

Exploratory Analysis of Operator:Robot ratio in Search and Rescue Missions

Alberto Valero¹ and Chiara Saracini² and Paloma de la Puente³
and Diego Rodríguez-Losada⁴ and Fernando Matía⁵

Abstract. Mobile robots are increasingly becoming an aid to humans in accomplishing dangerous tasks. Examples of such tasks include search and rescue missions, military missions, surveillance, scheduled operations (such as checking the reactor of a nuclear plant), and so forth. The advantage of using robots in such situations is that they accomplish high-risk task without exposing humans to danger: robots go where humans fear to tread. When dealing with robot teams for exploration missions, the human operator needs to control simultaneously multiple robots. This paradigm highly increases the difficulty of the operator task. In this paper, based on extensive experimental data, we are studying the problematic associated to the Single-Operator-Multiple-Robot Interaction. This analysis can help to ascertain the critical points in multi-robot control and supervision. We propose two autonomy management modes and analyse the optimal operator robot:ratio for each of them. The experiments have been done in a Search and Rescue scenario using USARSim.

1 Introduction

An HRI system should permit humans and remote robots to cooperate in order to accomplish cognitively-demanding tasks, working in the spectrum between a fully autonomous system and a fully tele-operated one. The HRI system should then improve the accomplishment of the task by benefiting from the capabilities of the artificial agent and the human. The increment of the degree of autonomy, and the robot-coordination techniques are conducting researchers to deploy teams of robots for exploration missions. This brings a new challenge to the Human-Robot-Interaction system: *how the information coming from robots distributed spatially needs to be integrated into a Graphical User Interface?* and, *how the robots can be controlled as singular entities and/or as a team?* The answers to these questions should permit one operator to be capable of controlling and/or supervising a team of robots deployed on an unknown environment. Consequently researchers are being encouraged to analyse, design, and test, multi-robot teams controlled by a single operator. Military missions, surveillance, scheduled operations, and so forth, are examples of application of this research.

In this paper we present the resulting analysis of two controlled experiments, in which a Search and Rescue Operation was simulated using USARSim. We modified the number of robots, from 1 to 4, for the first experiment, and from 1 to 3, for the second, and analysed the performance of the operators. The analysed data allows us to identify the critical points in multi-robot tele-operation. **The goal of this study is to set the basis for an adequate single-operator-multi-robot Interaction System design.**

2 Single-Robot vs. Multi-Robot for Exploration Missions

Developments in Graphical User Interfaces (GUIs) for tele-operating a mobile robot have mainly focused the single-robot paradigm. They are mostly concerned on how to provide the operator with the required Situational Awareness for controlling a remote mobile robot [18]. A multi-year study made in a rescue simulated competition [17], revealed poor interface designs. Yanco et al. designed and evaluated a video-centric interface [16], raising a great interest on the GUI design. The Human Robot Interaction INL research group⁶, designed a map centred interface [7], giving to the operator a first person point of view of the environment. Both interfaces were experimentally compared in [4]. The limitation of these two interfaces is that they are difficult to extend to the multi-robot paradigm without considerable re-design. In first mentioned GUI, the video information is tightly related with the robot localization, and thus unfeasible for managing a team of spread robots; the INL GUI, map centred, could be easier to extend, as the mapped environment is common to all robots, but it has been designed in order to keep the operator point of view mostly fixed over the robot, presenting consequently the same problem.

These works have set the basis for a good single-robot interface design. Now, the next step drives through multi-robot systems. There are many advantages of using multiple robots in complex exploration tasks like Search and Rescue: properly coordinated robots can cover faster an area than a single robot, the key problem to be solved is to coordinate the robots so that they simultaneously explore different regions of their environment [9][2]. SLAM can be benefited from the use of multiple-robot sensing by reducing the uncertainty associated to localization and map building [10][12]. To act effectively under uncertainty it is required to accurately estimate the state of the environment, providing information of the dynamically changing situation; individual robots, with uncertain sensors, may not be able to accurately determine the current situation, the team as a whole

¹ Centro de Automática y Robótica, Universidad Politécnica de Madrid, email: avalero@etsii.upm.es

² Lab. of Cognitive Science and Psychology, SAPIENZA - Università di Roma, Italy, email: chiara.saracini@uniroma1.it

³ Centro de Automática y Robótica, Universidad Politécnica de Madrid, email: paloma.delapuerta@upm.es

⁴ Centro de Automática y Robótica, Universidad Politécnica de Madrid, email: diego.rlosada@upm.es

⁵ Centro de Automática y Robótica, Universidad Politécnica de Madrid, email: fernando.matia@upm.es

⁶ <http://www.inl.gov/adaptiverobotics/>

can exchange information to perform a better situation assessment by verifying hypothesis.

