A Framework for Engineering Human/Agent Teaming Systems

Rick Evertsz, John Thangarajah
RMIT University, Melbourne, Australia
{rick.evertsz, john.thagarajah}@rmit.edu.au

Abstract

The increasing capabilities of autonomous systems offer the potential for more effective teaming with humans. Effective human/agent teaming is facilitated by a mutual understanding of the team objective and how that objective is decomposed into team roles. This paper presents a framework for engineering human/agent teams that delineates the key human/agent teaming components, using TDF-T diagrams to *design* the agents/teams and then *present* contextualised team cognition to the human team members at runtime. Our hypothesis is that this facilitates effective human/agent teaming by enhancing the human's understanding of their role in the team and their coordination requirements. To evaluate this hypothesis we conducted a study with human participants using our user interface for the StarCraft strategy game, which presents pertinent, instantiated TDF-T diagrams to the human at runtime. The performance of human participants in the study indicates that their ability to work in concert with the non-player characters in the game is significantly enhanced by the timely presentation of a diagrammatic representation of team cognition.

Introduction

Recent advances in autonomy have resulted in artificial agents that can tackle roles that were previously only feasible for humans. Many applications require humans to work together as a team, and the potential for autonomous agents to become part of such teams has led to research into the problems of human/agent teams. Human/agent teams present a broad range of challenges, including: (i) sharing situation awareness between human and artificial team members (Miller et al. 2014), (ii) explainable agency (Langley et al. 2017), (iii) explainable planning (Fox, Long, and Magazzeni 2017), and (iv) designing teams to better support human/agent teams (Gao, Cummings, and Solovey 2016).

The work described here is part of a long-term air domain training simulation program. These simulations, for years, have relied on human support personnel (known as the *white force*) to take on various roles in the scenarios (e.g. wingmen or hostile entities). The white force typically outnumbers the trainees by a wide margin, and so these training simulations are very resource intensive. To overcome this, we are investigating the creation of the white force as a collection of human/agent teams, where a combination of humans and artificial autonomous agents work together collaboratively to drive the training scenarios. In these training scenarios, the tactics are typically pre-defined and well understood. More generally, humans use pre-defined tactics in many team-oriented domains, for example in sports, emergency management and air-combat tactics. Pre-defined tactics are most effective in contexts where time is limited and there are commonalities that allow the tactics to be re-applied successfully. The tactics must be familiar to the team members and this reduces the communications overhead because there is no need to explain the course of action.

With this in mind, our research on human/agent teams addresses how to *engineer* multi-agent systems so that the human can be incorporated as an effective member of the team. In particular, our stakeholders are interested in applications where the human is a *peer* rather than a *supervisor* of the agents. We have focused on using diagrammatic methods of representing and presenting the team's structure and decision making so that the human can see a visual overview of their role in the team, what goals they should be tackling at any given time, and the extent to which their activities need to synchronise with teammates. For the representation, we investigated the application of TDF-T (Evertsz, Thangarajah, and Papasimeon 2017) to the wider problem of engineering human/agent teams. We explored how to contextualise the TDF-T design diagrams by determining which aspects to *instantiate* and present to the human at runtime. Our hypothesis was that such contextualised diagrammatic representations of team cognition can significantly enhance the human's ability to collaborate with the peer agents.

This paper makes the following contributions: (i) the important human/agent teaming parameters are identified, including how to design models to support human/agent teaming; (ii) a novel, contextualised diagrammatic approach is proposed for the runtime presentation of team behaviour to the human; and (iii) the effectiveness of the approach is evaluated with users in the context of a human/agent real-time strategy game.

Background

In principle, human/agent interaction can span a wide spectrum, from control of the agent's concrete *actions* by the human, through supervisory control in terms of abstractions such as *goals*, to team-oriented applications where the human is one among a number of peers, some of which are artificial. Joint activity can involve various types of mutual behaviour (Bradshaw, Feltovich, and Johnson 2012), in particular: co-allocation, cooperation and collaboration.

Teamwork. In the Multi-Agent Systems (MAS) field, Joint Intentions Theory (Cohen and Levesque 1991) is probably the most influential perspective on the behavioural dimension of teamwork. It proposed that team behaviour should be modelled in terms of the collective constructs: *joint intention*, *joint commitment* and *joint persistent goal*. The notion of joint intention is taken to mean a joint commitment by the agents to pursue a collective goal within the context of some shared mental state. Effective team performance relies on having a mutual understanding of the joint goal and the current situation, and having mechanisms for coordinating the activities of the team members. Human/agent teams present further challenges because it is more difficult to share situation awareness and coordinate activity. Whereas artificial agents can typically transfer and share data structures and can be controlled in software, humans must rely on the external environment for communication and are free to act independently of the team; this complicates timely coordination with the artificial agents in the team.

