Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Planning to Fail - Reliability Needs to Be Considered *a Priori* in Multirobot Task Allocation

Stephen B. Stancliff, John Dolan
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA, USA
{sbs,jmd}@cmu.edu

Ashitey Trebi-Ollennu
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA
Ashitey.Trebi-Ollennu@jpl.nasa.gov

***Abstract*—The reliability of individual team members has a substantial and complex influence on the success of multirobot missions. When one robot fails, other robots must be retasked to complete the tasks that were assigned to the failed robot. This in turn increases the likelihood of these other robots failing, since they have more work to do. Existing multirobot task allocation systems consider robot failures only after the fact–by replanning after a failure occurs. We hypothesize that it should be important to consider robot reliabilities when generating an initial plan. In this paper we test this hypothesis in the context of the multirobot exploration problem. We take a simple exhaustive planner and compare the plan it chooses against the optimal plan that takes into account robot failures and the backup plans that occur after failure. Our results show that for this problem domain, making an initial plan without regards to individual robot reliabilities results in choosing a suboptimal plan most of the time, and that the difference in mission performance between the chosen plan and the optimal plan is usually substantial. In brief, in order to successfully plan we must 'plan to fail'.**

***Index Terms*—Multirobot systems, reliability, task allocation.**

I. Introduction

In order for multirobot systems to operate successfully in the real world, they must be able to deal with the inevitable failures of team members. Reference [1] summarizes historical failure data for small field robots and reveals that the robots were either broken or under repair approximately half of the time. Given such poor reliability, it seems likely that any multirobot planning system that does not consider robot failures will perform poorly for real-world missions.

There has been a substantial amount of work in the area of detecting and recovering from robot failures (e.g., [2], [3], [4]), and several multirobot mission planning systems provide mechanisms for reallocation of tasks among surviving team members after a robot failure (e.g., [5], [6], [7]). However, all of these methods are reactive rather than predictive, dealing with failure only after it occurs. While it is important in dynamic real-world environments to be able to replan after robot failure, we hypothesize that it should also be useful to consider the probabilities of robot failures when making initial task assignments. As an abstract example, one would not want to assign a robot that has a high chance of failure to a critically important task. We are not aware of any existing work that addresses the use of robot reliability to improve multirobot task allocation in this way.

In this paper we examine the multirobot exploration problem in order to evaluate whether it is beneficial to consider the possibility of robot failure *a priori* in multirobot task allocation. We compare the plans chosen by a simple exhaustive planner that is capable of reallocating tasks after robot failure against the optimal plans that take into account the probabilities of robot failures and the backup plans that occur after failure. We examine planner performance under two different utility functions: minimizing mission duration and minimizing total distance travelled.

Our results show that the task allocations chosen when robot reliability is ignored often produce suboptimal performance in scenarios where robot failure is likely to occur. In other words, in most real-world scenarios.

One key feature of our method is that we do not introduce arbitrary reliability requirements into the mission specifications, but instead show that reliability needs to be considered in order to optimize an *existing* utility metric.

We conclude that robot reliability needs to be considered in multirobot task allocation even when reliability is not itself an explicit performance metric for the mission.

II. Background

Mission planning for multirobot systems is an active research area with substantial literature describing many different mission planners. One measure by which the performance of multirobot mission planning methods can be compared is how they deal with robot failures. Reference [6], for example, describes a mission planning system that is able to recover from robot failure because tasks can be reallocated. In this system, tasks are "auctioned" to the robot with the highest bid (or lowest, depending on the utility function). This system allows for task reallocation during the mission when new information changes the valuation of tasks. For instance, if a robot suffers a component failure that impairs its ability to perform its assigned tasks, it will change its valuation for those tasks, and it can then subcontract tasks to another robot that has a better valuation for those tasks.