From the operator point of view, the control and supervision of a team of robots brings a challenging novelty with many associated difficulties. It is not only a question of operator's available time. He does not only need "free" time to switch control from one robot to another, and supervise them sequentially, he also needs to process all the information coming from the deployed robots and integrate it in order to have a global Situation Awareness of the scenario and the robots [18]. He must be able to send the proper commands, not only to the single robots, but also to the team, in order to carry out a coordinated action, and so forth. The more the Interaction System enhances the operator in this task, the better he will perform.

Robots, considered as entities are individual sources of information, which cannot be "processed" by a human if not properly integrated. One operator cannot tele-operate simultaneously several robots. If he just switch control from one to another, the performance of the operation is highly reduced [3]. Adams et al. address this problem proposing not to consider the single robots, but the team as a whole entity, integrating the information from the team and sending team commands [15]. The limitations of this approach is that it presupposes that the robots are able to autonomously carry out the tasks assigned to them by the team coordination layer, what is not feasible in hazardous scenarios like Urban Search and Rescue. Goodrich et al., propose to dynamically adjust the autonomy level to keep the operator workload within tolerable ranges responding to environment and workload changes [6]. This would minimize the time required by each robot, permitting the operator to switch among them. The limitation is that the robots are not treated as a team. Burke and Murphy, based on field studies consider that an operator is not suitable for driving more than one robot simultaneously in *real* Search and Rescue Missions [1]. Scholtz et al. experimented in autonomous off-road driving, analysing the number of times that an UGV requires operator assistance to complete the task. Based on this analysis they conclude that one operator can only supervise two UGVs simultaneously in outdoor environments [11]. Fong et al. proposed a system in which the operator is just a supervisor, who is asked for help by the robot when his assistance is required [5]. Unfortunately no study of the scalability of Fong's system was made.

Based on experimental data, we analyse the case of multi-robot control in Search and Rescue Missions. This application scenario is more complex than the off-road autonomous navigation studied by Scholtz et al. as in their application the UGV should not deal with exploration. We are more flexible than Burke and Murphy, not applying all the hazardous conditions that a *real* mission would present. We have used as test-bed the NIST simulator: USARSim; the "emergency scenarios" are those used in the 2006 RoboCup Competition⁷.

3 Interface description

Our first interface prototype was inspired by the works of [16] and [7], which are probably the most evaluated designs in the literature for single robot GUIs. In addition, we analysed the multi-robot problem, and re-designed their GUIs to support the control and supervision of a team of robots. This interface was used in the 2008 RoboCup Competition, for controlling a team composed by three ground robots and an aerial vehicle. An experimental evaluation of the first prototype can be seen in [14].

The second prototype includes two displays:

The Complex Display (Fig. 1(a)): It includes allocentric and egocentric views of the scenario. It consists of three views: a *Local View of the Map*, that can be set to have the robot facing up, or to have the map north oriented; a *Global View of the Map* giving a bird's eye view of the explored area, and a *pseudo-3D View* giving a first person view, including the video feed-back on its real position in the environment.

The Team Display (Fig. 1(b)), which gives a bird's eye point of view of the explored area with the robots within, for an easier team control and supervision. The operator can zoom a particular area or see the whole explored area. He can switch from one robot to another by clicking on it, and set the target point clicking on the map. Additionally he can tag the map by adding information about the scenario. Victims found, snapshots taken, tagged areas, robots path, a-priori maps, and other retrieved information, can be visualized on the map.

The operator can control the robot in four modalities: *tele-operation*, *safe tele-operation*, *shared control* and *autonomy*. In the *safe tele-operation* mode the system prevents the robot from colliding with obstacles, limiting the speed according to the obstacles distance. In the *shared control* mode the operator sets a target point for the robot by directly clicking on the map, which the robot must try to reach. The *autonomy* policy depends on the mission. For Search and Rescue, our robot explores autonomously the area visiting the unknown places and taking pictures of detected victims [2].

4 Interface Evaluation

As we said at the beginning, an effective operation of multiple robots requires that the operator's time be distributed among the robots; this allows him to supervise the whole team and intervene whenever this is required. The experiment we are presenting here aimed to evaluate scalability of the mechanism of autonomy adjustment that we implemented. Furthermore, subjects were asked to fill out questionnaires as a way of measuring the usability of the interface for multiple-robot control and supervision.

4.1 Experiment Design and Procedure

Forty subjects (thirty-five males, five females) participated in the experiments. All of them were either master students, PhD students, or senior researchers in the field of engineering. None of them had had previous experience with the interface. The subjects were randomly divided into four groups. Each subject in a given group had to accomplish a Search and Rescue Mission while controlling a team of robots (*number of robots* $\in [1 - 4]$). Every subject went through a twenty-minute training program to acquire a basic knowledge of the functionalities provided by the interface. The training scenario was taken from the RoboCup Competition and was similar in difficulty to the scenarios used for the experiments.

The subjects were asked to run two experiments. In the first experiment, they had to explore an unknown office building (indoor scenario) in order to search for victims. They were given 15 minutes. After completing this experiment, they filled out a usability questionnaire. In the second experiment, they were asked to explore an unknown urban outdoor area (with the same number of robots), also with the aim of finding victims. In this paper we will only analyse the results concerning the indoor scenario. Figure 2 shows a screen-shot of the indoor scenario.

