly possible to study the interactions among teams of heterogeneous robots. To date, most such research on multi-robot teamwork assumes that robots share a common coordination protocol. However, as the number of different companies and research labs producing robots grows, and especially as long-term autonomous capabilities become more common, it becomes increasingly likely that robots will have the occasion to collaborate with previously unknown teammates in pursuit of a common goal. When engaging in such *ad hoc teamwork* [10], robots must recognize and reason about their teammates' capabilities.

Although much of the initial research on ad hoc teamwork has taken a theoretical perspective, it has been argued that ad hoc teamwork is "ultimately an empirical challenge" [10]. In order to facilitate such empirical ad hoc teamwork research, this paper introduces a series of "drop-in player challenges" that the authors helped to organize at RoboCup[1] 2013, a well established multi-robot competition. These challenges brought together real and simulated robots from teams from around the world to investigate the current ability of robots to cooperate with a variety of unknown teammates.

In each game of the challenges, robots were drawn from the participating teams and combined to form a new team. These robots were not informed of each other's identities, and thus had to adapt quickly to their teammates over the course of a single game so as to discover how to intelligently share the ball and select which roles to play. Teams from around the world submitted teammates for this challenge.[2]

This paper introduces the drop-in player challenges as a novel testbed for ad hoc teamwork and facilitates future research in this area. After specifying the ad hoc teamwork problem and introducing the three substrate RoboCup domains in Section II, its main contributions are: 1) the detailed format and rules of the challenges (Section III); 2) an introduction of a team strategy used in each of the challenges, (Section IV); and 3) detailed results and analyses of the largest scale ad hoc teamwork experiments conducted to date (Sections V and VI). Section VII situates this work in literature, and Section VIII concludes. The paper's main purpose is to serve as a strong starting point for future large-scale experimental ad hoc teamwork research, both in RoboCup, and in other multi-robot domains.

II. Overview and Domain Description

Robot soccer[1] has served as an excellent research domain for autonomous agents and multiagent systems over the past decade and a half. In this domain, teams of autonomous robots compete with each other in a complex, real-time, noisy and dynamic environment, in a setting that is both collaborative and adversarial. RoboCup includes several different leagues, each emphasizing different research challenges.

A. Ad Hoc Teamwork

During the more than 15 years of annual RoboCup soccer competitions, participants have always created full teams of agents to compete against other teams. As a result, they have been able to build in complex, finely-tuned coordination protocols that allow them to agree upon which player should go to the ball and the other players' roles.

While the majority of multiagent research in general focuses on creating coordinated teams that complete shared tasks, ad hoc teamwork research focuses on creating agents that can cooperate with unknown teammates without prior coordination. Rather than creating a whole team of agents

[1] http://www.robocup.org/

[2] Videos of the challenges are at http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/AustinVilla3DSimulationFiles/2013/html/dropin.html

that share a coordination protocol, we assume that each developer can only create a single agent or a small subset of agents on the team; the other agents are under the control of other developers. The objective of each agent is to be capable of adapting to any teammates it may encounter while working on a shared task, in this case winning soccer games.

B. RoboCup Standard Platform League (SPL)

In the Standard Platform League (SPL),[3] teams compete with identical 57.3cm tall Aldebaran Nao humanoid robots,[4] as shown in Figure 1. Since teams compete with identical hardware, the SPL is essentially a software competition. In the SPL, games are played with 5 robots per side on a 6m by 9m carpeted field and last for two, 10 minute halves. Robots play completely autonomously and are able to communicate with each other via a wireless network.

Fig. 1: UT Austin Villa's Nao robots (in pink) at RoboCup 2013.

C. RoboCup 2D Simulation League

As one of the oldest RoboCup leagues, 2D simulation soccer has been well explored, both in competition and in research. The domain consists of teams of 11 autonomous agents playing soccer on a simulated 2D soccer field shown in Figure 2. The field measures 105m by 68m with robots having a radius of 30cm. Games have two 5 minute halves having a total of 6,000 simulation steps each lasting 100 ms. The agents receive abstract sensory information about the game, including the position of the ball and other agents, from the central server. After processing this information, the agents select abstract actions such as dashing, kicking, and turning. 2D soccer abstracts away many of the low-level behaviors required for humanoid robot soccer, including walking and computer vision, instead focusing on higher-level aspects of playing soccer such as multiagent coordination and strategy.

