DRDC-RDDC-2015-P187

Static Target Search Path Planning Optimization with Heterogeneous Agents

Jean Berger
DRDC Valcartier
Quebec City, Canada
jean.berger@drdc-rddc.gc.ca

Nassirou Lo
T-OptLogic Ltd.
Quebec City, Canada
nassirou.lo@t-optlogic.com

Mohamed Barkaoui
Carthage University
Bizerte, Tunisia
mohamed.barkaoui@fsb.rnu.tn

Abstract—**As discrete multi-agent static open-loop target search path planning known to be computationally hard recently proved to be solvable in practice in the homogeneous case, its heterogeneous problem counterpart still remains very difficult. The heterogeneous problem introduces broken symmetry reflected by dissimilar sensing ability/capacity, agent capability and relative velocity and, is further exacerbated when operating under near real-time problem-solving constraints, as key decision variables grow exponentially in the number of agents. Departing from the homogeneous agent model already published, new integer linear and quadratic programming formulations are proposed to reduce computational complexity and near-optimally solve the discrete static search path planning problem involving heterogeneous agents. The novelty consists in taking advantage of typical optimal path solution property to derive new tractable problem models. At the expense of a slightly accrued computational complexity, the proposed quadratic integer program formulation conveys considerable benefit by keeping key decision variables linear in the number of agents. The convexity property of its defined objective function further allows ensuring global optimality when a local optimum is computed. Special agent network representations capturing individual agent decision moves are also devised to simplify problem modeling and expedite constraint modeling specification. As a result, cost-effective quadratic program implementation for realistic problems may be achieved to rapidly compute near-optimal solutions, while providing a robust bound on solution quality through Lagrangian relaxation.**

Keywords: search path planning, search and rescue, quadratic programmimng, linear programming, heterogeneous agents

I. Introduction

Search path planning aimed at detecting a static target is common to a variety of civilian and military problem domains. These include homeland security and emergency management to name a few, and are mostly instantiated as search and rescue (SAR) problems. The discrete SAR or so-called optimal searcher path problem basically defined for a static or stationary target setting is known to be NP-Hard [1]. SAR problem taxonomy typically include the following search classes and/or characteristics: one-sided search in which targets remain passive or non-responsive regarding searcher's actions, two-sided search, describing a diversity of target attitudes or behaviors (cooperative, non-cooperative or anti-cooperative) toward the searcher, stationary Vs. moving target search, discrete Vs. continuous time and space search (efforts indivisibility/divisibility), static/dynamic as well as open and closed -loop decision models. Other features may further define specific SAR instances, namely, sought objective, perfect/imperfect searcher's sensing capability, target and searcher multiplicity and diversity. Early work was first pioneered by search-theoretic approaches [2], [3] based on mathematical framework mostly focusing on effort allocation (time/energy spent per visit) rather than explicitly path-centric decision models. Efforts were then increasingly spent to refine algorithms to solve dynamic problem settings and variants [4]-[7]. In other respect, many useful search path planning contributions alternatively derive from the robotics domain namely, robot motion planning [8], terrain acquisition [9], [10] and coverage path planning [11]-[13]. Reported work on robot motion planning has primarily focused on coverage [14], [15] problems, proposing constrained shortest path type solutions. It typically looks at uncertain environment search problems for sparsely distributed static and unknown targets and obstacles, assuming very limited prior domain knowledge. Although different in nature and complexity, related work on pursuit-evasion problems has been independently referenced as well [16]. For various taxonomies and comprehensive surveys on target search problems from the field of search theory, distributed artificial intelligence and robotic control including pursuit-evasion settings, we further direct the reader to the references [17],[5], [18]-[20] and [16] respectively.

However, regardless of the extensive work contributions published so far on various models, enduring SAR problem computational complexity still persists [5], [20]. Approximate problem-solving techniques and heuristics recently proposed may satisfactorily work for specific constraints (e.g. hard constraints relaxation) or conditions to keep the problem manageable, but ultimately face the curse of dimensionality [21]-[23],[5],[20],[24],[25]. Still delivering approximate solution, related methods struggle to convincingly estimate real performance optimality gap for practical size problems, casting some doubts on their reported efficiency. But, recent progress proved discrete static open-loop target search path planning with homogeneous agents to be efficiently manageable in practical cases [26] while providing a bound on optimal solution quality. However, solving efficiently the multi-agent heterogeneous problem remains nonetheless elusive. Considered as a natural and simple extension of the homogeneous case, the multi-agent heterogeneous search path planning problem is often mistakenly perceived as a trivial generalization of the homogeneous formulation. Consequently and regardless of its prevalence, it received little attention so far and continues to be largely overlooked in scientific literature. However, agent capability diversity and broken symmetry induced by dissimilar sensing capability and relative velocity, substantially magnifies computational complexity, making it intractable.

In this paper, new mixed-integer linear and quadratic programming (MIP) formulations are alternately proposed to near-optimally solve the discrete static search path planning problem with heterogeneous agents**.** In that setting, a team of centrally controlled heterogeneous agents searches an area (grid) to maximize probability of target detection (probability of success), given a prior cell occupancy probability distribution. The search team is characterized by agent-dependent traveling speed and imperfect sensing capabilities. Imperfect readings in this context mainly refer to missed target events, and exclude false positive observations. However, agent sensor readings are assumed to be false-positive free given the nature of the searched target, meaning that a positive observation readily and unambiguously conducts to target detection and identification. The open-loop property of the problem confers objective function separability over cells. This consideration enables objective function coefficients pre-computation in advance resulting in a new, simple and convenient mixed-integer formulation. Proposed approaches take advantage of typical optimal path solution properties to derive tractable problem models. It namely exploits the fact that path optimality does require a limited number of visits per cell in practice as the gain for additional visits quickly drops in comparison to other prospect or competing site visits. Agent proximity from promising cells which tends to make more likely near future additional visits by the same agent is also exploited. Combining these properties makes cell assignment to a limited number of agents quite reasonable and desirable in practice, while preserving solution optimality. Typically aimed at efficiently maximizing target detection probability of success, the proposed quadratic program turns out to be particularly attractive and simple in most problem conditions, reducing substantially the relative number of additional decision variables required in comparison to alternate linear formulations. Its objective function conveys the suitable convexity property guaranteeing global optimality when solving the problem. Easily extendable to accommodate additional number of agents to service a given cell, the revisited decision model is also more appealing computationally offering efficiency gain. Agent network representations describing possible individual agent actions over time are further proposed to simplify problem modeling and considerably ease constraint specification. Cost-effective implementation of the quadratic program allows for practical size problems to rapidly compute a near-optimal solution, while providing an upper bound on the optimal path quality using Lagrangian programming relaxation. The computable upper bound defines an objective measure to fairly compare performance gap assessment over various techniques. Computational results prove the proposed approach very efficient and to significantly outperform limited myopic search path planning for a random sample of problem instances.