While it is important to recover from robot failures, it would be better to minimize the likelihood of such failures in the first place. The mission planning system can influence these likelihoods since the initial assignment of tasks to robots plays

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

TABLE I. Robot and target parameters

	x	y	P_t
Robot 1	4	12	0.99
Robot 2	14	3	0.99
Target 1	1	1	—
Target 2	3	5	—

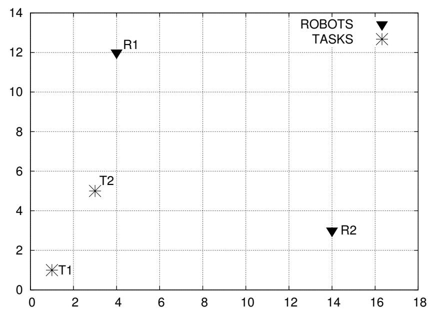

Fig. 1. Exploration mission

TABLE II. Plan durations

Plan	$d(R1)$	$d(R2)$	d_{plan}
A $(R_1T_1 + R_1T_2)$	15.9	0	15.9
B $(R_1T_1 + R_2T_2)$	11.4	11.2	***11.4***
C $(R_2T_1 + R_1T_2)$	7.62	13.2	13.2
D $(R_2T_1 + R_2T_2)$	0	17.6	17.6
E $(R_1T_2 + R_1T_1)$	11.5	0	11.5
F $(R_2T_2 + R_2T_1)$	0	15.7	15.7

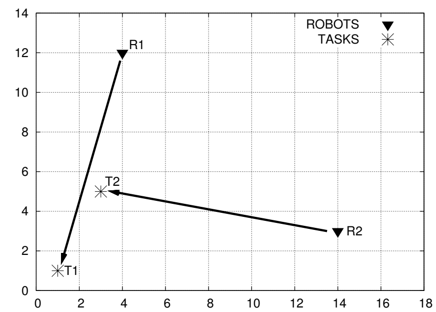

Fig. 2. Chosen plan (Plan B)

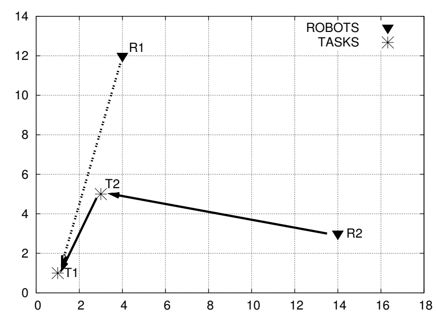

Fig. 3. Backup plan for Plan B when Robot 1 fails (dashed line represents robot failure)

a role in determining the probabilities of robot failures during the mission. For example, assigning a robot with a damaged drive motor to a task that requires it to travel a long distance results in a higher probability of that robot failing than if it was assigned to a task that required less travel.

One way to incorporate such reliability concerns into multirobot mission planning would be to introduce a reliability component into the utility function used by the planner. For example, [8] provides methods for predicting probabilities of failure for mobile robots during mission tasks. Incorporating such probabilities directly into the planner utility function is unsatisfying for two reasons. The first is the incommensurability of different components of a utility measure—how do we combine dollars spent, meters travelled, and probability of failure into a single metric? The second is that establishing a numeric reliability requirement is itself a difficult problem that has been minimally explored for the mobile robot domain—i.e., how do we decide if the reliability requirement for a mission should be 95% rather than 96%?

In order to avoid these difficulties, we take a different approach—rather than devising utility metrics that incorporate reliability, we instead look at how robot reliability affects the utility metrics already being used. In plain language, what we are *not* doing is taking "Find the solution with the shortest time" and turning it into "Find the solution with the shortest time that also meets reliability level X" but instead turning it into "Find the solution with the shortest *expected* time" where the expected time takes into account the alternative outcomes that occur when robots fail.

III. Illustrating Example

Consider a simple multirobot exploration mission with two identical robots and two locations to be visited (Fig. 1). The goal of the mission is for all target locations to be visited in any order by any robot in the shortest total mission time. Time is assumed here to be proportional to distance traversed.