Fig. 2: A screenshot of a 2D soccer simulation league game.

[3] `http://www.tzi.de/spl/`
[4] `http://www.aldebaran.com`

D. RoboCup 3D Simulation League

The RoboCup 3D simulation environment is based on SimSpark,[5] a generic physical multiagent systems simulator. SimSpark simulates realistic physics using the Open Dynamics Engine (ODE) library.[6] The robots used are 57cm tall models of the Aldebaran Nao[4] which receive abstract perceptual information and send torque commands for their motors.

Each robot has 22 degrees of freedom, each equipped with a perceptor and an effector. Joint perceptors provide the agent with noise-free angular measurements every simulation cycle (20 ms), and joint effectors allow the agent to specify the torque and direction in which to move a joint. Although there is no intentional noise in actuation, there is slight actuation noise that results from approximations in the physics engine. Abstract 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.

Games consist of two 5 minute halves of 11 versus 11 agents on a 20m by 30m field. Figure 3 shows a visualization of the simulated robot and the soccer field during a game.

Fig. 3: A screenshot of the Nao-based humanoid robot (left), and a view of the soccer field during a 11 versus 11 game (right).

III. Challenge Descriptions

This section describes the first main contribution of this paper, namely the format and rules of the drop-in player challenges held at RoboCup 2013. These rules encouraged new ad hoc teamwork research in a competitive setting.

A. SPL Challenge

The SPL challenge[7] required five players per side, and given that six teams participated, each team contributed one or two drop-in players. If two players from the same team were used, they played on the same side. Both teams were composed of randomly selected drop-in players, and each competitor participated in four drop-in games lasting 5 minutes each. Shorter games were used to allow for more games to be played in the allotted time. In normal SPL games, a goalie is specified at the start of a game. In the

[5] `http://simspark.sourceforge.net/`
[6] `http://www.ode.org/`
[7] Full rules of the SPL challenge can be found at `http://www.tzi.de/spl/pub/Website/Downloads/Challenges2013.pdf`

challenge, the first defensive player to enter the goal box became the goalie for the remainder of the game.

During the challenge, players were allowed to communicate with each other using a simple protocol, but this communication was not required. This protocol allows for communicating the locations of the player and the ball, the variance (uncertainty) of the player and ball locations, the ball's velocity, the time since the ball was last seen, and whether the robot is fallen or penalized.

The SPL challenge was scored using two metrics: average goal difference and average score from three judges. The two scoring metrics were combined to determine the overall winner of the SPL challenge. Human judges were used to help identify good teamwork abilities in agents and alleviate the effects of random variance given the limited number of games played. For each game, each judge was asked to score each player between 0 (poor) and 10 (excellent). The judges were instructed to focus on teamwork capabilities, rather than individual skills, such that a less-skilled robot could still be given top marks for good teamwork.

B. 2D Challenge

For the 2D drop-in player challenge,[8] each team contributed two drop-in field players to a game where both teams consisted of drop-in field players. Games consisted of two 5 minute halves with teams of 7 players, rather than the standard 11 players. The number of players was changed because the majority of teams are based on the same code release (agent2d [1]) which provides default formations for 11 player teams that would provide implicit coordination for these teams. The seventh player on each team was a common goalie agent from the agent2d release, and no coaches were used in the challenge. Drop-in players were encouraged to use the default agent2d communication protocol for communication, but this was not required.

Players were scored exclusively by their average goal difference across all of their games; no human judges were used. To accurately measure their performance, every team played at least one game against opponents from each other team. A total of nine teams participated in the challenge. Game pairings were chosen by a greedy algorithm that attempts to even out the number of times agents from different teams play with and against each other, shown in Alg. 1. This algorithm is general and can be applied to other ad hoc teamwork settings. The algorithm terminates when all agents have played at least one game against opponents from every other team.