The content of the paper is organized as follows. The static open-loop search path planning problem with heterogeneous agents is first outlined in Section II. It briefly defines the main problem characteristics and the pursued objective. A preliminary mixed-integer linear programming formulation model is then introduced in Section III. It describes the main modeling concept combining network representation and mathematical modeling. An agent path reconstruction procedure is also depicted. Revisited linear and quadratic problem formulations are then presented in Section IV. The main intent is to ultimately reduce computational complexity and obtain a simplified tractable decision model to efficiently solve the original problem. Section V briefly reports computational results. The value of the advocated MIP quadratic programming (QP) approach is compared to an alternate myopic heuristic for reference purposes. A summary of the main highlights and conclusive remarks are finally given in Section VI.

II. Problem

A. *Target Search Path planning with Heterogeneous Agents*

The discrete centralized static search path planning problem consists in a team of n heterogeneous agents standing as mobile sensors, aimed at finding a static target located in a specific area of interest over a predetermined time horizon T. Search capability diversity reflected by agent (and cell) - dependent conditional probability of detection and relative velocity mainly characterizes team heterogeneity. The objective of the search and rescue mission consists in having the search team maximize the probability of target detection. The static feature of the problem refers to a system steady state, which assumes that the environment does not significantly evolve or similarly go through state transitions during problem-solving. Pictured as a grid representation, the search region defines a bi-dimensional mesh N composed of $|N|$ regular (square) cells, populated by a single stationary target. The target is assumed to occupy a single cell and its precise location is unknown. Based on domain knowledge, a prior target location probability density distribution defines individual cell occupancy, describing a grid cognitive map. The cognitive map or uncertainty grid is a knowledge base portraying a local environment state representation, capturing

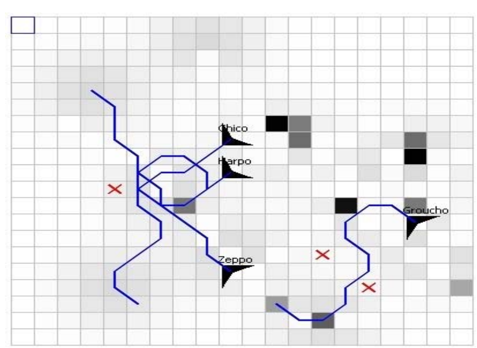

Figure 1. A cognitive map for a 4-agent team over episode *t*. Beliefs on target cell occupancy are represented through multi-level shaded cell areas. Anticipated agent plans are reflected as possible upcoming paths.

target occupancy belief distribution and agent positions during a given period. A typical cognitive map is displayed in Figure 1 for a specific time interval t. Without loss of generality, the target is assumed to be located in the bounded environment. Accordingly, should the target be located outside the sought region, a virtual inaccessible and non-observable cell would be artificially appended to the grid with the corresponding prior belief. As a result, target cell occupancy probabilities over the grid sum up to one.

Cell visit time defining episode duration is assumed to be agent-dependent but constant over the grid. Air vehicles are assumed to fly at slightly different altitudes to avoid colliding with one another. Other agent types resort to a predefined hierarchical authority structure to resolve eventual moving conflicts. A search path solution conveys agent path plans maximizing overall target detection.

B. *Path Planning*

Exploiting agent's position (cell location) and possible legal moves information, a centralized decision-making process determines best agent's decision at each episode. Possible decisions at each step include all possible moving directions to alternate neighbor cells. It is obviously assumed that the agent may decide to stay in the current host cell during the next episode. The objective primarily consists in selecting base-level control action moves across the grid in order to maximize probability of success (target detection).

C. *Measure of Performance - Probability of Success*

A suitable metric to measure search path plan performance is defined as the probability of successful target detection by the team of searching agents at the end of the task. It may be expressed by summing up the products of conditional probability of detection and probability of target occupancy over visited grid cells.

The probability of success (*POS*) for a particular path solution can then be stated as follows:

$$POS = \sum_{c \in N} p_{c0} \left(1 - \left(\prod_{k=1}^{n} \left(1 - p_{cck}\right)^{l_k(c)} \right) \right) \quad (1)$$

where p_{c0} refers to the current probability/belief of target cell occupancy, whereas p_{cck} is the probability that agent k visiting cell c correctly detect the target, given that the target is present in cell c. $l_k(c)$ refers to the overall number of visits in cell c by agent k. Agent sensors are assumed to be free of false-positive readings, leading systematically to negative observation outcomes when vacant cells are visited. This is basically due to the nature of the target being sought, which can be easily and unambiguously identified on a positive sensor reading. It should be noticed from Eq. (1) that solution quality (*POS*) remains invariant over cell visit ordering, and sequence of agent visits for a particular cell. This property confers to a solution that any particular cell permutation preserving path feasibility from a given path, are equivalent. In other words, only visited cells ultimately matter, and not when those sites are specifically visited. More computationally convenient, the dual probability of failure *POF* (1-*POS*) or non-detection is alternatively defined by:

$$POF = \sum_{c \in N} p_{c0} \left(\prod_{k=1}^{n} \left(1 - p_{cck}\right)^{l_k(c)} \right) \quad (2)$$

In the current setting, agent sensor's range defining visibility or footprint (coverage of observable cells given the current sensor position) is limited to the cell being searched.

III. Preliminary Mixed-Integer Linear Programming Model Formulation

A. *Graph Representation*

Mainly inspired from a previous work [26], an agent network representation is exploited to simplify problem and constraint modeling ultimately reducing run-time, as it implicitly captures key constraints by construction. Such constraints comprise itinerary path length bounded by relative velocity over time horizon T, legal moves, and single route agent solution (multiple disconnected subtours agent solution are prohibited) which could unnecessarily degrade problem-solving performance should they be explicitly included in the analytical decision model.

