Mission Accomplished: Sliding Autonomy for Team Fault Tolerance

Fang Tang, Ethan Ito, and Ian Stodart
Computer Science Department
California State Polytechnic University, Pomona
Email: {ftang|ehito|icstodart}@cpp.edu

Abstract— **This paper describes the application of sliding autonomy to robotic control such that the robot team can accomplish the mission under many new or unexpected situations. Our approach extends the traditional sliding autonomy by including a new mode that support human-robot peer-to-peer collaboration. We validated our approach on physical robots through a simplified search and rescue task, and demonstrated that the human-robot team's overall performance can be improved under the sliding autonomy control.**

I. Introduction

Multi-robot systems are widely used in today's robotic applications because of their advantages over single robot systems [1], such as improvements in robustness, reliability and efficiency. With similar reasons, we can also extend a multi-robot system to a multi-agent system, where the human operators can also be a team member. Increased autonomy will reduce human intervention, thus reduce the operating cost and increase human safety. However, challenges also arise with this type of complex system, such as mechanisms for coordinating team members, fault tolerance of the team and the operator control of the team. While researchers strive to build a fully autonomous system to perform various tasks, the robot team inevitably faces many unforeseen circumstances in an open and dynamic world. The team can either adapt to the dynamics through some life-long learning process, or by seeking help from the more competent human operator. Our goal is to develop a human-robot team with collaborative robots assisting the human operator to accomplish tasks while adapting to new or unexpected situations with the help from the human operator.

As a motivating example, consider a search and rescue task after a disaster. A specific area (map given but with discrepancy due to the impact of a disaster) needs to be explored and victim locations can be identified. Multiple robots can be assigned to search different areas of the environment. A robot with a laser range sensor can localize itself in the environment and uses its camera to detect any object of interest. However, when these sensors are distributed on multiple robots, these robots will need to work together to accomplish the job. Whether a subteam will be formed or not is highly dependent on the sensing and computational capabilities of the team members. We expect that robots can accomplish the tasks autonomously most of the time. However, human operators may be needed to help the robots in special situations. For example, a robot has a degraded performance in localization since the environment is not consistent with the existing map. Additionally, a robot cannot handle special conditions when its sensor is obstructed by a piece of cloth, or its wheels get stuck, which are quite common on a search and rescue site.

Our approach to the above problem is to apply sliding autonomy to control the multi-robot team. Our sliding autonomy approach features levels of control from fully autonomous operations, to human-intervened operations and pure teleoperation. Additionally, there is a great potential for human and robots to work together side by side and for each team member to contribute to the task objective based on their capabilities. Thus, we introduce a peer-to-peer interaction mode for the human operator to work closely with the robot team as peers rather than just as supervisors. Our sliding autonomy control interface allows human operators to monitor the task execution status, intervene to improve efficiency and react to unforeseen issues. It also enables the system to seamlessly switch between different levels of control. It helps establish the interaction between humans and robots by allowing them to influence each other's action selection and decision making.

The goal of this research is to build an operator interface that enables interoperability between multiple robots and a single human operator without putting a heavy load on the operator. The major contribution of this work lies in the following areas: 1) the introduction of peer-to-peer interaction mode to sliding autonomy; 2) the development of a task allocation framework involving human operators; and 3) the physical experiment to demonstrate peer-to-peer interaction under sliding autonomy. We validated our proposed sliding autonomy approach on a search and rescue task with physical robots. We demonstrated that our system can easily switch between different levels of controls. The overall team performance can be maintained across different situations.

The remainder of the paper is organized as follows. We discuss the related work in Section II and describe the details of our approach in Section III. In Section IV, experiments are performed to validate our approach. We finally conclude our work in Section V.

II. Related Work

Sliding autonomy has been widely used in controlling a single robot. It was also referred to as adjustable autonomy [4] and mixed initiative teaming [5] in earlier work. The paper in [4] describes the potential use of adjustable autonomy in space missions. The motivation of peer-to-peer interaction rather than supervisory interaction has been presented in [5]. In some cases, th system is composed of discrete autonomy levels, varying from pure teleoperation to full autonomy of robots. In other cases, the system is composed of a sliding autonomy levels which can be decided by varying a set of parameters. As a example, the work in [6] describes how human inputs and robot inputs can be combined to generate a dynamic mode between the two extreme modes of autonomy and teleoperation.

