Exploration strategies based on Multi-Criteria Decision Making for search and rescue autonomous robots

Nicola Basilico
Politecnico di Milano, Italy
basilico@elet.polimi.it

Francesco Amigoni
Politecnico di Milano, Italy
amigoni@elet.polimi.it

ABSTRACT

Autonomous mobile robots are considered a valuable technology for search and rescue applications, where an initially unknown environment has to be explored to locate human victims. In this scenario, robots exploit exploration strategies to autonomously move around the environment. Most of the strategies proposed in literature are based on the idea of evaluating a number of candidate locations according to *ad hoc* utility functions that combine different criteria. In this paper, we show some of the advantages of using a more theoretically-grounded approach, based on Multi-Criteria Decision Making (MCDM), to define exploration strategies for robots employed in search and rescue applications. We implemented some MCDM-based exploration strategies within an existing robot controller and we experimentally evaluated their performance in a simulated environment.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics—*Autonomous Vehicles*

General Terms

Algorithms

Keywords

Mapping, search and rescue, exploration strategies

1. INTRODUCTION

In search and rescue with autonomous mobile robots, an initially unknown environment has to be explored and searched for human victims [4]. *Exploration strategies* that drive the robots around the partially known environment on the basis of the available knowledge are fundamental for achieving an effective behavior. The mainstream approach for developing exploration strategies is based on the idea of incrementally exploring the environment by evaluating a number of candidate observation locations according to an utility function and by selecting, at each step, the next best observation location. Exploration strategies differ in the utility functions they use to evaluate candidate locations. Although in multirobot exploration the evaluation of candidate observation locations is closely related to their coordinated allocations to the available robots, in this paper we focus only on evaluation of candidate observation locations. In systems proposed in literature, this evaluation is performed using utility functions that aggregate multiple criteria measuring different aspects of the locations and that are rarely based on a theoretical ground.

Cite as: Exploration strategies based on Multi-Criteria Decision Making for search and rescue autonomous robots, N. Basilico and F. Amigoni, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. 99-106.

In this paper, we apply a decision-theoretical tool, called *Multi-Criteria Decision Making* (*MCDM*), to define exploration strategies for search and rescue. Using decision-theoretical tools, on the one hand, contributes to the further assessment of the science of robotics and, on the other hand, provides practical advantages in the definition of effective exploration strategies. Although MCDM has been already applied to map building with a single robot [5], we deem that its application to multirobot search and rescue represents a significant contribution since it addresses a more challenging setting for exploration strategies, where the primary objective is not to build an accurate map of the physical space but to search the environment for locating the largest number of victims in a limited amount of time. Differently from map building, in search and rescue settings operations must be performed quickly, privileging the amount of explored area over the map quality. To the best of our knowledge, this is the first attempt to apply MCDM to search and rescue.

We consider a situation in which a team of robots have to search an initially unknown environment for victims. Since no *a priori* knowledge about the possible locations of the victims is assumed to be available, we can reduce the problem of maximizing the number of victims found in a given time interval to the problem of maximizing the amount of area covered by robots' sensors in the same time interval. Broadly speaking, the robots operate according to the following steps: (a) they perceive the surrounding environment, (b) they integrate the perceived data within a map representing the environment known so far, (c) they decide where to go next, and (d) they go there and start again from (a). We propose to use MCDM for addressing step (c), namely for defining the exploration strategy. In our experiments, we implemented the proposed approach as a modification of a publicly available controller used for the RoboCup Rescue Virtual Robots Competition [18]. In this way, on the one hand, we can focus on the development of exploration strategies (step (c)) exploiting an already tested framework for steps (a), (b), and (d) and, on the other hand, we can fairly compare our strategies with that originally used in [18].

2. RELATED WORK

Robotic exploration is a broad concept that can be defined as a process that discovers unknown features in environments by means of mobile robots. Exploration is employed in several tasks, like map building [16], search and rescue [15], and coverage [8]. For example, in map building the features to be discovered are the obstacles and the free space, while in search and rescue can be the locations of victims or fires. *Exploration strategies* are used to move autonomous robots around environments in order to discover their features. In this paper, we are interested in exploration strategies employed for discovering the physical structure of environments that are initially unknown. In these scenarios, we do not know *ex ante* the complete set of the possible locations that the robots can reach. We explicitly note that, as a consequence, we cannot employ some exploration strategies, like those proposed in [11] and [13], which require an *a priori* knowledge on the possible observation locations. In the following, we survey a representative sample of the countless exploration strategies that have been proposed in literature.

Unsurprisingly, most of the work on exploration strategies for discovering the physical structure of environments has been done for map building. The mainstream approach models exploration as an incremental Next Best View (NBV) process, i.e., a repeated greedy selection of the next best observation location. Usually, at each step, an NBV system considers a number of candidate locations on the frontier between the known free space and the unexplored part of the environment (in such a way they are reachable from the current position of the robot) and selects the best one [20]. The most important feature of an exploration strategy is how it evaluates candidate locations in order to select the best one.

In evaluating candidate locations, different criteria can be used. A simple one is the distance from the current position of the robot [20], according to which the best observation location is the nearest one. However, most works combine different criteria in more complex utility functions. For example, in [14] the cost of reaching a candidate location p is linearly combined with its benefit. Measuring the cost as the distance $d(p)$ of p from the current location of the robot and the benefit as an estimate of the new information $A(p)$ acquirable from p, the global utility of p is computed as:

$$u(p) = A(p) - \beta d(p), \tag{1}$$

where β balances the relative weight of benefit versus cost (authors show that choosing β within the interval $[0.01, 50]$ does not causes significant variations in the exploration performance). Another example of combination of different criteria is [9], in which distance $d(p)$ and the expected information gain $A(p)$ of a candidate location p are combined in an exponential function

$$u(p) = A(p)e^{-\lambda d(p)} \tag{2}$$

(where λ is a parameter that weights the two criteria). In [1], a technique based on relative entropy is used to combine traveling cost and expected information gain. In [17], several criteria (such as uncertainty in landmark recognition and number of visible features) are combined in a multiplicative function. In [12], traveling cost to reach a location is used as the main criterion for evaluating candidate locations, while the utility of the locations (calculated according to the proximity of other robots) is used as a tie-breaker.