Each robot is defined by an (x, y) location and a reliability P_t, which is the probability of surviving a one-unit traverse. Each target is defined by its (x, y) location. The robot and target parameters used for this example are listed in Table I.

For a small number of robots and tasks, it is feasible to exhaustively enumerate the possible task assignments and then calculate the distance that each robot must traverse to accomplish each plan (Table II). The plan duration d_{plan} is equal to the greatest distance that any robot travels during that plan. The plan with the smallest duration is then chosen. In this example Plan B (Fig. 2) would be chosen.

Now consider what happens when a robot fails while executing this plan. If Robot 1 fails, then Robot 2 is assigned to perform Task 1 after completing Task 2 (Fig. 3). If Robot 2 fails, then Robot 1 is assigned to Task 2 after completing Task 1 (Fig. 4). We assume here that tasks are not interrupted, so that new tasks are assigned to surviving robots only after the completion of their current tasks.

Examining Fig. 3 and Fig. 4, we see that while the backup plan in the first case is reasonable, the second one is inefficient because the optimal plan for Robot 1 to visit both locations is to perform Task 2 first. This inefficiency is what is being missed when reliability is not considered in the original task allocation.

In order to take into consideration both primary and backup plans, we calculate an expected duration (d_{exp}) for each primary plan by multiplying the duration and probability for each alternative outcome and then adding these together. Here we assume that the duration for a failed robot–task pairing is equal to the entire duration of that task. In other words, replanning only happens after a robot has failed to meet a deadline.[1]

For Plan B this gives:

$$\begin{aligned} \text{Primary plan duration} &= 11.4 \\ \text{Backup 1 (Fig. 3) duration} &= 15.7 \\ \text{Backup 2 (Fig. 4) duration} &= 15.9 \\ P(R_1T_1 \text{ success }) = (P_t)^{11.4} &= 0.891 \\ P(R_2T_2 \text{ success }) = (P_t)^{11.2} &= 0.894 \\ P(\text{Primary plan}) = P(R_1T_1) \times P(R_2T_2) &= 0.797 \\ P(\text{Backup 1}) = P(\overline{R_1T_1}) \times P(R_2T_2) &= 0.097 \\ P(\text{Backup 2}) = P(R_1T_1) \times P(\overline{R_2T_2}) &= 0.095 \end{aligned}$$

$$\begin{aligned} d_{exp} &= \frac{\sum P_i \times d_i}{\sum P_i} \\ &= \frac{(0.797)(11.4) + (0.097)(15.7) + (0.095)(15.9)}{(0.797 + 0.097 + 0.095)} \\ &= 12.3 \end{aligned}$$

The denominator in the last equation normalizes the expected duration to account for the fact that we are ignoring those cases where both robots fail. Our result is therefore an expected duration *assuming that the mission succeeds*.

Repeating these calculations for each primary plan gives the results shown in Table III. We see here that a different plan (Plan E,Fig. 5) has the shortest expected duration. Looking at the backup plans for Plan E (Fig. 6 and Fig. 7), we see that these plans are optimal plans for the surviving robot in each case. It is this optimality of backup plans that is being missed when reliability is ignored in the initial task allocation.

[1]This assumption will fail when all of the following conditions are met: (1) A robot has a failure that prevents completion of its task but does not eliminate its ability to communicate with other robots; (2) the robot is capable of self-diagnosis in order to recognize that a failure has occurred; (3) the mission requirements allow the robot to communicate its failure (i.e., not a mission where terrain blocks communication, not operating under radio silence in enemy territory, etc.). For most real-world missions, these conditions will not all be met, so the failure of one robot will be detected by its failure to communicate success at some time greater than or equal to the expected task completion time. For realistic missions, therefore, the assumption is conservative.

