This is a preprint version of the article: Faigl, J., and Kulich, M. (2015). **On Benchmarking of Frontier-Based Multi-Robot Exploration Strategies**. *Proceedings of the European Conference on Mobile Robots 2015*, which has been published in a final form at https://doi.org/10.1109/ECMR.2015.7324183.

On Benchmarking of Frontier-Based Multi-Robot Exploration Strategies

Jan Faigl
Faculty of Electrical Engineering
Czech Technical University in Prague
Technicka 2, 166 27 Prague, Czech Republic
Email: faiglj@fel.cvut.cz

Miroslav Kulich
Czech Institute of Informatics, Robotics and Cybernetics
Czech Technical University in Prague
Zikova 1903/4, 166 36 Prague, Czech Republic
Email: kulich@ciirc.cvut.cz

***Abstract*—This paper tackles an evaluation and comparison of frontier-based exploration strategies to create a grid map of unknown environment by a team of autonomous mobile robots. A strategy is considered as a set of procedures to determine promising goal candidates, allocate them to the robots, and select the next navigational goal for each robot. A mobile robot is a complex system with many components that affect the mission performance and a comparison of different strategies in real experiments can be performed only for a particular system setup and with only partial controllability of important parameters. Therefore, the reproducibility and repeatability of such a comparison are not satisfied. In this paper, we propose a methodology for evaluating exploration strategies and provide a benchmark for a comparison of frontier-based approaches in a well-defined evaluation environment. The proposed methodology is demonstrated on a comparison of five state-of-the-art task-allocation strategies in multi-robot exploration.**

I. Introduction

Multi-robot exploration can be considered as a problem to efficiently navigate a robot or a team of collaborating robots in an unknown environment to acquire information about some studied phenomena. The basic variant of this problem is to collect information about an unknown environment and to create a map of the environment. Such a map can be used to find objects of interest located in the environment, which is one of the fundamental problem to address search and rescue missions, where the primary objective is to find possible victims as quickly as possible.

Within this context, exploration strategies are procedures that provide new navigational goals for each robot in the team. Then, the robots are navigated to the goals until new goals are determined. A more frequent determination of the navigational goals may utilize new information available and thus the mission can perform better.

Regarding the mission performance, there are several approaches ranging from work on designing off-line trajectories to approaches focused on autonomous navigation and related localization aspects. Authors usually demonstrate that their new approach is working and in some sense provides a “better” performance than the selected state-of-the-art algorithms that are most typically implemented within a particular framework developed by the authors for the specific robotic system [1]. Due to the specific details of the framework and the dependency of the system performance on various aspects of the whole robotic system, such comparisons and made conclusions are less general because practical experiments are tightly related to a particular robotic system used for the experimental evaluation. Even though a practical deployment is an indispensable part of the experimental system verification in realistic scenarios, simulations are becoming widely accepted by the robotics community as a part of the appropriate experimental methodology [2] that follows experimental principles well established in science [1].

978-1-4673-9163-4/15/$31.00 © 2015 IEEE

It has been shown [3], [4] that the exploration performance in a practical deployment depends on available computational resources. Hence, more sophisticated and demanding algorithms can provide worse results than simpler and fast-to-compute approaches because of limited computational power currently available on-board of mobile exploring units. Available computational power is still increasing, and therefore, an influence of the current lack of resources has to be fixed in the evaluation methodology. This motivates us to consider ideas in experimental computing [5], multi-robot exploration [4], and simulation in robotics [6], [1], [2] to propose a benchmark of various exploration strategies.

The paper is organized as follows. The related work on experimenting and benchmarking in computing and robotics is presented in the next section. An overview of the proposed methodology regarding experimental principles [1] and views of experiments [5] is discussed in Section III. The proposed methodology is presented in Section IV and a use case of its application in evaluation of five exploration strategies (briefly introduced in Section V) is described in Section VI. Concluding remarks are given in Section VII.

II. Related Work

There is progress in the development of the experimental methodology in both fields: robotics and informatics [1], [7], [5], [8]. A key question is how a simulation can be employed in the experimental verification, evaluation, and study of complex robotic systems, because it is clear that a practical experimentation using real robots is time consuming, costly, and also very hard to replicate with the identical properties, especially for multi-robot scenarios. Therefore, simulators are used to study a particular aspect of the robotic system [2]. For exploration, there are custom simulators [9] or available frameworks like Player/Stage [10] and USARSim [6], [11].

Although there are still discussions what is the role of experiments in computing, we follow principles of experimental methodology originating in natural sciences that includes aspects of: (1) *comparison*; (2) *reproducibility*; (3) *repeatability*; (4) *justification*; and (5) *explanation* [1]. It is clear that a simulation can provide better controllability of the experiment and thus it definitely supports these aspects.

The crucial issue of the experimental methodology is the problem how to compare different approaches. One way is to implement selected approaches within the same framework and perform several trials in the same scenario, e.g., like in [9], [3]. Simulations provide a better controllability of the scenario than a practical experiment with real robotic systems. The studied performance indicators, such as the time to explore the environment, can be evaluated statistically using ANOVA test [12] or Wilcoxon test [3]. However, methods may perform differently in different type of environments and an aggregation of the results into a single indicator is not straightforward without a reference value.