The above strategies aggregate different criteria in utility functions that are defined *ad hoc* and are strongly dependent on the criteria they combine. In [2], the authors dealt with this problem and proposed a more theoretically-grounded approach based on multi-objective optimization, in which the best candidate location is selected on the Pareto frontier. Besides distance and expected information gain, also overlap is taken into account. This criterion is related to the amount of old information that will be acquired again from a candidate location. Maximizing the overlap can improve the self-localization of the robot. The work presented in this paper follows the same theoretically-grounded approach and, as described in Sections 3 and 4, tries to employ MCDM in search and rescue applications.

Compared with exploration strategies for map building, only few works proposed exploration strategies for autonomous search and rescue. A work that explicitly addressed this problem is [18], which proposes to combine the distance $d(p)$, the expected information gain $A(p)$, and the probability of a successful communication $P(p)$ from a candidate location p in the following utility function:

$$u(p) = \frac{A(p)P(p)}{d(p)}. \tag{3}$$

This strategy has been employed, with good results, in different RoboCup Rescue Virtual Robots Competitions. In this work we experimentally compare the exploration strategies developed with our approach with that proposed in [18], which is explicitly devoted to the same goal. Another exploration strategy for search and rescue is reported in [6], where a formalism based on Petri nets is used to exploit *a priori* information about the victims' distribution (e.g., if they are uniformly spread or concentrated in few clusters) to improve the search.

3. MULTI-CRITERIA DECISION MAKING

When designing an effective exploration strategy for exploring initially unknown environments, the main challenge is to achieve a good global (long-term) performance by means of local (short-term) decisions that are made on the basis of partial knowledge. In our scenario, the partial knowledge is given by the current map built by the robots and short-term decisions are made by evaluating a number of alternatives, i.e., candidate observation locations on the frontiers between the explored and unexplored space, and by selecting the best one. The "goodness" of an observation location can be measured with respect to multiple criteria, as we have seen in the previous section. The number of criteria that can be considered is, in principle, unlimited. As the tasks the robots perform become more complex (think, for example, of an exploring robot that has also to find victims, localize fire sources, communicate with a base station, and so on), this number is likely to increase.

In this work, we explicitly consider the evaluation of candidate locations as a multi-objective (or multi-criteria) optimization problem. We have a set C of candidate locations among which we want to choose the "best" one. We denote the set of n criteria considered in the evaluation process as $N = \{1, 2, \ldots, n\}$. Given a candidate $p \in C$ we denote with $u_i(p) \in I$ its *utility* with respect to criterion $i \in N$, where $I \subseteq \mathbb{R}$ represents the set of possible utility values. Note that we assume that all utilities have values over the same set I. The larger the utility $u_i(p)$, the better location p satisfies criterion i. Each candidate p can be associated to a vector of n elements, namely its utilities, $u_p = (u_1(p), u_2(p), \ldots, u_n(p))$. The problem of selecting the "best" candidate observation location comes down to the problem of selecting the optimal candidate location p^* from C.

Dealing with this multi-criteria scenario, the optimality of candidates involves the concept of *Pareto frontier*. Formally, the Pareto frontier of C can be defined as the largest subset $P \subseteq C$ such that for every $p \in P$ there is not any candidate $q \in C$ with $u_i(q) > u_i(p)$ for all $i \in N$. A candidate $q \in C \setminus P$ is said to be *Pareto-dominated* and can be safely discarded, since at least a preferable candidate is guaranteed to exist in P. Therefore, choosing a candidate on the Pareto frontier P is a fundamental requirement to select a "good" candidate. The actual selection is performed via a *global utility function* $u(p) = f(u_p) = f(u_1(p), u_2(p), \ldots, u_n(p))$ that combines together utilities in an aggregate value (well-known examples are the arithmetic and weighted mean). Since computing the Pareto frontier P can be computationally expensive (especially when the number of candidates grows), the selection is usually done by looking directly at the initial set C, namely $p^* = \arg\max_{p \in C} f(u_p)$. It can be easily shown that if $f()$ is a non-decreasing function in every one of its n arguments, then p^* is guaranteed to be on the Pareto frontier. As the previous section shows, the mainstream approach followed in literature to define global utility functions is to combine a pre-determined number of criteria in an *ad hoc* form. Despite it is not explicitly mentioned, almost all these methods are Pareto optimal, since a non-decreasing global utility function is a "natural" choice.

In the following section, we describe Multi-Criteria Decision Making (MCDM) as a general method for defining global utility functions and we discuss some of its advantages and properties that make it a valid tool for defining exploration strategies for autonomous mobile robots.