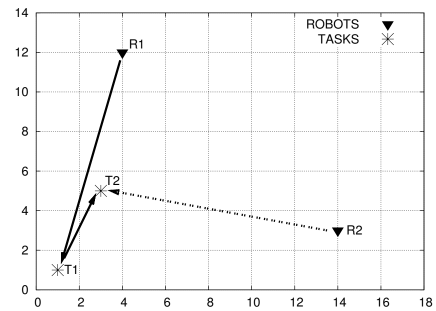

Fig. 4. Backup plan for Plan B when Robot 2 fails

TABLE III. Naive and expected durations

	Plan	d_{naive}	d_{exp}
A	$R_1T_1 + R_1T_2$	15.9	15.8
B	$R_1T_1 + R_2T_2$	***11.4***	12.3
C	$R_2T_1 + R_1T_2$	13.2	13.2
D	$R_2T_1 + R_2T_2$	17.6	16.8
E	$R_1T_2 + R_1T_1$	11.5	***11.9***
F	$R_2T_2 + R_2T_1$	15.7	15.2

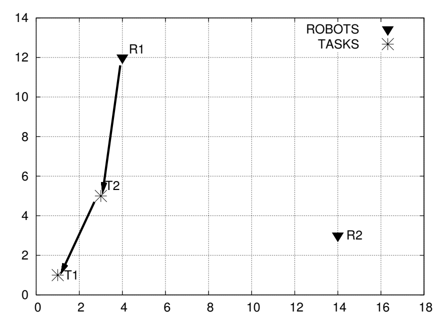

Fig. 5. Plan with shortest expected duration (Plan E)

IV. Simulation Results

The preceding example demonstrated a case where the plan chosen by a reliability-naive planner is not the optimal plan. In order to determine whether this is an anomalous case or if it is the norm, we implemented the methods described in the preceding section in software. This software was used to evaluate mission configurations with varying world sizes, team sizes, task counts, and robot reliabilities.

For each mission configuration (gridsize, P_t, number of robots, task count), numerous trials (on the order of 100k) were run, with robot and target locations randomized for each trial. The primary output of the simulation was the percentage of trials in which the planner chose a suboptimal plan. A simulation run was terminated when this value converged. Repeatability of our results was within roughly 2%.

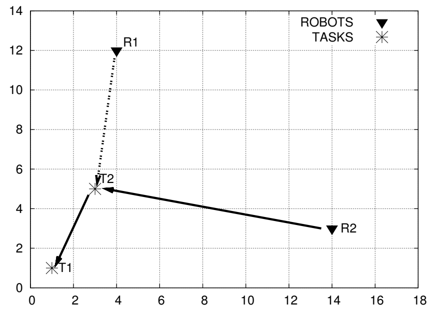

Fig. 6. Backup for Plan E when Robot 1 fails during Task 2

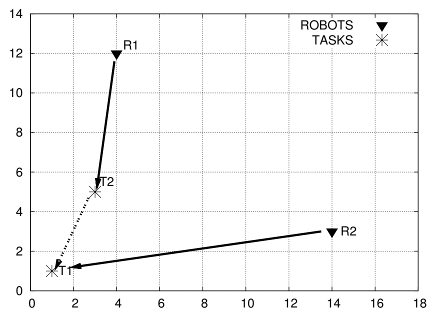

Fig. 7. Backup for Plan E when Robot 1 fails during Task 1

A. Minimax Utility Function

The analysis in this section follows the example in Section III in seeking to minimize the overall mission duration by minimizing the maximum distance travelled by any individual robot (minimax utility function).

Fig. 8 shows the results for several values of P_t for a 50×50 world with two robots and four targets. We see from this figure that even when the robots are fairly reliable ($P_t = 0.99$) the planner selects a suboptimal plan about 66% of the time, and the error rate increases rapidly with decreasing robot reliability.

Fig. 9 shows the results for different world sizes with a constant value of P_t. We see here that planner performance is poor even for a small 50×50 world, and that the error rate increases with increasing world size.

