RT4T: A Reconfigurable Testbed for Joint Human-Agent-Robot Teamwork

Matthew Johnson, Jeffrey M. Bradshaw, Daniel Duran, Micael Vignati, Paul J. Feltovich
Florida Institute for Human and Machine Cognition (IHMC)
40 South Alcaniz Street
Pensacola, FL 32502 USA
+1 (850) 202-4462
{mjohnson, jbradshaw, dduran, mvignati, pfeltovich}@ihmc.us

Catholijn Jonker, M. Birna van Riemsdijk
EEMCS
Delft University of Technology
Delft, The Netherlands
+31 6.48.87.52.07
{c.m.jonker, m.b.vanriemsdijk}@tudelft.nl

ABSTRACT

In this paper, we describe a reconfigurable testbed for experiementation on joint activity in mixed human-agent-robot teamwork (HART). The testbed was originally inspired by the classic AI planning problem of Blocks World (BW) extended into what we call Blocks World for Teams (BW4T) [1] and now with more generality and power into RT4T, a Reconfigurable Testbed for Teams. By teams, we mean at least two, but usually more human , agent, or robot members. We describe the results of two experiments using BW4T, one showing the results of increasing autonomy without addressing interdependence and the other addressing soft interdependence as a performance factor. We introduce RT4T and a new teamwork measurement schema.

General Terms

Measurement, Design, Experimentation, Human Factors

Keywords

Joint Activity, Coordination, Teamwork, Coactive Design, HART

1. INTRODUCTION

As a means for inexpensive, rapid evaluation of hypotheses relating to human-robot teamwork, IHMC has developed a simulation testbed for joint activity While there have been plenty of multi-agent system (MAS) testbeds, there are very few testbeds specifically designed for arbitrary sized heterogeneous (human and agent) teams. This testbed is similar in some ways to MICE (Michigan's Intelligent Coordination Experiment) [2] in that it addresses a simple domain. BW4T and RT4T are similar to Gamebots3D [3] in that we focus on human participation.

Despite the similarities, there are some significant differences in our testbeds with previous work by others. MICE is a discrete event 2D only testbed. BW4T is a continuous event 2D only testbed. Gamebots is a continuous event 3D testbed. RT4T is a continuous event 3D testbed. MICE and Gamebots tend to focus on post-activity performance metrics (see Section 5 below). In contrast, both BW4T and RT4T use interdependence as a way to understand both design-time and runtime issues. For example, BW4T can use block color to vary interdependence within the same activity. In addition, BW4T and RT4T are more suited to explore the variety of communications possible in teamwork. For example, Gamebots provides only a basic set of communication options, including enumerated messages like "Defend the base," "Hold your position," and "Cover me." No rationale for the choice of these messages is given and it is not clear whether the message set is extensible. In BW4T and RT4T, the choice of messages is configurable and motivated by theoretical considerations.

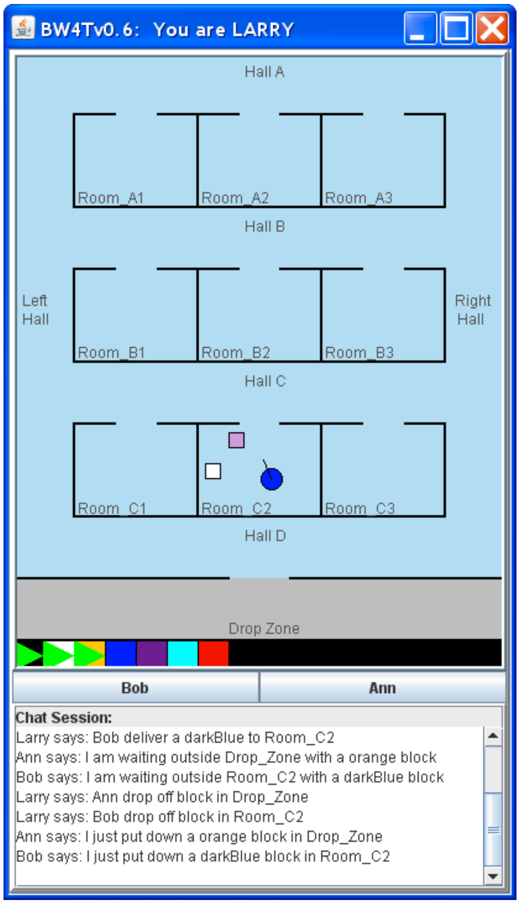
An impressive capability of our testbeds is the ability to address problems and scenarios of increasing complexity in both taskwork and teamwork. The fact that we can deploy the simulation equivalent of real robots in our testbeds sets us apart from approaches such as Colored Trails.

2. INITIAL VERSION OF BW4T

We will describe the simulation environment with reference to the classic AI planning problem of Blocks World (BW) [4]. BW has been a popular test domain with the Planning community because of its simplicity and was borrowed by the Distributed AI (DAI) and Multi-Agent Systems (MAS) community to study distributed planning and coordination. We extend BW into what we are calling Blocks World for Teams (BW4T). Teams consist of at least two, but usually more members. Additionally, we do not restrict the membership to artificial agents, but include and in fact expect human members. Study of *joint activity of heterogeneous teams* is the main function of the BW4T testbed. The goal of the BW4T game is to "stack" colored blocks in a particular order. To keep things simple, the blocks are unstacked to begin with, so unstacking is not necessary.

