Preliminary Results in Sliding Autonomy for Assembly by Coordinated Teams

Jonathan Brookshire, Sanjiv Singh and Reid Simmons
Robotics Institute
Carnegie Mellon University
Pittsburgh, USA
{brookshire, ssingh, reids}@cmu.edu

***Abstract*—We are developing a coordinated team of robots to assemble structures, a task that cannot be performed by any single robot. Even simple operations in this domain require complex interaction between multiple robots and the number of contingencies that must be addressed if the team is to act completely autonomously is prohibitively large. This scenario forces incorporation of a human operator. Ideally we would like a seamless interface between the robots and the operator such that the operator can interact with the system by helping it be more efficient or get out of a stuck condition or performing a task that the robots are not capable of themselves. We use an architecture that implements "sliding autonomy" to accomplish these goals. The system of robots can be fully autonomous as long as all is well. The system is capable of accepting input from the operator at any time, especially when it is unable to recover from a failure. We motivate this scenario with results from an extended series of experiments we have conducted with three robots that work together to dock both ends of a suspended beam. We show the difference in performance between a completely tele-operated system, a fully autonomous system, and one in which sliding autonomy has been incorporated.**

Keywords- multi-robot coordination, assembly, structures, architectures, sliding autonomoty, tele-operation.

I. INTRODUCTION

Increasingly, robotic systems perform tasks that humans cannot accomplish in environments where humans cannot operate. Search and rescue, manufacturing, construction, and planetary exploration offer only a small sample of the complex tasks and the hostile environments in which robots are expected to function. The intricate nature of these tasks may require a team of robots capable of monitoring and guiding their own progress with a high level of autonomy. On the other hand, unpredictable environments and circumstances may exceed the abilities of the autonomous system and, as a result, also demand tele-operator intervention or assistance. In this way, a system that combines elements of both autonomy and human control becomes necessary to assure performance and quality.

Our primary objective is to develop fundamental capabilities that enable multiple heterogeneous robots to work together, and with humans, in flexible, robust ways to deal with contingencies and to improve overall efficiency. The basic concept is for robots to accomplish tasks, either autonomously or through tele-autonomy, which can be mixed and matched together to achieve significant levels of coordinated behavior. Decisions about when to switch between autonomous and tele-operated control can be made smoothly, both by humans and by the robots themselves.

This concept, which we term sliding autonomy, addresses one of the main areas of difficulty in current human-robot interaction: Typically control is highly inflexible, reflecting an "either-or" constraint. That is, operating structures are either pure tele-operation or pure autonomy. As complexity of the task increases, so does our inability to anticipate every contingency in which the robot will find itself [2]. In order to overcome these further complexities, then, it is essential to consider a mixed initiative system in which humans are able to collaborate seamlessly with a team of robots. Further, while autonomous operation is important for teams of robots performing complex functions, there will be many situations in which they will not successfully perform, or not even know how to perform, a given subtask. In addition, it is impossible to anticipate the types of failures that might occur when a team of heterogeneous agents collaborates and to devise autonomous recovery strategies for all of them ahead of time. In these cases, and others as well, human intervention can help.

As part of developing sliding autonomy, we are designing the architectural framework and techniques that enable a human to take over control of a given subtask, while the robot(s) maintain local autonomy. This architecture allows the robot to maintain cognizance of the overall task – monitoring it to determine when it has been completed (in which case it should autonomously move on to the next task) or when it has failed (in which case it should attempt recovery actions, perhaps in concert with the human operator). Figure 1 shows examples of shifting control between an operator and the system. This approach has several distinct advantages over other approaches to human-robot interaction. The robot maintains local autonomy even during tele-operation. In most other approaches, the task context is lost when the robot cedes control. When autonomous control is returned to the robot, it has little idea of what the user has done and how those actions fit in with the overall goals and tasks. Our approach allows for a smooth hand-off between (any part of) the system and a human operator. This approach is truly mixed

The research reported was supported by National Science Foundation Grant ECS-0233698.

initiative. The robot can request help if it determines it is outside its range of expertise; but, at the same time, an operator can request to take over control of some task whenever he/she desires.