These results make intuitive sense because both larger world size and lower robot reliability increase the likelihood of robot failure during the mission, and with higher likelihood of robot failure the influence of the backup plans on the expected duration is greater.

The effects of task count (Fig. 10) and team size (Fig. 11) on planner performance are as expected, with more tasks and more robots leading to more robot failures, which in turn leads to poorer plan selection.

B. Differences in Plan Durations

The above results show that the planner often chooses a suboptimal plan, but it is perhaps more important to know *how much worse* the chosen plan is in comparison with the

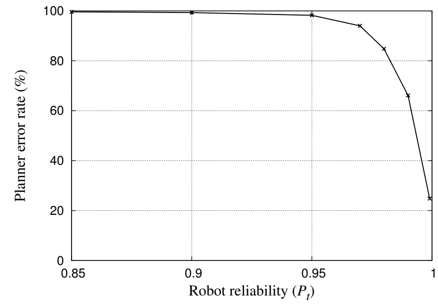

Fig. 8. Suboptimal allocations as a function of robot reliability (50×50 world, two robots, four targets)

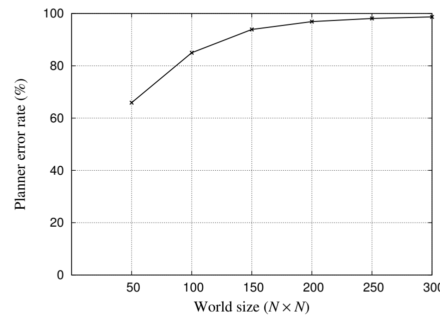

Fig. 9. Suboptimal allocations as a function of world size ($P_t = 0.99$, two robots, four targets)

optimal plan. Fig. 12 shows, for the same mission scenario as Fig. 10, the average percent difference in duration for the chosen plan versus the optimal plan for those cases where the planner chooses a suboptimal plan.

Looking at Fig. 10 and Fig. 12 together, we see that for a 100×100 world, $P_t = 0.99$, two robots, and four tasks a suboptimal plan is chosen 85% of the time, and the chosen plan in those cases is on average 44% longer than the optimal plan.

C. Overall Planner Performance Metric

We can combine these results (percent error rate and percent increase in duration) into a single measure of planner performance by multiplying them together, giving the expected increase in plan duration (Fig. 13).

Figures 14–16 revisit the effects of team size, world size, and robot reliability in terms of this overall performance metric. These figures show that for very simple missions (few robots, few tasks, small world) with reliable robots, the expected performance penalty from ignoring reliability is under 25%, but it increases rapidly with increasing mission complexity.

D. Minisum Utility Function

The utility function used in the previous examples seeks to minimize the overall mission duration. This may not be the

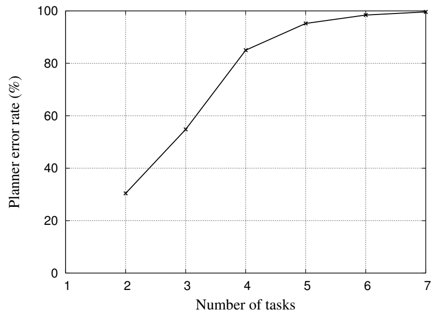

Fig. 10. Suboptimal allocations as a function of task count (100×100 world, $P_t = 0.99$, two robots)

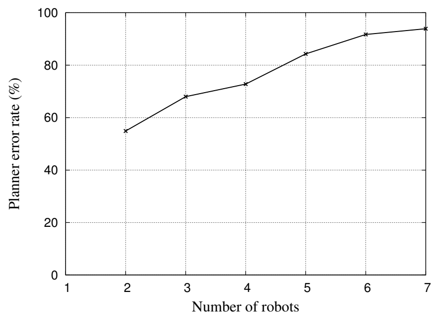

Fig. 11. Suboptimal allocations as a function of team size (100×100 world, $P_t = 0.99$, three targets)