Authors of [13] propose to evaluate performance of the exploration as the competitive ratio, i.e., the ratio of the length of the path found by the studied algorithm to the length of the optimal path determined off-line for the known environment. Determination of the optimal continuous exploration tour (closed path) has been shown to be NP-hard [14], and therefore, authors developed approximation algorithm based on A* to find an optimal exploration path in a grid based environment for a single mobile robot with the limited visibility. Then, having the estimation of the optimal off-line performance P_o, the competitive ratio of an on-line algorithm a can be computed as P_a/P_o, where P_a is the performance of the on-line exploration algorithm a in the same environment. In particular, the traveled distance is the used as the performance indicator for the exploration [13].

Another approach is to determine a reference solution as the best found solution for each particular scenario using thousands of performed trials with different exploration strategies [4]. Such a solution represents a realistic lower bound of the required time to explore the whole environment that can be achieved by one of the evaluated strategies. This also allows to aggregate the results from different scenarios albeit the selected solution depends on the performance of the on-line algorithm that can be probably always worse than off-line solution found for the known environment.

An important aspect of evaluating exploration strategies is the frequency of sensing and in-situ decision-making. Authors of [12] experimentally confirm that, generally, a higher frequency provides better results, i.e., a shorter exploration path. On the other hand, it may not be the case of the computationally demanding methods because of limited computational power. Hence, a less sophisticated strategy may perform better than a more demanding approach on the same hardware because of a more frequent decision-making.

The computational burden can be decreased by approaches like [15]; however, in practical deployment, the exploration performance is always affected by this issue, which disfavors computationally expensive strategies. Computational resources are still improving; hence, such an evaluation can be considered as a less general.

A. Practical experimentation vs simulation

Probably the most criticized aspect of using simulations in robotics is the level of the realism and how much are the obtained results generalizable into a practical deployment. On the other hand, the valuable benefits of simulations are the reproducibility and repeatability [2] in addition to a statistical evaluation of many trials. Here, we can argue that such criticisms are mostly from the authors that are focused on robotic issues related to the sources of uncertainty in sensing and acting.

Such uncertainties are typically manifested in the precision of the localization. The exploration can be considered within the context of SLAM [16] and an exploration strategy has to trade-off between exploring new areas and navigation to the previously visited locations to decrease the localization uncertainty [17]. However, ongoing improvements of localization techniques provide sufficiently precise localization for indoor structured (e.g., office like) environments [18]. For example, teams participating in the CAROTTE multi-robot exploration challenge reported the most important part of their system design was to efficiently share the workload among the team members [19]. Hence, it seems that the well matured localization techniques allow to focus on further challenges of the exploration related to multi-robot coordination, limited communication, and efficient data sharing.

III. Experimental Principles

This section is intended to provide a background for the experimental evaluation techniques and how these principles are addressed in the proposed methodology.

A. Experimental Principles

Although there are still discussions what is the role of experiments in computing, we follow principles of experimental methodology originating in natural sciences that includes aspects of [1]:

Comparison – is one of the main principles that allows to measure and decide which approach provides better results. However, it is also a question in what type of the environment (setup) and in which measure the approach is better. Three basic types of comparisons can be identified [20]: 1) usage of the same implementations used in the previous experiments; 2) development of a new implementation based on the available description; 3) usage of the results reported by other authors in their publications. In all cases, a precise description of the mission setup is very important.

Reproducibility – is mostly related to the description of the experiment that must be sufficiently detailed to allow replication of the experiment by other researchers.

Repeatability – expresses the ability to get the same outcome from several trials and thus it is related to the controllability of the experiment, i.e., all the important influences that affect the measured indicator are fixed or under specified bounds.

Justification and *Explanation* represent aspects that allow to make a general conclusion based on the experimental data collected, which is not an artifact of the particular experimental setup [1]. To further support that, a comparison with a reference or optimal solution may be provided.

B. Views on Experimentation

Five views on experimental computer science are discussed in [5]. Although we do not aim to provide a definition of a particular type of experiment, we provide a brief comment on each view regarding experimentation in the robotic exploration and exploration strategies themselves.

Feasibility Experiment shows that a particular approach works without specification of assumed conditions. It provides no comparison with other approaches.

Trial Experiment includes elaborated results under different conditions and provides performance indicators.

Field Experiment is an experiment performed in real-world conditions and it should include challenges to show robustness of the solution in realistic scenarios.

Comparison Experiment is a comparison of various solutions providing a conclusion that one approach is better than another. For robotic exploration, a simple greedy selection of the next-best goal can be applied and thus each experiment should provide at least a comparison with such a simple solution.

Controlled Experiment is the way how to provide generalizable results and conclusions. All variables that can influence the results of the experiment should be under control and specified in the experimental protocol.

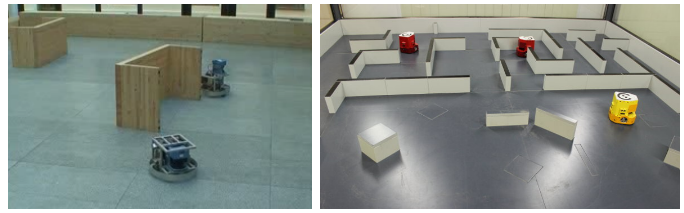
IV. Evaluation of Exploration

The proposed evaluation methodology is based on the aforementioned principles, our practical experience, and literature review. Although, the general recommendations of good experimental methodology can be found in several papers, we aim to provide additional contribution towards further development in strategies for multi-robotic exploration. Therefore, we built our approach on three pillars, where for each pillar, we also provide particular implementation artifacts to support ease of use of the proposed methodology.