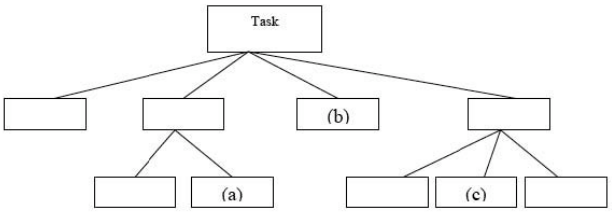


Figure 1. Mode shift between operator and system. A task is decomposed hierarchically. Control can shift in three instances: (a) human initiated - the operator might take control if dissatisfied by the progress of the system; (b) scripted - a subtask is designated a priori as one to be performed by the operator; the system turns over control to an operator at the appropriate time and picks up operation when the operator indicates completion; (c) robot initiated - the system continuously estimates confidence in its own state and the chance of completing the task; upon encountering low confidence, it autonomously drops down in levels of autonomy and asks for operator assistance.

I. Related Work

Our architectural approach differs from most other work in multi-robot systems, in which the robots are either loosely coupled agents, with little or no explicit coordination (e.g. [1], [4], [20], and [21]) or are tightly coordinated by a highly centralized planning/execution system. Many multi-robot tasks are characterized by close coordination of the robots. One approach is to coordinate the robots centrally (e.g. [3] and [18]). Doing so, however, introduces a single-point source of failure, and does not work well in high latency situations. An alternative approach, somewhat more reliable and flexible, is to distribute control. This remedy, however, makes it difficult to achieve tight coordination. Under our scheme, individual robots can autonomously solve many problems either by themselves or by negotiating with each other, without having to invoke a high-level planner. These characteristics reduce the need for inter-robot communication and improve overall reliability. As such, our approach is similar to work in which coordination strategies are explicitly represented and reasoned [17]. Our architecture also supports dynamic team formation. Coordination occurs between agents filling specific roles in the structure of the team, and roles can be dynamically assigned to agents. In terms of human collaboration with robots, there are several relevant research efforts. The COBOT project seeks to make manually operated machines more intelligent by providing guidance so that the operator does not have to finesse control. Typically, the human provides the force input, while the system steers the mechanism into the right place [14] and [26]. A research effort in getting large numbers of people (hundreds) to collaborate to achieve a trajectory-tracking task is discussed in [10]. This type of system is essentially tele-operation because the robot is being controlled explicitly. The authors show that variation in performance is based on the number of collaborators of the task. A more closely related system is described by Fong et al. in which the robot and the user participate in a dialogue [9]. The robot can ask the operator to help with localization or to clarify sensor readings. The operator can also make queries of the robot. This framework assumes that the robot is capable of performing all tasks as long as it has full state information. Another effort has examined the effectiveness of an operator when controlling a robot at different levels of autonomy given increasing inattention to the robot [11]. Scerri has proposed an architecture for sliding autonomy applied to a daily scheduler [23]. The autonomous system attempts to resolve timing conflicts (missed meetings, group discussions, personal conflicts, etc.) among some set of team members. Members could adjust the autonomy of the system by indicating their intent to attend gatherings or willingness to perform tasks. The term sliding autonomy is interchangeable with adjustable autonomy as presented by Dorais et al. [5]. The authors provide several examples in which sliding autonomy will be essential for space operations where demands on the operator must be focused and minimized. Using a roving eye and a (fixed) manipulator similar to ours, Kortenkamp et al. developed and tested a software infrastructure that allows for sliding autonomous control of a robot manipulator [19]. The task involved a pick-and-place operation during which sliding autonomy allowed the operator to recover from visual servoing errors, participate in high-level planning, and tele-operate to complete tasks beyond autonomous capabilities. Our work extends these experiments with a more complex assembly task and a greater level of sliding autonomy between robots.

I. Approach

We are testing these ideas in the context of a team of robots that work together to assemble a physical structure that requires operations that cannot be performed by any single robot. The robot team includes a mobile manipulator (a skid-steered ATRV with a five degree-of-freedom manipulator), the NIST Robocrane (a six DOF inverted Stewart platform), and a mobile robot equipped with stereo cameras (Figure 2) [25]. The beam and the robots are marked with fiducials that allow the roving eye to determine the relative distance between the fiducials (Figure 3). This information is continually transmitted wirelessly to both the other robots so that they can use this information to move.