The basic graph description is summarized as follows. Let $\mathcal{G}_k=(\mathcal{V}_k, \mathcal{A}_k)$ be a directed acyclic graph (the grid network) coupled to agent $k \in \eta=\{1,...,n\}$. $\mathcal{V}_k$ denotes a set of vertices defining possible agent k's states. A state (c,t) refers to the cell location $c \in N$ (position) an agent is operating into during episode $t \in \mathcal{T}_k=\{0,1,...,T_k -1\}$, where T_k symbolizes agent k's path length. $\mathcal{A}_k$ designates the set of arcs (i,j) connecting a prior state i to a posterior state j, where $i,j \in \mathcal{V}_k$. An arc captures an episodic state transition resulting from an agent's action selected in the set $A=\{a_1, a_2,…, a_9\}$ of possible moving directions to alternate neighbor cells including its host location itself. Accounting for team member speed diversity, agent network k involves T_k stages (or moves) covering all feasible paths mapping possible state transitions. T_k depends on s_k, the relative speed of agent k with respect to maximum agent team member speed ($\max_k s_k = 1$). Accordingly, agent k network includes $T_k=s_kT$ periods. For agent network generation convenience, an integer T_k approximation is assumed as no partial observations resulting from incomplete search episodes can be explicitly accounted for. This consideration remains consistent with the fact that a promising cell is still interesting with respect to other competitive sites, whether or not it is visited for a partial or complete period. The approach is rather convenient, as probability of success primarily focuses on cell visit cardinality as opposed to detailed cell visit sequence patterns. As a result, agent-dependent time interval duration has no impact computationally. Fictitious origin (o) and destination (d) location nodes are artificially introduced to conveniently define legitimate path in the graph. A partial view of an agent network is exhibited in Fig. 2 over two successive episodes t and t+1. An integer binary flow decision variable x_{ijk} related to an arc $(i,j) \in \mathcal{A}_k$ defines a basic agent path's construct. Accordingly, agent k's path solution includes

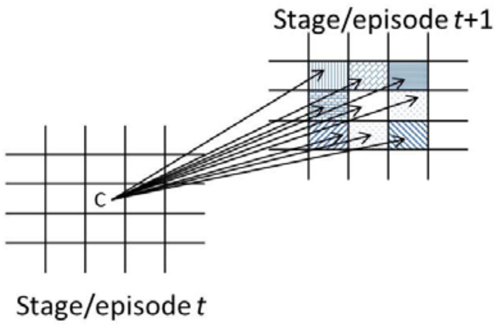

Figure 2. Agent network representation describing state nodes and possible transitions over consecutive episodes t and t+1. Nodes denote agent states (position c, episode t). Arcs capture state transitions defining legal moves to neighboring/host cells over successive episodes. Agent path is constructed by moving along the arcs from stage 1 to T.

arcs $(i,j) \in \mathcal{A}_k$ when x_{ijk} = 1. These flow decision variables are coupled to alternate integer binary visit decision variables v_{jlk} reflecting that l visits on cell j are part of the physical agent k's path solution (v_{jlk} =1) in minimizing expected non-detection over the grid. From the initial state $i_0(k)$, a feasible agent path may be built traveling along arcs connecting o to d nodes, episodically instantiating flow decision variables. The strategy to duplicate the grid structure at each episode to uniquely define agent states (position c, episode t) is very convenient and efficient to generate solutions that implicitly satisfy path length constraint and exclude unsuitable solution composed of disjoint subtours. The resulting network structure depicting a directed acyclic graph may then be used to construct a legal agent path solution through a temporal sequence of binary integer flow decision variable instantiations. An agent network is defined by $|N|\ T_k$ nodes and $|N|\ T_k\ |A|$ arcs.

B. *Mathematical Model*

As cell probability of target detection uniquely depends on local visits being conducted at the site, problem formulation for the *POF* objective function expressed in (2) proves to be separable. Decision variables may then be partitioned in subsets with separate contributions to the objective function. Separability makes feasible the pre-computation of cell target detection probability contribution values in advance, since these contributions essentially rely on local visits on a cell c.

The parameters and variables used to specify a basic problem model formulation are described as follows:

Parameters:

- η : set of heterogeneous agents $\{1,2,\ldots,n\}$.
- N: set of cells defining the grid search area $\{1,2,..,|N|\}$.
- T: problem time horizon.
- p_{c0}: initial belief of cell c target occupancy over the grid.
- V: maximum number of visits on cell c by an agent.
- s_k: relative speed of agent k to maximum agent team member speed.
- p_{cck}: conditional probability of 'correct' target detection on a visit in cell c by agent k given that the target is located in c.
- $\mathcal{T}_k$: set of time intervals defining the time horizon for agent k $\{0,1,...,T_k-1\}$.
- T_k: number of time intervals or stages defining agent k network ($T_k = s_k T \leq T$).
- $x_{o\ i0(k)\ k}$: initial position of agent k. Agents are assumed to be positioned in different cell locations.

Decision variables:

- $z_{cl_1l_2\ldots l_n}$: binary decision variable. $z_{cl_1l_2\ldots l_n}$ = 1 depicts a path solution involving l_k visits by agent k ($k \in \eta$) on cell c (otherwise 0).
- v_{clk}: binary decision variable. v_{clk} =1 corresponds to a cumulative number of visits l on cell c by agent k at the end of time time horizon T_k (otherwise 0).
- x_{ijk}: binary decision variable. x_{ijk} = 1 indicates network state transition from i to j by agent k. Solution including arcs $(i,j) \in \mathcal{A}_k$ for which x_{ijk} = 1 define agent k's path.

Accordingly, a naïve mixed-integer linear program derived from Eq. (2) may be formulated as a preliminary model:

$$\underset{\{z_{cl_1l_2\ldots l_n}\}}{Min} \sum_{c\in N}\sum_{l_1=0}^{V}\sum_{l_2=0}^{V}\cdots\sum_{l_n=0}^{V}\left(p_{c0}\prod_{k\in\eta}(1-p_{cck})^{l_k}\right)z_{cl_1l_2\ldots l_n} \quad (3)$$

Subject to the linear constraint set:

$$z_{cl_1l_2\ldots l_n} \geq \sum_{k\in\eta} v_{cl_kk} - (n-1) \quad c\in N, l_k \in \{0..V\}, k\in\eta \quad (4)$$

$$\sum_{(i,j(c))\in\mathcal{A}_k} x_{ijk} - \sum_{l=0}^{V} l v_{clk} = 0 \qquad c\in N, k\in\eta \quad (5)$$

$$\sum_{l=0}^{V} v_{clk} = 1 \quad c\in N, k\in\eta \quad (6)$$

Initial agent position

$$x_{o\,i_0(k)\,k} = 1 \qquad k\in\eta, i_0(k)\in\mathcal{V}_k \quad (7)$$

Initial/final path condition

$$\sum_{i\in\mathcal{V}_k} x_{oik} = 1 \qquad k\in\eta \quad (8)$$

$$\sum_{i\in\mathcal{V}_k} x_{idk} = 1 \qquad k\in\eta \quad (9)$$

Flow conservation:

$$\sum_{i\in\mathcal{V}_k\cup\{o\}} x_{ijk} - \sum_{i\in\mathcal{V}_k\cup\{d\}} x_{jik} = 0 \quad k\in\eta, j\in\mathcal{V}_k, (i,j)\in\mathcal{A}_k \quad (10)$$