In order to study joint activity of heterogeneous teams in a controlled manner, we extend the basic BW problem in a few ways. First, instead of having only one player, as usual in BW, for BW4T we allow multiple players as in the DAI and MAS work. Our approach is different in that players can be combinations of both human and artificial agents. Second, instead of having all the blocks visible on a table, we hide them in a series of rooms. Agents can only see blocks that are in the same room as they are (though this feature can be changed if the experimenter desires). This feature is usually added to force the coordination to be explicit, i.e., to force coordination through communication. Coordination can frequently occur through observation of the environment and non-verbal cues. While implicit coordination is another valuable area of study, these cues can be very difficult to detect and measure. Restricting the visibility will force explicit communication. A restricted chat window is provided for communication. By controlling the goal and the communication options, we can influence the need for coordination and type of coordination available during the joint activity.

The most important variation on the problem we have made is to allow multiple players to work jointly on the same task. We control the observability between players and the environment. The degree of interdependence that is embedded in the task is represented by the complexity of color orderings within the goal stack. The task environment (Figure 1) is composed of nine rooms containing a random assortment of blocks and a drop off area for the goal. The environment is hidden from each of the players, except for the contents of the current room. Teams may be composed of two or more players, each working toward the shared team goal. Players cannot see each other, so coordination must be explicit through the chat window. The task can be done without any coordination, but it is clear that coordination (i.e., the players managing their interdependence) can be beneficial.



Larry says: Bob deliver a darkBlue to Room_C2
Ann says: I am waiting outside Drop_Zone with a orange block
Bob says: I am waiting outside Room_C2 with a darkBlue block
Larry says: Ann drop off block in Drop_Zone
Larry says: Bob drop off block in Room_C2
Ann says: I just put down a orange block in Drop_Zone
Bob says: I just put down a darkBlue block in Room_C2

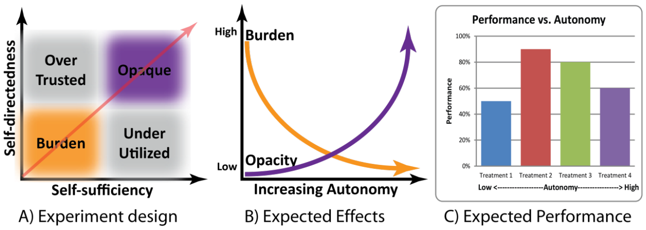
Figure 1. BW4T – Two-player example

3. INITIAL EXPERIMENTS

3.1 Experiment 1: Adding Autonomy Without Addressing Interdependence

A common suggestion for how to improve human-agent systems is to increase the level of agent autonomy. This solution is also commonly proposed for future systems. It is true that additional increments of agent autonomy *might*, in a given circumstance, reap benefits to team performance through reduction of human burden.

However, there is a point in problem complexity at which the benefits of autonomy may be outweighed by the increase in system opacity when interdependence issues are not adequately addressed. The fundamental principle of Coactive Design is that, in sophisticated human-agent systems, the underlying interdependence of participants in joint activity is a critical factor in human-agent system design [9, 10, 11]. Another way to state this is that in human-agent systems engaged in joint activity, the benefits of higher levels of autonomy cannot be realized without addressing interdependence through coordination.



A) Experiment design B) Expected Effects C) Expected Performance

Figure 2. A) Illustration of our experimental design approach. B) Expected effects of increasing autonomy on the burden of managing the agent and the opacity of the agent to other task participants. C) Expected performance under treatment conditions of increasing autonomy, due to the competing factors of agent management burden and agent opacity.

Objective and Expected Results. Our goal was to demonstrate this claim empirically. We attempted to rule out over-trust in automation as a failure factor by ensuring that the agent players never made mistakes and that they exhibited reasonably intelligent behavior. We also attempted to ensure that the interaction between the human and the agent could be at a relatively high level of abstraction—i.e., that the agent's capabilities for autonomy were not under-utilized. We did not want an agent capable of completing the mission autonomously managed at a low level akin to teleoperation. To this end, we provided an interface appropriate to agents' capabilities. These elements of our experimental design are illustrated in Figure 2 (A).

Figure 2 (B) illustrates the general trends we expected to find in our results. We anticipated that the management burden the agent player imposed on the human player would decrease as agent autonomy increased. Such a finding would be no surprise, since reduction in human workload is both the common expectation and the major motivation for automation. However, we also anticipated that, without support for managing interdependence issues, the opacity of the work system to task participants would grow with increasing autonomy. Due to these competing factors of burden and opacity, we expected an inflection point in team performance, where the benefits of increasing autonomy eventually would be completely offset by the negative side effects of opacity. In other words, we predicted that the highest level of autonomy would not demonstrate the highest level of team performance, consistent with the general shape of the notional bar graph shown in Figure 2 (C)

General Description of the Experiment. For each run of this experiment [5], we had a single human participate in a joint activity (collecting colored blocks in a specified sequence) with a single agent player. Both the human and the agent controlled a robot avatar. The agent teammate was directed by the human (i.e., participant or user) at levels of autonomy that varied in each

HRI 2015 Human-Robot Teaming Workshop, March 2, 2015, Portland, OR, USA.

experimental condition. The agent was designed to perform reliably and with reasonably intelligent behavior. This means that the self-directedness is always sufficient for the self-sufficiency and thus the system cannot be over-trusted. This experiment also limited the command interface for each level to the highest possible command set, thus preventing under-utilization. As such, we were looking only at the burdensomeness and opacity of the system.