C. 3D Challenge

For the 3D drop-in player challenge,[9] each participating team contributed two drop-in field players to a game. Games lasted for two 5 minute halves with both teams consisting of drop-in players. No goalies were used for the challenge to increase the likelihood of goals being scored.

Each drop-in player could communicate with its teammates about the ball and their own positions, but using the protocol was optional. The challenge was scored solely by the average goal difference received by an agent across all games it played. Four drop-in games were played during the challenge, and each of the ten participating teams played in every game. Team pairings for games were determined by Alg. 1 so each drop-in player played at least one game versus each other player.

IV. Drop-In Player Strategies

In addition to creating the rules for the drop-in player challenges, UT Austin Villa also participated in these challenges. This section describes the strategies we employed in these large-scale ad hoc teamwork experiments. We can only analyze the UT Austin Villa strategies as there was no mechanism to collect other teams' strategies. At the 2014 RoboCup competition drop-in player challenge participants will be asked to submit brief descriptions of their strategies.

A. SPL Strategies

In the main SPL competition, UT Austin Villa's robots coordinate by communicating their positions and bids to play as chaser (player who goes to the ball) based on their relative positions to the ball. The robot with the highest bid becomes chaser, and the remaining players are assigned to the remaining roles, such as defender, forward, or midfielder. These roles are assigned based on the priority of the roles and the robots' distances from the roles' locations.

For the drop-in player challenge, our robots follow a similar strategy. Our robots estimate their teammates' bids to be chaser based on their communicated locations, ignoring robots that do not communicate. The player with the highest estimated bid is assumed to be the chaser. Remaining players are assumed to assign themselves to roles based on their locations similarly to above, though there is no guarantee that they will follow these assumptions. While a more sophisticated approach might better estimate the teammates' roles, this approach worked well in initial tests.

As in the main competition, our robots reasoned about passing and positioning to receive passes, adapting to their

Algorithm 1 Drop-In Team Pairings

Input: *Agents (single agents or sets of agents teams contribute to a game)*

```
games = ∅
while not allAgentsHavePlayedEachOther() do
    team1, team2 := ∅
    for i := 1 to AGENTS_PER_TEAM do
        team1 ← getNextAgent(Agents \ {team1 ∪ team2})
        team2 ← getNextAgent(Agents \ {team1 ∪ team2})
    games ← {team1, team2}
return games
getNextAgent(AvailableAgents)
Select agents using the following ordered preferences:
    1) Played fewer games.
    2) Played against fewer of the opponents.
    3) Played with fewer of the teammates.
    4) Played lower max number of games against/with any one opponent/teammate.
    5) Played lower max number of games against any one opponent.
    6) Played lower max number of games with any one teammate.
```

[8] Full rules of the 2D challenge can be found at http://www.cs.utexas.edu/~AustinVilla/sim/2dsimulation/2013_dropin_challenge/2D_DropInPlayerChallenge.pdf

[9] Full rules of the 3D challenge can be found at http://www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/2013_dropin_challenge/3D_DropInPlayerChallenge.pdf

teammates dynamically. Furthermore, during the team's kick-off, the robot considers a variety of set plays, and prefers plays that pass the ball to its teammates when its teammates report that they are in opportune positions. If the other players do not cooperate by moving to receiving positions, the robot executes set plays that do not require their help, such as burying the ball deep near the opponents' goal.

B. 2D Strategies

For the 2D competition, UT Austin Villa builds on the agent2d [1] base code release which provides a fully functional soccer playing agent team. We modified this code base to use our own dynamic role assignment system [8] which attempts to minimize the makespan (time for all agents to reach target home positions) while also preventing collisions among agents. An important aspect of the drop-in challenge is for an agent to be able to adapt to the behaviors of its teammates: for instance if most of an agent's teammates are assuming offensive roles, that agent might better serve the team by taking on a defensive role. UT Austin Villa's dynamic role assignment system implicitly allows for this adaptation to occur as it naturally chooses roles that do not currently have other agents nearby.