The human/agent team problem overlaps with the more general issue of how to integrate heterogeneous agents into a cohesive team. TEAMCORE (Tambe et al. 1999) and Machinetta (Scerri et al. 2003) integrate heterogeneous agents by leveraging proxy agents to implement a homogeneous team layer that handles the coordination aspects independently of the heterogeneous agents.

Human Teams. Studies of exclusively human teams, e.g. (Espinosa et al. 2004), reveal that coordination can be both *explicit* and *implicit*. Explicit coordination occurs through mechanisms, such as commonly understood team tactics, plans and procedures, and also through verbal and non-verbal communication. Implicit coordination is achieved via meta-cognitive methods such as using shared mental models that allow teammates to anticipate each other's activities and needs, thereby fostering coordinated action (Espinosa et al. 2004). Such implicit coordination is possible when one is able to infer what one's teammates intend to do, and one way humans do this is by applying a *theory of mind* that incorporates the concepts of *belief*, *desire* and *intention* (Dennett 1987).

The Team as an Intentional Entity. The question then is: how do humans conceptualise teams and team decision making? Following Searle (Searle 1997), we argue that humans view the team as, in some sense, having its own mental state. A human team member will want to know what the *team* is meant to achieve, not merely what the individual agents are doing; an understanding of the overall joint objective and course of action being followed is key to understanding the team's behaviour.

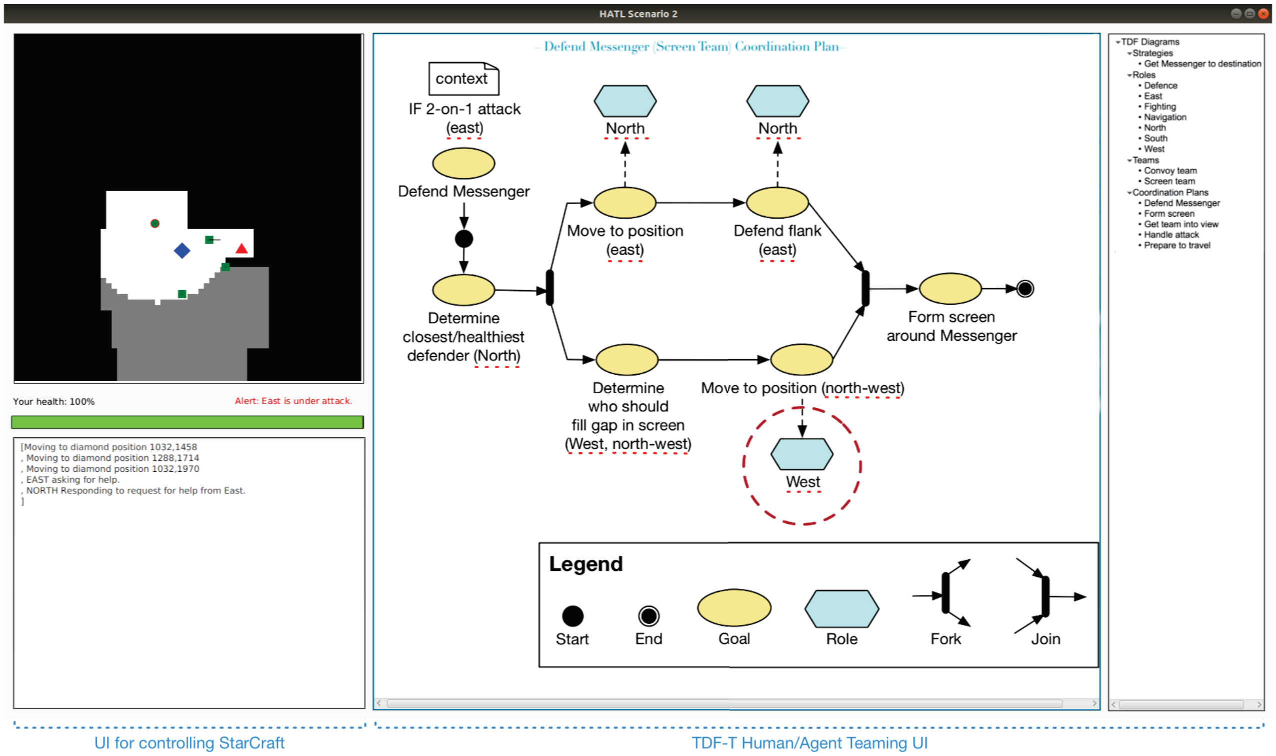
Humans talk about the team in a way that gives it equivalent status to such abstractions as: *agent*, *belief*, *desire* and *intention*. When talking about a group of individuals working towards a common goal, humans talk about the team as if it were an entity with its own beliefs, desires and intentions. For example, one says that the *soccer team is trying to win the game*, or that the *platoon doesn't know that it is about to be attacked*. Although humans understand that the team is not a physical entity, it reduces cognitive complexity to treat it as if it is.

