Simulating Team Tutoring in Multiagent Environments

Max Johnson, Ning Wang, & David V. Pynadath

University of Southern California

maxwelsj@usc.edu,nwang@ict.usc.edu, pynadath@ict.usc.edu

Abstract. A good team functions like a well-oiled machine. Teammembers train individually and together in order to do well as a team. Realistic simulations can offer safe and repeatable environments for teams to practice without real-world consequences. However, instructional support is often needed to help the team and individuals in case of mistakes and impasses and to guide the team on the path to success. In our work, we designed a simulated learning environment for teams of autonomous agents using PsychSim. The simulation provides a testbed for developing tutoring strategies suited for team training and for the skills it aims to engender. The simulation implements a "capture-the-flag" scenario, where a team of agents (the Blue team) must work to capture the flag being defended by an opposing team of agents (Red team). While the scenario is simple, the tutoring strategies to be used by a tutoring agent can be complex and dynamic. For example, what type of student behavior is considered a mistake and what should the tutoring agent instruct the student agents to do instead? In this paper, we will discuss the simulation experiments we designed to uncover tutoring strategies.

Keywords: collaborative learning, team-based training, intelligent agents, social simulation

1 Introduction

Individual mastery and mastery as a member of a team are two fundamentally critical concepts in teams. The nature of team dynamics necessitates that each member be proficient in not only their individual role, but also their ability to communicate and adjust to their teammates. In order to achieve this level of proficiency at a task, team members must train both individually and as a team. Due to the prevalence of team tasks in today's society, particularly in medical care, emergency responses, and the military, team-based training has been explored and refined over a long history, particularly with the use of simulations (e.g., [10], [18], and [26]). While realistic simulations can offer safe and repeatable environments for teams to practice without the real-world consequences, they are often not enough to ensure learning without instructional support. Providing this support in teams has its unique challenges, such as deciding who to target (individual vs. team), communication channel (private vs. public), and the timing of the feedback (immediate vs. delayed). These and other variables can greatly impact how such support is received by the team and the efficacy of the feedback [33]. The correct decision as to these actions often depends on the team structure (e.g., with leadership or leaderless) and what the team is trying to learn (e.g., task-related vs. teamwork related, for review, see [7] and [28]), and incorrect decisions can lead to feedback being ignored or worse, causing a negative impact on the team's learning [34].

A simulation of team training and the influence of instructional feedback on team members and a team is desired to mitigate the cost and resources needed for testing with human participants. We have developed a testbed containing such a simulation where team members are modeled as virtual agents in a collaborative learning setting where they can learn from experience to improve team performance, as well as interact with a tutor agent. In collaborative learning, there is an emphasis on each individual of the team training how to collaborate to improve as a whole [28], as opposed to cooperative learning wherein members try to maximize learning of other team members. However, our simulation testbed is not limited to collaborative learning only.

Instructional support in team tutoring often can be adapted to the structure of the team being tutored. For example, tutorial feedback for a team with a vertical leadership structure is more likely to cater to members based on their level in said structure. In horizontally organized teams, the feedback is likely to be designed for a group of peers [1]. When a team is actively engaged in learning, team members communicate among themselves to discuss best actions, ask each other questions, and explain their reasoning. In our simulation testbed, we build upon both instruction from a tutor and feedback from peers and their own experience.

In this paper, we discuss a multiagent simulation testbed for experimentation to explore team-tutoring strategies. This testbed forms a foundation for developing and testing automated team tutor agents. In the testbed, a team of simulated agents attempt to complete a collaborative task with or without a tutoring agent. In order for a tutor agent to be of any help to our team, we first need to know what it should teach. If we do not know what our team should be doing in order to win, then we have no basis for teaching them how to win. Thus, the focus of our work will be determining what exactly our tutor should teach the team. This paper details our work in the design of the testbed, and our work in uncovering what the tutor agent should teach.