The first pillar is a specification of the exploration framework, which provides a general concept about what the exploration strategy is and what algorithmic parts influence the exploration performance. The second pillar is the benchmark that consists of environments, particular frameworks and methods, reference solutions, and statistical evaluation. The third pillar can be considered as an experimental protocol that is a precise specification of the particular trials, experimental methods and all parameters of the exploration system. Regarding the role of simulation and experimental principles, we propose to consider evaluation using different levels of realism to clarify what is the purpose of the experiment.

Level–0 fixes all parameters and aspects of the exploration and represents a completely controlled evaluation environment. It allows to evaluate strategies independently on computational resources. Under these constraints, performance indicators like the real required time to explore the environment cannot be directly measured, instead, the traveled distance is used as the performance indicator.

Level–1 represents evaluation using a software simulator like Player/Stage or USARSim. The main difference from the Level–0 is that it provides direct measurement of the real exploration time based on physical simulations and models of the robot motion. It also includes influence of the computational requirements of the studied strategies albeit the simulator can be slowed down.



(a) (b)

Fig. 1. Examples of real mobile robots in evaluation of autonomous robotic exploration under Level–2 or realism.

Level–2 is an experiment with real mobile robots; however, in a well defined environment, where the localization is not an issue and the deployment allows to repeat the experiments several times. Such an environment can look like in Fig. 1a or it can be a remotely accessible system for a robotic e-learning like [21], which is shown in Fig. 1b. These deployments allow to consider practical issues of real mobile robots, but they still provide controlled environment without communication issues. Particular limits in communication range and reliability can be realistically simulated.

Further levels of realism are considered as *field experiments* with increasing challenges in navigation and decreasing controllability of the experimental parameters.

A. Exploration Framework

The problem to decide where to navigate mobile robots can be decomposed into several sub-problems, where a solution of each such a problem may affect the overall system performance. We call a set of procedures to address the sub-problems as an exploration strategy and the procedures are the basic building blocks of the exploration framework. In the proposed methodology, we provide the framework supplemented by implementations of the selected algorithms [22].

For m robots $\boldsymbol{R} = \{r_1, \ldots, r_m\}$, the exploration can be considered as an iterative procedure as follows. Notice, the highlighted parts of the procedure denote the important algorithms that may affect the performance significantly.

1) Initialize the model of the environment and set the initial plans to $\mathcal{P} = (P_1, \ldots, P_m)$, where $P_i = \{\emptyset\}$ for each robot $1 \leq i \leq m$.
2) `Repeat`
   a) **Navigate robots** using the plans $\mathcal{P}$;
   b) Collect new measurements;
   c) Update the navigation map $\mathcal{M}$;

   `Until` **replanning condition is met**.
3) **Determine goal candidates $\boldsymbol{G}$** from $\mathcal{M}$.
4) `If` $|\boldsymbol{G}| > 0$ **assign goals to the robots**
   - $(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle)$=assign$(\boldsymbol{R}, \boldsymbol{G}, \mathcal{M})$, $r_i \in \boldsymbol{R}, g_{r_i} \in \boldsymbol{G}$;
   - **Plan paths** to the assigned goals $\mathcal{P} = \text{plan}(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle, \mathcal{M})$;
   - Go to Step 2.
5) Stop all robots or navigate them to the depot (all reachable parts of the environment are explored).

B. Problem specification

The listed procedure can be used for different exploration missions; however, for simplicity, we consider frontier-based exploration to create a grid map of the environment as quickly as possible. Thus, the evaluation criterion is the time to create a map of the whole environment (using occupancy grid [23]), which can be measured as the longest exploration path traveled by an individual robot of the team.

In this discussion of benchmarking exploration strategies, we prefer to simplify the problem and thus we consider centralized approach of multi-robot frontier-based exploration with homogeneous mobile robots equipped with an omnidirectional sensor with a limited sensing range ρ. The cells of the grid map are in one of three possible states: free, obstacle, and unknown; and a cell is called frontier cell if it is a free cell that is incident with at least one unknown cell.

Although the considered expression of the problem is relatively simple, there are still several parameters of the exploration procedure to provide a complete specification of the problem. Most of the reports on exploration are focused on algorithms how to select the next navigational goal to share the workload among robots; however, the exploration procedure also depends on the following aspects.

The first aspect is a resolution of the grid map which is also related to the path planning algorithm because a shape of the path affects what the robot can explore during navigation to its current goal. Here, we can imagine standard algorithms like Dijkstra's and A*, Voronoi Diagram or wave-front propagation techniques like Distance Transform (DT) [24], and its improved variant EEDT [25] and Fast Marching method [26].