Each robot plays a role in docking a beam securely between two uprights. The Robocrane provides the heavy lifting capability and large workspace to grossly maneuver the beam, while the mobile manipulator finely positions the beam into the docking clamps using a coordinated resolved motion rate control to drive the ends. The mobile stereo cameras provide feedback for visual servoing and can be moved to focus on different aspects of the operation. Figure 4 shows three steps in the operation of docking both ends of the beam.

Coordinating these robots requires synchronization of a task tree that spans robots. This is done using a task executive that orchestrates the task across the robots [24]. For example the two-ended beam docking described here requires the execution of the tree shown in Figure 5. The task is complex enough that there are many different failure

modes, some of which occur very infrequently. We have run the beam docking experiment hundreds of times and have analyzed the types of errors that can occur--occasionally, the visual servoing fails because the roving eye loses track of a fiducial and cannot acquire it once again. Some times, a misestimation of distance causes a (near) collision between robots and the experiment has to be aborted. Sometimes, the robot arm gets stuck on the beam and the system can't easily disambiguate this situation from the normal operation. Very occasionally an intermittent electrical problem resets one of the robots, and, a software bug causes a crash of a vital process on the mobile manipulator.

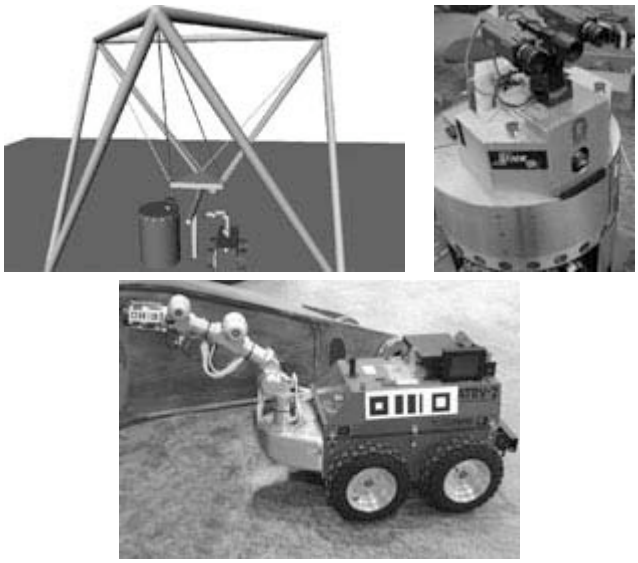


Figure 1. Experimental testbed consisting of 6 DOF crane (top), mobile manipulator (middle), and roving eye (bottom)

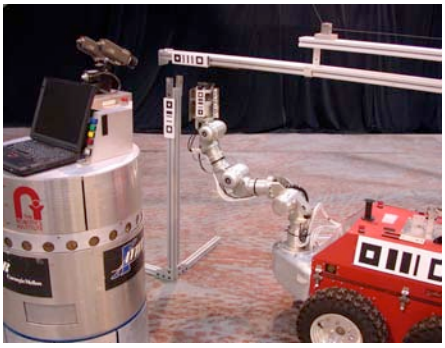


Figure 1. Tracking fiducials by the roving eye robot. Fiducials are mounted on the fixed structure, on the beam being emplaced, and on the mobile manipulator

As discussed above, sliding autonomy provides support at several different levels of operator interaction. These different levels require that each task be capable of functioning in both autonomous and tele-operated modes. For example, the leaf node that characterizes the mobile manipulator's dock-beam operation sometimes fails to dock the beam securely due to errors in the visual tracking. When the robot initiates human control due to this failure, possibly after some number of re-tries, the dock task must switch modes and accept control from the tele-operator (ignoring inputs from the visual tracking routine). Although the system is performing the same function, docking the beam, the task is accomplished differently in the autonomous and operator modes.



Figure 1. Steps in the assembly process: (top)Crane brings the beam close to the upright supports. (middle) Mobile manipulator grasps one end and docks it in one support. (bottom) The mobile manipulator turns around, drives to the second support, In this case, the mobile manipulator guides the beam in the horizontal direction while the crane lowers the beam into place.