More research works started applying sliding autonomy to multiple robots. According to [7], when a human operator needs to control a multi-robot team, it is more efficient if robots have coordinated control rather than manual control. They used the USARSim for their experiments. The work in [8] performed four experiments to compare the sequential management style with the playbook management style when controlling multiple robots and showed that adjustable autonomy can improve the performance of both styles.

Our work is based on the sliding autonomy approach introduced in [9]. This research work breaks the sliding control into four modes of pure autonomy, mixed-initiative, system-initiative and teleoperation. They explored the issues of how to apply sliding autonomy to a multi-robot team by enabling the robots to request for help, bringing extra situation awareness and maintaining coordination of the team. They applied their control approach to a multi-robot assembly problem, which requires coordination among heterogeneous robots. Different from this work, our work introduces a new component of peer-to-peer interaction mode.

The work in [10] also introduced peer-to-peer human-robot interaction into the sliding autonomy. Based on the work in [9] and [11], they identified six capabilities for enabling sliding autonomy on a treasure hunt task. Their experiments showed that sliding autonomy can improve team performance by allowing different team configurations (including the human operator) to accomplish the same task. Our work also addresses the similar issue but with a different perspective. Particularly, our approach enables a human operator to participate in the task allocation process that will facilitate the operator to proactively help the robot when available.

III. THE APPROACH

Our system consists of four main components, as shown in Figure 1: an operator control interface on a desktop for remote monitoring and the sliding autonomy of operation control, an operator control interface on a mobile device for a close interaction with the robot on the site, a task auctioneer that enables the operator to enter tasks and assign them to the appropriate subteam, and a robot controller that supports sensing, acting and bidding for the robot.

A. Operator Control Interface

The operator control interface allows an operator and robots to communicate explicitly. Human operators and robots exchange information depending on different levels

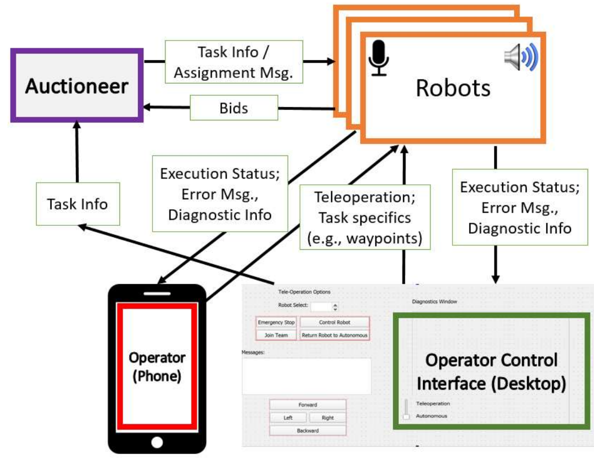


Fig. 1: The four components of our system with the flow of information.

of interaction between them. Data communication is bi-directional. At a high level, an operator can assign tasks to robots with task specification such as defining the goal position for the robots to reach. Robots inform operators of their current task-execution status, for example, informing the operator the completion of its current task. At a low level, the operator may teleoperate robots through direct commands, or exchange sensing or computational information with robot as needed.

Additionally, robots inform the operator of their possible failures so that the operator can assist them in assessing or recovering from the failures. Based on the flow of information that are required to accomplish a task, humans and robots can communicate with each other under different modes. For example, an operator can help a robot find its way home by teleoperating it with direct commands or giving the relative goal position to the robot.

The operator control interface also supports task allocation by passing the high-level tasks to our task auctioneer, which then makes appropriate mapping between tasks and agents (both robots and the operator). To support our search and recuse task, the interface displays agent locations, waypoints on a path, and a map of the environment. The operator can request to get a visual representation of the sensor readings in order to diagnose any potential problems.

We are also in the process of developing a control interface on a portable mobile device in order to facilitate peer-to-peer human-robot interaction. We anticipate that an operator can carry the portable interface to the work site, which allows an operator to directly send/receive further information to/from the robot, such as sending teleoperation commands or receiving sensing data.