The robots are navigated towards the assigned goal by following the planned path to the goal; however, the speed of the robot depends on its kinematics and used motion controller. For the Level–0 of realism we assume the robot is capable of omnidirectional motion and visits each grid cell of the found path. The expected exploration time can be then estimated from the length of the traveled path in the grid and average velocity of the robot. However, for higher levels of realism, we need to deal with the local motion planner (and controller) that uses current sensor measurements to determine the robot's forward and angular velocities. Notice that such a planner (e.g., SND [27]) may decrease the robot velocity in proximity of obstacles (to increase safety in narrow passages), while a robot may move significantly faster in open space areas. The character of the environment and planned paths may affect the average speed of the robot during a particular mission.

Based on the impact of the frequency of the decision-making [12], we identify two limiting cases: (a) **goal replanning** (GR) – the assignment of newly determined goals whenever a robot reaches its previously assigned goals, (2) **immediate replanning** (IR) – determination and assignment of new goals whenever a previously assigned goal is no longer a frontier as its surroundings have been explored.

In a case the robots collect high quality images of the explored areas that cannot be transmitted remotely [19], we can further distinguish two additional variants of the exploration which may influence the exploration time: **open paths** (OP) and **return to the depot** (RD).

TABLE I. SPECIFICATION OF THE EXPLORATION PROCEDURE

Parameter	Value
Sensor model	Laser range finder with sensing range ρ with omnidirectional/limited field of view
Environment map	Grid map with the cell size 0.05 m
Path planning	DT with a ray-shooting simplification [3]
Local navigation	Discrete movements in the grid / SND [27]
Depot return	Yes (RD)
Decision-making	Goal / Immediate
Coordination	Centralized
Communication	Full without restriction

Finally, we also found out another detail that influences the required time to explore the whole environment. It is related to the identification of the frontiers and determination of the goal candidates. In [28], authors consider frontier cells organized into single connected components. Then, for each such a component a number of representatives is determined, but only if the component consists of more than a given number of frontier cells (denoted as nfc in the rest of this paper). Although a low value of nfc does not affect the ability to explore the whole environment (e.g., nfc=2), it may provide significantly faster exploration, because robots avoid navigation to areas represented by a small set of frontier cells, e.g., corners of rooms. The value of nfc is therefore important to compare different approaches under the same conditions.

A list of the discussed parameters of the exploration procedures is depicted in Table I, except the methods to determine the goal candidates and assign them to the robots that are discussed in Section V.

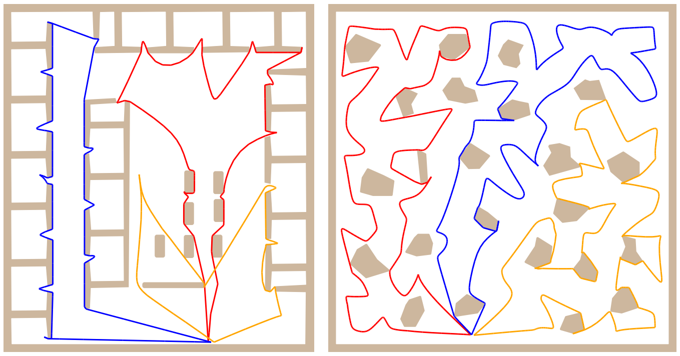
C. Scenario Setup

Further aspect of the evaluation is a specification of the mission setup – the environment to be explored and initial positions of particular robots. A representative environment is important and to support further generalization of the results a set of randomly generated environments with particular percentage of obstacles can be used, e.g., using a generator like in [29]. However, regarding the evaluation of exploration strategies, there are basically two types of environments: 1) office like with long corridors and many relatively small rooms; 2) and large open space areas with obstacles. Moreover, an environment should contain loops to demonstrate coordination of robots, as long corridors are not a significant difficulty for finding an efficient solution.

Regarding the environment and a shape of the robot, it is worth mentioning that path planning can be performed in an enlarged representation of the environment currently being explored, e.g., using Minkowski sum of the current map with a disc representing a shape of the robot. Thus, some parts of the environment may appear as unreachable, and therefore, dimensions and shape of the robot body is also important regarding evaluation of the mission performance.

D. Benchmarks

Benchmark in multi-robot exploration is a set of artifacts that enable evaluation of the exploration strategies regarding the aforementioned experimental aspects and views. In this paper, we do not provide a strict set of the artifacts that have



(a) *jh* (b) *potholes*

Fig. 2. Example of reference solution for 3 robots with sensing range ρ=3 m; notice, the solutions are found in enlarged maps of the environment to respect dimensions of the robots.

to be used, we rather provide initial solutions that can be used and further extended.

A set of maps is the first part of the benchmark. Due to the limited space, we consider only two environments called *jh* and *potholes* to cover the office-like environments and open-space areas, see Fig. 2.

The additional part of the benchmark are the particular algorithms for: path planning, task-allocation, and determination of the goal candidates. Four algorithms for path planning are provided: DT, EEDT, Voronoi, and modified DT with a simple ray-shooting technique to improve the solutions.

The task-allocation algorithms provided as a part of the benchmark are briefly described in the next section. Besides, we also provide three methods to determine goal candidates: all frontiers, the method RFE introduced by the authors of [28], and its modification called ANR [4].