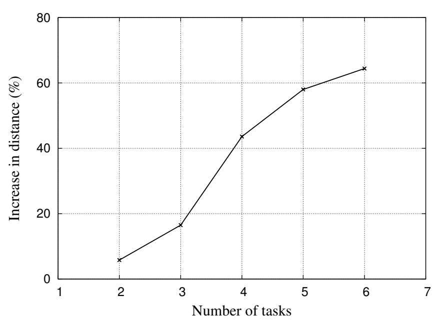

Fig. 12. Average increase in duration over optimal plan as a function of task count. (100×100 world, $P_t = 0.99$, two robots)

most important task allocation criterion for all robot missions. For example, in a planetary exploration mission it is usually more important to conserve resources (e.g., fuel, electricity) than to meet deadlines.

We modified our software to minimize the total distance travelled by all the robots together (assuming that power consumption is directly proportional to distance travelled) and repeated the above analyses. The results in terms of number of tasks and number of robots are shown in Fig. 17 and Fig. 18.

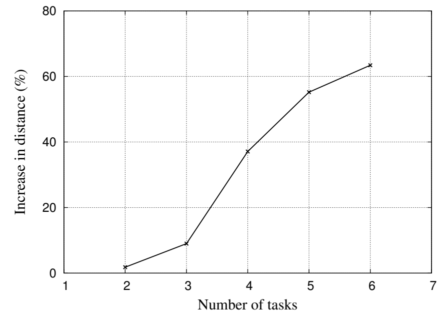

Fig. 13. Expected increase in duration over optimal plan as a function of task count. (100×100 world, $P_t = 0.99$, two robots)

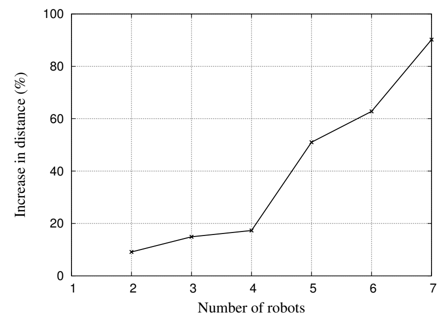

Fig. 14. Expected increase in duration over optimal plan as a function of team size (100×100 world, $P_t = 0.99$, three targets)

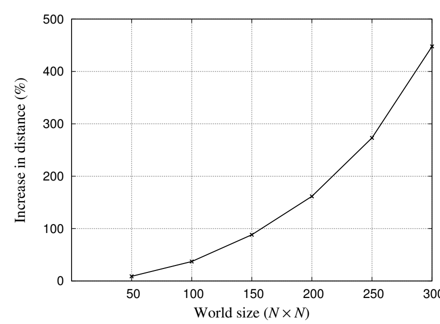

Fig. 15. Expected increase in duration over optimal plan as a function of world size ($P_t = 0.99$, two robots, four targets)

Comparing these figures with Fig. 13 and Fig. 14 shows that the planner is better at choosing the optimal plan under the minisum utility function, but there is still a large performance penalty for complex missions.

It makes intuitive sense that the planner performs better under the minisum function because minimizing the total distance travelled also tends to minimize robot failures.

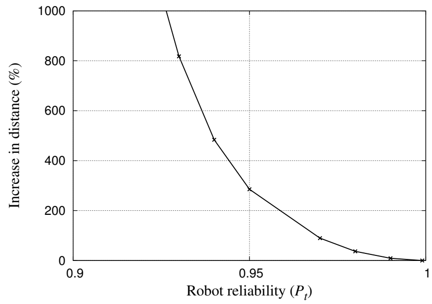

Fig. 16. Expected increase in duration over optimal plan as a function of robot reliability (50×50 world, two robots, four targets)

