The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Using Simulation to Predict Multi-robot Performance on Coverage Tasks

Shameka Dawson, Briana Lowe Wellman, and Monica Anderson

***Abstract*— Simulations are typically used to model a problem and find a solution before real world testing. They speed up the validation process and allow researchers to modify their code accordingly. However, a problem occurs when simulation results are not consistent with real world results. Researchers have found inconsistencies due to odometry error and team size. However, no research has studied the effects specific to robot teams that affect the realism of multi-robot experiments.**

This paper shows how simulation results vary from experimental results when conducting multi-robot experiments. Simulation and real experiments are performed using different environments and cooperation paradigms. Results show that specific environmental features and cooperation paradigms significantly affect the usefulness of simulated results when predicting performance of real robot teams.

I. Introduction

Validation of robot control algorithms is achieved by performing simulations, real experiments, or both. Robot simulators include 2-dimensional [1], [2], 3-dimensional and physics-based [3], [4], [5] modeling, which are typically used for two specific reasons: 1) testing and debugging, and 2) prediction.

Prediction attempts to model the real world by observing patterns in simulations and relating them to real experiments. Simulations provide predictions on how a robot will interact with its environment. Real robot behavior and performance can then be determined based upon observations of the simulations. However, these predictions do not always hold true in real experiments. Therefore, it is not "wise to rely on simulation alone" [6].

Researchers acknowledge that error models are approximate, but there are some circumstances where simulations can predict relative performance [7]. Prediction of multi-robot performance in simulation is important because of the cost and time involved in acquiring, maintaining, and running multiple robots. To this end, it is important to understand when simulations accurately predict performance and when they do not.

There are a number of reasons why simulations differ from real experiments (i.e. sensor noise, odometry error, etc.). In [8], it is asserted that the way in which a robot interacts with its environment accounts for a huge difference between simulated and real robots. Robots in simulation generally perform perfectly because their sensors and actuators return perfect results. Whereas, robots in reality are subject to imperfect sensor and actuator readings and odometry error. Therefore, experiments that perform successfully in simulation may not always perform successfully in reality. In these cases, simulation results will not accurately portray how a robot may perform in a real environment. In order to mitigate these differences, a certain amount of sensor noise can be added to the simulation.

In addition, having multiple robots in an experiment provides another significant reason as to why simulations differ from real experiments. Multi-robot experiments include additional factors such as interference and latency that affect performance. However, researchers often neglect this fact.

In this paper, we explore the discrepancies between multi-robot simulations and experiments that do not exist in single robot experiments. We also propose a preliminary model of several factors that affect multi-robot performance that should be considered in simulation.

This paper is organized as follows. Section II presents related work. Our approach and experiments are discussed in Section III and IV, respectively. Results and analysis are discussed in Section V. Finally, the conclusion is presented in Section VI.

II. Related Work

Prior work indicates that researchers are concerned with the correlation between simulation and physical experiments. Jakobi et al. [9] conducted a study comparing levels of noise in simulation to noise in a real environment. They found that different levels of noise used in simulation resulted in different behaviors in the real environment. They concluded that the noise level must be correct in order to get a realistic simulation. However, they determined that a thorough set of empirical data must be gathered to produce accurate parameters to be used in simulation. But it is not always possible to correlate robot-environment interactions under complex circumstances.

In some instances, sensors are dependent upon each other. In [10], Meeden found a correlation between certain sensors. A hybrid model was developed that combined independent and dependent sensor noise to account for different amounts of correlation. Their results imply that the hybrid noise model transfers better from simulation to the real world than an independent noise model. However, they note that it will get more difficult to construct accurate simulations as robots become more complex.

Melhuish et al. [11] performed a patch sorting study in both simulation and a real multi-robot system using minimalist robots. Experiments in simulation showed better

This work was supported in part by the following NSF grants: IIS-0846976 and CCF-0829827.

S Dawson, B Wellman, and M Anderson are with the Department of Computer Science, University of Alabama, Tuscaloosa, Alabama 35487 USA (email: {dawso003, lowe002}@crimson.ua.edu; anderson@cs.ua.edu).

performance than those run on the real robot. They say that changing light levels caused difficulties with using infrared sensors which may have resulted in the poorer performance. However, they claim that their study shows that a successful transfer of their simulated results to the real world is possible. Nevertheless, their experiments were run on robots with minimal sensing abilities, not complex multi-robot systems.