Algorithm for Agent Behavior. The algorithm chosen as the basis for the agent behavior reflects the most common approach we observed for human players of the game. This algorithm was chosen because we felt it would be easily understandable and predictable for most human players. The algorithmic solution is shown on the left side of Figure 3. The main goal (a color sequence) is composed of several subgoals (individual colors). To achieve any given subgoal, one simply finds the block of the appropriate color and delivers it. Note that these tasks need not be performed in sequence or by the same player. For example, a player could first find all the blocks and then deliver them. Alternatively, one player could find a block and another could deliver it. The overall task can be thought of as being composed of several *find* tasks and several *deliver* tasks, which are themselves composed of some decision and action primitives. The action primitives include going to a room, entering the room, going to a block, picking up a block, and putting down a block. The two main decisions are: 1) whether to look for a block or to deliver a block, and 2) which room to go to in order to look for a block. The agent player is designed to perform its task "perfectly," meaning it will perform any assigned task efficiently and will make rational decisions based on a complete and accurate recollection of where it has been and what it has seen in the past. It will also report when a task is completed. To be consistent, it *only* reports the completion status when an assigned task is completed, and does not provide any additional information.

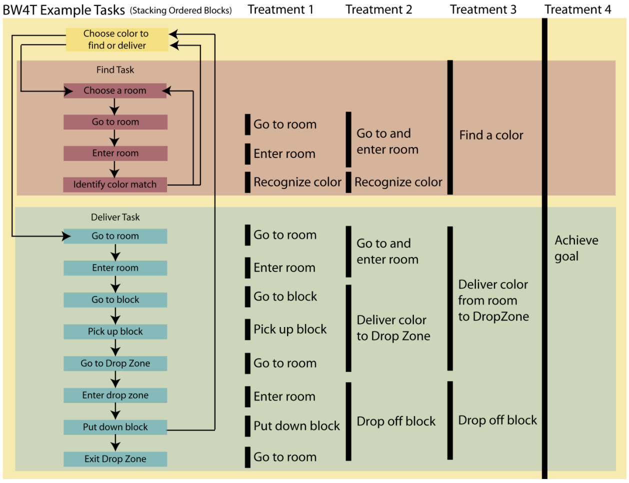


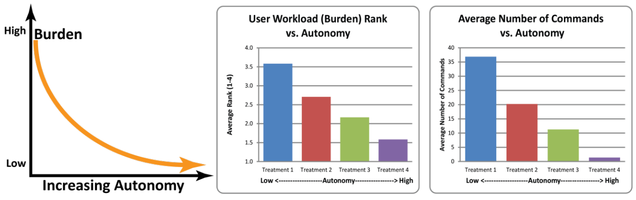
Figure 3. Autonomy Treatments for Experiment 1

Autonomy Treatments. In Treatment 1, the human made all decisions and initiated all actions for the agent player. In essence, the human was manually controlling two robot avatars. This corresponds to Sheridan's lowest level of autonomy. For Treatment 2 we automated most actions of the agent. All decisions remained with the human. We expected this automation would be preferred because it was reducing burden without adding opacity. Treatment 3 had all of the autonomous actions from Treatment 2 and also added an autonomous decision (i.e., which room to search). This increased opacity in two ways. First, the human is no longer aware of all of the decisions because one of them has been automated. Second, the robot has to make the decision without the same information the human had available when making the decision for the agent. Treatment 4 added automation of the remaining decision, making the task "fully autonomous." This corresponds to Sheridan's highest level of autonomy.

Experimental Design. 24 participants (17 male and 7 female) were selected from a student population at TU Delft, with an age range of 19-39. We employed a complete randomized block design based on the autonomy treatment, with each participant performing each treatment once. The data are cross-classified by k = 4 autonomy treatments and b = 24 blocks, consisting of the individual participants. All participants received a demographic survey. They were trained on the game until they demonstrated proficiency by completing a simplified version of the task. Next they performed a series of trials, one for each treatment. The participant filled out a brief survey at the end of the experiment, evaluating team burden, opacity, performance, and preference in each treatment.

Results. Our results include quantitative numeric data as well as subjective ranking data. For the former, we use standard approaches for normal data. For the ranked data, we used the nonparametric Friedman test.

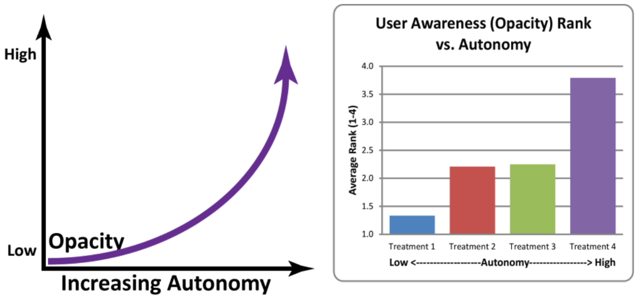
*Assessing Burden***.** Our hypothesis predicted a decrease in agent management burden as autonomy increased from treatments 1 to 4. This is depicted in Figure 4 (A). We asked the participants to rank how demanding it was to work with the agent in each condition, on a scale of 1 (least demanding) to 4 (most demanding). The results, shown in Figure 4 (B), indicate a very clear decrease in burden as autonomy increased. As a second, independent measure of burden, we also counted the number of commands the human player had to give to the agent teammate in each condition. Figure 4 (C) shows the results, which correlate with the subjective assessment.



A) Expected change in burden B) Qualitative burden measure C) Quantitative burden measure

Figure 4. (A) Expected change in burden as autonomy increases (B) Subject ranking of agent management workload (burden) as autonomy increases across experimental treatments. (C) Average number of commands (Burden) as autonomy increases.