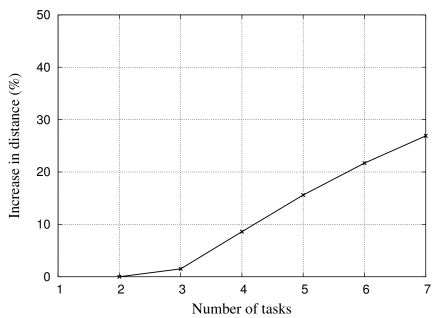

Fig. 17. Expected increase in distance over optimal plan as a function of task count (Minisum, 100×100 world, $P_t = 0.99$, two robots)

V. Summary and Future Work

In this paper we tested the hypothesis that failing to consider robot reliabilities in multirobot task allocation will lead to suboptimal plans. For the simple instances of the multirobot exploration problem evaluated here, our results show that the task allocation chosen when robot failure is ignored often produces suboptimal performance in scenarios where robot failure is likely to occur. In other words, in most real-world scenarios.

A key feature of these results is that we have slipped reliability "in through the back door" in that we have not introduced arbitrary reliability requirements into the mission specifications but have instead shown that reliability needs to be considered in order to optimize an *existing* utility metric (in these examples, total mission duration or total energy expenditure). We conclude that robot reliability needs to be considered in generating multirobot task allocations even when reliability is not itself an explicit performance metric for the mission.

The most significant shortcoming of these results is that they were obtained through brute-force solutions to fairly simple problems. Real-world multirobot problems are usually too complex for brute-force solutions, which is why the literature

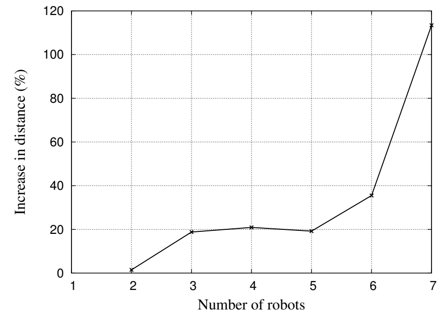

Fig. 18. Expected increase in distance over optimal plan as a function of team size (Minisum, 100×100 world, $P_t = 0.99$, three targets)

describes many heuristic planning methods (e.g., [6], [7]).

The obvious question to be addressed in future work is whether *a priori* reliability information is useful in the context of a heuristic planner. The first part of this problem is to determine if *incomplete* knowledge of backup plans can still provide an improvement in task allocation. The second is to determine if backup plans can be considered without excessively increasing the computational complexity of the task allocation problem.

Ultimately, we hope that this work will lead to integration of reliability estimation into existing mission planners so that we can "plan to fail" rather than "failing to plan."

Acknowledgment

Part of the research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] J. Carlson and R. Murphy, "Reliability analysis of mobile robots," *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 1, pp. 274–281, Sept. 2003.

[2] L. Parker, "Alliance: an architecture for fault tolerant multirobot cooperation," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 2, pp. 220–240, Apr 1998.

[3] G. Kaminka and M. Tambe, "Social comparison for failure detection and recovery," in *Intelligent Agents IV; Proceedings of ATAL-97*. Springer Verlag, 1997, pp. 12–7.

[4] V. Verma, G. Gordon, R. Simmons, and S. Thrun, "Real-time fault diagnosis [robot fault diagnosis]," *Robotics and Automation Magazine, IEEE*, vol. 11, no. 2, pp. 56–66, June 2004.

[5] B. Brumitt and A. Stentz, "Grammps: a generalized mission planner for multiple mobile robots in unstructured environments," *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 2, pp. 1564–1571, May 1998.

[6] B. Dias, M. Zinck, R. Zlot, and A. Stentz, "Robust multirobot coordination in dynamic environments," *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 4, pp. 3435–3442, May 2004.

[7] B. Gerkey and M. Mataric, "Sold!: auction methods for multirobot coordination," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 5, pp. 758–768, Oct 2002.

[8] S. Stancliff, J. Dolan, and A. Trebi-Ollennu, "Mission reliability estimation for multirobot team design," *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 2206–2211, Oct. 2006.