A method for transferring a control system for a bipedal robot from simulation to the physical world is presented in [12]. Although their method showed a similar mapping between simulation and the physical robot, there were some inconsistencies where the physical robot performed worse. They reported that motor wear and battery power were the causes of the differences. However, these are issues that are not considered in a simulation but are very relevant to real robot performance.

In [13], [14], [15], it is shown that as the number of robots increase, performance decreases when coordinating multiple robots in a search task. This is because of the limited communication bandwidth and the computational requirements when dealing with multiple robots. It was determined that interference and message exchange affect multi-robot experiments.

Although researchers believe that simulators have significant benefits [16], [17], simulations still raise several concerns. It is stated that a simulation's accuracy relies on the robot's hardware, the algorithm, and operating environment. It was suggested that making certain assumptions about these elements reduce the authenticity of simulations.

III. Approach

The traditional approach structures robot control systems into functional modules such as perception, planning, and modeling. However, many researchers use a behavior-based decomposition that layers task achieving behaviors such as obstacle avoidance and wander [18]. A behavior-based robot is designed to operate in dynamic environments because its reactions are determined by what is sensed without necessarily modeling the environment structure that provides the sensor readings. As it senses the environment, it computes what it senses, and then acts on what is computed (see Figure 1). This structure often includes higher-level behaviors that are not purely reactive that model and hold state such as mapping and path planning.

Simulations are an important component of software validation. Robot controllers are tested within a simulated environment to verify properly coded semantics. In addition, simulation is often used to predict controller performance under a set of constraints. Specifically, the environment is varied along with the robot configuration to quantify performance under a more generalized set of parameters.

Error modeling is a significant component of obtaining reasonable performance prediction within single robot simulations. Sources of error such as sensor noise and wheel slippage have an impact on performance when conducting experiments. Although, researchers acknowledge these sources, they are not always modeled in simulation. When these factors are modeled, a simplified model of error relative to sensor magnitude distributed uniformly or normally is often thought as sufficient [19].

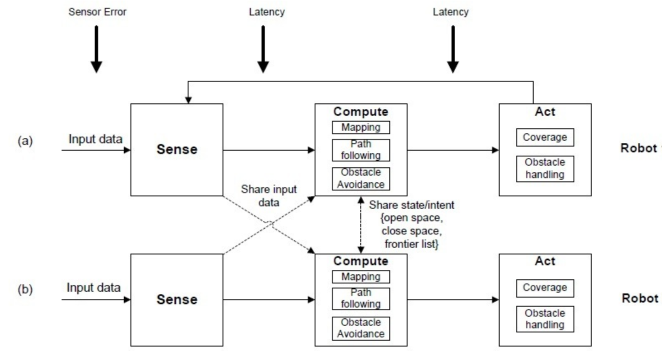

Fig. 1. A behavior-based model consists of a sense, compute, and act state. A behavior-based robot chooses its actions depending on what is sensed in the environment. (a) Single robot and (b) Multiple robots.

A. Single Robot: Predicting Performance

When predicting performance in simulation, we focus on modeling factors that significantly affect the performance of controllers. There is inherent uncertainty at points where the robot interacts with the environment: sensing and actuation. This uncertainty causes variance in real performance. In addition, performance is biased through interaction with non-modeled factors changing real controller performance. This phenomenon is discernible when simulated results are better or different than real robot results.

Figure 1a illustrates where uncertainty is introduced in the behavior-based model for a single robot. A robot receives (or senses) information about its environment such as laser and distance data. Therefore, sensor error affects how the robot senses the environment. The robot then calculates its intent based on the calculated current state of the system. Based upon these calculations, the robot executes the desired behavior (coverage or obstacle avoidance). Latency is injected because of the delay between sensing, computation time, and execution time. Along with environmental configuration [20], these factors (see Figure 2) influence real performance enough to be fairly consistently modeled in simulation.

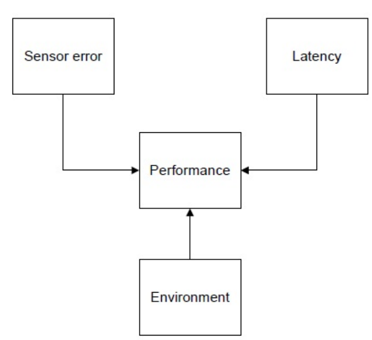

Fig. 2. A model of the issues that affect the performance of a single robot system.