3.1 Combining Criteria with the Choquet Integral

We introduce and motivate the proposal of MCDM by considering the important aspect of the dependency between criteria, that is often neglected by global utility functions. Criteria that are used to evaluate candidate locations are not always independent. For example, think of criteria that estimate the same feature using different methods, like two criteria that estimate the distance of a candidate location from the current position of the robot according to the Euclidean and Manhattan distance. Intuitively, when combining them into a global utility function, their overall contribution to the global utility of a candidate location should be less than the sum of their individual ones. In this case, a *redundancy* relation holds between criteria. A dual situation occurs when two or more criteria are very different and, in general, can be hardly optimized together. In this case, a *synergy* relation holds between criteria, and their overall contribution should be considered larger than the sum of the individual ones. An example involves the estimated information gain and the overlap. These criteria can be considered synergic, since large utilities for both are very difficult to achieve by a single candidate and candidates that satisfy both criteria reasonably well should be preferred to candidates that satisfy them in an unbalanced way. In order to consider these issues we need a way to define a global utility function that accounts for redundancy and synergy between criteria when combining them. MCDM provides a general aggregation method which can deal with this and with other aspects and that exploits the *Choquet integral* to compute global utilities [10]. Let us introduce it.

We first introduce a (total) function $\mu : \mathcal{P}(N) \to [0,1]$ ($\mathcal{P}(N)$ is the power set of set N) with the following properties: $\mu(\{\emptyset\}) = 0$, $\mu(N) = 1$, and if $A \subset B \subset N$, then $\mu(A) \leq \mu(B)$. That is, μ is a normalized *fuzzy measure* on the set of criteria N that will be used to associate a weight to each group of criteria. The weights specified by the definition of μ describe the dependency relations that hold for each group of criteria. Criteria belonging to a group $G \subseteq N$ are said to be redundant if $\mu(G) < \sum_{i \in G} \mu(i)$, synergic if $\mu(G) > \sum_{i \in G} \mu(i)$, and independent otherwise.

The global utility $f(u_p)$ for a candidate p is computed as the discrete Choquet integral $\mathcal{C}()$ with respect to the fuzzy measure μ using p's utilities:

$$f(u_p) = \mathcal{C}(u_p) = \sum_{j=1}^{n} (u_{(j)}(p) - u_{(j-1)}(p)) \mu(A_{(j)}), \quad (4)$$

where $(j) \in N$ indicates the j-th criterion according to an increasing ordering with respect to utilities, i.e., after that criteria have been permutated to have, for candidate p,

$$u_{(1)}(p) \leq \ldots \leq u_{(n)}(p) \leq 1.$$

It is assumed that $u_{(0)}(p) = 0$. Finally, the set $A_{(j)}$ is defined as

$$A_{(j)} = \{i \in N | u_{(j)}(p) \leq u_i(p) \leq u_{(n)}(p)\}.$$

Using $\mathcal{C}(u_p)$ to compute global utilities allows to consider criteria's importance and their mutual dependency relations.

3.2 Some Properties of MCDM

In this section, we discuss a number of properties of the proposed MCDM approach. A first general feature of the Choquet integral is that, differently from *ad hoc* global utility functions, it can be applied to any number of criteria. Indeed, rigorously speaking, $\mathcal{C}()$ as defined in (4) is not an aggregation function, for which the number of arguments is fixed *a priori*, but an *aggregation operator*. An aggregation operator is a collection of aggregation functions, one for each number n of criteria to be combined. For example, the arithmetic and weighted means are aggregation operators since they basically specify an aggregation technique for every possible number of criteria, while global utility functions like (2) and (3) are aggregation functions suitable only for the set of criteria they have been tailored for. In this sense, we can say that an aggregation operator is more general than an aggregation function. An obvious advantage of using an aggregation operator instead of an aggregation function is the increased flexibility, because adding and removing criteria can be accomplished preserving the way in which they are combined. As we will discuss in the next sections, this feature enables easy refinements of the exploration strategies and facilitates some experimental activities such as assessing the impact of removing or including a criterion.

$\mathcal{C}(u_p)$ enjoys several other properties [10]. Here, we briefly discuss some properties that are significant in connection with the definition of exploration strategies and that characterize MCDM as a suitable approach to define global utility functions.

Increasing monotonicity in each argument

For all $u_p, u'_p \in I^n$,

- if $\forall i \in N,\ u_i(p) \leq u'_i(p)$, then $\mathcal{C}(u_p) \leq \mathcal{C}(u'_p)$,
- if $\forall i \in N,\ u_i(p) < u'_i(p)$, then $\mathcal{C}(u_p) < \mathcal{C}(u'_p)$.

This property can be exploited to guarantee that the maximization of $\mathcal{C}()$ over the set of candidate locations C will select a Pareto optimal candidate. As we discussed before, almost all aggregation functions proposed in literature for exploration strategies satisfy this property.

Stability for linear transformations

For all $u_p \in I^n$ and $r, s \in \mathbb{R}$ with $r > 0$ such that, for all $i \in N$, $ru_i(p) + s \in I$, it holds that

$$\mathcal{C}(ru_1(p) + s, ru_2(p) + s, \ldots, ru_n(p) + s) = r\mathcal{C}(u_1(p), u_2(p), \ldots, u_n(p)) + s.$$

This property ensures the independence of the particular scale in which utilities are measured (up to a linear transformation). In this paper we assume, without any loss of generality, that utilities have values in $I = [0, 1]$; however, any other common scale would have been equivalent. In general, this property is rarely satisfied by aggregation functions proposed in literature, where often criteria are measured with respect to different scales and combined without any normalization (see, for example, [9] and [18]).

Continuity

Given n, the corresponding aggregation function $\mathcal{C}()$ is continuous on I^n. This property prevents the global utility to exhibit irregular variations with respect to small changes of the utility values that are aggregated. When the global utility is computed by adopting exponential or fractional functions (see (2) and (3)), this property is satisfied.

Idempotence

If, for a given p, all $u_i(p) = u \in I$, then

$$\mathcal{C}(u_1(p), u_2(p), \ldots, u_n(p)) = \mathcal{C}(u, u, \ldots, u) = u.$$

This property assures a sort of consistency, namely, if all the criteria are satisfied with the same degree u, then the global utility is u. This property is rarely exhibited by the aggregation functions used in literature, with the drawback that the particular form in which criteria are combined can introduce a bias in the evaluation, for example by implicitly giving more importance to some criteria to the detriment of others.