C. 3D Strategies

For the drop-in challenge, UT Austin Villa's agent goes to the ball when it is closest and tries to move the ball towards the opponent's goal. If our agent is not closest to the ball, it waits two meters behind the ball as this was found in [8] to be an important supporting position due to the prevalence of dribbling in the competition. Since two UT Austin Villa agents are always on the same team and neither of them may be closest to the ball, the two agents are often moving to the same target position, but avoid each other using collision avoidance as described in [8].

As not all agents are adept at self-localization, our agent tracks the trustworthiness of other teammates' observations. This assessment is done throughout the game by recording the accuracy of the teammates' messages about their location and the ball's location, comparing them to values observed by our agent. Should the average accuracy fall below a set level, our agent disregards that agent's information when building a model of the world. Unfortunately, as no other teams used the shared communication protocol for this challenge, this feature was only used in our preliminary tests.

During kickoffs, agents can teleport to anywhere on their own side of the field before play begins. As shown in [9], long kicks that move the ball deep into the opponent's side provide a substantial gain in performance. Therefore, prior to our team's kickoff, our agents would teleport to a random position on the field and wait for their teammates to move to their positions. Then, if no teammate is next to the ball, our agent would teleport near the ball to take the kickoff.

V. CHALLENGE RESULTS

In addition to the challenges themselves and the strategies we used to participate, the third main contribution of this paper is a presentation and analysis of the results of the RoboCup 2013 drop-in challenges. In this section, we present the raw results of the challenges as well as additional tests we performed to further gauge the competitors' performance. Section VI further analyzes these results.

A. SPL Results

In the SPL drop-in player challenge, six teams participated in four 5-minute games. As discussed in Section III-A, the overall winner of this challenge was determined via two metrics: average goal difference and average human-judged score. Challenge scores and rankings can be seen in Table I.

Team	Avg Goal Diff	Norm Goal Diff	Avg Judge Score	Final Score	Rank (Goal, Judge)
B-Human	1.17	10	6.67	16.67	1 (1,1)
Nao Devils	0.57	4.9	6.24	11.14	2 (3,2)
rUNSWift	0.67	5.71	5.22	10.94	3 (2,4)
UT Austin Villa	-0.29	-2.45	6	3.55	4 (4,3)
UPennalizers	-0.57	-4.9	4.48	-0.42	5 (5,5)
Berlin United	-1.29	-11.02	3.38	-7.64	6 (6,6)

TABLE I: Final scores and rankings for the SPL drop-in challenge.

B. 2D Simulation Results

Nine teams participated in the 2D drop-in player challenge with seven games being played in total. Due to the noise of 2D simulation games, seven games are not enough for the results to be statistically significant. Therefore, following the competition, we also replayed the challenge with the released binaries over many games including all combinations of the nine teams contributing two agents each. There are $(\binom{9}{3} * \binom{6}{3})/2 = 840$ combinations given that each drop-in team is made up of agents from three different teams. Five games were played for each combination, resulting in 4,200 games.

	RoboCup		Many Games	
Team	AGD	Rank	AGD	Rank
FCPerspolis	2.4	1	3.025 (0.142)	1
Yushan	2.25	2	2.583 (0.141)	2
ITAndroids	2.0	3	1.379 (0.152)	5
Axiom	1.2	4	1.315 (0.148)	6
UT Austin Villa	.25	5	1.659 (0.153)	4
HfutEngine	-0.2	6	-2.076 (0.152)	7
WrightEagle	-1.6	7	-6.218 (0.129)	9
FCPortugal	-2.2	8	-3.379 (0.150)	8
AUTMasterminds	-2.8	9	1.711 (0.152)	3

TABLE II: Avg. goal difference (AGD) with standard error shown in parentheses and rankings for the 2D drop-in player challenges.

The competition results and extended results are shown in Table II. The difference between these results show that only playing seven games does not reveal the true rankings of teams as the last place team during the challenge at RoboCup, AUTMasterminds, finished third overall when playing thousands of games. We do however acknowledge that as we are not running games on the same exact machines as were used at RoboCup, there is the potential for agents to behave differently in these tests.