B. Multiple Robots: Predicting Performance

Is predicting performance in multi-robot systems as simple as correctly modeling individual factors? Certainly sensor error, latency and environment all affect team performance since team performance is an aggregation of individual performance in some ways. However, we conjecture that there are important factors specific to multi-robot systems that affect performance. Because as [21] states, predicting how a multi-robot system behaves is more complex than predicting a single robot system.

For instance, researchers have studied team size as a factor in performance. Algorithm evaluation often includes varying team sizes to allow for generalization of results [13]. Rybski et al. [14] shows that limited bandwidth limits the effective team size. Propagation of localization error between robots allows robots to share incorrect state information resulting in reduced coverage percentages. This factor is usually modeled through individual sensor error models.

C. Expanding the Model

It is instructive to consider how the single robot behavior model changes when dealing with multi-robot teams. Figure 1b shows possibilities for how multiple robots share data and cooperate. Robots can share sensor data, calculated/modeled state and/or intent such as intended search targets. Not only are actions dependent on what a single robot senses, but also shared data between robots. These additional links to and between the behaviors can create additional processing in a resource constrained environment. Increasing the processing load in a computationally constrained environment may increase latency. However, it is not understood whether this increase in latency affects end performance more than existing single robot latency.

Other performance affecting factors may be identified by understanding the components in the ACT state. In behavior-based systems, computation results in control of the motors being given to one of the behaviors. In a coverage algorithm, usually the behaviors that want motor control can be classified as either obstacle handling or coverage-based. For simplicity, algorithms that use a combination are not initially considered. Due to the subsumptive nature of such approaches, the performance time can be described as $t_A + (1 - t_A)$ where t_A is the time avoiding and $1 - t_A$ is the remaining time for coverage activities. If we assume that coverage primarily occurs during the execution of coverage behaviors and that coverage performance is linear to time spent covering, it follows that any time spent avoiding reduces coverage performance over time.

Some factors that influence time spent avoiding do not change by the addition of robots such as obstacle avoidance (environment-based). However, interference resulting in time spent avoiding occurs when multiple robots try to occupy the same space. We propose that both cooperation paradigm and environmental configuration influence robot interference enough to impact t_A and the resulting coverage performance.

Cooperation paradigms allow robots to share information and divide work in such a way to choose different paths or search targets, thus impacting interference. Environmental configuration determines the number, placement and size of obstacles resulting in possibly increased robot obstacle avoidance. However, obstacles may also affect robot interference by causing robots to spread out more, lessening the effect of interference and degradation of search performance.

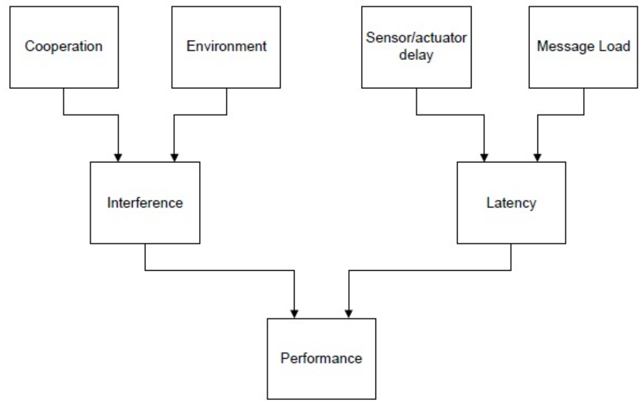

Fig. 3. This preliminary model shows specific issues that affect multi-robot performance.

These observations allow us to propose an updated model (see Figure 3) for predicting performance in multi-robot coverage tasks. To investigate this model, experiments in simulation and with real robots were conducted focusing on specific parts of the model.

In this paper, we will examine the following hypotheses:

* Cooperation paradigms and environmental configuration influence interference.
* Interference affects how well simulation approximates performance of a multi-robot experiment.
* Latency from message loads affects multi-robot performance.