B. Sliding Autonomy

Our sliding autonomy component resides in the operator control interface, which implements different levels of interactions to enable the human operator and the robot team

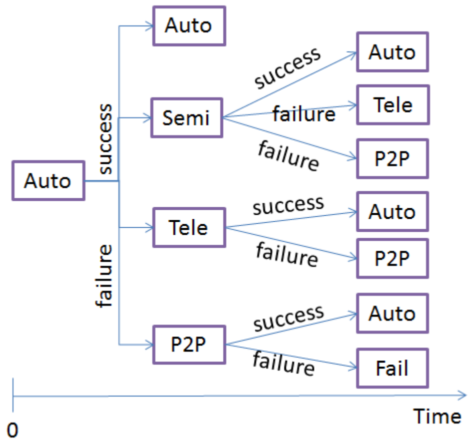


Fig. 2: Sliding Autonomy of Controls. Here, failure can be triggered either by the robot or the operator.

to accomplish a task in a collaborative manner. Under sliding autonomy, the system can dynamically switch between autonomy, semi-autonomy, teleoperation, and peer-to-peer interaction modes depending on the situation. The different levels or modes represent an increasing level of human involvement in task execution. By default, all robots start in an autonomous mode. Full autonomy is most desirable when the robots have the capabilities to handle the task with efficiency. However, we also recognize that robots work in a dynamic world with unforeseen uncertainties, which cannot be easily handled with an autonomous solution. Thus, human operators can assist the operation of robots at critical times when robots face irresolvable issues. They can change the mode of operations at anytime when an autonomous mode has a degraded performance or fails, see Figure 2.

The operator gets to select the appropriate level of control depending on his/her work load and the status of task execution. Our goal is not to overload the operator's job; especially when a single operator needs to attend to the entire multi-robot team rather than a few robots. In a semi-autonomous mode, a human operator is typically expected to share critical information with the robots (e.g., a new way point) or provide minimum guidance or assessment on the current situation (e.g. the camera view is blocked). Thus it only requires bounded levels of human interaction but not full attention. The human operator can still multi-task while helping the individual robot. The teleoperation mode provides more precise control by constantly sending direct commands and controls, it however requires operator's full attention. There is also the challenge of providing enough situation awareness.

For cases that robots cannot handle solely by themselves even with the shared information or direct control, the human operator will need to physically work with the robots as peers to accomplish the task. We include this peer-to-peer mode in our model, which is the mode that will require the operator's full attention and mechanisms for the peers to communicate and interact with each other. No matter what mode the robot is in, the system will resume in an autonomous mode after the problem is resolved. The core concept behind the sliding autonomy approach is to combine the control methods of both autonomy and teleoperation in a way that best highlights each control method's advantages.

To enable the switch between different operation modes, the human operator will oversee the execution of the entire team via a graphical interface on a base station. Each robot will also constantly monitor its own task execution. If a robot detects a certain error condition that it cannot recover from while under the autonomy mode, it can request help from a human operator to guide it to a condition the system can recover from. The human operator can also dismiss the request if he/she is currently busy with other higher priority jobs. Meanwhile, if the human operator observes a robot behaving incorrectly or inefficiently, the human operator can intervene the control of the robot to improve its behavior.

There are three different initiative models for incorporating human assistance: 1) the operator helps a robot when the help is requested. This is typical when the operator is currently busy with other errands and only responds when necessary; 2) the operator helps a robot when he/she detects that the help is needed; and 3) the operator helps a robot proactively whenever he/she can help. The latter two models are common when the operator is free of other errands and thus can intervene in order to improve the team performance. The ability of both the robot and the operator to issue a control switch helps the system respond to many types of situation.

With sliding autonomy, we expect that the robots can behave autonomously most of the time with occasional requests for cooperation at critical time, thus very few human operators and little operation time are expected. This type of system should solve the problems of full autonomy and teleoperation without the associated drawbacks of the two. With the addition of the peer-to-peer interaction mode, we augment the traditional sliding autonomy to handle more complex situations when the other modes do not meet the requirement.

C. Peer-to-Peer Interaction

When peers work together, there's usually both explicit and implicit communication between them. With implicit communication, the operator can observe the status of the task or the robots since they directly work together at the task site. However, this observation is often qualitative but not quantitative. Robots can also observe the status of the tasks or the actions of the operators through sensory feedback. Passive communication is always a challenging task for the robots. We rely on the environment to provide the media of communication. For example, a robot can keep running a self-diagnosis of its laser reading, when it detects any sudden change in its readings, it will alert the operator. In a highly collaborative task between robots and the operator, robots can also obtain sensory feedback through the environment by observing the task execution status, for example, the angle of a box when two agents manipulate the box.

Explicit communication is also a favorable approach when there's reliable communication and the bandwidth is abundant. With the proposed portable operator control interface, robots and an operator can exchange task relevant information with each other, such as locations or error messages. These quantitative data, when available, are always better for the robot to use. As for human operators, it is more natural for them to communicate with the robots through dialog rather than reading some encoded messages. We will also enable the robots to have speech synthesis and recognition capabilities.