2 Related Work

Research on such support in the context of team training is relatively scarce in comparison to the growing abundance of research on automatically-generated instructional support for individual learning (for review, see [3]). Early research in this topic focuses on creating simulation environments that allow teams to practice together. One such effort, the Advanced Embedded Training System (AETS), is an intelligent tutoring system built for an Air Defense Team on a ship's Combat Information Center to learn how to utilize the command and control system [38]. In AETS, multiple users train as a team while receiving assessment and feedback on an individual basis. A human tutor then takes this feedback and offers team-based feedback. A similar effort is the Steve agent-based training simulation for emergency response on a military vessel [26], where Steve agents can serve as a tutor as well as an individual team member. This allows the simulation to support a team of any combination of Steve agents and humans to train together, learning to complete tasks through communication between team members.

In a more recent example, one team training simulation testbed implements a scenario where a team of three completes errands following a shopping list in a virtual mall, called the Multiple Errands Test [34]. This testbed was used in a study in which privacy (Public vs. Private) and audience (Direct vs. Group) of feedback and other such variables showed no influence on team performance. Even more recently the Recon testbed, built with the Generalized Intelligent Framework for Tutoring (GIFT) [7], was developed to explore the collaborative team task of reconnaissance [2]. Once again, this testbed was used by researchers to experiment with different targets (individual vs. team) for feedback within 2-person teams [14]. On simulating students as virtual agent, the SimStudent project developed an approach that could accurately model a single student's cognitive processes for one-on-one intelligent tutoring system research [16]. This work shows the promise of using simulated students, albeit in one-on-one tutoring scenarios. These examples all point to a resurgence of research into automated tutorial support for team training.

Our testbed simulates the training process, similar to the training that takes place in the aforementioned work. Agents learn to improve both their own and the team's performance from their own experience, by observing other agents, by communicating with teammates, and via the guidance of a tutor agent. Existing formalisms within the body of multiagent research on simulating teamwork and learning represent team goals, plans, and organizations that operationalize decision-making found in human teams [6, 9, 30]. Embedding these mechanisms within intelligent agents has enabled the construction of high-fidelity simulations of team behavior (e.g., simulated aircraft performing a joint mission [31]). As uncertainty and conflicting goals are prevalent in most team settings, decision-theoretic extensions of these models incorporating quantitative probability and utility functions captured these dynamics effectively [24, 32]. In addition, the use of reinforcement learning (among other methods) to derive agents' models through experience in a decentralized fashion has been incorporated to accurately model how team members can arrive at a coordinated strategy through their individual experience [5, 20, 29].

3 PsychSim

We have built our testbed using the multiagent social simulation framework, PsychSim [15, 21]. PsychSim grew out of the prescriptive teamwork frameworks cited in Section 2 (especially [24]), but with a different aim toward being a descriptive model of human behavior. PsychSim represents people as autonomous agents that integrate two multiagent technologies: recursive models [8] and decision-theoretic reasoning [11]. Recursive modeling gives agents a Theory of Mind [37], to form complex attributions about others and incorporate such beliefs into their own behavior. Decision theory provides the agents with domain-independent algorithms for making decisions under uncertainty and in the face of conflicting objectives. We have used PsychSim to model a range of cognitive and affective biases in human decision-making and social behavior (e.g., [22, 23]).

Another motivation behind the use of PsychSim is its successful application within multiple simulation-based learning environments. The Tactical Language Training System (TLTS) is an interactive narrative environment in which students practice their language and culture skills by talking to non-player characters built upon PsychSim agents [27]. We also used PsychSim's mental models and quantitative decision-theoretic reasoning to model a spectrum of negotiation styles within the ELECT BiLAT training system [12]. Additionally, UrbanSim used a PsychSim-driven simulation to put trainees into the role of a battalion commander undertaking an urban stabilization operation [17]. In SOLVE, PsychSim agents populate a virtual social scene where people could practice techniques for avoiding risky behavior [13, 19].

We have also used PsychSim to build experimental testbeds for studying human teamwork. In one such testbed, we used a PsychSim agent to autonomously generate behaviors for a simulated robot that teamed with a person, in a study of trust within human-robot interaction [35, 36]. Another PsychSim-based testbed gave four human participants a joint objective of defeating a common enemy, but with individual scores that provided some impetus for competitive behavior within the ostensible team setting [25]. We build upon PsychSim's capability for such experimental use in the expanded interaction of the current investigation.