⁷ Available at <http://usarsim.sourceforge.net>

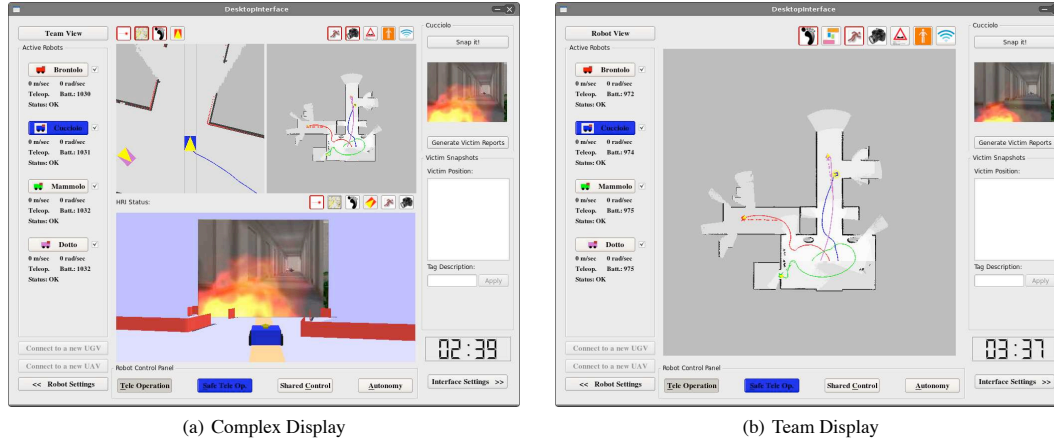


Figure 1: Desktop-interface.



Figure 2: Indoor Scenario used in the Experiments

4.2 Data analysis

The focus of our analysis is the influence of the factor *number of robots* $\in \{1, 2, 3, 4\}$. The variables studied are:

- The area explored divided by the total time of the mission.
- The percentage of the total time in which the robots were operating (moving) simultaneously.
- The relation between the time the robot remained stationary in *Shared Mode* and the total time in that mode;
- The relation between the time the robot remained stationary in *Tele-operation Mode* and the total time in that mode;

4.2.1 Area Explored

A measure of operator performance is the area covered by the team of robots. When an operator is controlling several robots simultaneously, he may neglect some of them, thus decreasing the time of simultaneous operation, as we have just seen. That said, this "low utilization" might be justified if the area explored by the robots is significantly large.

We made an Analysis of Variance (ANOVA) for the explored area⁸. The best level of the factor corresponds to one robot, but it is not significantly different from 2, 3 or 4 robots, because the intervals are overlapped. Figure 3(a) shows this result.

4.2.2 Simultaneous Operation

When controlling a team of robots, the operator must try to use all of the robots simultaneously. As he can tele-operate only one robot at a time, the rest of the robots have to be working at some level of autonomy: *Shared Control*, when a target point is set, or, when it is not, then *Full Autonomy*. In the best case scenario, all the robots composing the team should be operating (moving) at any given time. An effect of operator neglect on the system performance can be seen in [3]. The present study measures the amount of time that all the robots were moving simultaneously.

An ANOVA was run. The main effect of the *number of robots* is significant. Fig. 3(b) shows the main effects of *number of robots*⁹. The simultaneous operation is significantly greater when the operator is controlling one robot.

4.2.3 Stop Condition Analysis

We also found it useful to analyse the conditions in which a robot was neglected by the operator. A robot is neglected when the operator is not aware of the state of the robot. This may occur in two different situations: A robot has finished its task and so is waiting for a new command; a robot is stalled. As the number of robots increases, the operator controlling a team can lose "*Robot State Situation Awareness*" [4]. This lack of SA indicates an overloaded operator who can be expected to yield a worse performance.

The lack of SA can increase in severity depending on the level of autonomy on which the robot is working. Robot inoperability can occur in a variety of different situations:

⁸ Due to the the extension of this paper the detailed ANOVA data is not presented, as it has been considered more explanatory the figures of the confidence intervals. For detailed data please refer to [13]

⁹ Bonferroni adjustment has been applied to solve the multiple comparisons problem.

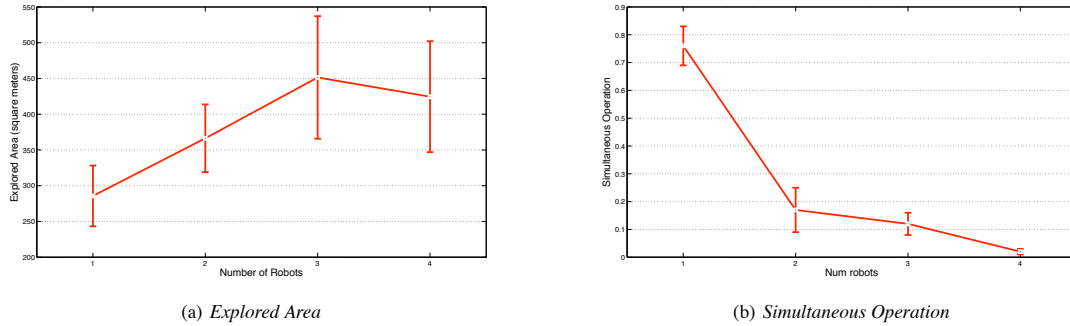


Figure 3: Means and 95% Bonferroni intervals.