C. 3D Simulation Results

Ten teams participated in the 3D drop-in player challenge, and four games were played in total. Game results in the 3D simulator also tend to have high variance, so results from only four games are not statistically significant. Therefore, we once again replayed the challenge using the released binaries using all combinations of teams. The total number of possible different drop-in team combinations is $(\binom{10}{5} * \binom{5}{5})/2 = 126$ as each drop-in team is made of agents from five different teams. Five games were played for each combination, resulting in a total of 630 games. Results from the competition and the extended analysis are shown in Table III.

	RoboCup		Many Games	
Team	**AGD**	**Rank**	**AGD**	**Rank**
BoldHearts	1.5	1	0.178 (0.068)	4
FCPortugal	0.75	T2	1.159 (0.060)	1
Bahia3D	0.75	T2	-0.378 (0.068)	7
Apollo3D	0.75	T2	0.159 (0.068)	5
magmaOffenburg	0.25	5	0.254 (0.068)	3
RoboCanes	-0.5	6	-0.286 (0.068)	6
UT Austin Villa	-0.75	T7	0.784 (0.065)	2
SEUJolly	-0.75	T7	-0.613 (0.066)	9
Photon	-0.75	T7	-0.425 (0.068)	8
L3MSIM	-1.25	10	-0.832 (0.065)	10

TABLE III: Avg. goal difference (AGD) with standard error shown in parentheses and rankings for the 3D drop-in player challenges.

VI. Challenge Analysis

This section presents deeper analysis of the results of the challenges, with a particular eye towards identifying which ad hoc teamwork strategies proved to be most effective.

A. SPL Analysis

The winners of the drop-in challenges should be the players who displayed the best teamwork abilities, not necessarily the best low level skills. Hence, we compare teams' performances in the drop-in player challenge to their performance in the main competition. Overall, the challenge's results were well correlated with the main competition's results. UPennalizers and Berlin United finished near the bottom in the drop-in challenge, and they also finished in the lower part of the main competition.[10] Notably, B-Human performed best in terms of their drop-in and main ranks as well as in the human judges' scores, indicating that their teamwork and adaptability performed well in both settings.

B. 2D Analysis

To further analyze the results from the 2D drop-in player challenge we again compare teams' performances in the drop-in player challenge to their performance in the main competition. As the main tournament only includes a relatively small number of games, and thus rankings from the main competition are typically not statistically significant, we ran 1,000 games of our team's main competition binary against each of the other teams' released main competition binaries who participated in the drop-in player challenge. This process gives us a statistically significant baseline for comparing the performance of teams on the task of playing standard team soccer. Results are shown in Table IV, using the drop-in rankings from the larger analysis.

While Table IV shows that there is not a direct correlation between ranking in the drop-in player challenge compared to standard team soccer, there is a trend for agents that perform better at standard team soccer to also perform better at the drop-in player challenge. Excluding the outlier team WrightEagle, the top half of the teams for the drop-in player challenge had an average rank of 4.25 when playing against UT Austin Villa while the bottom half of the teams had an average rank of 6.75 against UT Austin Villa.

[10] Main competition results can be found at https://www.tzi.de/spl/bin/view/Website/Results2013

	Drop-In	Main	Against UT Austin Villa	
Team	**Rank**	**Rank**	**Rank**	**AGD**
FCPerspolis	1	5	4	2.923 (0.056)
Yushan	2	2	3	3.616 (0.063)
AUTMasterminds	3	4	2	5.289 (0.084)
UT Austin Villa	4	8	8	0 (self)
ITAndroids	5	7	7	0.442 (0.060)
Axiom	6	3	5	1.248 (0.072)
HfutEngine	7	9	9	-6.07 (0.175)
FCPortugal	8	6	6*	*
WrightEagle	9	1	1	6.537 (0.25)

TABLE IV: Avg. goal difference (AGD) with standard error shown in parentheses and rankings for the 2D drop-in player challenge, main competition, and playing against UT Austin Villa. *We were unable to run the released FCPortugal binary and thus used their relative ranking from the main competition.