3.3 Generality of MCDM

Another important advantage of MCDM is its generality. Indeed, different aggregation operators turn out to be particular cases of the Choquet integral, up to a proper choice of weights for the fuzzy measure μ. For instance, a class of aggregation operators that can be expressed with the Choquet integral are *weighted means*. A weighted mean is defined as $\sum_{i=1}^{n} w_i u_i(p)$ where w_i is the weight of criterion i and $\sum_{i=1}^{n} w_i = 1$. This aggregation operator can be obtained from Choquet integral by setting $\mu(\{i\}) = w_i$ for all $i \in N$ and by constraining μ to be additive:

$$\mu(S) = \sum_{i \in S} w_i \ \ \forall S \in \mathcal{P}(N).$$

Note that additivity of μ reflects independence between criteria, namely joint contributions are exactly the sum of marginal ones. Therefore, weighted means should be considered suitable when such independence between criteria holds. Moreover, the arithmetic mean and the k-th criterion projection can be obtained as further particular cases of weighted means by imposing $w_i = 1/n \ \forall i \in N$ and $w_k = 1, w_i = 0 \ \forall i \in N \setminus \{k\}$, respectively. In the context of exploration, this means that the strategy proposed in [14] and based on (1) can be viewed as a special case of MCDM-based exploration strategies. Moreover, also the global utility function proposed in [12] can be viewed as a special case of MCDM, basically being a k-th criterion projection.

A second class of aggregation operators that are special cases of the Choquet integral is composed of *ordered weighted means*. An ordered weighted mean is defined as $\sum_{j=1}^{n} w_j u_{(j)}(p)$ (i.e., a weighted mean in which w_j is the weight of the j-th criterion according to an increasing ordering of utilities). An ordered weighted mean aggregation operator can be obtained from the Choquet integral by setting $\mu(\{i\}) = w_i$ for all $i \in N$ and by defining $\mu(S)$ according to:

$$\mu(S) = \sum_{i=n-|S|+1}^{n} w_i \ \ \forall S \in \mathcal{P}(N).$$

Some further particular cases of ordered weighted means that can be modeled with a proper choice of weights w_i are the minimum and maximum (when $w_1 = 1$ and $w_n = 1$, respectively), the median (when $w_{\frac{n}{2}} = w_{\frac{n}{2}+1} = 0.5$ and n is even or when $w_{\frac{n+1}{2}} = 1$ and n is odd), and the arithmetic mean excluding the two extremes (when $w_1 = w_n = 0$ and $w_i = \frac{1}{n-2} \ \forall i \in N \setminus \{1, n\}$). This shows the possibility offered by MCDM of obtaining completely different global utility functions (and, as a consequence, different behaviors of the robot) by simply setting weights μ. In this sense, we say that MCDM constitutes a general approach for defining exploration strategies.

4. MCDM-BASED EXPLORATION STRATEGIES FOR SEARCH AND RESCUE

We apply the proposed MCDM approach to search and rescue, where mobile robots are deployed in an initially unknown environment with the goal to explore it and locate human victims within a limited amount of time. As discussed in Section 1, this appliation domain offers a challenging scenario to test exploration strategies.

We implemented MCDM-based exploration strategies in an existing robot controller for search and rescue applications. We looked at the participants to the RoboCup Rescue Virtual Robots Competition where different teams compete in developing simulated robotic platforms operating in Urban Search And Rescue scenarios simulated in USARSim [7] (an high fidelity 3D robot simulator). From an analysis based on availability of code and performance obtained in the competition, we selected the controller developed by the Amsterdam and Oxford Universities (Amsterdam Oxford Joint Rescue Forces, AOJRF[1]) for the 2009 competition [19]. The reasons for implementing MCDM-based exploration strategies in an existing controller are that we can focus only on the exploration strategies, exploiting existing and tested methods for navigation, localization, and mapping and that we have a fair way to compare our exploration strategies with that originally used in the controller. In the following we describe the original controller and how we modified it to implement MCDM-based strategies.

4.1 The AOJRF Controller

In this section, we describe some of the controller's features that are relevant to the scope of this paper (please refer to [18] for a complete description).

The controller manages a team of robots. The robotic platform used is a Pioneer P3AT, whose basic model and sensors are provided with the USARSim simulator. The map of the environment is maintained by a base station, whose position is fixed in the environment, and to which robots periodically send data. The map is two-dimensional and represented by two occupancy grids. The first one is obtained with a small-range (typically 3 meters) scanner and constitutes the *safe area*, i.e., the area where the robots can safely move. The second one is obtained from maximum-range scans (typically 20 meters) and constitutes the *free area*, i.e., the area which is believed to be free but not yet safe. Moreover, a representation of the *clear area* is also maintained as a subset of the safe area that has been checked for the presence of victims (this task is accomplished with simulated sensors for victim detection). Given a map represented as above, a set of boundaries between safe and free regions are extracted and considered as frontiers. For each frontier, the middle point is considered as a candidate location to reach. The utility of a candidate location p is evaluated by combining the following criteria:

- $A(p)$ is the amount of the free area beyond the frontier of p computed according to the free area occupancy grid;
- $P(p)$ is the probability that the robot, once reached p, will be able to transmit information (such as the perceived data or

[1] `http://www.jointrescueforces.eu/`

the locations of victims) to the base station (whose position in the environment is known), this criterion depends on the distance between p and the base station;

- $d(p, r)$ is the distance between p and current position of robot r, this criterion can be calculated with two different methods: $d_{EU}()$, using the Euclidean distance, and $d_{PP}()$, using the exact value of the distance returned by a path planner.