4 Team-based Training Simulation

In our testbed, we implement a "capture-the-flag" scenario. In the scenario, a team of trainees learn how to work together to capture a goal location being defended by a team of opponents. Both the trainees and opponents are represented as PsychSim agents. In the preliminary testing described here, both the blue team and the red team consist of three agents. The three blue agents are not assigned any distinct roles. In this scenario, agents can be "tagged" by opposing agents if they are adjacent in one of the four main directions. Any agent that is tagged three times is eliminated from play and can no longer act in the scenario.

PsychSim represents the decision-making problem facing the agents as a Partially Observable Markov Decision Process (POMDP) [11]. Partial observability accounts for the fact that the agents cannot read each other's minds and that they may have incomplete or noisy observations of the environment. However, in this presentation, we make the environment itself completely observable, reducing the domain to a Markov Decision Process (MDP) instead. An MDP is a tuple *(S A, P, R)*, with *S* being the set of states, *A* the set of actions, *P* the transition probability representing the effects of the actions on the states, and *R* the reward function that expresses the player's preferences.

The state of the world, *S*, represents the evolution of the game state over time. We use a factored representation [4] that allows us to separate the overall game state into orthogonal features that are easier to specify and model. The locations of the agents and of the goal are specified by *x* and *y* coordinates on a grid. The grid is 7×7 in the specific configuration described here, but obviously other grid sizes are possible (see Figure 1).

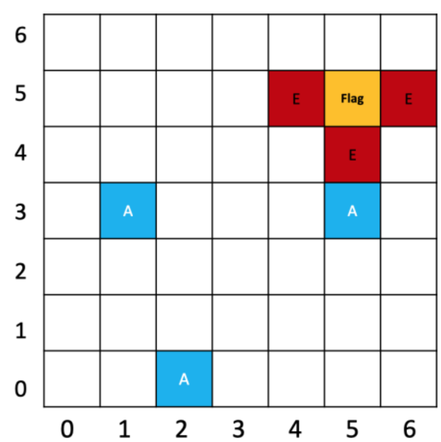

Fig. 1. A mid-mission screenshot of the "capture-the-flag" scenario. The blue team agents are located at [1,3], [2,0] and [5,3], while the red team agents are located at [4,5], [5,4] and [6,5] and the goal is located at [5,5].

The actions, *A*, available to the agents are moves in one of the four directions, attempting to "tag" an opponent in one of the four directions, or waiting in their current location. The transition probability, *P*, represents the effect of the agents' movement decisions, which we specify here to succeed with 100% reliability. In general, the *P* function can capture any desired stochastic error (e.g., due to terrain or visual conditions).

Each blue team agent has three potentially conflicting objectives within its reward function, *R*: minimizing its distance to the goal (i.e., to try and reach the goal), maximizing the number of times Red agents are tagged (i.e., remove opponents from play), and minimizing the number of times that they get tagged (i.e., avoid being removed from play). The red agents also have three potentially conflicting objectives: maximizing opponent distance to the goal (i.e., keep opponents away from the goal), maximizing the number of times blue team agents are tagged (i.e., remove opponents from play), and minimizing the number of times that they get tagged (i.e., avoid being removed from play). Thus, each agent has three conflicting objectives within its reward function, and the weights assigned to each determine their relative priority. Modifying these weights will change the incentives that each agent perceives.

Having specified this scenario within the PsychSim language, we can apply existing algorithms to autonomously generate decisions for individual agents [11]. Such algorithms enable the agent to consider possible moves (both immediate and future), generate expectations of the responses of the other agents, and compute an expected reward gain (or potentially loss) for each such move. It then chooses the move that maximizes this expected reward. Importantly, this algorithm can autonomously generate behavior without any additional specification, allowing us to observe differences in behavior that result from varying modeling parameters (e.g., the relative priority between objectives).