Sliding autonomy has successfully been implemented in several parts of the system and all three modes of human interaction have been demonstrated:

- *pre-assigned tasks*: driving the mobile robot around in our construction site is not an easy task to automate so our system hands over control to the operator to roughly bring the mobile manipulator into a position from which the roving eye can track it. As soon as the mobile manipulator is in view of the roving eye, it automatically starts tracking and the operator is able to turn control back to the operator.
- *human intervention*: When the roving eye loses one of the fiducials it goes into an exhaustive search mode scanning its entire environment for the fiducial. The operator can interrupt this search and take over control of the camera and point it to towards the general direction of the fiducial. Once again, as soon as the fiducial is found, autonomous operation can resume.

- *failure recovery*: In those cases where the robot is stuck and can't proceed after having tried the preplanned contingencies, it gives up and asks for help. We have tested this case by forcing the system to fail in docking by blocking the docking clamps. Once the mobile manipulator has failed three times, it turns the system over to the operator under tele-operated control.

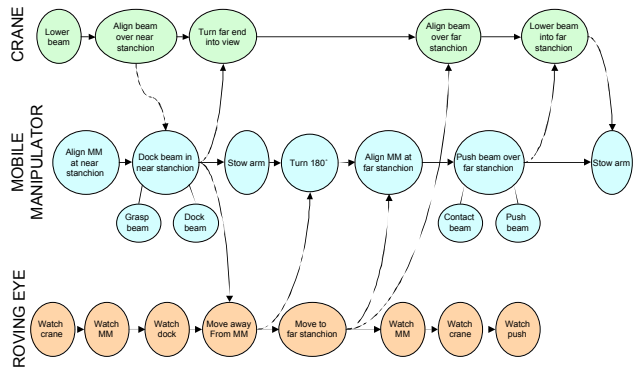


Figure 2. Task tree that specifies coordination across multiple robots to perform assembly of a beam.

An advantage of sliding autonomy is the ability to have the autonomous system monitor the operator and, conversely, to have the operator monitor the autonomous system. Consider the code fragment suggested for a visual servoing operation (Figure 6). If the operator took control of such a task, he would be responsible for both moving toward the goal position and determining arrival at the destination (within some tolerance)

monitor

```
error = desired_position
      – current_position
if (error < tolerance)
    then done = true;
```

action

```
error = desired_position
      – current_position
move(error)
```

Figure 3. Visual servoing operations use feedback from the Roving Eye to move closer to the goal pose. If the goal is within a certain tolerance distance, the goal is considered satisfied

To develop a mode whereby the autonomous system could monitor an operator, we have identified a distinction between monitor and action elements. A monitor task is responsible for determining when a goal criterion has been satisfied. An action task is responsible for moving the system closer to some goal state. The separation of monitor and action components for visual servoing is shown in Figure 7. The monitor block is responsible for determining when the position of the end-effector is within some tolerance distance of the goal. The action block is responsible for moving the arm closer to the goal.

```
error = desired_position - current_position
     if (error < tolerance)
        then done=true
        else move(error)
```

Figure 4. Separating the visual servoing operation into monitor and action components allows a much finer variety of sliding autonomy. A monitor block is responsible for determining when the goal is achieved. An action block is responsible for progressing the system closer to the goal.

When a task is separated into action and monitor components the operator is able to take over either component while the system continues with the other, enabling a mode of shared control. In general, direct control of several robots simultaneously while attempting to achieve some related goal places a significant burden on the operator. For example, during the dock of the second end of the beam, the Crane and Mobile Manipulator must work concurrently to position the beam. The second end dock is a task that is subject to failure and benefits from the addition of sliding autonomy, but requiring the operator to servo the Mobile Manipulator and Crane together is unacceptable.

Our solution is such that when the operator moves the Mobile Manipulator, the Crane follows accordingly. An ideal implementation would slave the Mobile Manipulator and Crane to a third input where the operator specifies beam's movements directly. We suggest that, in general, the goal during teleoperation is to concentrate the operator's interaction toward satisfying the goals (e.g., beam placement) rather than achieving indirect sub-goals (e.g., robot positions).

II. RESULTS

To validate our hypothesis that sliding autonomy will increase a system's overall efficiency and performance, we have compared fully autonomous, sliding autonomous, and tele-operated versions of the system. Efficiency and performance are quantified by the number of successful completions and the time needed to complete the task. The assembly task has been executed 50 times in each of the control modes.