Given these criteria, the global utility for a candidate p is calculated using function (3). We will refer to the exploration strategy using this global utility function as the "AOJRF strategy".

The allocation of candidate locations to robots is performed with the following algorithm, which is executed by each robot independently, knowing (from the base station) the current map and the positions of other robots [18]:

1. compute the global utility $u(p, r)$ of allocating each candidate p to each robot r using (3) where $d(p, r)$ is calculated using the Euclidean distance $d_{EU}()$ (namely using an underestimate of the real distance),
2. find the pair (p^*, r^*) such that the previously computed utility is maximum, $(p^*, r^*) = \arg\max_{p,r} u(p, r)$,
3. re-compute the distance between p^* and r^* using $d_{PP}()$ with the path planner (namely considering the real distance) and update the utility of (p^*, r^*) using such exact value instead of the Euclidean distance,
4. if (p^*, r^*) is still the best allocation, then allocate robot r^* to location p^*, otherwise go to Step 2,
5. eliminate robot r^* and candidate p^* and go to Step 2.

The reason behind the utility update of Step 3 is that computing $d_{PP}()$ requires a considerable amount of time. Doing this for all the candidate locations and all robots would be not affordable in the rescue competition, since a maximum exploration time of 20 minutes is enforced.

4.2 Developing MCDM-based strategies

We now describe the changes we made to the original controller to include our MCDM-based strategies.

MCDM

criteria	$\mu()$	criteria	μ
A	0.5	A, d	0.95
d	0.3	A, P	0.7
P	0.2	d, P	0.4

MCDMb

criteria	$\mu()$	criteria	$\mu()$
A	0.4	d, P	0.25
d	0.25	d, b	0.35
P	0.1	P, b	0.25
b	0.25	A, d, P	0.75
A, d	0.75	A, d, b	0.9
A, P	0.5	A, P, b	0.75
A, b	0.65	d, P, b	0.45

MCDMw

criteria	$\mu_1()$	$\mu_2()$
A	0.6	0.4
d	0.1	0.5
P	0.3	0.1
A, d	0.8	0.95
A, P	0.9	0.5
d, P	0.3	0.5

Table 1: Weights used for the MCDM-based strategies.

The first MCDM-based strategy we propose adopts the same criteria of the AOJRF strategy (i.e., A, P, and d, as described above), but combines them with the MCDM approach. Basically, we replace function (3) with function (4), with the weights reported in Tab. 1 (top-left). We call this the "MCDM strategy".

Choosing a particular set of weights can be tricky. In this phase, the designer considers the application domain and defines the importance of single and groups of criteria. We remark that searching for the "best" set of weights is an ill-posed problem in the context of MCDM. MCDM is not a method to determine the best exploration strategy, but provides a flexible and general tool to combine criteria. Therefore, we assigned weights manually, considering the search and rescue context. For example, the MCDM strategy assigns more importance to A than to P and d (see Tab. 1 (top-left)), pushing the robot to discover new areas, even covering long distances or risking a loss of communication. The joint contribution of d and P is inhibited by establishing redundancy between them. On the other side, a synergy holds between d and A, privileging locations satisfying these criteria in a balanced way. This manual method for assigning weights does not scale well with the number n of criteria. Indeed, $2^n - 2$ weights have to be assigned. However, specification of weights is done at design-time and there are semi-automated techniques to compute weights for large sets of criteria [10].

To apply MCDM, utilities have to be normalized to the chosen common scale $I = [0, 1]$. We note that the robot's decision at any step depends only on C and not on previous decisions and previous sets of candidate locations. Hence, we use a linear relative normalization. For example, given a robot r, the utility of a candidate p related to the distance $d()$ is normalized using $u_d(p, r) = 1 - (d(p, r) - \min_{q \in C} d(q, r))/(\max_{q \in C} d(q, r) - \min_{q \in C} d(q, r))$. This poses a problem for normalizing the updated utility in Step 3, since it would require to determine the path for every candidate location, making the 20 minutes limit too strict to achieve an acceptable performance (recall that $d_{PP}()$ is computationally expensive). To deal with this problem we use the following procedure in Step 3: once computed $d_{PP}(p^*, r^*)$, we normalize it by using the previously calculated values $d_{EU}(p, r^*)$ for other candidates $p \in C$.

The second MCDM-based strategy we propose shows the flexibility of MCDM in adding a new criterion, i.e., the robot's battery remaining charge b. Explicitly considering the battery can improve exploration by preventing the robot from making decisions it cannot complete (e.g., selecting a location not reachable with the residual energy). To compute $u_b(p)$ we need an estimate of the energy spent for reaching p. We consider a very simple model in which the power consumption is translated in a time interval. In order to estimate the time needed to reach a location p we consider the path the robot should follow in terms of linear segments and rotations. By approximating the linear and angular velocities of the robot as constants, we can derive estimates of the time $b(p)$ needed to reach p. Obviously, the smaller $b(p)$ the larger $u_b(p)$. Notice that b and d show an evident dependency relation given by the fact that long traveling distances often correspond to long times. However, despite this similarity, including b in the set of criteria can, to some extent, provide more informed decisions since it captures also the difficulty for covering a path which generally is not captured by d (consider, for example, short but winding paths that could require lot of time and battery). Modeling a redundancy relation between these two criteria is the proper way to include both of them in the decision-making process without unbalancing decisions toward the common selection principle encoded in b and d. We denote the strategy including b as "MCDMb strategy", whose weights are reported in Tab. 1 (top-right). As it can be seen, the weight assigned to the set $\{d, b\}$ is lower than the sum of weights of b and d. Redundancy and synergy are also defined on sets of more than two criteria; for example, criteria d, P, and b are redundant and the weight of the set $\{d, P, b\}$ is smaller than the sum of the weights of its elements.