For the search task, a frontier-based algorithm [22] is used where robots recursively explore an unknown area while building a cellular representation [23]. Frontiers are the areas between unknown and open space. As robots detect frontiers, they store them in a list of areas to explore. Frontiers are visited to gain more knowledge about the environment, thereby recursively exploring.

For experiments with no communication, there is no direct communication between robots. Each robot relies on their own perception of the environment. With direct communication, robots explicitly broadcast messages to all other robots through point-to-point communication. Robots send messages to share whether an area is open or closed. A robot updates its map and frontier search list with the information in the messages.

To examine the influence of cooperation and environment on interference, experiments with robot teams with no communication and direct communication were conducted in several different environments. Performance for this research is measured by the time it takes a team to cover an area. The number of incidents that the robots avoided each other were calculated, then examined against performance.

To compare the effects of message loads, experiments were performed using two direct communication scenarios. In the first scenario, robots were allowed to transmit messages about a new area only once. In the second scenario, robots were allowed to broadcast more by transmitting information about an area every time step it was observed. The data collected includes the percentage of area covered, average times at 50% and 90% coverage, and average message delay for messages received.

IV. Experimental Setup

Our experiments used both simulated and physical robots for comparison. The control program, written in the C, was essentially the same for both the simulated and real experiments. The simulation environment used was Webots [4], a 3-D physics-based mobile robot simulator. While other 3-D simulators [5] could have been utilized, Webots was chosen as the tool for comparison because it provides a fast, easy to use, high fidelity simulation. It also includes typical features that are found in popular simulators.

The robots in simulation used global positioning sensor (GPS) for localization as well as a laser range finder. The simulations were performed on a Dual Core 2.33GHz Linux machine with 2GB of RAM. A wheel encoder noise (based on a Gaussian distribution) was added to the trials run in simulation to compensate for error in the real world.

K-team Koala robots were used for physical experiments. The real robots were equipped with a Hokuyo URG laser range finder and a Hagisonic StarGazer Localization System (used to mitigate sensor error). The robots were also equipped with a Dual Core 1.6GHz machine running Ubuntu with 2GB of RAM. The range of the laser used for both experiments was 2m.

We conducted three sets of multi-robot experiments: two to test the effects of interference (NO COMMUNICATION and DIRECT COMMUNICATION) and the third to test the effect of latency (MESSAGE LOADS). A team of three robots was dispersed into an unknown environment to search via frontier exploration. For each experiment, five trials were run in the physical environments and 20 trials were run in simulation for each environment.

The six 6m x 6m environments used for the NO COMMUNICATION and DIRECT COMMUNICATION experiments are shown in Figure 4. Environments 2 - 4 all have the same amount of free space with one 1.5m x 1.5m obstacle strategically placed in the environment. Environments 5 and 6 both contain four 0.9m x 0.9m obstacles. The environmental configuration for the MESSAGE LOADS experiment was 6m x 6m with a T-shaped obstacle that represent walls of a building (see Figure 5). The robots were placed along the same wall one meter apart from each other.

V. Results and Analysis

Results and analysis are presented in this section.

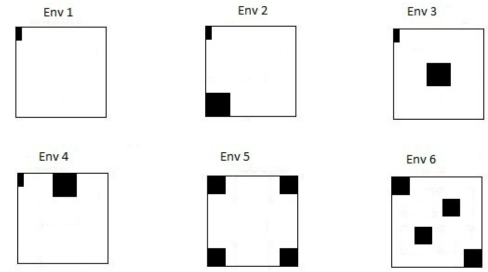

Fig. 4. The six environments used to test the effect of interference.

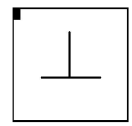

Fig. 5. Environmental configuration for the experiments testing the effect of message loads on latency.

A. Interference

1) Cooperation: To examine the hypothesis that different cooperation paradigms influence interference, experiments were performed using two cooperation paradigms: NO COMMUNICATION and DIRECT COMMUNICATION.

NO COMMUNICATION (see Table I) resulted in more total interference than DIRECT COMMUNICATION (see Table II) in most environments for both the real and simulated robots. However, the discrepancies between the two are not comparable. The interference for NO COMMUNICATION does not correlate to the search time or area and is much larger than its simulation or experimental DIRECT COMMUNICATION counterparts in both frequency and duration.