For the autonomous trials, the system performed the assembly task as described in Figure 4 without any sort of operator interaction (aside from initialization). For the sliding autonomy trials, operators were allowed to intervene and perform a fixed set of tasks. These tasks included grasping, pushing, and docking the beam. The operator could also control the roving eye's visual search and take control of the mobile manipulator at several other times. During these experiments there were no pre-assigned operator tasks. When an operator intervened during a trial that would have been successful regardless (intervention may have served to accelerate success), we term that discretionary intervention. Intervention during trials that would have otherwise failed is termed mandatory intervention. Finally, for the tele-operated trials, four operators each performed fifteen iterations of the assembly. The primary input to the operator was the roving eye's video stream (i.e., the operators could not directly see the

system). The operator output control to each robot using several simple interfaces and a six DOF "space mouse." All operators were familiar with the system, but the skill levels differed (one skilled, two intermediate, and one novice). Each operator was allowed to perform several practice runs and the extreme performances were discarded for a total of 50 trials.

The failures occurred roughly evenly across the near end, swap ends, and far end segments of the experiment. For example, on very few occasions there was an electrical failure on the mobile manipulator that prevented movement. Errors in the visual servoing nearly caused a collision between two of the robots. A handful of times, a portion of the assembled structure broke apart and prevented further assembly, the roving eye irrecoverably lost sight of some fiducial, or some autonomous task failed to start properly. The nature and variety of errors does not clearly suggest a small or easy set of autonomous fixes. Yet, from the standpoint of a passive observer, the majority of these cases had a clear recovery solution. Considering the experiment where two robots nearly collided, the proper strategy would be to recognize the close proximity, stop before the collision, and back up one of the robots. Averting this situation autonomously might require increased visual accuracy, better motion control, or more complex obstacle avoidance. Although there is undoubtedly the potential to remedy this kind of error autonomously, for some systems the high cost of implementation seems to out weigh the 2% improvement in success rate. The timing results are described in Table 1.

TABLE 1: COMPARISON OF SUCCESS AND TIMING CHARACTERISTICS FOR TRIALS. MEAN ARE STANDARD DEVIATION ARE CALCULATED BASED ONLY ON SUCCESSFUL TRIALS.

	Successes (%)	***Mean Completion Time (min)***	***Std. Completion Time (min)***
Fully Autonomous	64	10	1.5
Sliding Autonomy	94	10	2
Discretionary Only	68	9.5	1.5
Mandatory Only	26	11	2.5
Tele-Operated	96	12.5	4

During the sliding autonomy trials, operator failure handling was never employed (in all cases the operator intervened before exceptions were generated). The overall success rate jumped to 94%, with only three failed experiments. The three failed experiments involved damaged hardware (proximity switches necessary for grasping were damaged) and two cases where erratic mobile manipulator movement (possibly the result of network failure) required a software reset. Strictly speaking, the erratic movements were recoverable with operator intervention, but we chose to power cycle the equipment for safety reasons.

The nature of failures– collisions, misalignments, and visual tracking errors– already suggests opportunities where human intervention can increase the success rate. Performance during the sliding autonomy experiments will also depend on the operator's attention to the system and the operator's interface [7]. As the workload of an operator increases (e.g., the operator is distracted) or the complexity of the world increases (e.g. the user interface is insufficient), the contribution of the operator will decline. To mitigate these effects in our experiments, monitoring and controlling the system will be our operator's only task. Furthermore, the operator interfaces will be tailored for each tele-operated task and will remain constant throughout the experiments. It is further interesting to note that sliding autonomy serves to both limit the workload of the operator (by employing operator assistance only when necessary) and reduce the complexity of the system (by focusing operator control to specific aspects of the system). Eventually, we propose to examine the quality of the operator's interface, perhaps as a function of information provided and demanded, with respect to the productivity of the operator. For example, a real-time video display may help navigate a vehicle around obstacles, but also requires a higher data rate and reduces tolerance to communication delays.