BDI-Based Team Modelling. Through the work of Bratman (Bratman 1987) on practical reasoning, Dennett's *intentional stance* (Dennett 1987) gave rise to the BDI (Beliefs, Desires, Intentions) paradigm, a popular modelling approach in the MAS community. In the BDI paradigm, agents have *desires* they wish to achieve, and use their *beliefs* about the world to inform the adoption of *intentions* that they commit to. We suggested above that humans naturally conceptualise a team as an *intentional entity with joint mental state*. We further argue that humans naturally conceptualise team cognition as being analogous to that of intentional individuals, namely humans, and for this reason we adopted the BDI paradigm to represent and present team behaviour in human/agent teams, and so used TDF-T (Evertsz, Thangarajah, and Papasimeon 2017) which is a BDI-based team modelling formalism, although other formalisms with similar teaming concepts could have potentially been used.

Case Study and Testbed

The case study from the user evaluation we conducted will be used for the diagrammatic examples of the representational and presentational guidelines in the next two sections, and so it is presented here. A StarCraft (Sta 2018) testbed was developed to provide a simple sandbox environment for investigating human/agent teams independently of our stakeholder's training simulation environment, which is in regular use and so is often unavailable for running experiments. Note that the case study does not rely on participants having pre-existing knowledge of StarCraft.

In the case study (Figure 1), the human plays the role of one of four Marines tasked with guarding a Messenger (blue diamond) who must reach a remote base that is situated to the north of the start location. Enemy combatants (red triangles) wait in ambush and try to kill the Messenger. The human (green circle) and the three Marine agents (green squares) must prevent this by working together to defeat the enemy combatants whenever they attack. The assigned strategy is to form a defensive screen around the Messenger. The Messenger is responsible for navigating to the destination. The game ends when either the Messenger is killed or the group reaches the destination. In general, a Marine (including the human) can defeat a single enemy combatant one-on-one, but will lose if faced with greater numbers. This means that if more than one enemy combatant attacks from a given direction, the Marines must fight together so that they are not outnumbered by the attackers.



```
[Moving to diamond position 1032,1458
, Moving to diamond position 1288,1714
, Moving to diamond position 1032,1970
, EAST asking for help.
, NORTH Responding to request for help from East.
]
```

Figure 1: StarCraft/TDF-T testbed with instantiated team coordination plan

Figure 1 shows the layout of the interface we developed. The lefthand column provides the game-view display (top) of the entities in the scenario, allows commands to be sent to StarCraft, and shows the health of the Marine controlled by the human (middle green bar) and the inter-agent messaging of the team (bottom). Here, the game-view display shows the human (green circle) defending the north-west, while two Marines (green squares) attack the enemy (red triangle) to the east. The black area represents areas not yet visited by the team, the grey area shows areas visited, and the white area shows the region that is currently within the line of sight of the human-controlled agent. The middle section of the UI shows the *instantiated* TDF-T coordination plan that is currently executing; for illustrative purposes, instantiated values are underlined in red (underlining is not shown in the actual UI). The righthand side provides a menu of TDF-T diagrams that the user can inspect if needed.

Engineering Human/Agent Teams

Although in all-agent teams it is generally accepted that the explicit modelling of teamwork is advantageous, e.g. (Tambe 1997), it is not essential. One can build MAS that do not embody explicit team structures and team plans, but instead represent the team and its coordination mechanisms *implicitly* through runtime inter-agent messaging. However, this is not practical in human/agent teams; in order to work effectively as part of the team, the human needs to have an overall understanding of the team's structure and the requirements for coordination as they relate to the human's role in the team. The best way to communicate these team-level aspects to the human will depend on the characteristics of the domain, for example, how much time is available. Nevertheless, we argue that, regardless of the domain and the available medium for human/agent communication, it is impractical to present a team-level view to the human if it is not explicitly part of the *design* of the human/agent system. For example, the required roles need to be delineated, as well as team hierarchies and team coordination plans.