Although robots individually optimize their paths to increase coverage, the lack of coordination of actions in NO COMMUNICATION results in robots choosing to search the same areas. Even in environments where the lack of obstacles would eliminate bottlenecks, we see the interference component is still large.

NO COMMUNICATION and DIRECT COMMUNICATION could be considered extreme cases of cooperation and therefore the differences in interference alone do not highlight the contribution of cooperation or non-cooperation to the reduction of interference or increases in performance. However, this phenomenon has been seen in [13] where two cooperative paradigms included a messaging component that varied in whether robots shared a global or local search list. Teammates that shared a global list tended to choose the same search targets which increased interference which in turn degraded overall time-to-cover. In contrast, the method based on non-shared search lists (locally discovered targets) had better time-to-cover, partially due to reduced avoidance time.

These findings validate the hypothesis that different forms of cooperation affect interference. When robot cooperation results in less interference, robots cover the area faster.

TABLE I

THE AVERAGE NUMBER OF TIMES THE ROBOTS INTERFERED WITH ONE ANOTHER AND THE AVERAGE LENGTH OF EACH OCCURRENCE FOR NO COMMUNICATION.

	Number of occurrences				Time per occurrence (s)				Interference time (s)	
	Real		Sim		Real		Sim			
Env	μ	σ	μ	σ	μ	σ	μ	σ	*Real*	*Sim*
1	5.2	1.79	2.1	1.21	16.87	8.61	6.63	3.97	78.4	15.4
2	3.6	1.82	1.7	1.26	13.33	9.11	6.53	6.15	46.2	12.8
3	2.0	1.00	1.8	1.28	11.57	2.59	7.25	8.54	24.0	14.0
4	3.8	3.77	1.5	1.79	12.38	8.93	3.54	3.93	63.0	9.9
5	5.4	3.44	1.4	1.35	8.06	6.03	8.74	15.88	28.8	13.2
6	1.6	1.95	1.3	1.16	6.48	7.34	15.25	17.95	10.0	22.6

TABLE II

THE AVERAGE NUMBER OF TIMES THE ROBOTS INTERFERED WITH ONE ANOTHER AND THE AVERAGE AMOUNT OF TIME EACH OCCURRENCE HAPPENED FOR DIRECT COMMUNICATION.

	Number of occurrences				Time per occurrence (s)				Interference time (s)	
	Real		Sim		Real		Sim			
Env	μ	σ	μ	σ	μ	σ	μ	σ	*Real*	*Sim*
1	0.6	0.55	0.15	0.37	6.6	11.52	0.9	2.99	6.6	0.9
2	0.2	0.45	0.20	0.41	0.6	1.34	0.7	2.30	0.6	0.7
3	0.6	0.89	0.90	0.97	0.5	0.71	3.9	4.45	0.8	6.3
4	0.4	0.55	0.05	0.22	7.8	11.63	0.3	1.34	7.8	0.3
5	0.4	0.55	0.45	0.51	1.4	2.19	1.8	2.44	1.4	1.8
6	0.4	0.55	0.35	0.59	18.8	28.72	0.9	3.21	18.8	0.9

2) Environmental Configuration: We also hypothesized that different environments influence performance through interference. Table I suggests that some environments resulted in more interference than others. For instance, environment 1 and 5 resulted in more occurrences of interference for the real robots. These are the environments with more free space in the center of the environment. Likewise, environments 3 and 6 (with obstacles obstructing the environment) had the least occurrences for the real robots.

The environment played a slightly different role in simulation for NO COMMUNICATION. While environment 1 had more occurrences than the other environments, there was not a huge distinction between the occurrences for the different environments in simulation. This is one indication of the differences between simulation and real experiments.

These results suggest that environmental configuration influences how robots interfere with each other. More importantly, they suggest that although obstacles create bottlenecks, they provide a mechanism for spreading robots in such a way that reduces interference. This result can be seen in simulation but is more pronounced in real experiments.

3) Effect on Performance: Using NO COMMUNICATION, the amount of time it took the robots to explore 90% of the environments was examined (see Figure 6). The real robots always took more time to explore than the simulated robots. For DIRECT COMMUNICATION, Figure 7 shows the average time the robots explored 90% of the environments. Like NO COMMUNICATION, the real robots took longer to explore than the simulated robots. Unlike NO COMMUNICATION, the real and simulated experiments track relative performance for more environments.