Assessing Opacity. Our hypothesis predicted an increased subject perception of opacity with increasing autonomy across the experimental conditions. This is depicted in Figure 5 (A). We expected this to be reflected in reports of subjects having more difficulty in understanding what was happening and in anticipating the agent's behavior as autonomy increased. An exit survey was used where subject were asked to rank their ongoing sense of awareness of current and future agent actions in the different conditions on a scale of 1 (most aware) to 4 (least aware). The results in Figure 5 (B) show opacity increasing with increasing autonomy as predicted. This confirms our prediction about opacity in this experimental setting, and validates the general expectation.



A) Expected change in opacity B) Qualitative opacity measure

Figure 5. (A) Expected change in opacity as autonomy increases (B) Average subjective ranking of awareness (opacity) as autonomy increases across experimental treatments.

Quantitative Performance Assessment. We performed three different quantitative performance assessments: time to complete task, idle time, and error rate.

The simplest performance metric is time to completion—i.e., delivering all the required blocks in the requested order. Figure 6 shows the results. At first glance, the results appear promising. We can clearly see the inflection point where performance begins to degrade rather than improve under conditions of increasing autonomy, consistent with the prediction of Figure 2 (C). The differences, however, were not statistically significant (p = 0.20). We believe that this is best explained by the fact that the task itself has a large amount of variance from run to run, and the penalty incurred by errors is less than the variance between runs. We note, however, that in 83% of the participants, the highest-autonomy condition (Treatment 4) was not the highest-performing condition by the time-to-completion criterion.

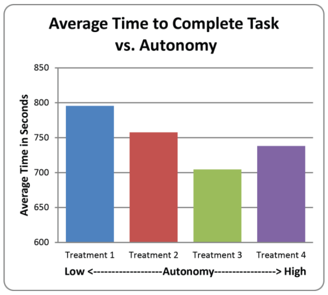


Figure 6. Time-to-completion as autonomy increases across treatments.

Another important performance measure is idle time (or wait time [6]). In the BW4T task, the agent player will be in near constant motion once a task has been assigned to it by its human teammate. Any idle time is indicative of inefficient use of the agent player (e.g., while it awaits the next command). Figure 7 (A) shows the results of average idle time for the agent player. There is a clear and significant decrease in idle time from treatment 1 to 4. On the surface, this could be taken as indicating more effective use of the agent player by the human, and thus suggesting improved performance. However, this is not borne out by the time-to-completion results. Additionally, we note that the amount of work done is fairly consistent across treatments. For example, the number of rooms entered and the number of boxes delivered does not change much across treatments. This also makes sense when one looks at the human player's idle time, shown in Figure 7 (B). There is a slight decrease in idle time as the burden is reduced, but not much, and certainly not on the order of the change seen in the agent player. This indicates that the interaction efficiency [6] is not that significant. This could be due to an effective interface, but it also can be due to the ability to multi-task and complete interactions concurrent with motion. The interesting takeaway lesson from this result is that "keeping your agent busy" does not equate to improved performance.

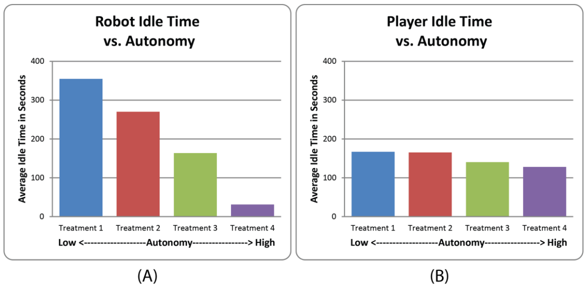


Figure 7. (A) Average agent player idle time across treatment conditions. (B) Average human player idle time.

For some kinds of tasks, error rate can be a good way to compare performance. We measured this in three ways. Our first was the amount of time that both players spent holding the same color block (Figure 8 (A)). Since, for this experiment, the goals were composed of unique colors (no repeats), this represented a measure of some fraction of overall redundant activity or inefficiency in task performance. This type of error, for the most part, only occurred in treatment 4 and is a side effect of the high opacity of the highest-autonomy condition. These results are no surprise, since this is the only treatment in which the agent player can make its own decision about which block to pick up. However, this does emphasize that functional differences matter when automating tasks [7].

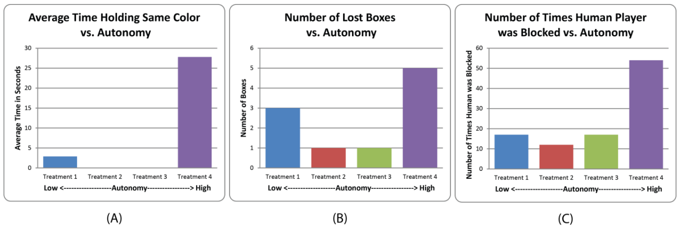


Figure 8. (A) Average time holding the same color (inefficiency) (B) Number of lost boxes (C) Number of times a human player was blocked by their agent partner while trying to enter a room

A second measure of error is the number boxes lost—i.e., dropped in the hallway or placed in the drop zone erroneously. Since BW4T is very simple, there were not many mistakes made by the human players, but of the ten lost boxes, 50% of them occurred in treatment 4 and 30% occurred in treatment 1, as shown in Figure 8 (B). The boxes lost in treatment 1 were most likely due to the high workload imposed by the minimal amount of autonomy. However, treatment 4 does not have the obvious workload challenges of treatment 1. In fact, it was clearly ranked as the least burdensome, so why would it have the highest occurrences of errors? We believe the high error rate is a side effect of the high opacity of the highest-autonomy condition.

Our third measure of error was the number of times a player was blocked while entering a room. This measure is indirect because it is possible that the most efficient act would be to wait outside a blocked door, but in general it indicates poor coordination. As shown in Figure 8 (C), the human player was blocked in treatment 4 much more often, indicating significantly more coordination breakdowns than any other treatment.