Over the last two years, together with our stakeholders, we have analysed a number of human/agent teaming scenarios with a view to distilling the important team-level modelling requirements; for example, we explored how the human can team with autonomous unmanned air and ground vehicles to provide an effective base protection capability. This resulted in the following engineering methodology for human/agent teams. The methodology addresses two key aspects of engineering human/agent teams: (i) *what* to represent, and (ii) *how* to represent those artefacts.

What Team Artefacts to Represent

Here we describe the key team-level artefacts that need to be represented to support human/agent teams and we also motivate their inclusion in the methodology; these are *strategies*, *role enactments*, *team structures*, *team beliefs* and *team coordination plans*.

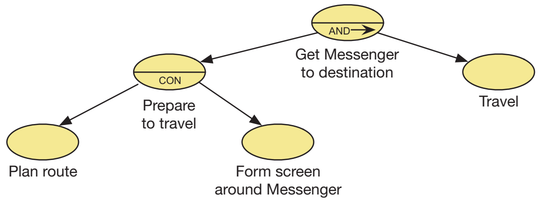


Figure 2: Overall strategy

Strategies: The high-level decomposition of the team goal into sub-goals is important because it enables the human team member to gain a rapid appreciation of the team's general approach to the problem. In TDF-T, this goal tree is represented as a *strategy*. A strategy decomposes the main goal into its sub-goals without specifying *how* those sub-goals are achieved or by *whom*. Figure 2 shows the case study strategy for taking a Messenger to a destination and shielding them on the way. The two sub-goals of `Get Messenger to destination' are tried sequentially from left-to-right, and the `Prepare to travel' goal has two concurrent sub-goals.

Role enactments: Goals represent what can be achieved by the system being modelled. In practice, goals are adopted by particular entities, whether individuals or teams. When modelling a system, it is useful to group related goals into *roles* (related in the sense that a role filler should take on the responsibility for achieving that group of goals). An entity's role not only represents a *responsibility* that it has, but also *functionality* that it must possess; after all, for a team to function effectively, its members should only take on the responsibility for roles they are capable of fulfilling. This is important in human/agent teams because it provides the human team member with a view of what goals they are expected to be able to achieve as part of their role in the team. Figure 3 shows a TDF-T Role Enactment diagram with the goals that the `Defence' role is responsible for (enacted by the `Screen' team, which the human will be a part of).

Team structures: In order for the human to fulfil their position in the team, it is important for them to understand the team's structure, and which agents/teams perform the various roles required. Figure 4 from the case study shows the structure of a `Convoy' team that gets formed in response to the `Form convoy' goal. It consists of a `Defence' and a `Navigation' role. The `Navigation' role is enacted by a `Messenger' agent and the `Defence' role by the `Screen' sub-team, which is decomposed into four roles enacted by `Marine' agents. Note that these diagrams show agent/team *types* rather than *instances*, and will need to be instantiated at runtime.

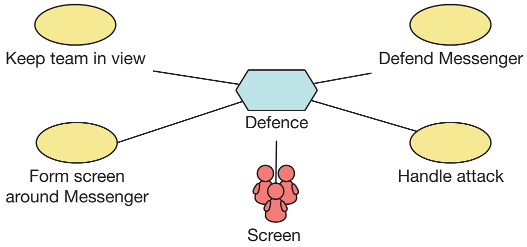


Figure 3: Role enactments

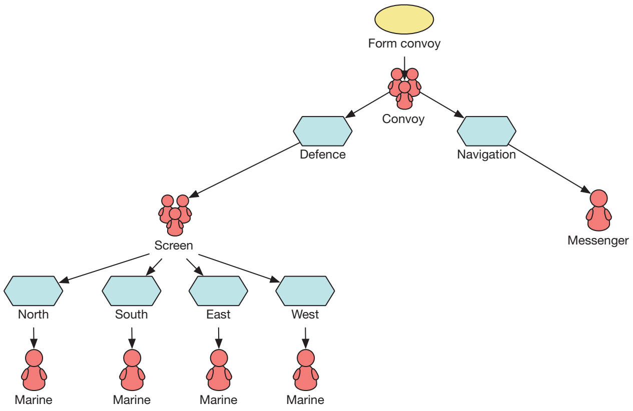


Figure 4: Team structure

Team beliefs: To function effectively, the team may need to share information about the current situation; in TDF-T this is represented in team belief sets (due to space constraints, not shown here). The members of the team can read/write from/to their team's belief sets, and in this way access and augment shared data within the team.