Maximum path length

$$\sum_{(i,j)\in\mathcal{A}_k} x_{ijk} = T_k \qquad k \in \eta \tag{11}$$

Continuous decision variables:

$$z_{cl_1l_2\ldots l_n} \geq 0 \quad c \in N, l_k \in \{0..V\}, k \in \eta \tag{12}$$

Binary decision variables

$$v_{clk} \in \{0,1\} \quad c \in N, l \in \{0..V\},\ k \in \eta \tag{13}$$

$$x_{ijk} \in \{0,1\} \quad k \in \eta, (i,j) \in \mathcal{A}_k \tag{14}$$

The objective function shown in (3) refers to the probability of non-detection (1-*POS*) over cell *c* assuming at most *V* visits at the site by a given agent. The bound *V* is selected arbitrarily large such that (3) may safely capture the optimal solution. Constraints are governed through equations (4)-(15). Constraints set (4) define a specific configuration (sequence) of team member's visits on cell *c* for feasible path solutions. It is worth noticing that minimization of (3) and inequality (4) guarantee the integrality of the numerous non-negative variables *z*. For a given agent *k*'s path solution, the coupling constraints (5) map number of visits and incoming arcs to a site *c*. The arc (*i*,*j*(*c*)) relates to agent *k* state transition terminating in position *c*. Constraints (6) simply represent the number of visits paid by agent *k* on site *c*. Path solution origin and end destination points are uniquely specified through constraints (7)-(9). Flow conservation captured by constraints (10) for visited state nodes balance the number of arcs coming inward and those going outward. Constraints (11) ensure a T_k-move path solution for agent *k*, but is ultimately redundant, serving little purpose as path length constraints are implicitly accounted for in the graph. Non-negative continuous *z* variables reflect a specific pattern of agent visits on cell *c*. Finally, v_{clk} and x_{ijk} represent binary decision variables referring to the number of visits *l* on cell *c* by agent *k,* and related state transition along arcs (*i*,*j*) at each move respectively. v_{clk} = 1 indicates agent *k*'s path solution involving *l* visits on cell *c*. The assignment $x_{ijk} = 1$ involves an agent *k* legally transitioning from state *i* to *j*.

C. *Path Reconstruction Procedure*

Once a solution has been computed, specific agent networks are reused to reconstruct individual agent paths. A particular agent path is then reconstructed using its own agent network and its instantiated flow decision variables to sequentially move from one stage to the next. T_k-move path generation for an agent *k* consists in successively exploring its related graph moving along instantiated arcs from its initial to its terminal state inserting cell visits, one at a time. Computational complexity of the team path reconstruction algorithm is linear with *n* ($O(nT)$).

D. *Receding Time Horizon*

Dynamic path solution can be periodically constructed over receding horizons as displayed in Fig. 3, integrating real information feedback to revise cell target occupancy beliefs, and continually improve solution quality. At each cycle, a new sub-problem is defined incorporating the observation outcomes from the previous period, and the resulting solution expands the current partial path by one more step. The process is then repeated to generate the overall path solution over the entire time horizon. The approach takes advantage of fast sub-problem optimizations to build a near optimal solution.

The approach can be further described as follows. Time horizon *T* is divided in $T/\delta T$ periods with time horizons ΔT and related subproblems are sequentially solved over corresponding cycles. δT refers to the sub-period over which a path plan is further extended (ideally δT = 1) after each planning cycle. Consequently, on subproblem-solving completion, the overall current partial path solution for an agent is expanded by δT (subperiod δT) moves while correspondingly updating beliefs from latest sensor readings impacting the objective function (coefficients) along with agent positions for the upcoming period. Restricted path move insertions on a given cycle permit successive planning period overlaps and mitigate undesirable effects related to myopic path construction. A new subproblem is then periodically solved. Then, the whole process goes on until the deadline *T* is reached.

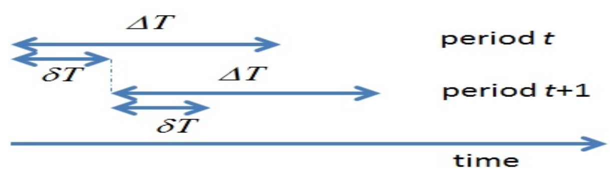

Figure 3. Time horizon *T* can be decomposed in $T/\delta T$ periods with receding sub-horizons ΔT. An agent path is constructively built, inserting δT new moves to the current partial solution, one cycle at a time.

IV. REVISITED MIP FORMULATIONS

A. *Analysis*

The value of the above proposed mixed-integer program lies in the fact that coefficients can be easily pre-computed in advance. Well-suited to handle small problem instances, the heterogeneous formulation (3)-(14) nonetheless remains complex, computationally hard and inconvenient as the number of variables *z* grows exponentially with the number of agents ($N(V+1)^n$), or when run-time is naturally constrained by near real-time application domain contingencies. Fortunately, in these cases, problem complexity may be further mitigated by exploiting key typical observations characterizing high-quality path solutions. First, it can typically be noticed over optimal path solutions that the number of visits required on a given cell generally remains restricted in practice, as the gain from repeated visits significantly drops in comparison to the potential benefit and appeal offered by competing neighbour cells. Imposing a limited number of visits per cell seems therefore quite satisfactory, suggesting the definition of an upper bound on a suitable/acceptable number of visits *V* to be conducted. Second, agent proximity from promising cells tends to make more likely near future additional visits by the same agent, as energy or cost requirement is expected to be low. Combining these two properties makes cell assignment to a limited number of agents fairly reasonable and justified in practice, while preserving solution near-optimality. In that regard, assuming two agents cell assignment subject to at most two visits per agent respectively seems conservative and quite satisfactory as posterior cell belief may abruptly decline. As a result, a simplified, tractable and more convenient model involving a polynomial number of variables in terms of problem attributes is proposed. Easily extendable to accommodate additional number of agents to service a given cell, the revisited decision model is also more appealing computationally offering tangible efficiency gain.