We asked the subjects to identify which team they felt performed best. Treatment 3 was the clear winner, with 63% of the participants selecting it as the best performing treatment (Figure 9 (A)). Only 17% of the subjects choose treatment 4 as the best performing.

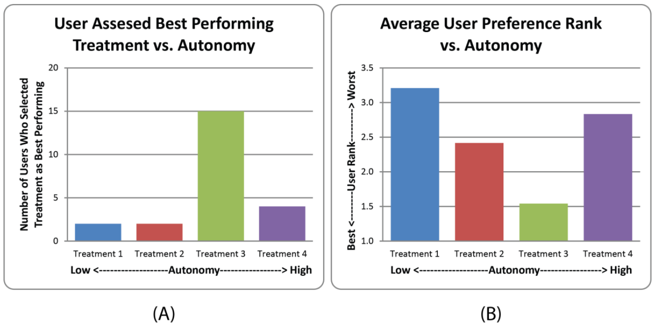


Figure 9. (A) User Assessment of Performance vs. Autonomy (B) User Preference vs. Autonomy

Human acceptance is an important component of overall system performance in tasks like ours. We asked the participants to rank the agents in each experimental condition with respect to their preference as to which one they would like to play with again, on a scale of 1 (most like to play with again) to 4 (least like to play with again).

Figure 9 (B) shows the results. Treatment 3 was preferred with statistical significance. This result also demonstrates the inflection point anticipated by the increasing opacity in the system from Figure 9 (C). We suspect this is because in treatment 3 the human holds the overall plan, most of the context, and exercises the greatest degree of creativity. In this context, transparency and control (directability) may be more important than autonomy (independent operation), especially in light of the particulars of the autonomous task.

We asked participants about the reasons for their rankings, and the responses were enlightening. Reasons for preferring Treatment 3 included:

- Shared information
- Able to anticipate
- Predictable
- Low burden
- Cleverest
- Automatic, but still have control

The first three reasons correlate with our predictions about opacity. The comment about low burden is interesting, because treatment 4 was objectively less burdensome. This comment suggests that there may be other types of burden besides the manual workload of tasking the agent. The comment about treatment 3 being cleverest is also interesting, because treatment 4 is objectively the most capable (clever) based on what the agent can do on its own. Perhaps this suggests that sometimes being more independent may not necessarily lead to being viewed as more clever. The final reason is also important because it relates to the broader issue. We focused on opacity in order to keep the experiment simple, but predictability, directability and other challenges in making automation a team player [8] are no doubt also affected by increased autonomy.

Summary. The results of our initial limited evaluation support our claim that increasing autonomy does not always improve performance of the human-machine system. In our example, increasing autonomy improved performance up to a point, but then there was an inflection point where performance decreased, depicted in Figure 10. We saw performance inflections in time, in error rates, and in user rankings. We propose that systems that fail to address interdependence adequate with have similar inflection points in performance. In the BW4T domain, this was principally due to opacity in the system, derived from increasing autonomy without accounting for the interdependence of the actions and decisions of the players and the coordination challenges this creates. Additionally, we showed how keeping an agent busy does not equate to improved performance, how human error rates are not only due to workload but can also be affected by opacity, and how user preference is not necessarily driven by reduced burden when other factors such as transparency, predictability and directability are relevant to the task. A key point to take away is that the ability to work *with* others becomes increasingly important as interdependence in the joint activity grows. It is possible that in complex and uncertain domains, this may be more valuable than the ability to work independently.

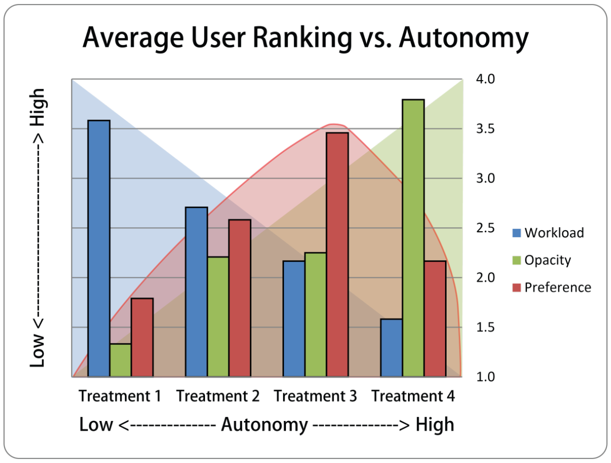


Figure 10. Performance inflection point demonstrated by results

It is obvious why opacity has such an effect on the system in the BW4T domain. The greater the autonomy of players, the greater the opacity, and hence the more room for coordination breakdowns. The independent activity in treatment 4 inhibited the team's ability to engage in what most people would consider "natural" coordination, resulting in a breakdown of common ground [8] and reduction in each player's individual situation awareness. This then caused suboptimal decisions and errors. While obvious in this simple, abstract domain, the problem remains prevalent in many systems today. Understanding the relationship of autonomy to interdependence is one step toward addressing the challenges facing future systems. We believe that consideration for interdependence while designing the autonomous capabilities of an agent can mitigate the effects demonstrated and will enable future systems to achieve greater potential.

3.2 Experiment 2: Soft Interdependence as a Performance Factor

In another experiment, we ran twelve subjects in various team sizes (2, 3, 4, 5, 6, and 8). The subjects were allowed to talk openly to one another. As the activity became more interdependent (more complex ordering of the goal stack), we noted an increase in the number of coordination attempts, as would be expected. We also noted some interesting aspects of the communication. Although only two basic tasks are involved, we observed a wide variety of communications. Of particular interest were the large number of communications that were about *soft interdependencies* and monitoring issues that were related to them.