Team coordination plans: Up to this point, the team's overall strategy, roles, structure and shared data have been defined, but the coordination of the team members' activities has yet to be specified. In TDF-T, how a team responds to an event is specified by one or more team coordination plans. The team coordination plan uses roles to reference the team members whose activities need to be coordinated. It procedurally specifies sequencing, concurrency and timing.

An instantiated team coordination plan from the case study was shown in Figure 1. An uninstantiated plan has variables rather than the values underlined in red in the Figure, for example `?DEFENDER' rather than `North'. This plan responds to the `Defend Messenger' goal in the context where there is a two-on-one attack, and begins by determining which role should defend (binding `?DEFENDER' to `North'). The plan then forks into two concurrent branches; the top branch sequentially and synchronously delegates the `Move to position' and `Defend flank' goals to the `North' role (bound earlier). Concurrently, the plan determines who should fill the resulting gap in the defensive screen (`West' in this example) and the position to be filled (`north-west'), and then delegates the `Move to position' goal to the `West' role. Once both branches are complete, they join and the team reforms the screen around the Messenger.

How to Represent Team Artefacts

Here we describe *how* to represent the key team-level artefacts for human/agent teams as well as why these representational rules are important to the successful implementation of human/agent teams. MAS are typically built with software engineering concerns in mind, such as maintainability, reusability and execution efficiency. However, we argue that the human's need to understand the models is pivotal when designing for human/agent team applications. Software engineering and human factors overlap to some extent, because a model that is designed to be understandable to a human team member at runtime is also likely to be more readable and therefore more maintainable. Nevertheless, although there is an intersection of the benefits from explainability and software engineering, there can be cases where focusing on one will come at the expense of the other, for example, if one focuses on developing an algorithm that is efficient, this could come at the expense of understandability.

Although a wider range of domains need to be investigated in order to make broad scientific claims, working with our stakeholders, we have found the following rules to be the most important in engineering teams that mix humans with agents:

1. **Use meaningful team-level artefact names.** It is essential to use standard domain terminology for the team-level artefacts in a model, for example, naming a goal `'Classify Detection'` rather than `'G1'`. At runtime, the meaning of presented artefacts should be immediately obvious to any human who is familiar with the domain. This rule typically requires that domain experts be involved in the model creation process.
2. **Make intentionality explicit.** Focus on expressing the *intentional* aspects of the team by building goal/sub-goal hierarchies rather than team plans that merely consist of sequences of actions;
3. **Structure diagrams to allow rapid visual scanning.** Although we have not developed quantitative guidelines, diagrams should be decomposed so that hierarchies are no more than three or four levels deep, and team coordination plans should comprise fewer than ten steps.

Presentation of Team Cognition

Apart from the representational aspects discussed in the previous section, the system also needs to present the relevant information to the human at the appropriate time. A number of factors will determine the effectiveness of the information presented to the human team member, including human factors such as cognitive load, which can be affected in turn by time pressure, user interface affordance, and stress resulting from the perceived consequences of failure. If there are only a few seconds to perform a task, then it is unlikely that the human will have time to look at the related TDF-T diagram. Similarly, if the user interface is difficult to use, then it will interfere with task execution, leading to poorer performance. Although these concerns lie outside the scope of this study, they present important challenges for the field of human/agent teams.

From an engineering standpoint there are three key aspects to consider in this regard:

- **Offline (prerequisite) information** - Presented prior to the humans joining a team of agents;
- **Online information** - Shown at runtime as the scenario progresses; and
- **User Interface** - The medium through which the human interacts with the system to receive the information and communicate with the members of its team and the overall system. This important aspect lies outside the scope of this study.

Offline (Prerequisite) Information

Based on earlier evidence that TDF-T is easy for people to understand (Evertsz, Thangarajah, and Papasimeon 2017), our hypothesis was that TDF-T diagrams would be an effective means of presenting important team-level information to the human so they can become familiar with the models offline. The guidelines we presented in the previous section foster a clearer representation, and this offline information was one aspect of our evaluation that we present ahead.