1) Up to 2 visiting agents decision model:

The additional parameters and variables required to define the decision model are given as follows:

Parameters:

B_{c0k}: probability of target non-detection contribution from agent k for an unvisited cell c by the entire search team.

$$B_{c0k} = p_{c0}/n \quad c \in N, k \in \eta$$

B_{clk}: probability of target non-detection on cell c resulting from l visits conducted by a single agent, namely agent k.

$$B_{clk} = p_{c0}\left(1 - p_{cck}\right)^{l} \quad c \in N, k \in \eta, l > 0$$

$C_{clkl'k'}$: probability of target non-detection on cell c resulting from l and l' visits conducted by exactly two agents, namely agent k and k'.

$$C_{clkl'k'} = p_{c0}\left(1 - p_{cck}\right)^{l}\left(1 - p_{cck'}\right)^{l'} \quad c \in N, k, k' \in \eta, \; l, l' > 0$$

Decision variables:

y_{clk}: non-negative continuous decision variable accounting for none site stopover, or cumulative visits resulting from a single searching agent only. $y_{clk}=1$ refers to l visits conducted by agent k on cell c.

$z_{clkl'k'}$: non-negative continuous decision variable capturing the contribution from two dissimilar agents. $z_{clkl'k'}=1$ indicates l and l' visits ($l,l' > 0$) by agents k and k' respectively on cell c.

A mixed-integer linear program is then formulated as follows:

$$\underset{\{y_{clk},\, z_{clkl'k'}\}}{Min} \sum_{c \in N} \sum_{k \in \eta} \sum_{l=0}^{V} B_{clk} y_{clk} + \sum_{c \in N} \sum_{k<k' \in \eta} \sum_{l>0}^{V} \sum_{l'>0}^{V} C_{clkl'k'} z_{clkl'k'} \quad (15)$$

Subject to the side constraints (5)-(11), (13)-(14) and:

$$y_{clk} \geq v_{clk} + \sum_{k' \neq k \in \eta} v_{c0k'} - (n-1) \quad \forall c, k \in \eta, l \geq 0 \quad (16)$$

$$z_{clkl'k'} \geq v_{clk} + v_{cl'k'} - 1 \quad \forall c, k < k' \in \eta, l, l' > 0 \quad (17)$$

$$y_{c0k}, y_{clk} \geq 0, \; z_{clkl'k'} \geq 0 \quad \forall c, k < k' \in \eta, l, l' > 0 \quad (18)$$

Eq. (15) includes non-negative zero/single agent and two-agent optional visiting contributions on a given cell in that order. The first term exclusively accounts for none and single agent visit per cell, whereas the second term alternatively captures the need for exactly two dissimilar agent visits on a given cell. Mutual exclusion of related terms is guaranteed by inequality (16)-(18) and non-negative coefficients $\{B_{clk}\}$ and $\{C_{clkl'k'}\}$ of the objective function, resulting in a single nonzero contributions for a given cell c. It should be noticed that cell visiting agents exceeding 2 is undesirable as it would unnecessarily and detrimentally increase the objective function, degrading solution quality. Cell-indexed coefficients simply indicate the probability of non-detection. This means that a cell may ultimately be visited up to $2V$ times, resulting from any combinations of two distinct agents. It can be readily verified that the null path solution which reflects unvisited cells by the entire team, expectedly results to a $\Sigma_c p_{c0} = 1$ outcome, trivially representing a target non-detection with certainty. The proposed decision model encompasses approximately $9N \Sigma_k T_k$ 'x', $N(V+1)n$ for both 'y' and 'v', as well as $Nn(n-1)V^2/2$ 'z' decision variables respectively.

2) Up to 3 visiting agents decision model:

The "up to 2 visiting agents" decision model is further extended to account for visits by up to three different agents. It consists in slightly revisiting the objective and related constraints at the expense of the introduction of additional non-negative continuous u variables.

The additional parameters and variables required to define the decision model are given as follows:

Parameters:

$D^0_{clkl'k'l''k''}$: probability of target non-detection on cell c resulting from l, l' and l'' visits conducted by three agents, namely agent k, k' and k'' respectively.

$$D^0_{clkl'k'l''k''} = p_{c0}\left(1 - p_{cck}\right)^{l}\left(1 - p_{cck'}\right)^{l'}\left(1 - p_{cck''}\right)^{l''}$$

$D_{clkl'k'l''k''}$: revisited probability of target non-detection contribution (non-positive) from the agents trio k, k' and k'' conducting l, l' and l'' ($l,l', l'' > 0$) visits respectively, to properly counterbalance unwanted pure 2-agent contributions ($\binom{3}{2}$ combinations) indirectly induced by Eq. (17).

$$D_{clkl'k'l''k''} = D^0_{clkl'k'l''k''} - C_{clkl'k'} - C_{clkl''k''} - C_{cl'k'l''k''}$$

Decision variables:

$u_{clkl'k'l''k''}$: non-negative continuous decision variable reflecting the contribution from three dissimilar agents. $u_{clkl'k'l''k''}=1$ refers to l,l' and l'' visits ($l,l',l'' > 0$) by agents k, k' and k'' respectively on cell c.

The resulting decision model is given as follows:

$$\underset{\substack{\{y_{clk}, z_{clkl'k'},\\ u_{clkl'k'l''k''}\}}}{Min} \sum_{c\in N}\sum_{k\in\eta}\sum_{l=0}^{V} B_{clk} y_{clk} + \sum_{c\in N}\sum_{k<k'\in\eta}\sum_{l>0}^{V}\sum_{l'>0}^{V} C_{clkl'k'} z_{clkl'k'}$$

$$+\sum_{c\in N}\sum_{k<k'<k''\in\eta}\sum_{l,l',l''>0}^{V} D_{clkl'k'l''k''}\, u_{clkl'k'l''k''} \tag{19}$$

Subject to the linear constraint set (5)-(11), (13)-(14), (16)-(18) and:

$$u_{clkl'k'l''k''} \le v_{clk}\,, \quad u_{clkl'k'l''k''} \le v_{cl'k'}\,, \quad u_{clkl'k'l''k''} \le v_{cl''k''}$$
$$c\in N, k<k'<k''\in\eta, l,l',l''>0 \tag{20}$$

$$u_{clkl'k'l''k''} \ge 0 \quad c\in N, k<k'<k''\in\eta, l,l',l''>0 \tag{21}$$

The objective (19) includes none, single agent, two-agent and three-agent optional visiting contributions on a given cell in that order. This condition is ensured by the combination of inequality (16)-(18), (20)-(21) and the non-negative coefficients $\{B_{clk}\}$ and $\{C_{clkl'k'}\}$ characterizing the objective function. Note that in order to reuse and naturally extend the objective (15), three-agent contributions require non-positive coefficients $\{D_{clkl'k'l''k''}\}$ compensating for unwanted additional 2-agent contributions inevitably induced through Eq. (17). This shows that a cell may eventually be subject up to $3V$ visits, resulting from any combination of three different agents. The model comprises approximately $9N\ \Sigma_k T_k$ 'x' s, $N(V+1)n$ 'y' and 'v', $Nn(n-1)V^2/2$ 'z', and finally $Nn(n-1)(n-2)V^3/6$ 'u' decision variables respectively.