We ran the simulation with a variety of configurations for our blue agents in order to evaluate our testbed's suitability for studying team training. Weaimed to verify that variations in an agent's reward function would lead to different behavior in that agent, and that certain behaviors would consistently lead to better or worse outcomes for the team. These configurations would inform us as to what rewards our tutor agent should aim to instill in the students to ensure future success. Our measurement for team success was relatively simple, for each simulation our team would receive a score of 1 if they reached the goal within 60 turns, and a score of 0 if they failed to do so. We chose this turn limit because it allowed teams with successful strategies enough time to win from any starting position, but limited teams enough that sub-optimal strategies would lead to worse outcomes. For each configuration of our blue agents, we ran simulations over 10 starting positions for our blue team agents and totaled the scores of each round. These starting positions were designed to be representative of the variety of different starting scenarios for our team, varying distance to the goal and distance between team members, for example, when the blue team starts close or far away from the flag, or when the blue team members start together or apart.

This measurement of success does not penalize team members for what might be considered individual failures, such as being far from the goal or being eliminated by a red agent, so long as the team achieves success. The overall team score over 10 trials is shown in Figure 2. The X axis represents the weight of reward of avoiding being tagged by a Red team agent, from least wanting to avoid being tagged (left) to most wanting to avoid being tagged (right). The Y axis represents the weight of attempting to tag the Red team agents, from least wanting to tag the Red team agents (bottom) to most wanting to tag the Red team agents (top). The reward weight for moving closer to the goal was kept constant at 1.0.

Fig. 2. Simulation Score with Varying Agent Reward Values. A value of 1 means that the agent's goal, either tagging the opponent or avoid being tagged by the opponent, is as important as reaching the goal (i.e. capturing the flag). The values on each axis range from 1/4 to 4 times as important as reaching the goal (i.e. capturing the flag).

Looking at the overall results in Figure 2, we see a couple of trends. First of all, the more focus a team put on avoiding opponents, the worse they tended to perform. However, this effect is somewhat mitigated by focusing more on tagging opponents. Hence, a team that prioritizes avoiding opponents but puts almost no emphasis on tagging opponents could not win from any starting position. A team was most successful when reaching the goal and tagging opponents were given equal weight, and avoiding opponents was not overly prioritized. This is what our tutor should teach our team in this scenario.

5 Discussion

In this paper, we outlined a testbed for exploring team tutoring strategies. We did this via a simplified "capture-the-flag" scenario, in which our focus was uncovering what a tutor should instruct our team of agents to do so that they would win. Our testing showed that our team was most successful when they did not overly prioritize tagging, avoiding opponents, or reaching the flag.

These results imply that our tutor should instruct our team to focus equally on tagging opponents and reaching the flag, but not to put too much emphasis on avoiding opponents. While a team with the correct strategy can win from any of our starting positions, team starting location currently plays a significant part in success or failure when a sub-optimal strategy is enacted. This is to say, when a team failed to win all ten rounds, it was due to losing rounds in which they started further from the flag. This, combined with the turn limit being the reason for our team's failures, largely explains why prioritizing avoiding opponents impacts success so negatively. A team that spends too much time staying out of reach of their opponents will struggle to reach the flag within 60 turns. Our tutor can use this understanding in order to better guide students towards this optimal prioritization of motivations.

In this section, we propose a series of modifications that would be valuable for studying collaborative learning and team training. First of all, we would like to explore a wider variety of starting positions. Many of our chosen starting locations have our team very close together, and adding more locations with agents split up in a variety of ways (such as two in one corner, one in another) could help ensure the robustness of our results and conclusions. Furthermore, using an agent framework like PsychSim gives us many dimensions along which we can enrich the reasoning of our learners. For example, in the current configuration, agents always succeed in any action they attempt. Tutoring students with varying skill will provide a more significant challenge. All of the agents also know each others' objectives, which is not a realistic model of human teamwork. Giving the agents uncertainty about the reward function of other agents introduces the need for communication among teammates. We can leverage our underlying agent architecture's existing algorithms for belief update [11] and communication [15] to explore alternate communication strategies to establish coherent joint beliefs among team members. In other words, our learning agents would expand their action space to include possible messages, such as "There is a 90% chance that the red agent is at (3,3)".