Probably the most important part of the proposed benchmark is the reference algorithm to find an off-line solution of the exploration problem for several mobile robots. Here, we consider the problem as a formulation of the watchman route problem with limited sensing range that stands as follows. For m robots and a given polygonal environment determine m paths from which the whole environment is covered using an omnidirectional sensor with the range ρ. This problem is known to be NP-hard, and therefore, we adopted the approximate algorithm [30] to consider m distinct initial positions of the robots. We consider this method only for the Level–0 of realism in a discrete time simulator for which particular reference solution is determined in a polygonal map that is created from the map of the environment being explored by applying Minkowski sum with a disc representing the shape of the robot. An example of solutions is depicted in Fig. 2.

V. Exploration Strategies

Five task-allocation procedures have been selected to show how the proposed evaluation method can be used to compare performance of the exploration strategies. The procedures have been already applied in multi-robot exploration by several authors and are well described in literature. Therefore only a brief description of them is presented here. Particular implementations of these methods are part of the provided benchmark to foster further development and evaluation.

Greedy Assignment (GA) – originally proposed by Yamauchi [31] has modified to select the closest not yet assigned goal to each robot sequentially in a random order.

Iterative Assignment (IA) – is a centralized implementation of the *Broadcast of Local Eligibility* algorithm [32], which assigns the goal according to an ordered sequence of all robot–goal pairs $\langle r, g\rangle$ using the associated distance cost. A not yet assigned goal with the lowest cost is assigned to the particular robot without a goal.

Hungarian Assignment (HA) – is based on the Hungarian algorithm [33] that provides optimal task-allocation for m robots and n goals, where the cost of the robot–goal assignment is stored in a cost matrix. For $m > n$, the IA algorithm is used. In a case $m < n$, additional (virtual) robots are added to the matrix with a very high cost of the assignment.

Multiple Traveling Salesman Assignment (MA) – is based on solution of the multiple traveling salesman problem (MTSP) [3] by the $\langle$*cluster first, route second*$\rangle$ heuristic based on the K-means clustering method. Once clusters are determined and assigned to the robots, the next navigational goal for each robot is determined as the first goal on the TSP tour to visit all the goals in the cluster that starts from the current robot's position. An approximate solution of the TSP is found by the Chained Lin-Kernighan heuristic [34] from the CONCORDE package [35].

MinPos algorithm consists of determination of the rank $r_{i,j}$ for each goal i and robot j. The rank $r_{i,j}$ denotes the number of robots that are closer to the goal candidate i than the robot j. The goal with the minimal rank to a robot is assigned as the next navigational goal of the particular robot. If there are several goal candidates with the same minimal rank to the robot, the closest goal candidate is selected [36]. Notice that even though MinPos is simple, it provides very similar (a little bit better) results as [37].

VI. Use Case

The proposed evaluation methodology has been employed for a comparison of the exploration strategies described in Section V to demonstrate how the method can be used. The comparison has been performed for three levels of the realism: 1) the precisely defined discrete-time simulator of the Level:0; 2) a more realistic simulator Player/Stage that allows to measure the real required time to explore the environment in the Level:1; 3) and finally using real robots in the remotely accessible SyRoTek platform [28] for the Level:2 of the realism. In all cases, the exploration is terminated when all the robots return to their initial positions.

A. Level–0: Discrete Simulator

In this case, all five task-allocation algorithms (GA, IA, HA, MA, and MinPos) have been deployed in the exploration of the *jh* and *potholes* environments and accompanied by the ANR method for determination of goal candidates. The sensing range has been selected from $\rho \in \{3\ \mathrm{m}, 5\ \mathrm{m}, 7\ \mathrm{m}\}$ and the number of robots $m \in \{3, 5, 7\}$. In addition, a small perturbation is added into the initial positions of the robots, which gives 20 variants of each problem. The IA, HA, and MinPos algorithms are deterministic, therefore only a single trial is performed by each algorithm for each exploration scenario defined by the environment, ρ, m, and the starting positions of the robots. GA and MA algorithms are stochastic and thus 20 trials are performed for each scenario. Besides, the evaluation has been performed for both limiting replanning conditions GR and IR, which gives 30 960 trials in total.

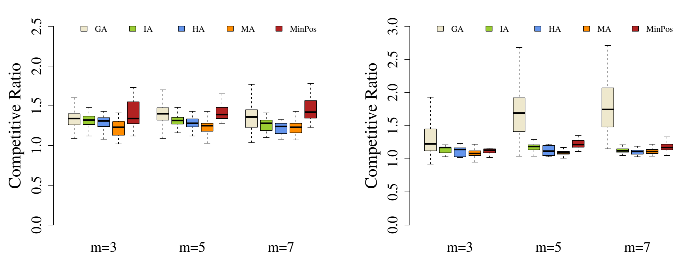
Regarding the high number of the trials, the performance indicators of the particular algorithms are considered as the five-number summaries of the competitive ratio of the longest exploration path in each trial and the reference value determined as a solution of the multiple watchman route problem (MWRP) with the minimization of the longest watchman route based on the approximate algorithm proposed in [30]. The particular reference solutions are depicted in Table II.

TABLE II. REFERENCE EXPLORATION PATHS IN METERS.