B. Model Simplification and Complexity Reduction

One can further reduce computational complexity maintaining the number of key decision variables linear in the number of agents and therefore keeping the problem manageable under more stringent temporal problem-solving constraints. This is achieved by eliminating z variables from the "up to 2 visiting agents" model (15), replacing the linear program formulation by a quadratic integer program (QP) ($\frac{1}{2}\ x^TQx + L^Tx$). The quadratic objective function to minimize is convex and turns out to be solvable exactly, as the problem formulation presents a semi-definite positive matrix Q, guaranteeing that a local path solution is also a global optimal solution. The complex original mixed integer linear programming model is therefore equivalent to the following quadratic integer program:

$$\underset{\{y_{clk}, v_{clk}\}}{Min} \sum_{c\in N}\sum_{k\in\eta}\sum_{l=0}^{V} B_{clk} y_{clk} +$$

$$\sum_{c\in N}\sum_{k<k'\in\eta}\sum_{l>0}^{V}\sum_{l'>0}^{V} C_{clkl'k'} \frac{\left(v_{clk}+v_{cl'k'}\right)^2 - \left(v_{clk}+v_{cl'k'}\right)}{2} \tag{22}$$

Subject to the side constraints (5)-(11), (13)-(14) and (16)-(18).

The non-negative coefficients $C_{clkl'k'}$ in the objective to be minimized naturally exclude path solution visits on a given cell c by more than a single agent pair, otherwise additional agent pair contributions would undesirably add-up, further degrading solution quality (increasing the function). The problem then consists in determining which agent couple is best fit to cover a promising cell. The revisited model formulation (22) is simple, elegant and compact with an appealing optimal solution property. It significantly reduces the number of decision variables required, making it quite adequate for near real-time practical problem-solving. The model involves approximately $9N\ \Sigma_k T_k$ 'x' variables and $N(V+1)n$ 'y' and 'v' variables respectively.

C. Special Case

The reduction of the heterogeneous model to the homogeneous agent formalism [26] for similar agents can be easily shown. Accordingly, as conditional probability of detection is agent-independent ($p_{cck} = p_{cc}$), and the equation $v^2_{clk} = v_{clk}$ is implicitly ensured by the integrality property of the variables v_{clk}, and that y_{clk} is implicitly binary as well, the quadratic objective function POF (22) may be further developed as follows:

$$POF = \sum_{c\in N}\sum_{k\in\eta}\sum_{l=0}^{V} B_{clk} y_{clk} +$$
$$\sum_{c\in N}\sum_{k<k'\in\eta}\sum_{l>0}^{V}\sum_{l'>0}^{V} p_{c0}\left(1-p_{cc}\right)^{l+l'} v_{clk}\, v_{cl'k'}$$

$$= \sum_{c\in N}\sum_{l=0}^{V} B_{cl} \underbrace{\sum_{k\in\eta} y_{clk}}_{w_{cl}} + \sum_{c\in N}\sum_{l>0}^{V}\sum_{l'>0}^{V} p_{c0}\left(1-p_{cc}\right)^{l+l'} \underbrace{\sum_{k<k'\in\eta} v_{clk}\, v_{cl'k'}}_{w_{cl+l'}}$$

As we deal with a minimization problem involving non-negative objective function coefficients there exists a single positive term combining indexes k and k', resulting in a binary result ultimately modeled using the new binary integer variable $w_{c\,l+l'}$.

$$POF = \sum_{c\in N}\sum_{l=0}^{V}\sum_{l'=0}^{V} p_{c0}\left(1-p_{cc}\right)^{l+l'} w_{c\,l+l'}$$

Given that $\sum_{l=0}^{V} v_{clk} = 1$, a single combination (l,l') similarly culminates in a nonzero (positive) $w_{c\,l+l'}$ contribution as well, leading to the final expression:

$$POF = \sum_{c\in N}\sum_{l=0}^{2V} p_{c0}\left(1-p_{cc}\right)^{l} w_{cl}$$

in which cells are subject to at most V visits per agent. The derived objective function expectedly conforms to the standard homogeneous problem model formulation proposed in [26].

D. Hybrid Model

Notice that a hybrid formulation combining the quadratic "up to 2 visiting agents" and "up to 3 visiting agents" decision models is also possible when either a few outstanding and more attractive cells significantly dominate over the grid, or target observability proves to be exceptionally or abnormally difficult in some areas. It consists in revisiting (19) using formulation (22), while restricting 3-agent optional visiting contributions on a given cell to a small subset of cells $N' \subseteq N$ likely to be visited very frequently, if any. These include highly dominating cells (large p_{c0}) as well as locations demonstrating very poor target observability/detectability (quasi-invisibility) to the entire search team due to specific conditions detrimental to sensor's capability (e.g. terrain, landscape, luminosity). The hybrid model is given as follows:

$$\underset{\substack{\{y_{clk}, v_{clk},\\ u_{clk\,l'k'l''k''}\}}}{Min} \sum_{c \in N} \sum_{k \in \eta} \sum_{l=0}^{V} B_{clk} y_{clk} + \sum_{c \in N} \sum_{k<k' \in \eta} \sum_{l>0}^{V} \sum_{l'>0}^{V} C_{clkl'k'} \frac{\left(v_{clk}+v_{cl'k'}\right)^2 - \left(v_{clk}+v_{cl'k'}\right)}{2} + \sum_{c \in N' \subseteq N} \sum_{k<k'<k'' \in \eta} \sum_{l,l',l''>0}^{V} D_{clkl'k'l''k''} u_{clk\,l'k'l''k''} \quad (23)$$

Subject to the linear constraint set (5)-(11), (13)-(14), (16)-(18), and (20)-(21).