For instance, it took the real robots the most time to explore environment 4 because the robots would interfere with each other resulting in less productivity. Figure 8 shows the paths of the robots in this environment. In this particular run, the robots spent a lot of time in the corner near the obstacle trying to maneuver around each other. As a result, two of the robots did not finish exploring the environment leaving the third robot to cover the rest of the area.

Figure 9 shows the coverage over time as robots covered environment 1 using NO COMMUNICATION. The same trend was seen in all six environments. The real robots started off covering the area at the same rate but eventually the simulated robots gained more coverage as time progressed. We believe that interference impacted the performance causing the real robots to lose coverage rate.

To formally investigate the hypothesis that unmodeled interference affects how well simulation approximates performance of a multi-robot experiment, we look at search time and coverage. If interference had an effect on real performance, we would expect to see a correlation between the differences between the real and simulated time-to-cover and the time spent interfering. In comparing the values for the environments for NO COMMUNICATION, we find a correlation of .77 which is significant at p=.005.

However, the same comparison for DIRECT COMMUNICATION does not show any correlation (r=.0645). Overall, interference in the DIRECT COMMUNICATION is quite low and is not a significant factor in coverage performance.

B. Latency

1) Message Loads: We hypothesized that latency from message loads affects multi-robot team performance. To compare the effects of message loads, experiments were per-

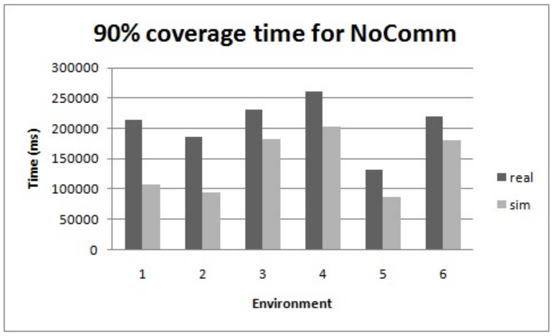

Fig. 6. Average time for the robots to explore 90% of each environment using no communication.

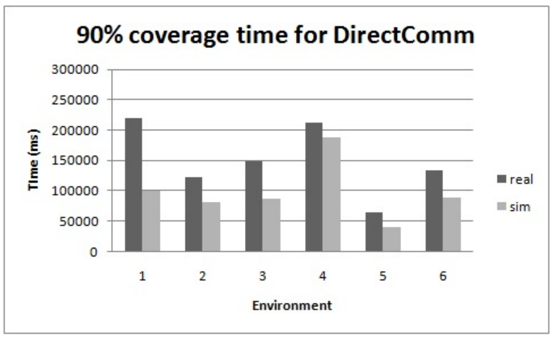

Fig. 7. Average amount of time for the robots to explore 90% of each environment using direct communication.

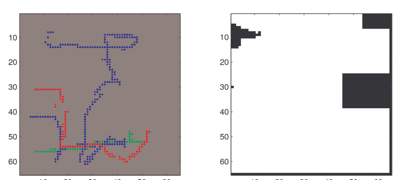

Fig. 8. Path of the real robot in environment 4 and the area that was covered (shown in white). Interference between the robots caused two of the robots to have poor performance.

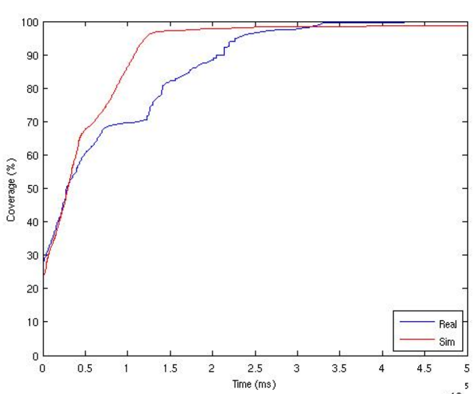

Fig. 9. Comparison of the average amount of area covered over time for environment 1 in the no communication experiments.

formed using two different direct communication scenarios: LESS MESS and MORE MESS.

The message delay averages are comparable between simulations and real experiments (see Table III). The message delay averages for real robot experiments are slightly higher than in simulation. There is a 0.344s and 0.372s difference between message delay for the LESS MESS and MORE MESS experiments, respectively. However, the standard deviations for the real robot experiments are significantly higher than those in simulation.