While the work discussed here focuses on simulations of how teams train together with virtual agents, it can help inform the design of intelligent team tutoring systems for real human teams. In conclusion, the multiagent testbed we have constructed uses a relatively simple coordination scenario as a jumping-off point for a wide variety of potential simulations of collaborative learning and team training that can have implications for intelligent tutoring systems for real-human teams.

References

1. Bonner, D., Gilbert, S., Dorneich, M.C., Burke, S., Walton, J., Ray, C., Winer, E.: Taxonomy of teams, team tasks, and tutors. In: GIFT Users Symposium. pp. 189 (2015).
2. Bonner, D., Walton, J., Dorneich, M.C., Gilbert, S.B., Sottilare, R.A.: The development of a testbed to assess an intelligent tutoring system for teams. In: AIED Workshop on Developing a GIFT (2015).
3. du Boulay, B.: Recent meta-reviews and meta–analyses of aied systems. IJAIED **26**(1), pp. 536–537. (2016).
4. Boutilier, C., Dean, T., Hanks, S.: Decision-theoretic planning: Structural assumptions and computational leverage. JAIR **11**(1), pp. 94. (1999).
5. Busoniu, L., Babuska, R., De Schutter, B.: A comprehensive survey of multiagent reinforcement learning. IEEE Transactions on SMC **38**(2), pp.156–172. (2008).

6. Cohen, P.R., Levesque, H.J.: Teamwork. Nous **25**(4), pp.487–512. (1991).
7. Gilbert, S.B., Slavina, A., Dorneich, M.C., Sinatra, A.M., Bonner, D., Johnston, J., Holub, J., MacAllister, A., Winer, E.: Creating a team tutor using gift. IJAIED pp. 1–28. (2017).
8. Gmytrasiewicz, P.J., Durfee, E.H.: A rigorous, operational formalization of recursive modeling. In: ICMAS. pp. 125–132. (1995).
9. Grosz, B.J., Kraus, S.: Collaborative plans for complex group action. AIJ **86**(2), pp. 269–357. (1996).
10. Heinrichs, W.L., Youngblood, P., Harter, P.M., Dev, P.: Simulation for team training and assessment: Case studies of online training with virtual worlds. World Journal of Surgery **32**(2), pp. 161–170. (2008).
11. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Planning and acting in partially observable stochastic domains. AIJ **101**(1), pp. 99–134. (1998).
12. Kim, J.M., Hill, Jr., R.W., Durlach, P.J., Lane, H.C., Forbell, E., Core, M., Marsella, S., Pynadath, D., Hart, J.: BiLAT: A game-based environment for practicing negotiation in a cultural context. IJAIED **19**(3), pp. 289–308. (2009).
13. Klatt, J., Marsella, S., Krämer, N.C.: Negotiations in the context of aids prevention: An agent-based model using theory of mind. In: IVA. pp. 209–215. (2011).
14. MacAllister, A., Kohl, A., Gilbert, S., Winer, E., Dorneich, M., Bonner, D., Slavina, A.: Analysis of team tutoring training data. In: MODSIM World. (2017).
15. Marsella, S.C., Pynadath, D.V., Read, S.J.: PsychSim: Agent-based modeling of social interactions and influence. In: ICCM. pp. 243–248. (2004).
16. Matsuda, N., Sekar, V.P.C., Wall, N.: Metacognitive scaffolding amplifies the effect of learning by teaching a teachable agent. In: Penstein Rosé, C., Martínez-Maldonado, R., Hoppe, H.U., Luckin, R., Mavrikis, M., Porayska-Pomsta, K., McLaren, B., du Boulay, B. (eds.) Artificial Intelligence in Education. pp. 311–323. Springer International Publishing, Cham. (2018).
17. McAlinden, R., Pynadath, D., Hill, Jr., R.W.: UrbanSim: Using social simulation to train for stability operations. In: Ehlschlaeger, C. (ed.) Understanding Megacities with the Reconnaissance, Surveillance, and Intelligence Paradigm, pp. 90–99. (2014).
18. Merién, A., Van de Ven, J., Mol, B., Houterman, S., Oei, S.: Multidisciplinary team training in a simulation setting for acute obstetric emergencies: A systematic review. Obstetrics & Gynecology **115**(5), pp. 1021–1031. (2010).
19. Miller, L.C., Marsella, S., Dey, T., Appleby, P.R., Christensen, J.L., Klatt, J., Read, S.J.: Socially optimized learning in virtual environments (SOLVE). In: ICIDS. pp. 182–192. (2011).
20. Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. JAAMAS **11**(3), pp. 387–434. (2005).
21. Pynadath, D.V., Marsella, S.C.: PsychSim: Modeling theory of mind with decision-theoretic agents. In: IJCAI. pp. 1181–1186. (2005).
22. Pynadath, D.V., Marsella, S.C.: Socio-cultural modeling through decision-theoretic agents with theory of mind. In: Nicholson, D.M., Schmorrow, D.D. (eds.) Advances in Design for Cross-Cultural Activities, pp. 417–426. CRC Press. (2013).
23. Pynadath, D.V., Si, M., Marsella, S.C.: Modeling theory of mind and cognitive appraisal with decision-theoretic agents. In: Gratch, J., Marsella, S. (eds.) Social emotions in nature and artifact: Emotions in human and human-computer interaction, chap. 5, pp. 70–87. Oxford University Press. (2014).
24. Pynadath, D.V., Tambe, M.: The communicative multiagent team decision problem: Analyzing teamwork theories and models. JAIR **16**, pp. 389–423. (2002).
25. Pynadath, D.V., Wang, N., Merchant, C.: Toward acquiring a human behavior model of competition vs. cooperation. In: I/ITSEC. (2015).