Stop in Shared Mode We analysed the time the robots were stopped in *Shared Mode*. When the robot has been sent to a target point (*Shared Mode*), the robot may be stalled. Most usually, robot standstill occurs when the robot has already reached the destination point. The robot remains stationary as it awaits new commands. This situation indicates an even greater lack of SA than the one previously described. After all, the operator knows that the robot will navigate autonomously only until it arrives at the target point (supposing that it does in fact reach this point), and he should therefore be able to estimate the time required to carry out the given task. Figure 4(b) shows the main effect of *number of robots*. The analysed data is significantly smaller when operating one robot. In addition, there are no significant differences between 2, 3 and 4 robots, since the intervals are not overlapped.

Stop in Tele-Operation The greatest lack of SA is manifested when a robot is left standing still in *Tele-operation Mode*. We analysed the $\log(Y)$. The *number of robots* is a significant factor. In Figure 4.2.3, the main effect of the *number of robots* is shown. The best performance in Tele-Operation is with one robot.

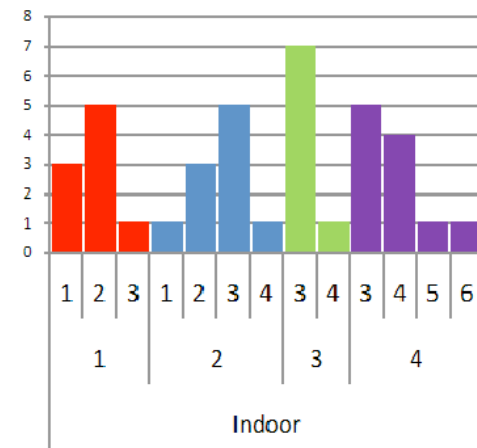
4.3 Questionnaires

After completing the task the subjects were asked to fill a questionnaire. We will not present all the collected data but only the subjective perception of the dimension of the team. That is, subjects were asked how many robots they think they can control in order to maximize the performance. This data is represented in Figure 5(a). Subjects scored each operation mode (1-worst, 5-best). This data is represented in Figure 5(b)

4.4 Discussion

4.4.1 Interface Design and Operation Modes

A first observation is that for one robot, the subjects almost always used the *Tele-operation Mode*. This suggests that the interface provides enough information to an operator to enable him to control a robot remotely and to manoeuvre it in a precise and secure way. If the operators' situational awareness had been insufficient for this task, the resulting feeling of lack of control, or of the inability to send precise commands, would have led them to prefer *Shared Mode* instead. In fact, a few subjects that showed small capabilities to drive



(a) Preferred Number of Robots. X Axis: Number of Robots Controlled. Y Axis: Preferred Number of Robots

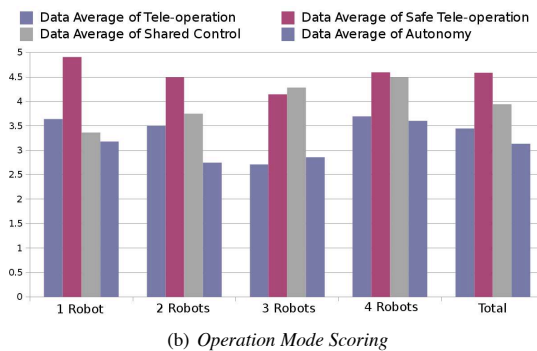


Figure 5: Questionnaires Analysis

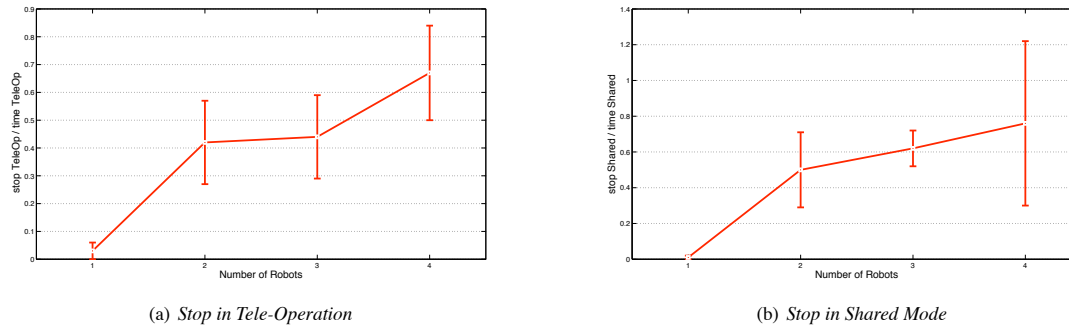


Figure 4: Means and 95% Bonferroni intervals.

the robot, used mostly the Shared Mode, commanding through target points.