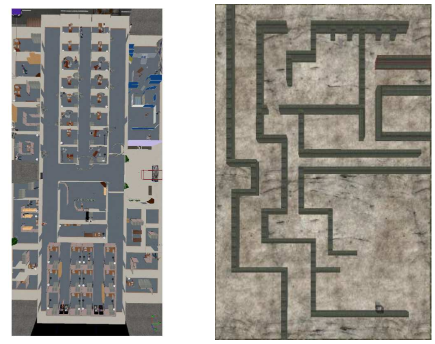
We also show how MCDM can be adopted for defining different *behaviors* in exploration. Broadly speaking, a behavior defines the preferences according to which the robot selects observation locations. Given a set of criteria, a behavior is associated to the particular set of weights of those criteria. By changing the weights during exploration, we can switch between different behaviors, varying the criteria's importance that drive robot decisions. This technique allow us to improve the exploration strategy's adaptability to different situations. Hence, we define a third MCDM-based strategy, called the "MCDMw strategy", whose weights are reported in Tab. 1 (bottom). This strategy encloses two different behaviors, given by the sets of weights denoted as μ_1 and μ_2, defined over the original set of criteria of the MCDM strategy (i.e., A, P, and d, as described above). In addition, we define the following policy for switching through behaviors. The weights defined by $\mu_1()$ are used during the first 10 minutes of search while those defined by $\mu_2()$ are used during the last 10 minutes. The first set of weights encodes an aggressive behavior oriented towards the maximization of the new area. This behavior is reasonable during the first part of the search when a long remaining time is left and the robot can privilege the amount of new area even if long paths have to be followed. Differently, the second set of weights induces a more conservative behavior. This behavior accounts for the fact that remaining time is short and gives more importance to distance ($\mu_1(d) = 0.1$ while $\mu_2(d) = 0.5$).

5. EXPERIMENTAL EVALUATION

In the first experiments we evaluate the performance of the MCDM strategy when compared with other strategies. We consider the AOJRF strategy (corresponding to (3)), the WS strategy (corresponding to (1) with $\beta = 1$), and the DIST strategy, by which locations are selected simply by minimizing d (i.e., choosing always the nearest location). AOJRF and WS are continuous and increasingly monotonic aggregation functions. These two strategies guarantee a Pareto optimal selection, however AOJRF strategy lacks in flexibility since including further criteria would require to re-define the aggregation technique, while WS can be considered as a special case of MCDM-based strategies (see Section 3.3). Nevertheless, AOJRF and WS have been proved to achieve good results in practice, therefore, by comparing the MCDM strategy with them, we aim at deriving insights on how performance changes when using a more theoretically-grounded way to define global utility functions. DIST is a very simple strategy that can be viewed as a particular case of MCDM-based strategy. Indeed, it can be obtained by restricting the set of criteria to the singleton d (see Section 3.3). By comparing MCDM and DIST we aim at confirming that making more informed local decisions actually results in a better global performance.

We considered teams of one or two robots, as in [18] (note that the maximum number of robots allowed in the RoboCup Rescue Virtual Robots Competition is 4). The robots are deployed in the two indoor environments of Fig. 1 that show different characteristics. Map A is cluttered and composed of corridors and many rooms, while Map B is characterized by the presence of open spaces. A configuration is defined as an environment, a team of robots deployed in it, and the exploration strategy adopted. For each configuration, we executed 10 runs (with randomly selected starting locations for the robots) of 20 minutes each. We assess performance by measuring the amount of free, safe, and clear area at each minute of the exploration. Due to space limitations, we report only data on safe area (free area is less significant and clear area is similar to the safe area).

Figs. 2 and 3 show the results of the first experiments with a team of one and two robots, respectively. Histograms compare the number of runs in which a strategy obtained the largest amount of safe area at the end of the 20 minutes exploration. Graphs show how the



(a) Map A (b) Map B

Figure 1: The maps used for tests.

mapped safe area varies with time (each point is the average over 10 runs).

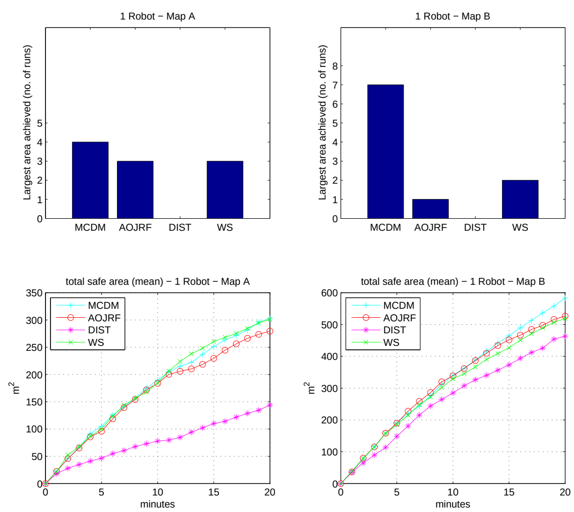


Figure 2: Comparison between MCDM and other exploration strategies with one robot.