We note that although there is a general top-down approach to navigating the diagrams, the users are free to view and study the models in any order, as suited to their expertise. For example, a novice may choose to peruse all of the diagrams in a top-down fashion, whereas an expert in the field may choose to focus on the team coordination plans or the belief structures.

A potentially significant advantage of this approach is that the diagrammatic method of presentation directly maps to the diagrammatic representation of the TDF-T models underlying the system's behaviour. This offers two major benefits: (i) humans can become familiar with the models offline, for example, common team coordination plans used by the system; and (ii) it is straightforward to pinpoint any unclear parts of the presented diagrams during *after action review* of the human/agent team's performance and then identify the relevant part of the underlying TDF-T model.

Online Information

The design diagrams represent the artefact *types* and their interrelationships. In attempting to use them to present team behaviour to human participants, it became clear that the use of types makes the diagrams too abstract for runtime presentation. Human reasoning is highly contextual, but the situational context is largely missing from the design diagrams, as they are intended to be general purpose, reusable designs. Therefore, in order to present them at runtime, they must be *instantiated* with the current context in which they are being applied.

In presenting these runtime instantiations of design models, we had to consider which of the many diagrams are to be instantiated and how to present them on-demand. After some trials, we have taken the following approach:

- The roles are assigned to human/agent *instances*.

- The team structures at the design stage can capture context-dependent alternatives, but at runtime, the instantiated team structure must be presented with the human's role(s) within the team clearly indicated.
- When a team is required to respond to an event, the relevant team coordination plan is instantiated and displayed, with a clear indication of the tasks required to be performed by the human. Note that not all of the team coordination plan might be instantiated at the start, but as the plan unfolds, further bindings will occur and the UI must update the diagram as the plan progresses.

This approach requires the following three components:

- an agent/team design tool,
- an agent/team programming language that implements the artefacts in the design, and
- middleware that extracts the relevant design diagrams, instantiates them and updates the instantiations by observing the runtime agent/team execution. This middleware must be integrated with the UI.

In our prototype, we use TDF-T as the design tool, the SARL agent language (Rodriguez, Gaud, and Galland 2014) for the implementation and we developed a proof-of-concept middleware integrated with a Java-based UI.

User Evaluation

Our research objective was to investigate the utility of TDF-T in supporting humans in human/agent teams. Following encouraging trials of the approach, we evaluated the hypothesis that the presentation of TDF-T diagrams at runtime will significantly enhance the human's performance in a human/agent team. To test this hypothesis, we developed the TDF-T/SARL/StarCraft testbed and case study described earlier.

In order to successfully defend the Messenger, the participant must (i) maintain formation, (ii) not move to defend the attacked east flank (the North agent is nearer), (iii) move to a north-west position to fill the gap left by the North agent who is fighting on the east flank, (iv) fight the attacker from the north, (v) move back west into formation, (vi) fight the west attacker while moving east to keep the team in view, and (vii) fight the enemy who attacks from the south-east. These seven checkpoints were the criteria by which the human's performance was evaluated. If the Messenger was killed at any point, the scenario was re-run from the checkpoint which comes immediately *after* the point where the participant failed in the previous run. In this way, each participant's performance was recorded for all seven checkpoints (see Figure 5 for a diagram showing all of the checkpoints apart from (i) and (v), which only relate to screen formation around the Messenger; their inclusion would unnecessarily complicate the diagram).

Experiments

Two experimental conditions were evaluated. In the TDF-T condition, TDF-T diagrams were presented, whereas in the Baseline (non TDF-T) condition, no TDF-T diagrams were shown. From a pool of 16, eight participants were randomly allocated to each condition; all had a computer science background. Both groups were given practice in using the testbed UI until they could interact with the game proficiently.

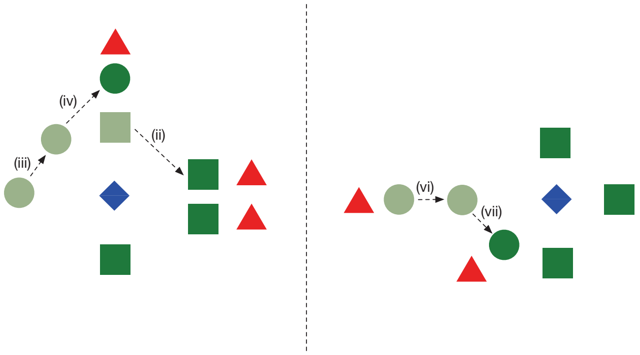


Figure 5: Scenario progression

TDF-T Condition:

The participants were given a written introduction to TDF-T, covering: strategy, role enactment, team structure and team coordination plan diagrams. Their understanding of TDF-T was then tested with a further series of the same types of diagram and a questionnaire. All of the participants completed the questionnaire successfully, indicating sufficient competence to understand the diagrams to be presented during the scenario run. This test also served as a way of evaluating the suitability of TDF-T diagrams for the case study.