Analysing the scoring the operator gave to the Operation Modes (Figure 5(b)) we conclude that the Tele-operation mode is the less appreciated. Operators considered this mode very dangerous, as collisions were not prevented at all. Instead the Safe Tele-operation Mode received a good scoring, which showed that the changes applied to the second version of the interface were successful. The Autonomy Mode received also a low scoring.

4.4.2 Operator Performance

The data analysis concludes that the optimal robot:human ratio, when account is taken of the utilization of robots and of the area explored, is 1:1. When we consider the utilization of the robots, this result is not surprising. When an operator controls more than one robot, he inevitably neglects some to a certain degree, which results in correspondingly less optimal utilization.

The use of several robots would be justified, even when they are not fully exploited, if the area explored increased with the number of robots. We had hypothesized that the area would expand to its with 2 or 3 robots, and would subsequently decrease due to an excessive operator workload. In actuality, the result of the experiments shows that the area covered by the robot team increases only when we use 3 or 4 robots, but its not better with 2 robots in relation with one. We believe that this result is caused by the great decrease in robot utilisation due to the shift from one robot to two. As we can see in Figure 3(b), when the operator is controlling two robots, they are moving simultaneously less than the half of the time. Thus, even if both robots explore different areas, the total area is not bigger than when an operator controls just one robot.

The *Stop Condition Analysis* helps us to understand why this performance fall-off occurs when the operator is using more than one robot. Let us suppose that the robot is working in *Shared Mode* (in other words, when the operator has set a target point that the robot must try to reach autonomously). If the operator is controlling multiple robots, he usually does not realize immediately that the particular robot has arrived at the target point. When the operator is controlling one robot, however, this does not happen, as he maintains supervision of the robot's actions during the entire task. Again, the problem arises only when he is controlling two or more robots. This is consistent with the usability questionnaires. Most users said that they felt

unable to keep track of the state and task of all the robots composing the team.

Something slightly different happened when the robot was left stationary in *Tele-operation Mode*. Direct observation of the experimental runs revealed to us the reason why this situation happened. When a robot was moving in *Autonomy Mode* or *Shared Control* and got stalled (that is, became unable to follow its path), as soon as the operator realized this, he switched control to the stalled robot, leaving the robot he was controlling previously in whatever state it happened to be in at the time of the switch. For example, if this old robot was in Tele-operation, it was left stationary, and the operator usually forgot about it and simply continued to manoeuvre the the new robot until he realized that he had forgotten the old one. When we asked the operators if they were aware of their mistake in this regard, they typically answered that it was very difficult to keep track of everything they were doing on account of the complicated nature of the task they were performing.

Analysing the subjective perception that the subjects had of the optimal team dimension (Figure 5(a)) we observe that operators considered they were able to command a team of up to 3 robots in the indoor scenario and up to 2 for the outdoor scenario. This perception changes with the number of robots they effectively commanded. Operators commanding 4 robots considered that the number was excessive, operators controlling 3 robots felt the number was ok, while for 1 and 2 robots they thought they could control more robots. We conclude that at this point the interface was not yet well designed for supporting robot teams, as the operator "optimism" indicates that he is capable of more. We concluded that the improvement should be made the autonomy adjustment management.

5 Autonomy Management Evolution

The major weakness that appeared in the previous study concerns the excessive amount of time the robots were neglected when they were not performing a task, that is, when they had stopped and were waiting for incoming commands. Taking account of system performance and the test subjects' input, we hypothesized that the problem might be caused by one of the following two factors:

- The robotic system was unable to complete the task autonomously and no error message was sent to the operator.
- The operation modes were not enough adequate to the operator's task

All these issues led to the development of a new interface prototype 3, focused on enhancing the operation modes, particularly the Shared Mode, understood as the spectrum between full autonomy and the Safe Tele-operation. We will describe these improvements in this paper. After describing the interface, we will present a set of controlled experiments performed for the sake of analysing the improvements.

5.1 Operation Modes Evolution

In the previous version there were four independent operation modes, and the operator decided which one was more suitable to the task requirements. In version 3, the management of the autonomy level changed. The system no longer waits for a "precise command" for each operation mode: target point in Shared Control, speed and jog in Tele-Operation. In the third version, the autonomy is adjusted in a multi-layered form. The layered system may be seen in Figure 6. The following list describes the different autonomy levels, beginning with full autonomy and ending with pure tele-operation (based on [8]).