The MCDM strategy discovered the largest area in the majority of runs, outperforming (on average) other strategies. According to an ANOVA test, the averages of the total safe area (in Map A) are statistically significantly different between DIST and each one of the other three strategies. Differences between MCDM, AOJRF, and WS are not statistically significant in Map A. In Map B, the MCDM strategy shows a statistically significant difference when compared to DIST and AOJRF, while the statistical difference between MCDM and WS is slightly acceptable. These findings reflect an interesting insight associated to the different characteristics of the two environments. Map A is cluttered and, exploring it, the robots deal with a relatively large number of frontiers among which to choose (30 candidate locations on average at each step with one robot and 40 with two robots). Map B is characterized by open spaces, resulting in a smaller number of candidate frontiers (5 candidate locations on average at each step with one robot and 8 with two). However, despite their large number, frontiers in Map A are very similar in the contribution they can give to the

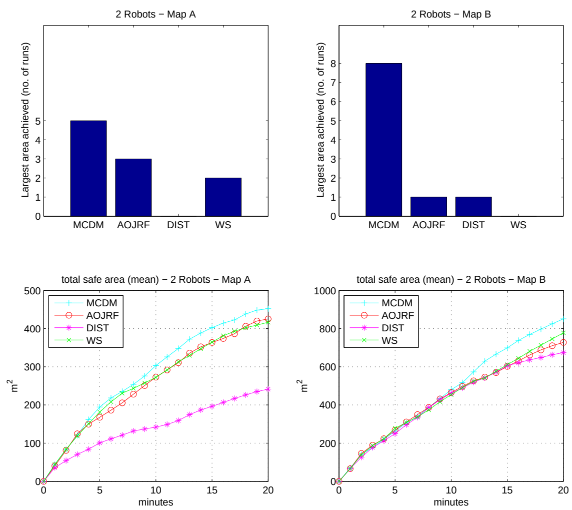


Figure 3: Comparison between MCDM and other exploration strategies with two robots.

explored area. Differently, in Map B the situation in which one alternative is remarkably better than others is more frequent. Consider, for instance, a frontier that lies close to an obstacle (from where an observation will return a small new area) and another one in front of an open space. In such situation, the benefits provided by a "right choice" would be more evident. This is what happens during the exploration of Map B, showing why differences between strategies are statistically significant in this environment. This basically confirms the single robot results presented in [5], enforcing the idea that when very different alternatives are present and making a good choice is very rewarding, MCDM-based exploration strategies achieve satisfactory results.

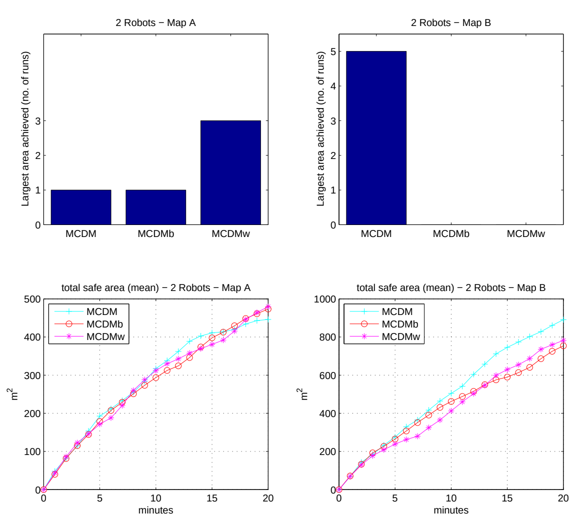
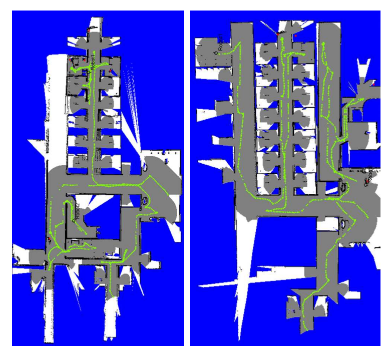


Figure 4: Comparison between the MCDM-based strategies.

Fig. 4 shows the performance of the three MCDM-based exploration strategies with two robots (we omit results with one robot, for which the same considerations can be drawn). A first comparison that is worth doing is between MCDM and MCDMb, to assess the impact of b's inclusion in evaluating a location. When adopted for exploring Map A, these two strategies performed similarly, not showing any statistically significant difference in the total safe area. However, the effects of introducing criterion b can be noted by looking at the final maps built by the robots. A representative example is shown in Fig. 5, which reports the two maps obtained with MCDM and MCDMb after a run. Considering that the criterion b pushes the robots to discard locations that require complicate paths with several rotating maneuvers, the robots save time avoiding to deeply explore corners, rooms, and other cluttered parts of the environment, preferring corridors and open spaces. The result is that the obtained map, from the one hand, is less precise but, from the other hand, is more representative of the general topology of the environment. This kind of map can be more useful to first responders in giving a broad idea of the topology of the environment (as discussed in [3]). The introduction of the criterion b does not show the same qualitative behavior in Map B, where the presence of open spaces makes intricate paths very rare. The employment of this criterion in an open space is not justified by the characteristics of the environment, showing an example where "too-much informed" local decisions could achieve a not so good global performance.



(a) MCDM (b) MCDMb

Figure 5: An example of maps obtained after an exploration.

Adopting different behaviors with the MCDMw strategy led to the best results in Map A. Roughly speaking, this strategy combines the benefits of MCDM and MCDMb strategies. In the first half of the exploration a more aggressive behavior is adopted, trying to maximize the explored area. Then, as the residual time decreases, the strategy becomes more conservative, trying to save time avoiding cluttered zones. In Map B, the employment of μ_1 in the first part of the exploration showed the main drawback of a very aggressive behavior: its vulnerability to decisions that happen to be not as good as expected. In a number of situations, μ_1 pushed the robots to cover long distances for reaching locations with potentially large amounts of new area that, due to information gain estimation errors, were not so informative once reached. This is the reason why MCDM and MCDMw curves are relatively separated in the first 10 minutes of exploration (Fig. 4).