They were then given a written account of the scenario that was a paraphrase of the description presented earlier in the **Case Study and Testbed** section, and were shown the strategy, role enactment and team structure diagrams for the scenario. The role enactment diagram was instantiated to show the role they would take on, and highlighted the goals their role was responsible for. The team structure diagram showed their role in the team hierarchy. During the running scenario, the relevant instantiated team coordination plans were displayed when the human was required to coordinate their actions with the rest of the team. After the game was over, they filled out a survey form with two questions that rated from 1-6: (i) how easy the TDF-T diagrams were to understand, and (ii) how helpful the participants found them to be during the scenario run.

Baseline (non TDF-T):

The participants were given the same written account of the scenario as the group in the TDF-T condition, but augmented with the following information: (i) the participant is responsible for defending the west flank of the team, and (ii) the participant has limited line of sight, and must keep the team in view so as not to lose sight of them. This extra information corresponds to what was shown to the participants in the TDF-T diagrams in the TDF-T condition.

Results and Analysis

Table 1 shows, for each checkpoint, the mean percentage of participants who correctly handled the checkpoint in each condition. The results indicate a clear advantage for the TDF-T group. In the baseline condition, at checkpoint (ii), 75% went east to fight the enemy, leaving the west flank exposed; this was despite being told they were responsible for the west flank. After the restart, they stayed on the west flank, but did not know to move north-west to fill the gap created by the North agent moving east. Also, none of the participants in the baseline condition kept the team in view as it moved east, and so they were all out of position when the south-east enemy attacked, and although the south-east enemy could be seen, the participants could not get there before the Messenger was killed.

In the TDF-T group, understandability of the diagrams was rated highly (μ_1 = 85%, σ_1 = 0.11), as was their assessment of how helpful the TDF-T diagrams were (μ_2 = 85%, σ_2 = 0.06). This subjective impression was backed up by: (i) all of them successfully completing the questionnaire that tested their understanding of TDF-T during the tutorial phase, and (ii) their superior performance in the game, despite only having a short time to learn the TDF-T notation.

We were curious as to why most of the non TDF-T group went to defend against the attack from the east (checkpoint (ii)), even though they were told their role in the team was to defend the west flank. From the debrief after the experiment, it was clear that it was just a knee-jerk reaction to the attack; they hadn't forgotten the instruction to defend the west flank, but perceived it as a surprise attack that would make the original strategy ineffective. We believe that the TDF-T group resisted the temptation to defend the east flank because the team coordination plan had popped up on the UI, and in some sense conveyed the fact that it was *current* rather than an outdated strategy that was invalidated by the surprise attack. This highlights an important aspect of how humans perform in teams; if they believe that the team has not fully taken the situation into account, they may elect to act independently.

In summary, the results clearly indicate the need for a framework, such as the one we have described, that uses diagrammatic representations and dynamic runtime presentations of team cognition, as it significantly enhances the human's ability to effectively collaborate and participate in such teams.

Checkpoint	**Mean% TDF-T**	**Mean% Baseline**
(i) maintain formation	100	87.5
(ii) do not go east	87.5	25
(iii) move north-west	87.5	0
(iv) fight north attacker	87.5	0
(v) back into formation	75	75
(vi) fight and move east	75	0
(vii) fight south-east enemy	75	0

Table 1: Mean percentage completing each checkpoint

Discussion

This paper presented a framework for engineering human/agent teams to support presenting team cognition to the human team members at runtime. This approach uses the same diagrammatic team representation to both *design* and *present* team decision making.

The key design concepts for human/agent team models were discussed, as well as general heuristics for designing the diagrams to be displayed. Note that we are not claiming that TDF-T is the optimal diagrammatic representation for supporting human/agent teams; another diagrammatic representation with similar concepts could potentially have been used instead.

We developed novel presentation techniques that instantiate the relevant diagrams at runtime so that they are grounded and show the human team member the relevant context. This allows the human to see their role in the team and fulfil their responsibilities at the appropriate junctures during plan execution.