An important aspect of the drop-in player challenge is for agents to adapt to the behaviors of their teammates, such as moving to defense when most teammates are playing offensive roles. As mentioned in Section IV-B, UT Austin Villa's dynamic role assignment system [8] implicitly encourages this adaptation. We tested a version of the UT Austin Villa agent that used a static role assignment rather then the dynamic system across the same 4,200 games. Compared to the dynamic assignments, using the static assignments dropped our agent's average goal difference from 1.659 (+/-0.153) to 1.473 (+/-0.157). We empirically found that most agents in the challenge used static role assignments, which may explain why UT Austin Villa performed better in the drop-in player competition than in the main competition.

C. 3D Analysis

To analyze results from the 3D drop-in player challenge we followed the same methodology used in the 2D analysis. To do so we ran at least 100 games of our team's main competition binary against each of the other teams' released main competition binaries who participated in the drop-in player challenge. Results for this are shown in Table V, using the drop-in rankings from the larger analysis.

	Drop-In	Main	Against UT Austin Villa	
Team	**Rank**	**Rank**	**Rank**	**AGD**
FCPortugal	1	3	2	-0.465 (0.023)
UT Austin Villa	2	2	1	0 (self)
magmaOffenburg	3	T5	5	-1.447 (0.026)
BoldHearts	4	T5	6	-1.607 (0.029)
Apollo3D	5	1	3	-0.698 (0.027)
RoboCanes	6	T5	7	-1.828 (0.031),
Bahia3D	7	10	10	-9.80 (0.110)
Photon	8	8	8	-4.59 (0.081)
SEUJolly	9	4	4	-1.133 (0.027)
L3MSIM	10	9	9	-6.05 (0.098)

TABLE V: Average goal difference (AGD) with standard error shown in parentheses and rankings for the 3D drop-in player challenge, main competition, and playing against UT Austin Villa.

Similar to the results in the 2D simulation league, Table V shows that there is not a strong correlation between rankings in the drop-in player challenge and ranking in the main competition. However, there is a trend that teams performing better at drop-in player soccer also do better at standard team soccer. The top half of the teams for the drop-in player challenge had an average rank of 3.4 against UT Austin Villa, while the bottom half's same average rank was 7.6.

To evaluate the importance of different parts of our drop-in player strategy, we created the following variants of the UT Austin Villa drop-in player agent:

Dribble: Agent only dribbles and never kicks.

DynamicRoles: Uses dynamic role assignment.

NoKickoff: No teleporting next to ball to take the kickoff.

We then tested them across the same 630 game extended drop-in player challenge used in Section V-C.

Agent	AGD
Dribble	1.370 (0.064)
UT Austin Villa	0.784 (0.065)
NoKickoff	0.676 (0.065)
DynamicRoles	0.568 (0.071)

TABLE VI: Average goal difference (AGD) with standard error shown in parentheses for agents playing in the extended 3D drop-in player challenge.

The results in Table VI show that the strategy of teleporting in to take the kickoff if no teammates are in place to do so improves performance. The results also reveal that by only dribbling, and not kicking, we would greatly improve our agent's performance in the drop-in player challenge. Considering that the UT Austin Villa 3D simulation team won both the 2011 and 2012 RoboCup world championships, and took second place in 2013, it can be surmised that the UT Austin Villa agent likely has better low-level skills than most other agents on its drop-in player team. Therefore, kicking the ball, and potentially passing it to a teammate, has a reasonable likelihood of hurting overall team performance.

VII. Related Work

Multiagent teamwork is a well studied topic, with most work tackling the problem of creating standards for coordinating and communicating. One such algorithm is STEAM [11], in which team members build up a partial hierarchy of joint actions and monitor the progress of their plans while communicating selectively. In [5], Grosz and Kraus present a reformulation of the SharedPlans, where agents communicate their intents and beliefs and use this information to reason about how to coordinate joint actions. In addition, SharedPlans provides a process for revising agents' intents and beliefs to adapt to changing conditions. While these algorithms have been shown to be effective, they require teammates to share their coordination framework.