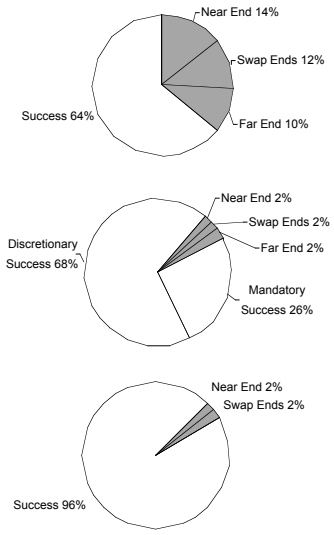


Figure 5. Success and failures from fully autonomous (top), sliding autonomous (middle), and tele-operated (bottom) trials. "Discretionary Success" trials are sliding autonomy trials that were successful with discretionary intervention. "Mandatory Success" trials were sliding autonomy trials that were successful only with mandatory intervention. Shaded wedges represent failed trials; empty wedges represent successful trails.

For the sliding autonomy trials, the average completion time was similar to that of the fully autonomous system, but the standard deviation was higher (see Table 1). On close consideration, however, this average and standard deviation are not directly comparable to those of the autonomous runs. The operator is now able to recover from disastrous runs that would not have weighed into the autonomous averages. Indeed, the successful trials must be subdivided into two groups: discretionary intervention and mandatory intervention. Of the original 50 runs, 68% were successful with only discretionary intervention – a number that compares with the 64% autonomously successful rate. The small decrease in average time (see Table 1) over the autonomous runs is the result of the operator intervening to quicken autonomous tasks. One common point of

intervention was manual completion of the autonomous visual search. Of the original 50 runs, 26% were successful only with mandatory intervention with an average of 11 minutes and a standard deviation of over 2.5 minutes. The large standard deviation is representative of the difference in the various errors. When the operator simply had to move the roving eye to bring a fiducial back into view, the time penalty was small. When the operator had to reposition robots to reconstruct parts of the assembly that had fallen apart, the time penalty was much larger. From these results, it is clear that there is a substantial increase in successful runs and a small, but noticeable, time decrease in the discretionary intervention trials. The success of the mandatory trials accounts for the increase in successful runs, but the completion of those tasks placed a heavy load on the operator.

III. CONCLUSIONS

Systems that rely solely on autonomy suffer from unexpected complications and excessive complexity. Tele-operated systems suffer from latency, bandwidth, and human limitations. Our goal is to develop an architectural framework that allows an operator to meaningfully and seamlessly participate in control of a multi-robot system. Our experiments todate show that sliding autonomy increases reliability over a completely autonomous system and task completion times are reduced over a manually operated system. While we have not measured cognitive loading on the operator, we believe that sliding autonomy has the prospect of significantly reducing the load on an operator. In the future, we intend to quantify the benefits of sliding autonomy in the assembly of a more complicated structure using a larger set of criteria. Another area we are examining has to do with representations of data presented to the operator.

ACKNOWLEDGMENT

The authors acknowledge the help of Brad Hamner, Brennan Sellner and Myung Hwangbo in getting the team of robots operational. Thanks to Metrica for the use of the robot manipulator.

REFRENCES

[1] R. Arkin, 1992 Cooperation without Communication: Multiagent Schema-Based Robot Navigation, Journal of Robotic Systems, 9:3, pp. 351-364.

[2] Bruemmer, D. J.; Dudenhoeffer, D. D.; Marble, J. Dynamic Autonomy for Urban Search and Rescue. AAAI Mobile Robot Workshop, Edmonton, Canada, August 2002.

[3] L. Chaimowicz, T. Sugar, V. Kumar, and M. Campos, An Architecture for Tightly Coupled Multi-Robot Cooperation, in Proceedings International Conference on Robotics and Automation, Seoul, Korea, May 21-26. 2001

[4] B. Donald, Distributed robotic manipulation: experiments in minimalism, In Proc. International Symposium on Experimental Robotics (ISER), Springer-Verlag, pp 11-25, 1995

[5] Dorais, G.; Bonasso, R.; Kortenkamp, D.; Pell, P.; and Schreckenghost, D. 1998. Adjustable autonomy for human-centered autonomous systems on mars. Presented at the Mars Society Conference.

[6] Doty, K. and Van Aken, R. Swarm robot materials handling paradigm for a manufacturing workcell. In IEEE ICRA. Volume 1, pages 778-782, 1993.