$N' = \{c \in N$: $(p_{c0}(1- p^{team}_{cc})^{2V} \geq 1/|N|)$ *OR*
$(\Sigma_{c'} p^2_{c'0} > 1/k_{thr}$ *AND* $p_{c0} \geq 1/k_{thr})$ $\}$

where p^{team}_{cc} refers to the second largest conditional probability of 'correct' target detection p_{cck} from the team, while k_{thr} is a constant ((1 – $1/|N|$ Σ_c p^{team}_{cc} $)^{2V}$ $|N|$) defining a threshold value conservatively bounding the number of cells that could require more than $2V$ visits (typically, 10< k_{thr}<15). The first condition defining *N'* is set to include high magnitude prior belief sites as well as interesting locations presenting very poor target observability or near invisibility to the team sensors. It captures cells still likely to be competitive after $2V$ visits (e.g. V=2) in a 2-agent cell visiting case, in comparison to an alternate nearby cell having on average a $1/|N|$ belief. The use of the second largest rather than the maximum conditional probability of 'correct' target detection from team sensor agents is very conservative. It deliberately includes as much cells prone to multiple visits as possible. As for the latter condition, it characterizes cells exhibiting significant belief dominance that could still have eluded satisfaction of the first condition. It consists in bounding the number of dominating cells beyond which domination can no longer be assumed. k_{thr} represents such an upper and is selected conservatively. Outstanding belief dominance appears when the variance Σ_c p^2_{c0} in prior probability distribution over grid cells exceeds a given threshold $1/k_{thr}$, meaning that multiple visits on few specific promising cells would inevitably take place. k_{thr} bounds the approximate number of promising or dominating cells, empirically given by $1/\Sigma_c p^2_{c0}$. In this case, *N*' includes alternate cells (at most k_{thr}) from the grid having initial prior belief values larger than $1/k_{thr}$, significantly reducing the number of additional variables required in the model. However, when initial belief variance magnitude is predictably negligible ($\Sigma_c p^2_{c0} \leq 1/k_{thr}$), the utilization of the "up to 2 visiting agents" model arguably appears quite legitimate. Likewise, a similar argument could be used to confirm path solution optimality obtained from the "up to 2 visiting agents" cell visit problem model. Correspondingly, *N'* would then refer to the set of cells composing the optimal "up to 2 visiting agents" model path solution visited more than once. In general, the utilization of the "up to 3 visiting agents" cell visit model (23) is very conservative and quite acceptable in practice to properly handle high variance prior probability distribution problems.

E. Problem-Solving

It is worth noticing that the revisited formulation (23) offers many advantages over most non-MIP modeling procedures, as the proposed quadratic program may efficiently generate a tight bound on the quality of the optimal path solution by relaxing variable integrality constraints (Lagrangian relaxation). This provides some comparative baseline measure to achieve performance gap analysis over highly competitive solutions, and suitably trade-off solution quality against run-time.

V. SIMULATIONS

A brief simulation experiment has been carried out to assess the value of the proposed MIP (QP) model (23). The value of the MIP (QP) model is measured in terms of optimality gap and computational run-time over a sample of scenarios. Performance comparison with an alternate myopic heuristic is also reported for illustration purposes. MIP (QP) is implemented and solved using the IBM ILOG CPLEX parallel Optimizer [27]. Path solutions from respective approaches are reported against terminal probability of detection optimality gap:

$$Opt\ gap = \frac{POS^* - POS_m}{POS^*} \quad (24)$$

*POS** reflects the computed optimal probability of success or an upper bound obtained from Lagrangian relaxation if not available, and POS_m the recorded performance for method *m*. Close optimality gap correlates with good performance.

A. Limited Look-Ahead Path Planning

A one-step look-ahead path solution construction method is proposed. The myopic heuristic simply consists in visiting the closest neighboring (including the current host) cell offering maximum gain. At each time step, the agent move presenting the highest reward is preferentially selected. The objective function is then updated accordingly. The process goes on repeatedly until all agents' decisions have been instantiated. The agent -cell visit allocation process is then reiterated for each episode over time horizon *T*. Computational run-time $\in$ O(*nT*).

B. Simulation Configurations

The following scenario configurations and computer set-up have been considered:

- Scenarios:

Prior cell occupancy belief distribution: exponential, uniform;
N: 10x10
$T=\sqrt{|N|}$ to ensure full grid visibility to the planner
Up to V=2 visits per cell by the same agent
k_{thr} = 15
Heterogeneous sensor agents:
Team size n = 5
Actions: 9 moves possible, toward host and neighbouring cells
Sensor parameters: p_{cc} runs into [0.75,0.85], $c \in N$
$T_k=T$ for all agents (worst-case)

- Hardware Platform:
  Intel (R) Xeon (R) CPU X5670
  Shared-memory multi-processing: 8 processors
  2.93 GHz
  Random Access Memory: 16 Go, 64 bits binary representation (double precision)

Focusing on the use of 10x10 grids more specifically may be easily motivated. Given that initial target cell occupancy probability values sum up to one, interest for comparative algorithm investigation and performance analysis expectedly decline for increasingly large grid as average belief magnitude decreases. Essentially, the larger the size of the grid, the smaller is the belief of cell occupancy. This condition inevitably leads either to high visit payoffs for a small number of sparsely distributed site locations, or comparable cell visit payoffs, for which any path construction algorithm likely demonstrates near equivalent performance. In both situations, a substantial computational effort required to build long path solution fragments expected to provide a low payoff may be too prohibitive and questionable. Therefore, we contend that large grid instances should be preprocessed hierarchically to generate aggregated and more convenient 10x10 cognitive map configurations. The approach would satisfactorily ensure adequate area coverage while permitting a meaningful comparative performance analysis assessment. This explains the selection to the proposed 10x10 grid size in the proposed investigation.

C. Computational Results

Simulation results are compiled in Table I for a few N=10x10 grid, 5-agent team, V=2 and time horizon T=10 scenarios. Each table entry is mapped to a specific problem instance labeled in the first column, against probability of success (*POS*=1-*POF*) and optimality gap performances reported for the myopic method and MIP (QP) respectively. Run-time for the myopic heuristic is negligible and has been overlooked. MIP (QP) run-time performance measured in seconds is compiled as well. Overall, the experiment shows MIP (QP) to be really fast with less than a 4.45% optimality gap. For reference purposes, the propposed MIP (QP) approach is shown to outperform the myopic procedure by more than 20% while indicating a conservative optimality gap and a required run-time less than two minutes. Reported run-time performance is comparable to the homogeneous agent case already presented in [26] for similar path solution space size given by $|A|^{nT} = 9^{nT}$. Note that computational performance gains might be further expected for alternate scenarios with additional cell visibility and/or accessibility constraints.