Sensing range	No. of Robots	Environment *jh*	Environment *potholes*
ρ=3 m	m=3	75	170
	m=5	59	120
	m=7	59	104
ρ=5 m	m=3	68	131
	m=5	57	100
	m=7	56	92
ρ=7 m	m=3	59	105
	m=5	57	87
	m=7	57	83

During the exploration, the paths are determined in the enlarged current map of the environment about a disc with the diameter 0.3 m that represents the shape of each robot. The same enlargement has been made for the reference solution of the MWRP. The occupancy grid and navigational grid have resolution 5 cm, i.e., the squared grid cell has dimension 0.05 m. The parameter nfc has been set to 0.

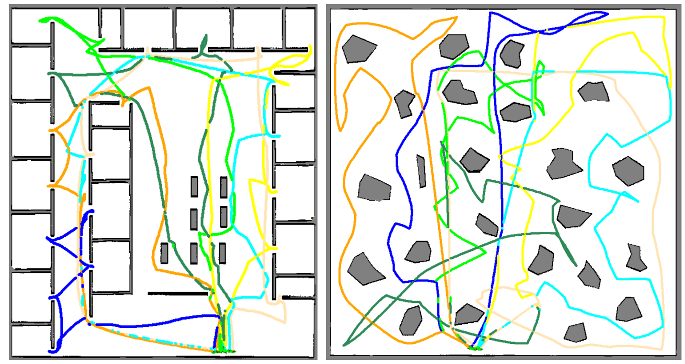
Although a high number of trials for different exploration missions have been performed, the competitive ratio allows to aggregate the results and show the overall performance indicators, e.g., to study how the algorithms scale with the number of robots m or the sensing range ρ. Due to limited space, only selected results are depicted in Fig. 3.



(a) ρ=5 m, GR (b) ρ=5 m, IR

Fig. 3. Competitive ratio for the GR and IR replanning conditions.

Generally, the IR condition improves performance for all exploration strategies; however, for an increasing number of robots m, the performance of GA is significantly worse than for the GR condition. It is because GA significantly changes the assignment of the goals to the robots between two assignments. Although the IR condition is unrealistically fast, it allows to identify such an "unstable" behaviour.



(a) *jh* (b) *potholes*

Fig. 4. Exploration paths for 7 robots.

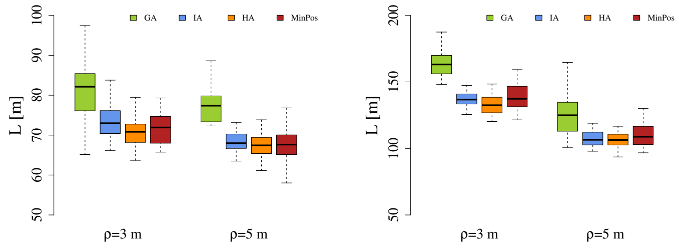
An identification of such instability of the strategy is not possible in a regular deployment using real-time simulator or real system, where the assignment is not performed with such a high frequency because of limited computational power. Besides, a high number of trials with randomized positions of the robot provides an opportunity to verify implementations of particular complex algorithms under various conditions.

Regarding a comparison of the task-allocation algorithms, the best performance provides the most computationally demanding MA algorithm based on the solution of the MTSP. On the other hand, differences between the algorithms are less significant with increasing frequency of the assignment.

B. Level–1: Player/Stage

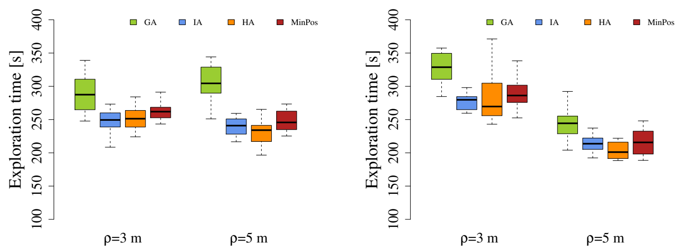
For the Player/Stage simulator, the considered environments are *jh* and *potholes* but the sensing range ρ is set to the value 3 m and 5 m, and simulations are performed only for m=7. The assignment strategies are GA, IA, HA, and MinPos accompanied by the ANR goal candidates determination and immediate replanning (IR) once a decision is made. In this case, 20 trials are performed for each strategy, as the simulation environment includes noise and thus perturbations are already included. All the other parameters are the same as for the Level–0, except the nfc, which is set to a more practical value 4 without an effect on the final created map, e.g., see examples of the created maps in Fig. 4.

Five-number summary of the longest exploration paths are depicted in Fig. 5. Under the Level–1 of the realism, we can measure the required exploration time that is depicted in Fig. 6. We can notice that the length of the exploration path corresponds to the required exploration time and thus travelled distances are representative performance indicators. However, we can also see that in particular cases, the averages and standard deviations are higher for the time than for the length of the exploration path. It is because it may happen that a robot is locally "stucked" due to the used SND controller, while the exploration paths have similar lengths. In addition, a higher exploration time may also be caused by the mutual avoidance of the robots. The results also indicate that a longer sensing



(a) *jh* (b) *potholes*

Fig. 5. Length of the longest exploration path for m=7 and sensing range ρ=3 m and ρ=5 m.



(a) *jh* (b) *potholes*

Fig. 6. Total required time to explore the whole environment and return all 7 robots to the depot.

range does not improve exploration time in the environment *jh* because of the environment structure.