TABLE III
COMPARISON OF MESSAGE DELAY (IN S) FOR MESSAGES RECEIVED.

	LESS MESS	σ	MORE MESS	σ
SIM	1.446	0.153	3.247	0.067
REAL	1.791	2.022	3.620	2.715

2) Effect on Performance: To examine the impact of message loads on performance, percentage of area covered and average times at 50% and 90% coverage are compared. The coverage times for simulations indicate that there is not a significant difference between when robots pass less or more messages (see Figure 10). For example, at 50% coverage, robots that sent more messages were out performing those that sent less messages (see Table IV). However, in the real robot experiments, time decreased as the number of messages increased (see Figure 11).

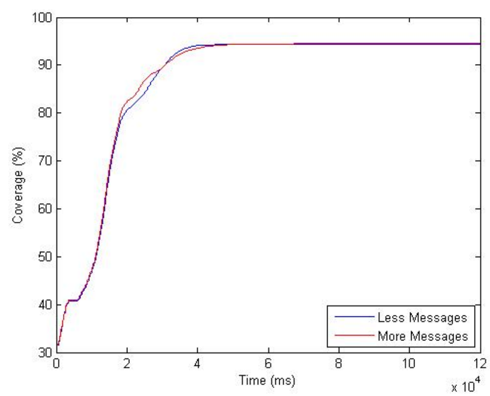

Fig. 10. Sim: Coverage Time for Less vs. More Messages Sent.

TABLE IV
SIM: AVERAGE COVERAGE TIME (IN S).

Messages	*50% cover time*	σ	*90% cover time*	σ
LESS	12.323	2.539	24.523	7.754
MORE	11.971	2.261	26.072	8.565

Tables IV and V show the average times robots reached 50% and 90% coverage. For the LESS MESS experiments, 90% coverage was reached in simulation 143.531s (2.39 min) faster than in real robot experiments. With the MORE MESS experiment, 90% coverage was reached in simulation 175.985s (2.93 min) faster than in real robot experiments.

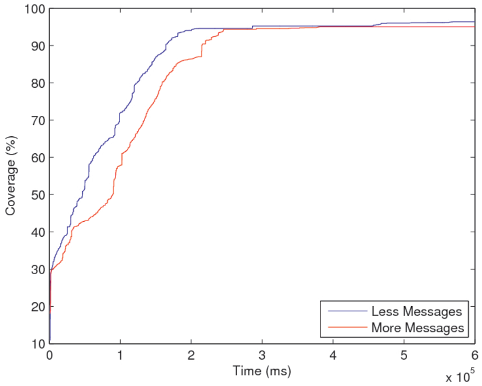

Fig. 11. Real: Coverage Time for Less vs. More Messages Sent.

TABLE V
REAL: AVERAGE COVERAGE TIME (IN S).

Messages	*50% cover time*	σ	*90% cover time*	σ
LESS	48.580	22.766	168.054	77.611
MORE	89.639	17.284	202.057	31.256

Results suggest that message delay has a greater impact on team performance in physical robot experiments than in simulation. The higher standard deviations for message delay in the real robot experiments indicate that it is more unpredictable and variable when using real robots. Therefore, when conducting simulations the effects of message loads should be modeled for more realistic performance prediction.

VI. CONCLUSION

In this paper, we present results that validate interference and message processing load as factors that affect real multi-robot performance but are not accurately modeled in simulation. The conclusion is that algorithms that show promise in simulation may not perform well on real robots. In contrast, algorithms that do not show significant performance gains in simulation may in fact provide performance increases in experiments by affecting one of the underlying unmodeled factors such as interference or reduction in message communication.

These results motivate research that looks at cooperation paradigms that reduce interference (through coordination or dispersion) without relying upon increased messaging paradigms. Approaches that use observation [24] like primate groups may be particularly suited to performance gains in real robots. In addition, the impact of environment configuration requires a much more careful study that we currently give it. Without the ability to quantify which environment features make coverage difficult, researchers cannot appropriately choose environments to validate algorithm performance in the general case.

VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of the following NSF grants: IIS-0846976 and CCF-0829827.