[7] Goodrich, M.A.; Crandall, J.W.; and Stimpson, J. L. Neglect Tolerant Teaming: Issues and Dilemmas. In proceedings of the 2003 AAAI Spring Symposium on Human Interaction with Autonomous Systems in Complex Environments.

[8] Terrence Fong, Nathalie Cabrol, Charles Thorpe, and Charles Baur, A Personal User Interface for Collaborative Human-Robot Exploration, In Proc. International Symposium on Artificial Intelligence, Robotics, and Automation in Space, June 2001, Montreal, Canada

[9] Fong, T.; Thorpe, C.; and Baur, C. Robot, Asker of Questions. In Robotics and Autonomous Systems. Vol. 42, 2003.

[10] Goldberg, K. and Chen, B. Collaborative control of robot motion: Robustness to error. In International Conference on Intelligent Robots and Systems (IROS). 2001.

[11] Goodrich, M.; Olsen, D.; Crandall, J.; and Palmer, T. Experiments in adjustable autonomy. In Proceedings of IJCAI Workshop on Autonomy, Delegation and Control: Interacting with Intelligent Agents. 2001.

[12] Goodrich, M.; Crandall, J.W.; and Stimpson, J. L. Neglect Tolerant Teaming: Issues and Dilemmas. In proceedings of the 2003 AAAI Spring Symposium on Human Interaction with Autonomous Systems in Complex Environments.

[13] Martin Hägele, Walter Schaaf, Evert Helms, Robot Assistants at Manual Workplaces: Effective Co-operation and Safety Aspects

[14] R. Brent Gillespie, J. Edward Colgate, Michael Peshkin, A General Framework for Cobot Control, 1999 International Conference on Robotics and Automation. Detroit MI.

[15] Hägele, M.; Schaaf, W.; and Helms, E. Robot Assistants at Manual Workplaces: Effective Co-operation and Safety Aspects. Proceedings of the 33rd ISRR (International Symposium on Robotics). October 7-11, 2002.

[16] D. Hershberger, R. Burridge, D. Kortenkamp, R. Simmons, Distributed Visual Servoing with a Roving Eye, In Proc. Conference on Intelligent Robots and Systems (IROS), Takamatsu Japan, October.

[17] J. Jennings, C. Kirkwood-Watts, 1998 Distributed mobile robotics by the method of dynamic teams, In Proc. Conference on Distributed Autonomous Robot Systems (DARS).

[18] O. Khatib, 1995 Force Strategies for Cooperative Tasks in Multiple Mobile Manipulation Systems, In Proc. International Symposium of Robotics Research, Munich, October.

[19] Kortenkamp, D.; Burridge, R.; Bonasso, P.; Schrenkenghoist, D.; and Hudson, M. An intelligent software architecture for semi-autonomous robot control. In Autonomy Control Software Workshop, Autonomous Agents 99, 1999.

[20] L. Parker, 1998 ALLIANCE: An Architecture for Fault Tolerant Multirobot Cooperation, IEEE Transactions on Robotics and Automation, 14:2, pp. 220-240, April.

[21] M. Mataric, 1992 Distributed Approaches to Behavior Control, In Proc. SPIE Sensor Fusion V, pp. 373-382.

[22] R. Simmons, 1994 Structured Control for Autonomous Robots, IEEE Transactions on Robotics and Automation, 10:1, pp 34-43, February.

[23] Scerri, P. and Pynadath, D. Towards Adjustable Autonomy for the Real World (2002).

[24] R. Simmons and D. Apfelbaum, 1998 A Task Description Language for Robot Control, In Proc. Conference on Intelligent Robotics and Systems, Vancouver Canada, October.

[25] R. Simmons, S. Singh, D. Hershberger, J. Ramos, T. Smith, First Results in the Coordination of Heterogeneous Robots for Large-Scale Assembly, In Proceedings, International Symposium on Experimental Robotics, Dec 2000, Hawaii.

[26] Wannasuphoprasit, W., Akella, P., Peshkin, M., Colgate, J.E., Cobots: A Novel Material Handling Technology (best paper award) International Mechanical Engineering Congress and Exposition, Anaheim, ASME 98-WA/MH-2, 1998.