26. Rickel, J., Johnson, W.L.: Virtual humans for team training in virtual reality. In: AIED. pp. 585. (1999).
27. Si, M., Marsella, S.C., Pynadath, D.V.: Thespian: Using multi-agent fitting to craft interactive drama. In: AAMAS. pp. 21–28. (2005).
28. Sottilare, R.A., Burke, C.S., Salas, E., Sinatra, A.M., Johnston, J.H., Gilbert, S.B.: Designing adaptive instruction for teams: A meta-analysis. IJAIED pp. 1–40. (2017).
29. Stone, P., Veloso, M.: Multiagent systems: A survey from a machine learning perspective. Autonomous Robots **8**(3), pp. 345–383. (2000).
30. Tambe, M.: Towards flexible teamwork. JAIR **7**, pp. 83–124. (1997).
31. Tambe, M., Johnson, W.L., Jones, R.M., Koss, F., Laird, J.E., Rosenbloom, P.S., Schwamb, K.: Intelligent agents for interactive simulation environments. AI Magazine **16**(1), pp. 15–40. (1995).
32. Tambe, M., Zhang, W.: Towards flexible teamwork in persistent teams. JAAMAS **3**(2), pp. 159–183. (2000).
33. Walton, J., Dorneich, M.C., Gilbert, S., Bonner, D., Winer, E., Ray, C.: Modality and timing of team feedback: Implications for GIFT. In: GIFT Users Symposium. pp. 190–198. (2014).
34. Walton, J., Gilbert, S.B., Winer, E., Dorneich, M.C., Bonner, D.: Evaluating distributed teams with the team multiple errands test. In: I/ITSEC. (2015).
35. Wang, N., Pynadath, D.V., Hill, S.G.: Building trust in a human-robot team. In: I/ITSEC. (2015).
36. Wang, N., Pynadath, D.V., Shankar, S., K.V., U., Merchant, C.: Intelligent agents for virtual simulation of human-robot interaction. In: HCI. pp. 228–329. (2015).
37. Whiten, A., Byrne, R.: Natural theories of mind: Evolution, development and simulation of everyday mindreading. B. Blackwell Oxford, UK. (1991).
38. Zachary, W., Cannon-Bowers, J.A., Bilazarian, P., Krecker, D.K., Lardieri, P.J., Burns, J.: The advanced embedded training system (AETS): An intelligent embedded tutoring system for tactical team training. IJAIED **10**, pp. 257–277. (1998).