C. Level–2: SyRoTek

We also consider deployment of the strategies using real robots in the SyRoTek system [21]. SyRoTek is a platform for e-learning and distant experimentation in robotics and related areas consisting of thirteen robots equipped with standard robot sensors (laser range-finders, sonars, odometry, etc.). The robots operate in the Arena of size 3.5×3.8 m and are fully programmable and remotely controlled. Provided global localization system, on-line visualization, interfaces to Player/Stage and ROS enable to perform long-term experiments without human assistance.

With the increased level of the realism we decrease the number of trials to five and restrict the exploration to three robots. Two strategies GA and HA are deployed with ρ=0.4 m and limited sensing view to 240° because of used sensor the HOKUYO URG-04LX laser range finder. The occupancy and navigational grids have resolution 2 cm and the nfc is set to 4. The maximal rotational velocity is set to $7^{\circ}/s$ to guarantee safe navigation in presence of other moving robots. The environment is without obstacles and exploration paths can be seen in Fig. 7. Average length of the longest exploration paths and average required exploration times are depicted in the table in Fig. 7.

In this case, the distance travelled does not correspond to the required exploration path and both strategies GA and HA provide similar performance. However, regarding the exploration time HA provides the expected better performance. It is because the low level motion controller SND is slow in turns and goals proposed by GA oscillate more frequently than these

Indicator	**GA**	**HA**
L – Traveled distance [m]	3.9	3.9
σ_L – std. dev.	0.65	0.68
T – Exploration time [s]	85	75
σ_T – std. dev.	3.88	5.48

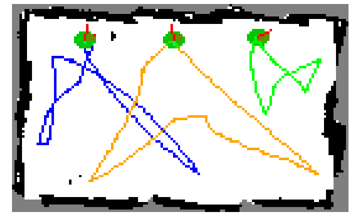


Fig. 7. Results for the deployment with real robots in SyRoTek.

provided by HA. This leads to frequent changes of the robots' heading and thus a slower movement.

VII. CONCLUSION

In this paper, an evaluation of multi-robot exploration has been discussed and methodology and benchmarks have been introduced. The benchmarks also include particular artifacts to support and encourage other researchers to standardize results in exploration approaches. The methodology is primarily focused on the frontier-based exploration problem to create a grid map of the environment; however, it can also be used for another variants of the robotic exploration.

Although the current results are presented for centralized approaches of multi-robot coordination, the proposed benchmarks with the reference solution as a lower bound may also be used for distributed strategies. They allow to measure efficiency of the distributed approach regarding the efficacy of the communication, which is a subject of our future work.

Finally, in this paper, we do not claim the proposed methodology is the ultimate approach, but we believe it can further support unification of the evaluation and benchmarking in multi-robot exploration.

ACKNOWLEDGMENTS

The work of J. Faigl is supported by the Czech Science Foundation (GAČR) under research project No. GJ15-09600Y and the work of M. Kulich by GAČR under research project No. GA15-22731S.

Computational resources were provided by the MetaCentrum under the program LM2010005 and the CERIT-SC under the program Centre CERIT Scientific Cloud, part of the Operational Program Research and Development for Innovations, Reg. No. CZ.1.05/3.2.00/08.0144.

REFERENCES

[1] F. Amigoni, M. Reggiani, and V. Schiaffonati, "An insightful comparison between experiments in mobile robotics and in science," *Autonomous Robots*, vol. 27, no. 4, pp. 313–325, 2009.

[2] F. Amigoni and V. Schiaffonati, "Good experimental methodologies and simulation in autonomous mobile robotics," in *Model-Based Reasoning in Science and Technology*, 2010, vol. 314, pp. 315–332.

[3] J. Faigl, M. Kulich, and L. Přeučil, "Goal assignment using distance cost in multi-robot exploration," in *IROS*, 2012, pp. 3741–3746.

[4] J. Faigl, O. Simonin, and F. Charpillet, "Comparison of task-allocation algorithms in frontier-based multi-robot exploration," in *Multi-Agent Systems - 12th European Conference, EUMAS*, 2014, pp. 101–110.

[5] M. Tedre and N. Moisseinen, "Experiments in computing: A survey," *The Scientific World Journal*, p. 11, 2014. [Online]. Available: http://dx.doi.org/10.1155/2014/549398

[6] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper, "Bridging the gap between simulation and reality in urban search and rescue," in *RoboCup*. Springer Berlin Heidelberg, 2006, pp. 1–12.

[7] R. Madhavan, R. Lakaemper, and T. Kalmar-Nagy, "Benchmarking and standardization of intelligent robotic systems," in *International Conference on Advanced Robotics (ICAR)*, 2009, pp. 1–7.

[8] C. Andujar, V. Schiaffonati, F. A. Schreiber, L. Tanca, M. Tedre, K. van Hee, and J. van Leeuwen, "The role and relevance of experimentation in informatics," Tech. Rep., 2013, [cited 7 July 2015]. [Online]. Available: http://www.informatics-europe.org/news-and-events/171-news-report-experimentation.html

[9] M. Juliá, A. Gil, and O. Reinoso, "A comparison of path planning strategies for autonomous exploration and mapping of unknown environments," *Autonomous Robots*, vol. 33, no. 4, pp. 427–444, 2012.