The utility of the approach was evaluated using a purpose-built TDF-T/SARL/StarCraft testbed, and the case study indicated that the diagrammatic team representation helps the human team member work in concert with the artificial agents.

We believe that a number of factors contributed to the effectiveness of the approach, including (i) the BDI model is natural to humans and they relate to it as an account of team-level cognition, (ii) the diagrammatic representation is intuitive and the human can easily understand the team structure, their own role(s) and goals, and the part they need to play in the current coordination plan, and (iii) the use of predefined models allows the human to understand the system in advance and quickly recognise the current approach and how it relates to the situation faced; this latter aspect is common in exclusively human teams, where there is often shared knowledge of standard tactics that work well in the domain.

In this paper, we have focused on how to engineer human/agent teams and present their cognition to human team members. Of course, there are many other aspects to human/agent teaming that we have not addressed, and they represent opportunities for complementing the work reported here. For example, transparency, performance, flexible autonomy, trust, psychological safety, group cohesion, maintenance of shared knowledge, learning and adapting to humans in the team, communication, interacting with physical environments and so on, to name a few. As human/agent teams become more pervasive, innovative solutions will be required to address these issues.

Acknowledgements

We are grateful to DST Group and the Defence Science Institute for their invaluable support of this work.

References

Bradshaw, J. M.; Feltovich, P. J.; and Johnson, M. 2012. Human-agent interaction. *The Handbook of Human-Machine Interaction: A Human-Centered Design Approach*.

Bratman, M. E. 1987. *Intention, Plans, and Practical Reasoning*. Cambridge, MA (USA): Harvard University Press.

Cohen, P. R., and Levesque, H. J. 1991. Teamwork. *Nous* 25(4):487–512.

Dennett, D. C. 1987. *The Intentional Stance*. MIT press.

Espinosa, A.; Lerch, F. J.; Kraut, R. E.; Salas, E.; and Fiore, S. M. 2004. Explicit vs. implicit coordination mechanisms and task dependencies: one size does not fit all. *Team cognition: understanding the factors that drive process and performance. American Psychological Association, Washington, DC* 107–129.

Evertsz, R.; Thangarajah, J.; and Papasimeon, M. 2017. The conceptual modelling of dynamic teams for autonomous systems. In Mayr, H. C.; Guizzardi, G.; Ma, H.; and Pastor, O., eds., *Conceptual Modeling*, 311–324. Cham: Springer International Publishing.

Fox, M.; Long, D.; and Magazzeni, D. 2017. Explainable planning. *arXiv preprint arXiv:1709.10256*.

Gao, F.; Cummings, M.; and Solovey, E. 2016. Designing for robust and effective teamwork in human-agent teams. In *Robust intelligence and trust in autonomous systems*. Springer. 167–190.

Langley, P.; Meadows, B.; Sridharan, M.; and Choi, D. 2017. Explainable agency for intelligent autonomous systems. In *AAAI*, 4762–4764.

Miller, T.; Pearce, A.; Sonenberg, L.; Dignum, F.; Felli, P.; and Muise, C. 2014. Foundations of human-agent collaboration: situation-relevant information sharing. In *2014 AAAI Fall Symposium Series*.

Rodriguez, S.; Gaud, N.; and Galland, S. 2014. SARL: a general-purpose agent-oriented programming language. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, volume 3, 103–110. IEEE.

Scerri, P.; Pynadath, D. V.; Schurr, N.; Farinelli, A.; Gandhe, S.; and Tambe, M. 2003. Team oriented programming and proxy agents: The next generation. In *International Workshop on Programming Multi-Agent Systems*, 131–148. Springer.

Searle, J. R. 1997. Responses to critics of the construction of social reality. *Philosophy and Phenomenological Research* 57(2):449–458.

2018. StarCraft. https://en.wikipedia.org/wiki/StarCraft. Accessed: 2019-11-14.

Tambe, M.; Shen, W.-M.; Mataric, M.; Pynadath, D. V.; Goldberg, D.; Modi, P. J.; Qiu, Z.; and Salemi, B. 1999. Teamwork in cyberspace: Using TEAMCORE to make agents team-ready. In *Proceedings of the AAAI spring symposium on agents in cyberspace*, 136–141.

Tambe, M. 1997. Towards flexible teamwork. *Journal of artificial intelligence research* 7:83–124.