D. Task Allocation

Task allocation is also an important component in our overall approach. The auctioneer receives high-level tasks from the operator and allocates the task one at a time to the appropriate agent(s). We implement a basic instantaneous task allocation approach based on the Contract Net Protocol [13]. Here is the process:

1) Task announcement: Initially, the human operator sends a task T to an Auctioneer agent built in the operator control interface, for examples reaching a series of way points. The auctioneer then announces each subtask (reaching one way point) to the agents. Each subtask t_i holds task specific information, such as the required capabilities and the goal location. In our search and rescue task, some example capabilities include exploration, point-to-point navigation and locating an object of interest.
2) Bid submission: Each idle agent submits a bid to the auctioneer, including its current distance to the goal, its capabilities and its corresponding cost.
3) Winner determination: Once bids are collected, the auctioneer then uses a greedy approach to determine the winning agent(s) for the current task and sends award messages to the agent(s). The winning agent(s) as a subteam should have the capabilities required by a task. Additionally, there are three other factors to consider. First, we favor subteams of smaller sizes. Second, we favor agents that are closer to the goal and thus ensure a faster completion time. Third, we favor subteams with a lower overall cost. These three factors are weighted to represent the preference of the operator. Any unsuccessful allocation will result in the subtask being reinserted back to the task queue.
4) Award acceptance: Winning agent(s) need to confirm with the auctioneer whether it will join the subteam to execute the task. In the case when award is rejected, the auctioneer selects the next subteam as determined in the winner determination process.

Note that the operator can also be one of the agents in the allocation process. The operator possesses a list of capabilities that he/she can subscribe at the beginning of the operation. The operator will always participate in task allocation. A bidder agent is embedded inside the operator control interface which does not require any interaction from the operator. Normally, an operator's capabilities are

Fig. 3: The iRobot Create robot.

associated with high costs, indicating the fact that it is costly to involve an operator in task execution. The costs can also be adjusted to show the willingness of the operator to participate. The motivation of involving humans in the task allocation process is to enable the operator to proactively help the robot whenever he/she can.

IV. Experiments

A. Experimental Setup

To validate our approach, we implemented the sliding autonomy control and applied it to a simplified search and rescue task with physical robots. In this task, a robot is given a map of the rescue site. It needs to travel to a specific location assigned by the human operator. The robot we used is an iRobot Create robot equipped with a Hokuyo laser range finder, and an Acer netbook equipped with a camera (see Figure 3). The laser range finder is used for localization in the given map. The camera can be used to locate a potential object of interest. We use Player as our robot controller [15].

To enhance situation awareness of the operator, data from two sources are visualized. On the main screen, we display the map of the environment, the robot's location and it's path (way points) to the goal. The data from the robot's laser sensor can be displayed in a separate window when requested. These two sources of information help the operator diagnose potential errors in the robot's localization and navigation process. For example, Figure 4 shows the interface when there's an error with the laser range data and the robot generated an alert to the operator. The operator then requested to see the laser data in a separate window (top right corner). The four modes of control are also shown in the interface. We designed the operator control interface using Qt [12].

We designed three test scenarios in order to show the sliding control of the robot:

1) The robot runs smoothly in an autonomous mode to complete the job.
2) The robot starts with an autonomous mode, but during operation it encounters an unexpected obstacle not specified on the existing map; in which case the operator detects the degraded performance in localization

TABLE I: Physical Experiments Results

Scenario	Operation Mode	Completion Time	Solution Quality	Workload
1. No Issue	Autonomous	68.6s	8 points	15
2. Unexpected Obstacle	Teleoperation	122.6s	8 points	28
3. Laser Error	Peer-to-Peer	104.6s	8 points	26

Fig. 4: The operator control interface used in our physical robot application. It displays the current robot coordinates and the goal coordinates together with the four modes of control. On the map, yellow grids are grids after obstacle growth. The dark blue dot represents the robot's current position. The green dot is one of its way points. The light blue dot represents the goal position. A small laser reading window (top right) can be opened on demand to diagnose the current situation. The small window on bottom right shows a partial local map which can be zoomed in/out.

and intervenes to change the robot's operation mode to the teleoperation mode.

3) The robot starts with an autonomous mode, but it encounters a laser measurement error during operation, which can not be resolved by the robot itself. The robot then alerts the operator. The operator diagnoses and resolves the issue on the site (peer-to-peer mode). Once the issue is resolved, the operator then changes the robot back to the autonomous mode.