On the other hand, ad hoc teamwork focuses on the case where the agents do not share a coordination algorithm. In [7], Liemhetcharat and Veloso reason about selecting agents to form ad hoc teams. Barrett et al. [3] empirically evaluate an MCTS-based ad hoc team agent in the pursuit domain, and Barrett and Stone [2] analyze existing research on ad hoc teams and propose one way to categorize ad hoc teamwork problems. Other approaches include Jones et al.'s work [6] on ad hoc teams in a treasure hunt domain. A more theoretical approach is Wu et al.'s work [12] into ad hoc teams using stage games and biased adaptive play.

In the domain of robot soccer, Bowling and McCracken [4] measure the performance of a few ad hoc agents, where each ad hoc agent is given a playbook that differs from that of its teammates. The teammates implicitly assign the ad hoc agent a role, and then react to it as they would any teammate. The ad hoc agent analyzes which plays work best over hundreds of games and predicts the roles that its teammates will play.

However, none of this previous research has evaluated their approaches with agents created by developers from around the world in a true ad hoc teamwork setting. Other RoboCup leagues have looked into ad hoc teamwork, including the small size league holding mixed team challenges.[11] Our work extends their results by including participation from many more teams, and with more than two teams per side.

VIII. Conclusions

This paper describes and documents the drop-in player challenges run in three of the leagues at the 2013 RoboCup competition. These challenges serve as a novel testbed for ad hoc teamwork, in which agents must adapt to a variety of new teammates without pre-coordination, and provided an opportunity to evaluate robots' abilities to cooperate with new teammates to accomplish goals in complex tasks. These were the first large scale pick-up games between teams each composed of agents designed by more than two sources with the goal to perform well as an ad hoc team. These challenges and the strategies introduced in them could generalize to other ad hoc teamwork scenarios. We believe that these drop-in challenges will serve as the starting point for many future drop-in challenges, and will also serve as a point of reference for designing new ad hoc teamwork testbeds.

Acknowledgments

We would like to thank Duncan ten Velthuis from the University of Amsterdam for helping to run the Standard Platform League challenge, Aijun Bai from the University of Science and Technology of China for running the 2D challenge, and Sander van Dijk from the University of Hertfordshire for helping to run the 3D 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). Patrick MacAlpine is supported by a NDSEG fellowship.

References

[1] H. Akiyama. Agent2d base code, 2010.

[2] S. Barrett and P. Stone. An analysis framework for ad hoc teamwork tasks. In *AAMAS '12*, June 2012.

[3] S. Barrett, P. Stone, S. Kraus, and A. Rosenfeld. Teamwork with limited knowledge of teammates. In *AAAI '13*, July 2013.

[4] M. Bowling and P. McCracken. Coordination and adaptation in impromptu teams. In *AAAI*, 2005.

[5] B. Grosz and S. Kraus. Collaborative plans for complex group actions. *Artificial Intelligence*, 86:269–368, 1996.

[6] E. Jones, B. Browning, M. B. Dias, B. Argall, M. M. Veloso, and A. T. Stentz. Dynamically formed heterogeneous robot teams performing tightly-coordinated tasks. In *ICRA*, pages 570 – 575, May 2006.

[7] S. Liemhetcharat and M. Veloso. Modeling mutual capabilities in heterogeneous teams for role assignment. In *IROS '11*, 2011.

[8] P. MacAlpine, F. Barrera, and P. Stone. Positioning to win: A dynamic role assignment and formation positioning system. In *RoboCup-2012: Robot Soccer World Cup XVI*. Springer Verlag, Berlin, 2013.

[9] P. MacAlpine, N. Collins, A. Lopez-Mobilia, and P. Stone. UT Austin Villa: RoboCup 2012 3D simulation league champion. In *RoboCup-2012: Robot Soccer World Cup XVI*. Springer Verlag, Berlin, 2013.

[10] P. Stone, G. A. Kaminka, S. Kraus, and J. S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *AAAI '10*, July 2010.

[11] M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:81–124, 1997.

[12] F. Wu, S. Zilberstein, and X. Chen. Online planning for ad hoc autonomous agent teams. In *IJCAI*, 2011.

[11] http://robocupssl.cpe.ku.ac.th/robocup2013:mixed_team_tournament