REFERENCES

[1] "Stage: A multiple robot simulator." [Online]. Available: http://playerstage.sourceforge.net/index.php?src=stage

[2] "Carmen: Robot navigation toolkit." [Online]. Available: http://carmen.sourceforge.net/home.html

[3] "Opensim: An open 3d robotics simulator." [Online]. Available: http://opensimulator.sourceforge.net/

[4] O. Michel, "Cyberbotics ltd - webots: Professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, pp. 39–42, 2004.

[5] "Gazebo: 3d multiple robot simulator with dynamics." [Online]. Available: http://playerstage.sourceforge.net/index.php?src=gazebo

[6] M. Brady and H. Hu, "Software and hardware architecture of advanced mobile robots for manufacturing," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 9, pp. 257–276, 1997.

[7] C. J. Pretorius, M. C. du Plessis, and C. B. Cilliers, "Towards an artificial neural network-based simulator for behavioural evolution in evolutionary robotics," in *Proceedings of the 2009 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists (SAICSIT)*, 2009, pp. 170–178.

[8] R. A. Brooks, "Artificial life and real robots," in *Proceedings of the First European Conference on Artificial Life*. MIT Press, 1992, pp. 3–10.

[9] N. Jakobi, P. Husbands, and I. Harvey, "Noise and the reality gap: The use of simulation in evolutionary robotics," in *Proceedings of Third European Conference on Artificial Life (ECAL'95)*. Springer-Verlag, 1995, pp. 704–720.

[10] L. Meeden, "Bridging the gap between robot simulations and reality with improved models of sensor noise," in *Proceedings of the Third Annual Genetic Programming Conference*, 1998, pp. 824–831.

[11] C. Melhuish, M. Wilson, and A. Sendova-Franks, "Patch sorting: Multi-object clustering using minimalist robots," in *Proceedings of 6th European Conference on advances in artificial life*. Springer, 2001, pp. 543–552.

[12] A. Boeing, S. Hanham, and T. Braunl, "Evolving autonomous biped control from simulation to reality," in *Proceedings of International Conference on Autonomous Robots and Agents (ICARA '04)*, 2004, pp. 440–445.

[13] M. Anderson and N. Papanikolopoulos, "Implicit cooperation strategies for multi-robot search of unknown areas," *Journal of Intelligent and Robotic Systems*, vol. 53, pp. 381–397, 2008.

[14] P. E. Rybski, S. A. Stoeter, M. Gini, D. F. Hougen, and N. Papanikolopoulos, "Performance of a distributed robotic system using shared communications channels," *IEEE transactions on Robotics and Automation*, vol. 18, no. 5, pp. 713–727, 2002.

[15] A. Rosenfeld, G. A. Kaminka, and S. Kraus, *A Study of Scalability Properties in Robotic Teams*. Springer, 2005.

[16] S. Balakirsky, S. Carpin, G. Dimitoglou, and B. Balaguer, "From simulation to real robots with predictable results: Methods and examples," in *Performance Evaluation and Benchmarking of Intelligent Systems*. Springer, 2009, pp. 113–137.

[17] T. Kyriacou, U. Nehmzow, R. Iglesias, and S. Billings, "Accurate robot simulation through system identification," in *Robotics and Autonomous Systems*, 2008, pp. 1082–1093.

[18] R. Brooks, "A robust layered control system for a mobile robot," *IEEE journal of robotics and automation*, vol. 2, no. 1, pp. 14–23, 1986.

[19] W. Burgard, D. Fox, D. Hennig, and T. Schmidt, "Estimating the absolute position of a mobile robot using position probability grids," in *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, 1996, pp. 896–901.

[20] B. Jung and G. S. Sukhatme, "Tracking targets using multiple robots: The effect of environment occlusion," *Autonomous Robots*, vol. 13, no. 3, pp. 191–205, 2002.

[21] M. J. Mataric, "Issues and approaches in design of collective autonomous agents," *Robotics and Autonomous Systems*, vol. 16, pp. 321–331, 1995.

[22] B. Yamauchi, "Frontier-based exploration using multiple robots," in *Proceedings of the second international conference on Autonomous agents*, 1998, pp. 47–53.

[23] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[24] B. Wellman, S. Downing, G. Moore, and M. Anderson, "Observation-based cooperation in mobile sensor networks," in *Proceedings of the first international conference on sensor networks and applications (SNA)*, 2009.