To measure the performance of our sliding autonomy control, we collected data on the task completion time (in seconds), the solution quality and the operator workload. The solution quality is represented as a value with a maximum of 8 points. 3 points are given if the completion time is less than 2 minutes (given our testing environment) and 5 points are given if the robot reached the goal. Based on the amount of time that requires an active participation of the human operator, the operator workload is determined by having the operator fill out the NASA TLX (Task Load Index) Survey [14], with scores ranging between 1 and 100. The higher the number, the more demanding a task is to accomplish.The data in this area is very subjective, but still shows expected trends.

B. Experimental Results and Discussion

We ran each scenario 3 times and averaged the results. Table I showed the completion time, the solution quality and the operator workload for each scenario.

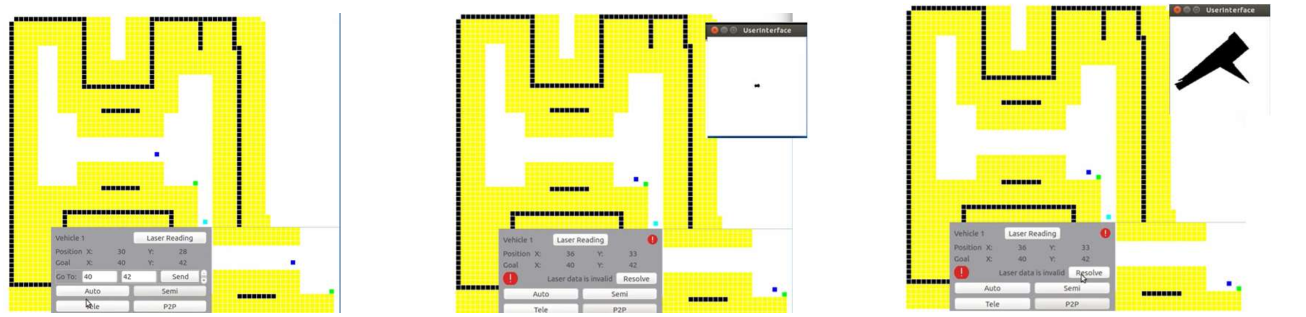
From the experimental results, we could see the solution quality is maintained across all scenarios, demonstrating the improvement of using sliding autonomy over the regular fully autonomous systems. A fully autonomous system may fail to localize in the second scenario and definitely will fail in the third scenario. The different levels of control allows the system to deal with degraded performance as well as faulty conditions. We showed that a single operator can flexibly switch between different modes to help the robots when necessary. By incorporating the peer-to-peer mode, we also showed that the team can improve its performance and even deal with situations that robots cannot handle by themselves.

When comparing the completion time, the teleoperation mode takes the longest time since the operator needs to use the limited visual clues (laser readings) to teleoperate the robot around obstacles. When the operator is not used to the directional orientation of the robot, the map or the interface, the teleoperation mode leads to a high workload. This motivate us to design better interface to enhance the situation awareness for the operator. The peer-to-peer mode also takes longer time and involves higher operator workload since it requires the operator's effort in diagnosing the potential error condition through the operator interface and resolving the issue on the site.

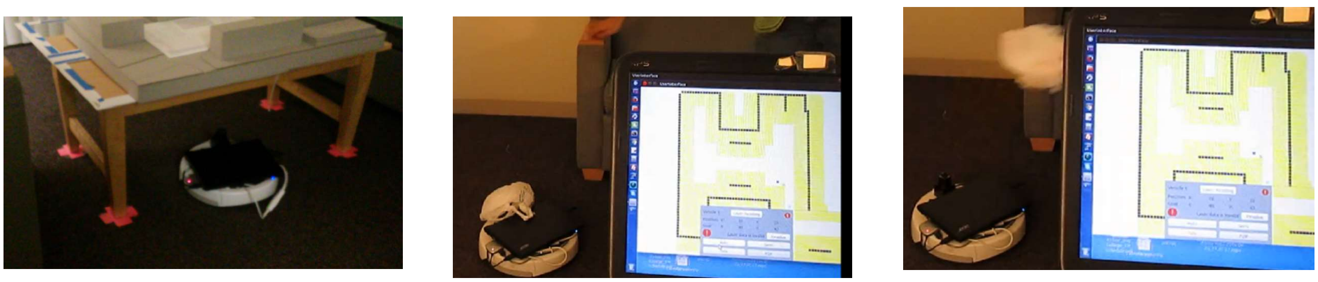
Figure 5 showed a typical run of the sliding autonomy control in the third scenario when there's a laser error. First, the operator entered the goal location through the semi-autonomous mode as shown in Figure 5d. The robot then started in the autonomous mode to navigate to the goal. In Figure 5e, the operator covered the laser with a plastic bag and the robot noticed its laser data error and alerted the operator. The operator selected to open the laser data window and diagnosed that there was a problem. The operator then decided to switch the robot's operation mode to peer-to-peer mode and resolved the issue by removing the bag from the laser (Figure 5f). The operator then changed the robot's mode back to the autonomous mode.