An example of a soft interdependency is the exchange of world state information. Since players could only see the status of their current room, they would exchange information about the location of specific colors. Although the task could clearly be completed without this communication, the importance of this soft interdependence is demonstrated by the frequency of its use. An example of monitoring in support of interdependence issues was when players provided or requested an update as a colored block was picked up. The frequency of both progress updates and world state updates are examples of the importance of addressing *supportive interdependence* in human-agent systems for joint activity. These types of exchanges typically accounted for approximately 60% of the overall communication and increased with the degree of interdependence required for a given problem. A final observation was that not only the amount of communication changed with the degree of interdependence in the task, but the pattern of communication varied as well. For example, during tasks with low interdependence, world state and task assignment were the dominant communications. As interdependence in the task (complexity in the ordering of the goal stack) increased, they both diminished in importance and progress updates became dominant.

4. RT4T: A RECONFIGURABLE TESTBED FOR TEAMS

In order to test more complex scenarios, we wanted to implement a new testbed that could tackle more challenging joint activity with increased ecological fidelity to envisioned real world applications. The domain initial domain we are focusing on is building clearing. While there are known tactics for room clearing by teams of soldiers, there is a potential for utilizing human-robot teams. Such teams would consist of human operators and semi-autonomous robots that would collaboratively clear a building while maximizing the capabilities of each of the team members.

Our scenario for this project is building clearing. In this scenario the team must ensure that the building is free of "*bad guys*" while escorting the "*good guys"* to safety. The team has to systematically cover the entire building as quickly as possible. Inside the rooms are some people (a few *good guys* and a few *bad guys*). The team must check every room thoroughly, remove all the people, and group the *good* and *bad* guys correctly.

We can now import 3D models of arbitrary complexity into the simulation (e.g., as shown in Figure 11), but for our current experiments we have kept the environment simple to avoid confounding effects of variations in player's spatial abilities. The simplified world is shown in Figure 12. We are also using a first person view (Figure 13) to more realistically represent the information that would be available to a player visually.

Our testbed is Java based, which means it is cross-platform compatible. The graphics engine used is the Java Monkey Engine (JME), which is a free open source graphics engine that is actively developed. It is a multi-player game environment that will allow for any number of human or robotic players in any combination. We have complete control of the environment, which will enable us to flexibly extend beyond our initial experimental scenarios to address multiple types of activities. This will give us a richer test and evaluation platform to investigate our teamwork metrics.

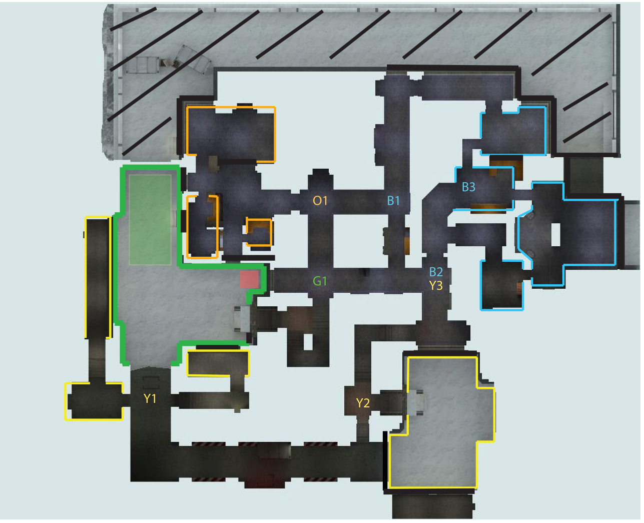


Figure 11. Complex building model

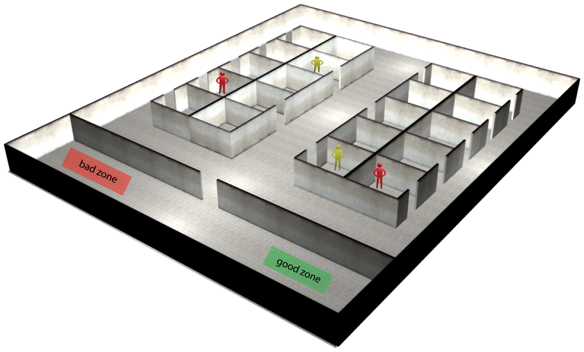
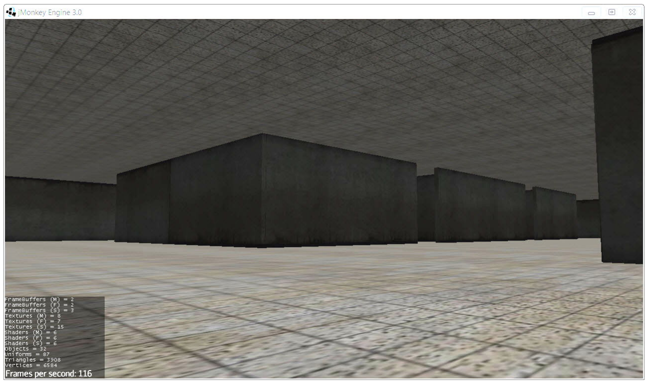


Figure 12. Simplified model for initial building clearing experiments



```
FrameBuffers (M) = 2
FrameBuffers (F) = 2
FrameBuffers (S) = 3
Textures (M) = 8
Textures (F) = 7
Textures (S) = 15
Shaders (M) = 6
Shaders (F) = 6
Shaders (S) = 6
Objects = 32
Uniforms = 87
Triangles = 3908
Vertices = 6584
Frames per second: 116
```

Figure 13. First person view

The RT4T testbed has a server that allows configuration of the scenario. The server interface, shown in Figure 14, allows configuration of the scenario and control over the game. It also allows for a complete view of the entire simulation independent of any of the players. The server logs all team data and the current version has a simple visualizer for experimental results, shown in Figure 15.