[10] B. P. Gerkey, R. T. Vaughan, and A. Howard, "The Player/Stage project: Tools for multi-robot and distributed sensor systems," in *Proc. of the 11th Int. Conf. on Advanced Robotics*, 2003, pp. 317–323.

[11] S. Mokaram, K. Samsudin, A. Ramli, and H. Kerdegari, "Mobile robots communication and control framework for usarsim," in *ICIAS*, vol. 2, 2012, pp. 540–544.

[12] F. Amigoni, A. Q. Li, and D. Holz, "Evaluating the impact of perception and decision timing on autonomous robotic exploration," in *ECMR*, 2013, pp. 68–73.

[13] A. Q. Li, F. Amigoni, and N. Basilico, "Searching for optimal off-line exploration paths in grid environments for a robot with limited visibility," in *AAAI Conference on Artificial Intelligence*, 2012.

[14] E. M. Arkin, S. P. Fekete, and J. S. B. Mitchell, "Approximation algorithms for lawn mowing and milling," *Comput. Geom.*, vol. 17, no. 1-2, pp. 25–50, 2000.

[15] M. Keidar and G. A. Kaminka, "Robot exploration with fast frontier detection: Theory and experiments," in *AAMAS*, 2012, pp. 113–120.

[16] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[17] D. Holz, N. Basilico, F. Amigoni, and S. Behnke, "Evaluating the efficiency of frontier-based exploration strategies," in *ISR/ROBOTIK*, 2010, pp. 1–8.

[18] E. Tsardoulias and L. Petrou, "Critical rays scan match slam," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 3-4, pp. 441–462, 2013.

[19] A. Bautin, P. Lucidarme, R. Guyonneau, O. Simonin, S. Lagrange, N. Delanoue, and F. Charpillet, "Cart-o-matic project: autonomous and collaborative multi-robot localization, exploration and mapping," in *Workshop on Planning, Perception and Navigation for Intelligent Vehicles, organized in conjunction with IROS*, 2013, pp. 210–215.

[20] F. Amigoni, M. Reggiani, and V. Schiaffonati, "Simulations used as experiments in autonomous mobile robotic," in *Workshop on The Role of Experiments in Robotics Research, organized in conjuction with ICRA*, 2010.

[21] M. Kulich, J. Chudoba, K. Kosnar, T. Krajník, J. Faigl, and L. Preucil, "SyRoTek - distance teaching of mobile robotics," *IEEE Trans. Education*, vol. 56, no. 1, pp. 18–23, 2013.

[22] J. Faigl and M. Kulich, "Framework for evaluation of frontier-based multi-robot exploration strategies," [cited 7 July 2015]. [Online]. Available: http://purl.org/faigl/mre_sim

[23] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, vol. 9, no. 2, pp. 61–74, 1988.

[24] R. A. Jarvis, "On distance transform based collision-free path planning for robot navigation in known, unknown and time-varying environments," in *Advanced Mobile Robots*, Y. F. Zang, Ed. World Scientific Publishing Co. Pty. Ltd., 1994, pp. 3–31.

[25] J. C. Elizondo-Leal, E. F. Parra-González, and J. G. Ramírez-Torres, "The exact euclidean distance transform: A new algorithm for universal path planning," *International Journal of Advanced Robotic Systems*, vol. 10, 2013.

[26] S. Garrido, L. Moreno, and D. Blanco, "Exploration of 2d and 3d environments using voronoi transform and fast marching method," *Journal of Intelligent and Robotic Systems*, vol. 55, no. 1, pp. 55–80, 2009.

[27] J. W. Durham and F. Bullo, "Smooth nearness-diagram navigation," in *IROS*, 2008, pp. 690–695.

[28] M. Kulich, J. Faigl, and L. Přeučil, "On distance utility in the exploration task," in *ICRA*, 2011, pp. 4455–4460.

[29] A. P. Tomas and A. L. Bajuelos, "Quadratic-time linear-space algorithms for generating orthogonal polygons with a given number of vertices," in *ICCSA (3)*, 2004, pp. 117–126.

[30] J. Faigl, "Approximate Solution of the Multiple Watchman Routes Problem with Restricted Visibility Range," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1668–1679, 2010.

[31] B. Yamauchi, "Decentralized coordination for multirobot exploration," *Robotics and Autonomous Systems*, vol. 29, pp. 111–118, 1999.

[32] B. B. Werger and M. J. Mataric, "Broadcast of local eligibility: behavior-based control for strongly cooperative robot teams," in *Autonomous agents*, 2000, pp. 21–22.

[33] C. Stachniss, "C implementation of the hungarian method," 2004, [cited 7 Jul 2015]. [Online]. Available: http://www.informatik.uni-freiburg.de/~stachnis/misc/libhungarian-v0.1.2.tgz

[34] D. Applegate, W. Cook, and A. Rohe, "Chained Lin-Kernighan for large traveling salesman problems," *Informs J. on Computing*, vol. 15, no. 1, pp. 82–92, 2003.

[35] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, "CONCORDE TSP Solver," 2003, [cited 28 July 2014]. [Online]. Available: http://www.tsp.gatech.edu/concorde.html

[36] A. Bautin, O. Simonin, and F. Charpillet, "Minpos: A novel frontier allocation algorithm for multi-robot exploration," in *ICIRA*, 2012, pp. 496–508.

[37] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.