V. CONCLUSIONS AND FUTURE WORK

We have described our sliding autonomy approach and its application on a search and rescue task. The human operator



(a) The operator enters a goal way point in the semi-autonomous mode. (b) Robot alerts the operator with suspicious laser readings. (c) Operator changes to the Peer-to-Peer mode to resolve the issue.



(d) The beginning of the mission. (e) The laser was covered during operation. (f) The operator removed the cover.

Fig. 5: A sample run of the sliding autonomy control to resolve the laser range error.

can monitor the status of the execution through a graphical control interface and intervene the operation at anytime to improve performance. The robots can also initiate a request for the operator's participation. We validated our approach on physical robots and demonstrated that the robot can accomplish tasks in different scenarios with the flexibility in changing operation modes.

As a future work, we will test our current sliding autonomy approach with a complex search and rescue tasks involving using cameras to locate potential objects of interest and test the task allocation with various goal destinations. We would like to extend the peer-to-peer interaction mode to enable dialog and speech recognition between the operators and robots. We would like to enhance the current graphical control interface to provide more situation awareness, such as including video streaming. To better facilitate the peer-to-peer interaction mode, we plan to develop a control interface on a portable mobile device such that human operators can carry it while interacting with the robots.

References

[1] L. E. Parker. Distributed Intelligence: Overview of the Field and its Application in Multi-Robot Systems. Journal of Physical Agents, Special Issue on Multi-Robot Systems, Vol. 2, No. 2, 2008.

[2] M. Lewis, J. Wang, and S. Hughes. USARSim: Simulation for the Study of Human-Robot Interaction. Journal of Cognitive Engineering and Decision Making, 2007.

[3] B. Gerkey and M. J. Mataric. A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems. International Journal of Robotics Research, Vol. 23, No. 9, 2004.

[4] G. Dorais, R. Banasso, et al. Adjustable Autonomy for Human-Centered Autonomous Systems on Mars. In Proceedings of the Mars Society Conference, 1998.

[5] J. Marble, D. Bruemmer, et al. Evaluation of Supervisory vs. Peer-To-Peer Interaction with Human-Robot Teams. In Proceedings of the Hawaii International Conference on System Sciences, 2004.

[6] M. Desai and H. A. Yanco. Blending Human and Robot Inputs for Sliding Scale Autonomy. In Proceedings of the IEEE International Workshop on Robots and Human Interactive Communication, 2005.

[7] J. Wang, M. Lewis, and P. Scerri. Cooperating Robots For Search and Rescue. In Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, 2005.

[8] M. A. Goodrich, T. W. McLain, et al. Managing Autonomy in Robot Teams: Observations From Our Experiments. In Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction, 2007.

[9] B. Sellner, F. Heger, et al. Coordinated Multi-Agent Teams and Sliding Autonomy for Large-Scale Assembly. In Proceedings of the IEEE - Special Issue on Multi-Robot Systems, Vol. 94, No. 7, 2006.

[10] M. B. Dias, B. Kannan, et al. Sliding Autonomy for Peer-To-Peer Human-Robot Teams. In Proceedings of the 10th Intelligent Conference on Intelligent Autonomous Systems, 2008.

[11] D. J. Bummer and M. Walton. Collaborative Tools for Mixed Teams of Humans and Robots. In Proceedings of the Workshop on Multi-Robot Systems, 2003.

[12] Qt Project. www.qt-project.org.

[13] R. Smith. The Contract-Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. In Proceedings of the IEEE Transactions on Computers, 1980.

[14] S. G. Hart and L. E. Staveland. Development of the NASA-TLX (task load index): Results of Empirical and Theoretical Research. Human Mental Workload, 1988.

[15] B. Gerkey, R. Vaughan, et al. Most Valuable Player: A Robot Device Server for Distributed Control. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001.