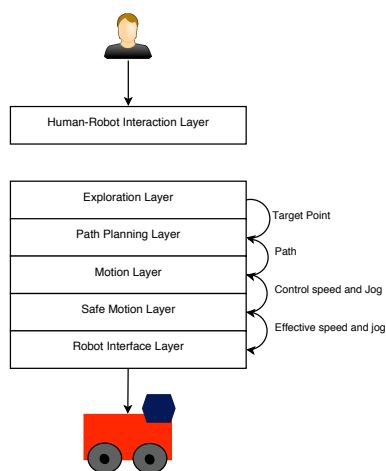


Figure 6: System Layers

1. The system proposes a target point;
2. if there is no input from the operator the system proposes a path to reach it, or
3. the operator sets a target point and the system proposes a path to reach it;
4. if the operator does not change the path the system follows the path, or
5. the operator sets a path and the system follows it;
6. the operator sets a control speed and jog and the system adjust the real speed and jog to avoid collisions;
7. the operator sets the real speed and jog.

From this list we can see that the operator has five command possibilities: set a target point (or sequence of target points), set a path, set a control speed and jog, set the real speed and jog. Depending on the operation mode of the robot: Tele-Operation, Safe Tele-Operation, Shared Control or Autonomy, the system will react differently to each input. The possibilities are:

Tele-Operation and Safe Tele-Operation. If the operator sets a target point or path, the robot changes to shared control.

Shared Control. In shared control mode, the system tries to reach the target point, or sequence of target points, set by the operator. This is done by means of two mechanisms: the calculation of a path and the control of robot motion to follow that path [2]. The operator can give the following commands.

- Speed Commands: The system remains in Shared Control, but while the operator continues sending speed commands, the robot's motion is not controlled by the system. When the operator stops sending commands, the system retakes motion control. This is very efficient when the robot is following a path and remains stalled, because the operator can get the robot out of its stalled condition without modifying its former task. As soon as the operator releases the control, the robot continues the previous task.
- The operator sets a path, the robot follows that path, and it then calculates a new path to the former target point, if there was one. In this way, if the operator disagrees with the path proposed by the system, he can modify it, and the robot adjusts its proposed path to the path set by the operator.

Autonomy. In full autonomy, the system chooses the most suitable target point in order to accomplish the mission [2]). The operator may send the following commands:

- Speed, Jog and Path, as in Shared Mode. The system gives the suitable control to the operator, but remains in autonomy, so that, when the operator stops commanding, the system keeps on working in autonomy. Like shared mode, this mode is suitable in situations in which the operator does not approve the actions taken by the system.
- The operator sets a target point or sequence of target points. The system tries to reach them, and, once they have all been reached, it keeps on exploring autonomously.
- The operator sets a desired path. The system tries to follow it, once it has completed the task, and if there are not target points, it keeps on exploring autonomously (if there is a desired path and a sequence of target points it will follow firstly the desired path, and afterwards it will try to reach the target points).

Each layer is in charge of performing an action. The more layers are working, the higher the autonomy level is. Thus, if all the layers are running, the system is working in full autonomy, while, if none of them is working, the operator has full control of the robot. The operator can command at every level of the layered system, substituting the actions of the higher layers. It is important to notice that this does not require him to change the operation mode. For example: The system may be operating in "Autonomy Mode," so that, by default, all the layers are running, but at any moment the operator can command at any level, while the system remains in "Autonomy Mode". At the same time, every layer provides feed-back indicating whether or not it has performed its respective task, so the operator has correct system state awareness and knows at which level he must act.

The new autonomy management is thought to provide the operator with two advantages:

- Fast recovery from navigation and exploration errors or bad performance. The operator can act at the error level, without re-configuring the robot task or changing the operation mode.

- Long-term commands and more granulated autonomy levels. The operator can set a sequence of target points, giving the robot a longer-term task. Moreover, since he can set the path, something that was not possible in previous version, he can send the robot along safe paths.

In this way, the system takes more advantage of the operator's expertise.

6 Evaluation

As we said at the beginning, an effective operation of multiple robots requires that an operator distribute his operating time among the robots, so that he can supervise the whole team and intervene whenever this is required. The experiment is equivalent to the previous one and aims to evaluate the scalability of the mechanism of autonomy adjustment that we implemented.

Forty-five subjects (forty-one males, four females) participated in the experiment. All of them were either master's students or Ph.D. candidates. The subjects were randomly divided into three groups. Each subject in a given group had to accomplish a Search and Rescue Mission while controlling a team of robots (*number of robots* $\in [1 - 3]$). Every subject went through a forty-minute training equivalent to the training performed for the previous experiment.

The participants were asked to explore an unknown office building (indoor scenario) in search of victims. This scenario was the same as that used in the previous experiment.

6.1 Data analysis

The focus of our analysis is the influence of the factor *number of robots* $\in \{1, 2, 3\}$. The variables studied were:

- The area explored divided by the total time of the mission.
- The percentage of the total time in which the robots are operating (moving) simultaneously in Shared Control Mode and in Tele-Operation.