Fig. 6 depicts an example of paths followed by a robot when employing the three MCDM-based strategies. The starting location in all the three cases is at the center of the top corridor. All the strategies initially drive the robot toward the right part of the top corridor until the first difference can be observed in the path of MCDMw. Its aggressive behavior pushed the robot to go back at the intersection with the vertical corridor to obtain a wide view over the free space. MCDM and MCDMb start to significantly differ in the bottom end of the central vertical corridor. More precisely, MCDMb's path resulted more regular than that of MCDM. Indeed, MCDM drove the robot to explore a sequence of rooms while, with MCDMb, the robot chose to enter the bottom horizontal corridor. MCDMw's paths avoided all the rooms in the right part of the environment (first 10 minutes) but performed a more detailed exploration in the left part of the map (last 10 minutes of the exploration). This example shows how obtained paths are coherent with the design principles of each strategy and demonstrates that the decision-theoretic framework of the MCDM-based strategies can provide some level of predictability.

Figure 6: Example of paths of MCDM-based strategies.

From our results, we can say that MCDM can be an effective method for defining good exploration strategies in search and rescue applications. Local decisions made with MCDM-based exploration strategies resulted in a comparable and sometimes better performance, when compared to other exploration strategies proposed in literature. In particular, MCDM showed significant improvements in situations (like those faced in Map B) where making the right decision is more rewarding. In addition, MCDM presents a remarkable flexibility in composing criteria that can be exploited to add new criteria or to define multi-behavioral strategies that can adapt to different situations.

6. CONCLUSIONS

In this paper, we have presented the application of the MCDM decision-theoretic approach to the definition of exploration strategies for search and rescue. We have shown that MCDM provides a general and flexible way for developing utility functions for evaluating candidate observation locations. Experimental results show that MCDM-based exploration strategies achieve a good performance, when compared with *ad hoc* strategies used in exploration.

Possible future work includes the development of automatic techniques to set the values of weights in MCDM, in order to further simplify the inclusion of new criteria in the evaluation of candidate locations, and the application of MCDM-based strategies to other domains, like planetary exploration. Another interesting direction is working on the robot-frontier allocation, trying to achieve a closer integration between evaluation of candidate locations and coordination of robots.

7. REFERENCES

[1] F. Amigoni and V. Caglioti. An information-based exploration strategy for environment mapping with mobile robots. *ROBOT AUTON SYST*, 5(58):684–699, 2010.

[2] F. Amigoni and A. Gallo. A multi-objective exploration strategy for mobile robots. In *Proc. ICRA*, pages 3861–3866, 2005.

[3] B. Balaguer, S. Balakirsky, S. Carpin, and A. Visser. Evaluating maps produced by urban search and rescue robots: Lessons learned from robocup. *AUTON ROBOT*, 27(4):449–464, 2009.

[4] S. Balakirsky, C. Scrapper, S. Carpin, and M. Lewis. Usarsim: a robocup virtual urban search and rescue competition. In *Proc. of SPIE*, 2007.

[5] N. Basilico and F. Amigoni. Exploration strategies based on multi-criteria decision making for an autonomous mobile robot. In *Proceedings of the European Conference on Mobile Robots (ECMR)*, pages 259–264, 2009.

[6] D. Calisi, A. Farinelli, L. Iocchi, and D. Nardi. Multi-objective exploration and search for autonomous rescue robots. *J FIELD ROBOT*, 24(8-9):763–777, 2007.

[7] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper. Usarsim: a robot simulator for research and education. In *Proc. ICRA*, pages 1400–1405, 2007.

[8] H. Choset. Coverage for robotics: A survey of recent results. *Ann. Math. Artif. Intell.*, 31(1-4):113–126, 2001.

[9] H. Gonzáles-Baños and J.-C. Latombe. Navigation strategies for exploring indoor environments. *INT J ROBOT RES*, 21(10-11):829–848, 2002.

[10] M. Grabisch and C. Labreuche. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. *4OR-Q J OPER RES*, 6(1):1–44, 2008.

[11] K. H. Low, J. Dolan, and P. Khosla. Adaptive multi-robot wide-area exploration and mapping. In *Proc. AAMAS*, pages 23 – 30, 2008.

[12] A. Marjovi, J. Nunes, L. Marques, and A. de Almeida. Multi-robot exploration and fire searching. In *Proc. IROS*, pages 1929 –1934, 2009.

[13] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser. Efficient informative sensing using multiple robots. *J ARTIF INTELL RES*, 34(1):707–755, 2009.

[14] C. Stachniss and W. Burgard. Exploring unknown environments with mobile robots using coverage maps. In *Proc. IJCAI*, pages 1127–1134, 2003.

[15] S. Tadokoro. *Rescue Robotics*. Springer-Verlag, 2010.

[16] S. Thrun. Robotic mapping: A survey. In *Exploring Artificial Intelligence in the New Millenium*, pages 1–35. 2002.

[17] B. Tovar, L. Munoz-Gomez, R. Murrieta-Cid, M. Alencastre-Miranda, R. Monroy, and S. Hutchinson. Planning exploration strategies for simultaneous localization and mapping. *ROBOT AUTON SYST*, 54(4):314 – 331, 2006.

[18] A. Visser and B. A. Slamet. Including communication success in the estimation of information gain for multi-robot exploration. In *Proc. WiOPT*, pages 680–687, 2008.

[19] A. Visser *et al.* Amsterdam Oxford joint rescue forces - team description paper - Virtual Robot competition - Rescue simulation league - robocup 2009. In *Proc. of RoboCup*, 2009.

[20] B. Yamauchi. A frontier-based approach for autonomous exploration. In *Proc. CIRA*, pages 146–151, 1997.