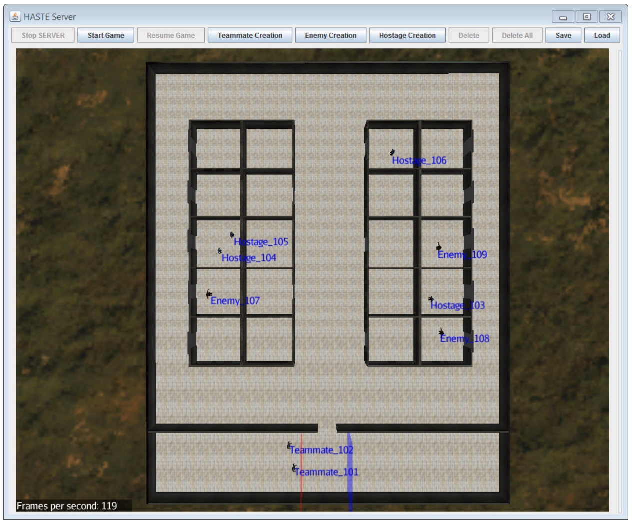
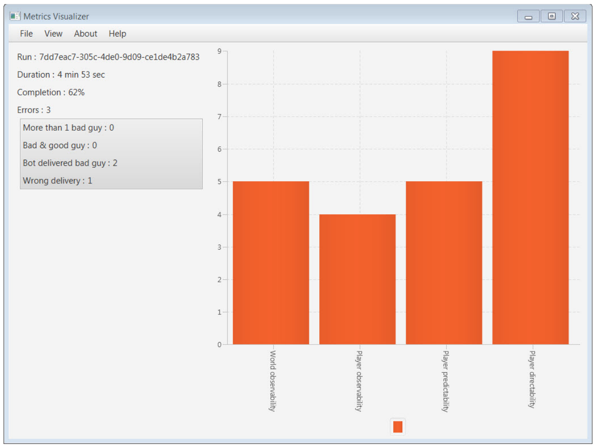


Figure 14. The RT4T server interface allows the configuration a scenario and provides an overview of the entire simulation in progress.



Run : 7dd7eac7-305c-4de0-9d09-ce1de4b2a783

Duration : 4 min 53 sec

Completion : 62%

Errors : 3

More than 1 bad guy : 0

Bad & good guy : 0

Bot delivered bad guy : 2

Wrong delivery : 1

Figure 15. Example RT4T visualization of experiment results.

5. CATEGORIZATION OF MEASURES

In developing a new suite of teamwork measures for our current experiments with RT4T, we have devised an initial high-level categorization that may prove useful in helping people understand and differentiate various types of experimental measurables (Figure 16). This categorization is based on temporal considerations, specifically:

1. **Design-time** measurables represent the *potential for effective work* afforded by a given HART system in a specific context. In principle, they can be measured before any activity begins. (e.g., *Flexibility*: How many different options do you have for sensing an obstacle? *Capacity*: How many objects can you sense at once? *Competence*: How accurately can you sense an object?).
2. **Run-time** performance measurables that represent a team's success in *leveraging design-time capabilities at run-time* in a contextually appropriate way. Effective teams rely on an understanding of the talents and styles of teammates, recognize opportunities and needs to make adjustments to teamwork and taskwork, and are efficient and effective in applying these changes.
3. **Post-activity** measurables include, among other things, tracking successes, failures, workload, and time to complete tasks.

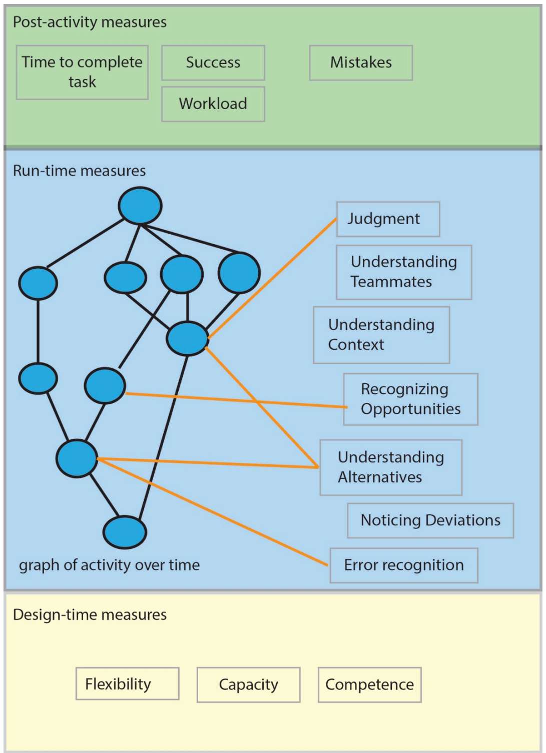


Figure 16. Categorization of Meastures

Our categorization relates to a key goal related to each phase of measurement: *adaptability potential, resilience,* and *robustness* (see Table 1).