6.1.1 Area Explored

The area explored was measured in square meters. The number of subjects in each group is 15. A one-way ANOVA was carried out to study if there were any significant differences. Confidence intervals can be seen in Figure 7

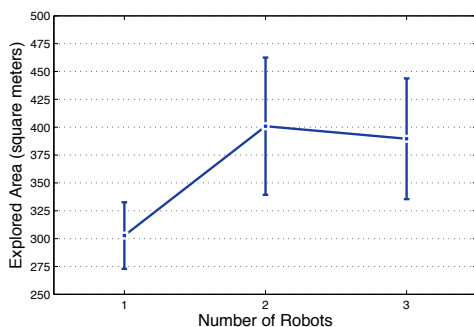


Figure 7: Explored Area - Confidence Intervals (95 %)

6.1.2 Stop Condition Analysis

The previous experiment showed that one of the reasons of a bad performance was that operator were neglecting some robots of the team. We hypothesized that this was caused by an insufficient autonomy management paradigm. We modified this paradigm in order to improve performance. This analysis, together with the previous one aims to analyse whether there is a significant improvement. We analysed stop condition in Tele-Operation and in Shared Mode. The results are displayed in the form of a graph in Figures 8(a) and 8(b).

6.2 Discussion

The major improvements made to the third version of the interface regarded the autonomy management and the new Shared Control commands. The results obtained for the previous version showed that the optimal number of robots for maximizing the area explored and minimizing the number of robots deployed was one. This experimental result led us to conclude that the interface did not provide the operator with suitable mechanisms for adjusting robot autonomy, which is crucial for the control of multiple robots. At the beginning of this paper, we presented a work of Scholtz et al., in which the authors conclude that the optimal operator ratio for off-road autonomous driving is 1:2. We set out to obtain a similar result for the current interface version. The analysis of the area explored reveals a considerable performance improvement over the previous version. While in the previous version increasing the number of robots did not lead to an increase in the area explored, in the current version the explored area is greater for 2 robots than for one robot. The differences are statistically significant. For 2 and 3 robots, there is no statistical difference (Figure 7).

The rest of the data studied enabled us to understand the key to this improvement. This phenomenon receives some explanation in light of the stop condition analysis (Section 6.1.2). In the previous version, there was a considerable jump in the stop condition between one robot and more than one robot, a jump that then entailed a corresponding significant loss of performance capability. In the data analysed above, this jump is mitigated, as can be seen in Figures 8(a) and 8(b). Furthermore, comparing the stop condition in Shared Mode in the previous interface with that of the current version, we see that the time the robots were stopped in Shared Mode is significantly less for 2 robots.

7 Conclusions

In this paper, we have evaluated the two different versions of our interface. This evaluation was presented on the basis of controlled user experiments.

The major change between the versions was the inclusion of new command capabilities for the Shared Mode as well as a different adjustable autonomy management. In version 2, the operator could set a target point only when working in Shared Control Mode. In version 3, the operator can set a sequence of target points or a path. Furthermore, each operation mode allows him to bypass any of its "proper" functionalities. For example, even if he is working in Autonomy Mode, the operator can set a target point, or temporarily control the speed and jog of the robot. These changes led to an improvement in the optimal operator robot:ratio. While in version 2 the performance of the operator decreased for more than one robot, in version 3 the operator achieves a better performance with 2 and 3 robots.

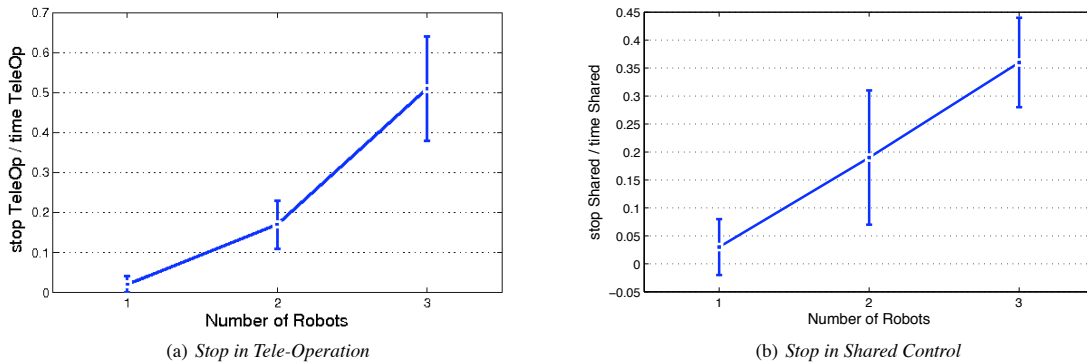


Figure 8: Means and 95% Bonferroni intervals.

We have learnt that commanding a team of robots requires proper shared control capabilities which enable the operator to act at the required autonomy level according to the task he must perform and the team state. If some of the robots are performing well in autonomy he can send higher level commands while commanding at a lower level those robots in most need of attention. Splitting the former Shared Mode into 3 commanding possibilities has been proven to be more suitable for this, as it has been seen in the experiments. Furthermore, long term commands, like setting the desired path or the sequence of target points decrease the negative effect of neglecting a robot.