TABLE I. RELATIVE PERFORMANCE OF THE MIP (QP) VS HEURISTICS FOR A SAMPLE DATA SET (10X10 GRID, 5-AGENT TEAM, V=2, T=10)

Instance	**Myopic Heuristic**		**MIP (QP)**		**Run-time (*s*)**
	POS	***Opt gap %***	***POS (1-POF)***	***Opt gap %***	
1	0.432	18.21	0.508	3.98	67.38
2	0.365	15.31	0.417	3.35	59.63
3	0.398	11.15	0.435	2.98	56.26
4	0.424	14.46	0.493	0.55	41.67
5	0.308	23.37	0.397	1.24	36.98
6	0.339	23.78	0.434	2.49	32.14
7	0.346	15.91	0.394	4.43	36.63
8	0.367	13.50	0.407	4.25	27.84

VI. CONCLUSION

By virtue of additional computational complexity intricacies induced by agent capability diversity and broken symmetry, search path planning modeling and problem-solving efficiency suddenly breaks apart when naturally evolving from a homogeneous to a heterogeneous sensing agent team setting. An innovative mixed-integer quadratic programming (QP) approach has been proposed to solve the discrete static open-loop search path planning (search and rescue) problem by a team of heterogeneous agents. The concurrent exploitation of suitable near optimal solution characteristics, network representation, and a problem-solving property of the revisited problem model highlight the novelty of the approach. An upper bound on the optimal path solution may also be estimated using Lagrangian relaxation providing additional guidance on convergence rate and acceptable run-time. Near-optimal solution can be efficiently computed for practical size problems.

Future work will consist to extend the heterogeneous agent team model to a moving target setting. Another direction aims at refining the target detection task using a belief network to represent increasingly complex observation models capturing multidimensional signal attributes and relationships from multiple dissimilar sensors, applicable to specific problem domains.

ACKNOWLEDGEMENTS

The reported work was co-funded by Defence Research and Development Canada Valcartier and T-OptLogic Ltd, Quebec City, Canada. The authors would like to thank anonymous reviewers who provided critical reviews to improve content presentation and readability of the paper.

REFERENCES

[1] K.E. Trummel and J.R Weisinger, “The complexity of the optimal searcher path problem”, Operations Research, 34 (2), 1986, pp.324-327.

[2] S. Benkoski, M. Monticino, and J. Weisinger, "A survey of the search theory literature", Naval Research Logistics 38, 1991, 469–494.
[3] L.D. Stone, Theory of Optimal Search, 2nd edition, Topics in Operations Research Series, INFORMS, 2007.
[4] R. Hohzaki, and K. Iida, "Optimal search plan for a moving target when a search path is given", Math. Japonica, 41(1), 1995, pp. 175-184.
[5] H. Lau, Optimal Search in Structured Environments, PhD Thesis, University of Technology, Sydney 2007.
[6] H. Lau, and G. Dissanayake, "Probabilistic search for a moving target in an indoor environment", In Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2006, pp. 3393-3398.
[7] H. Lau, and G. Dissanayake, "Optimal search for multiple targets in a built environment", In Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Edmonton, Alberta, Canada, 2005, pp. 228-233.
[8] J.-C. Latombe, Robot Motion Planning, ser. International Series in Engineering and Computer Science; Robotics: Vision, Manipulation and Sensors, vol. 124, Boston, Kluwer Academic Publishers, 1991.
[9] C. Choo, J. Smith, and N. Nasrabadi, "An efficient terrain acquisition algorithm for a mobile robot", in Proc. IEEE Int. Conf. Robotics and Automation, Sacramento, CA, 1991, pp. 306–311.
[10] A. Sankaranarayanan, and I. Masuda, "Sensor based terrain acquisition: A new, hierarchical algorithm and a basic theory", in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Raleigh, 1992, pp. 1515–1523.
[11] J. Svennebring, and S. Koenig, "Building terrain-covering ant robots: A feasibility study", Auton. Robots, 16 (3), 2004, pp. 313–332.
[12] S. Wong, and B. MacDonald, "A topological coverage algorithm for mobile robots", in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Las Vegas, 2003, pp. 1685–1690.
[13] S. Yang, and C. Luo, "A neural network approach to complete coverage path planning", IEEE Trans. Syst.,Man, Cybern. B, Cybern., 34(1), 2004, pp. 718–724.
[14] I. Rekleitis, et al., "Efficient boustrophedon multi-robot coverage: an algorithmic approach", Ann Math Artif. Intell. 52, 2008, pp. 109–142.
[15] W. Wagner, et al., "Distributed covering by ant-robots using evaporating traces", IEEE Trans. Robot. Autom., 15(5), 1999, pp. 918-933.
[16] T. Chung, G. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics: A survey", Autonomous Robots (AURO), 31(4), 2011, pp. 299-316.
[17] J.O. Royset and H. Sato, "Route Optimization for Multiple Searchers", Naval Research Logistics, 57 (8), 2010, pp. 701-717.
[18] T. H. Chung, J. Burdick, "Analysis of search decision making using probabilistic search strategies", IEEE Transactions on Robotics, 28 (10), 2012, pp. 132-144.
[19] Y. Jin, Y. Liao, A. Minai, and M. Polycarpou, "Balancing search and target response in cooperative unmanned aerial vehicle (UAV) Teams", IEEE Trans on Sys Man and Cybern. Part B, 36(3), 2006, pp. 571-587.
[20] G.A. Hollinger, Search in the Physical World. CMU-RI-TR-10-20, Robotics Institute, PhD thesis, Carnegie Mellon University, 2010.
[21] I. Abi-Zeid, O. Nilo, and L. Lamontagne, "Constraint optimization approach for the allocation of multiple search units in search and rescue operations", INFOR: Information Systems and Operational Research, 49, Number 1 / 2 2011, pp. 15-30.
[22] M. Morin et al. "The Optimal Searcher Path problem with a Visibility Criterion in Discrete Time and Space", Int. conference on Information Fusion, Seatle, USA, 2009.
[23] M. Morin, A.P. Papillon, F. Laviolette, I. Abi-Zeid, and C.G. Quimper, "Constraint Programming for Path Planning with Uncertainty: Solving the Optimal Search Path problem," in Proceedings of the 18th Conference on Principles and Practice of Constraint Programming, Québec, Qc, Canada, 2012, pp. 988-1003.
[24] G.A. Hollinger, and S. Singh, "GSST: Anytime Guaranteed Search with Spanning Trees", Autonomous Robots, 29(1), 2010, pp. 99-118.
[25] A. R.Washburn, "Branch and Bound Methods for a Search Problem", Naval Research Logistics, 45, 1998, 243-257.
[26] N. Lo, J. Berger and M. Noel, "Toward Optimizing Static Target Search Path Planning", Proceedings of the 5th IEEE Symposium on Computational Intelligence for Security and Defence Applications, Ottawa, Canada, July 2012.
[27] IBM ILOG CPLEX Optimizer, Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/cplex-optimizer/cplex-performance/ and IBM ILOG CPLEX V12.1, 2009. Available: ftp://public.dhe.ibm.com/software/websphere/ilog/docs/optimization/cplex/ps_usrmancplex.pdf