Table 1: Teamwork Goals Related to Each Phase of Measurement

Goal	Conceptual Definition
Adaptability Potential	The latent potential for a team to *perform its work in different ways*, enabled by *designing for flexible alternatives* that harness available capacity and competence (measured at design-time)
Resilience	The ability of a team to *recognize problems and opportunities* as the work is being performed, to *successfully analyze appropriate alternatives for a given context*, and then to *change resources, roles, and goals* efficiently and effectively (measured at run-time)
Robustness	The ability for a team to *maintain effectiveness across a range of tasks, situations, and conditions* (measured at post-activity time)

Why is this important? First, a given measure can only give you certain types of information. For example, post-activity analysis usually provides a direct measure of performance (e.g., efficiency or effectiveness), while other measures are used in an attempt to help us learn about possible factors that may contribute to performance. In other words, post-activity measures often cannot tell you the cause unless linked to an indirect measure from one of the other two categories. As an example, consider two robots that must navigate a course. They both complete the course in five minutes, so can we conclude that their performance was equivalent? Of course not.

Fortunately, the other measures of design-time and run-time provide additional insight. Consider that the first robot may only have a top speed (design-time measure) of half that of the second robot. This would indicate that the second robot should have done better. Consider that the second robot may have made several bad navigation choices (run-time measure) that caused this. Fundamentally, the reason behind all of these proposed measures is to ferret out the potential causes that are driving the performance measures. Most of these are run-time measures (e.g. what are you going to do next?). Some also touch on design-time measures if they are focused on understanding different player capabilities (e.g., who could assist you?) instead of the run-time version that also includes current context (e.g., who could assist you right now?).

Another nice characteristic of this way of categorizing is that it provides us a way to design experiments. For example, we may want to focus on flexibility, so we ensure the experiment is designed to have a low competence requirement. We could also control for flexibility and measure how well the system can recognize opportunities and leverage them.

The next phase of our work will be focused on refining and formalizing the measurement framework through further experimentation. RT4T will provide invaluable help in understanding which teamwork measures are most important and how these measures relate to one another.

6. ACKNOWLEDGMENTS

The work on BW4T was performed as part of Matthew Johnson's doctoral dissertation at the Technical University of Delft. Our thanks to Danielle Ward of Aptima, Inc for her contributions to the scenarios and ideas for RT4T in the context of our joint work on the Human and Autonomous Systems Team Evaluation (HASTE) project, sponsored by the Office of the Secretary of Defense (OSD14.1-AU3). This work was also sponsored in part by funding from the NASA RNM project, the National Robotics Initiative, and the DARPA Robotics Challenge.

7. REFERENCES

[1] Johnson, M., Jonker, C., Riemsdijk, B. van, Feltovich, P. J., & Bradshaw, J. M. (2009). Joint Activity Testbed: Blocks World for Teams (BW4T). *Engineering Societies in the Agents World X*. Berlin: Springer Verlag, pp. 254-256.

[2] Durfee, E. H., Montgomery, T. A., MICE: A Flexible Testbed for Intelligent Coordination Experiments; In Proceedings of the 1989 Distributed AI Workshop (1989).

[3] Adobbati, R. Marshall A. N., Scholer A., Tejada, S.; Gamebots: A 3D Virtual World Test-Bed For Multi-Agent Research; In Proceedings of the Second International Workshop on Infrastructure for Agents, MAS, and Scalable MAS (2001).

[4] Terry Winograd was the first one to formulate this problem. For a brief overview, e.g., http://en.wikipedia.org/wiki/SHRDLU.

[5] Johnson, M., Bradshaw, J., Feltovich, P., Jonker, C., van Riemsdijk, B., & Sierhuis, M. (2011). The Fundamental Principle of Coactive Design: Interdependence Must Shape Autonomy. In M. De Vos, N. Fornara, J. Pitt, & G. Vouros (Eds.), *Coordination, Organizations, Institutions, and Norms in Agent Systems VI* (Vol. 6541, pp. 172-191). Springer Berlin / Heidelberg.

[6] Crandall, J. W., and M. L. Cummings. 2007a. Identifying predictive metrics for supervisory control of multiple robots. *IEEE Transactions on Robotics – Special Issue on Human-Robot Interaction* 23 (5):942-951.

[7] Johnson, Matthew, Jeffrey M. Bradshaw, Paul J. Feltovich, Robert R. Hoffman, Catholijn Jonker, Birna van Riemsdijk, and Maarten Sierhuis. "Beyond cooperative robotics: The central role of interdependence in coactive design." *IEEE Intelligent Systems 26*, no. 3 (May/June 2011): 81-88.

[8] Klein, Gary, David D. Woods, J. M. Bradshaw, Robert Hoffman, and Paul Feltovich. "Ten challenges for making automation a "team player" in joint human-agent activity." *IEEE Intelligent Systems* 19, no. 6 (November-December 2004): 91-95.

[9] Johnson, M., J.M. Bradshaw, P. J. Feltovich, C. M. Jonker, M. B. van Riemsdijk, and M. Sierhuis. Coactive design: Designing support for interdependence in joint activity. *Journal of Human-Robot Interaction*, Vol. 3, No. 1, 2014, 43-69.

[10] Bradshaw, J.M, Robert R. Hoffman, Matthew Johnson, and David D. Woods. The Seven Deadly Myths of "Autonomous Systems." *IEEE Intelligent Systems*, May/June 2013 (vol. 28 iss. 3), 54-61.

[11] Johnson, M., Bradshaw, J.M., Hoffman, R. R., Feltovich, P. J., and Woods, D. D. Seven Cardinal Virtues for Human-Machine Teamwork: Examples from the DARPA Robotic Challenge. *IEEE Intelligent Systems*, November/December 2014 (vol. 29 iss. 6), 74-80.