The third version of the interface was used in the last RoboCup Edition (2009, Austria), in the Rescue Simulation League. It received a Technical Award for the most innovative interface. A fourth version has been developed introducing new functionalities in the *Full Autonomous* mode. Further work will include the conducting of user experiments to evaluate the improvements brought by Version 4. We hypothesize that, with the new functionalities offered in Autonomy Mode, operators will use it more often. The goal is to increase optimal team size to three or even four robots.

REFERENCES

- [1] J. L. Burke and R. R. Murphy, 'Human-robot interaction in usar technical search: Two heads are better than one', in *IEEE International Workshop on Robot and Human Interactive Communication*, pp. 307–312, (2004).
- [2] D. Calisi, A. Farinelli, L. Iocchi, and D. Nardi, 'Multi-objective exploration and search for autonomous rescue robots', *Journal of Field Robotics, Special Issue on Quantitative Performance Evaluation of Robotic and Intelligent Systems*, **24**, 763–777, (August - September 2007).
- [3] Jacob W. Crandall, Michael A. Goodrich, Dan R. Olsen, and Curtis W. Nielsen, 'Validating human-robot interaction schemes in multitasking environments', *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, **35**(4), 438–449, (2005).
- [4] Jill L. Drury, Brenden Keyes, and Holly A. Yanco, 'Lassoing hri: analyzing situation awareness in map-centric and video-centric interfaces', in *Proceedings of the Second ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, pp. 279 – 286, (2007).
- [5] Terrence Fong, Charles E. Thorpe, and Charles Baur, 'Advanced interfaces for vehicle teleoperation: Collaborative control, sensor fusion displays, and remote driving tools', *Auton. Robots*, **11**(1), (2001).
- [6] Michael A. Goodrich, Timothy W. McLain, Jeffrey D. Anderson, Jisang Sun, and Jacob W. Crandall, 'Managing autonomy in robot teams: observations from four experiments', in *Proceedings of the Second ACM SIGCHI/SIGART Conference on Human-Robot Interaction, HRI 2007, Arlington, Virginia, USA, March 10-12, 2007*, pp. 25–32, (2007).
- [7] Curtis W. Nielsen and Michael A. Goodrich, 'Comparing the usefulness of video and map information in navigation tasks', in *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction, HRI 2006*, pp. 95–101, (2006).
- [8] Raja Parasuraman, Thomas B. Sheridan, and Christopher D. Wickens, 'A model for types and levels of human interaction with automation', *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, **30**(3), 286–297, (2000).
- [9] Sarvapali D. Ramchurn, Alex Rogers, Kathryn Macarthur, Alessandro Farinelli, Perukrishnen Vytelingum, Ioannis A. Vetsikas, and Nicholas R. Jennings, 'Agent-based coordination technologies in disaster management', in *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Demo Proceedings*, pp. 1651–1652, (2008).
- [10] Diego Rodriguez-Losada, Fernando Matia, Agustin Jimenez, and Ramon Galan, 'Local map fusion for real-time indoor simultaneous localization and mapping', *Journal of Robotic Systems*, **23**(5), 291 – 309, (April 2006).
- [11] Jean Scholtz, Brian Antonishek, and Jeff Young, 'Operator interventions in autonomous off-road driving: effects of terrain', in *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics: The Hague, Netherlands, 10-13 October 2004*, pp. 2797–2802, (2004).
- [12] Sebastian Thrun and Yufeng Liu, 'Multi-robot slam with sparse extended information filters', in *Robotics Research, The Eleventh International Symposium, ISRR, October 19-22, 2003, Siena, Italy*, pp. 254–266, (2003).
- [13] Alberto Valero, *Evolutionary Design of Human-Robot Interfaces for Teaming Human and Mobile Robots in Exploration Missions*, Ph.D. dissertation, Sapienza - University of Rome, January 2010.
- [14] Alberto Valero, Gabriele Randelli, Chiara Saracini, Fabiano Botta, and Massimo Mecella, 'The advantage of mobility, mobile tele-operation for mobile robots', in *Proceedings of the AISB-HRI Symposium "New Frontiers in Human - Robot Interaction (To appear)*, (March 2009).
- [15] Lovekesh Vig and Julie A. Adams, 'A framework for multi-robot coalition formation', in *Proceedings of the 2nd Indian International Conference on Artificial Intelligence, Pune, India, December 20-22, 2005*, pp. 347–363, (2005).
- [16] Holly A. Yanco and Jill L. Drury, '"where am i?" acquiring situation awareness using a remote robot platform', in *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics: The Hague, Netherlands, 10-13 October 2004*, pp. 2835–2840, (2004).
- [17] Holly A. Yanco and Jill L. Drury, 'Rescuing interfaces: A multi-year study of human-robot interaction at the aaai robot rescue competition', *Auton. Robots*, **22**(4), 333–352, (2007).
- [18] Holly A. Yanco, Jill L. Drury, and Jean Scholtz, 'Awareness in human-robot interactions', in *Proceedings of the IEEE Conference on Systems, Man and Cybernetics, Washington, DC, October 2003*, (2003).