An Information-theoretic –based Integer Linear Programming Approach for the Discrete Search Path Planning Problem

Jean Berger[†], Nassirou Lo[‡], Abdeslem Boukhtouta[†], Martin Noel*

[†] DRDC Valcartier, Quebec City, Canada, jean.berger@drdc-rddc.gc.ca

[‡] T-OptLogic Ltd. Quebec City, Canada, nassirou.lo@t-optlogic.com

* UQAM, TELUQ, Quebec City, Canada, noel.martin@teluq.uqam.ca

Abstract – *Discrete search path planning is known to be a NP-Hard problem, and problem-solving methods proposed so far rely on heuristics with no way to reasonably estimate solution quality for practical size problems. Departing from traditional nonlinear model formulations, a novel information-theoretic –based approach using integer linear programming (ILP) is proposed to optimally solve the discrete open-loop centralized search path planning problem with anticipated feedback, involving a team of homogeneous agents. The approach takes advantage of objective function separability and conditional probability independence of observations to efficiently minimize expected system entropy. A network representation is exploited to simplify modeling, reduce constraint specification and speed-up problem-solving. The novelty of the approach consists in capturing false-alarm observations explicitly while proposing an original and efficient way to linearize the underlying non-linear expected entropy function required to properly represent target location uncertainty, making for the first time practical problems tractable. The proposed ILP formulation rapidly yields near-optimal solutions for realistic problems while providing for the first time, a robust lower bound through Lagrangian relaxation. Long planning problem horizon may be dynamically adapted by periodically solving new problem instances incorporating actual observation outcomes from previous episodes over receding horizons. Computational results clearly show the value of the approach in comparison to a myopic heuristic over various problem instances.*

DRDC-RDDC-2015-P105

Keywords: combinatorial optimization, search path planning, information theory, network flow, linear programming, open-loop with anticipated feedback

1 Introduction

Persistent surveillance and reconnaissance tasks in sensor agent networks are key to constructing recognized domain pictures over a variety of civilian and military domains. A commonly encountered subtask often involves target search. Applications include homeland security, emergency management, search and rescue/respond, urban operations, terrain mapping, monitoring, surveillance and reconnaissance and intelligence/information gathering. However, despite recent technological information system advances, efficient information gathering and search still remain computationally hard. Search can indeed be reduced to a multiple knapsack problem, known to be NP-Hard in the strong sense.

Search theory [2], [22] pioneered original work contributions on the search path planning problem. Proposed approaches initially focused on the effort allocation decision problem laying emphasis on time spent per visit over all key areas rather than overall search path cost per se when operating under itinerary constraints. Promoting a mathematical framework to primarily develop analytical solutions for basic static problems, attention has progressively shifted toward algorithmic performance issues and contributions for dynamic problem settings [8], [14]. In other respect, the robotics field has also seen the emergence of a significant body of work contributions on search path planning. Key domains such as robot motion planning [13], terrain acquisition [4], [20], [21] and coverage path planning [26], [27] rapidly emerged as leading areas. Motion planning mainly embraced search path planning from a constrained shortest path perspective to handle coverage problem instances [19], [24]. Reported contributions focused on problem features involving environment uncertainty, obstructing obstacles and limited exploitable domain knowledge, searching a number of stationary targets sparsely distributed over an area of interest. Robotic work inspired from cooperative control theory and artificial intelligence alternatively covers an extensive body of research on distributed decision-making, cooperative control, and multiagent coordination problems in resource-bounded uncertain environment. In that case, research mainly focuses on cooperative search and distributed information-gathering for a team of agents [18], [12], [7], [23], [6]. The reader is advised to look at recent search path planning surveys and taxonomies published in search theory [14] and artificial intelligence/distributed robotic control [9] for more details.

Computational complexity using exact methods to solve sequential decision problem formulations tend to scale exponentially with search time horizon, grid and agent team sizes. Dynamic programming [14], [9], [15], [3] and tree – based search techniques [1], [10] represent such methods. They work fine for various settings and particular constraints or conditions, but eventually demonstrate a high level of complexity leading to combinatorial explosion and poor scalability, setting the stage for the promotion of new approximate and heuristic procedures. Accordingly, preliminary proposed methods simply attempts to manage problem complexity by relaxing hard constraints. More sophisticated techniques based on branch and bound [17], [25], [16] and path finding A* and variants were then further developed and refined. However, the determination of good heuristics to compute tight bounds for long-term solution quality estimation remains fundamentally difficult [14].

The proposed work focuses on a centralized class of discrete time and space stationary multi-target search path planning problem [28]. It deals with some of the problem complexity limitations overlooked or induced by simplifying assumptions mostly considered by search approaches advocated so far. Such assumptions typically ignore one or several combinations of the following conditions: partial environment state observability, anticipated feedback information exploitation, imperfect sensing agent capabilities (e.g. false-alarm detection), realistic agent team cardinality and bounded computational resources. In other respect, published frameworks reporting on problem-solving heuristics efficiency mostly fail to provably estimate real performance gap to the optimal for practical size problems. A new approach bringing robust performance gap assessment improvement, might significantly impact system/team performance when trading-off computed solution quality against run-time.

In this paper, we propose a new information-theoretic –based integer linear programming approach to solve the discrete open-loop centralized search path planning problem with anticipated feedback. The 'open-loop with anticipated feedback' terminology essentially relates to offline planning execution while accounting for projected (conditional) rather than real cell visit action outcome. Anticipated feedback information is aimed at enriching open-loop classical model formulations which deliberately overlook feedback information outcome. The intent is to improve solution quality exploiting conditional information outcome, while reducing expected computational complexity limitations usually induced from closed-loop problem formulations (e.g. dynamic programming, and partially observable Markov decision processes). In that setting, a team of centrally controlled homogeneous agent with imperfect sensing capabilities searches an area (grid) to minimize target cell occupancy uncertainty, given a prior cell occupancy probability distribution. Grid entropy separability over cells and anticipated conditional observation probability independence enable efficient objective function pre-computation leading to a new and convenient ILP formulation. A network representation is used to further mitigate modeling complexity, greatly simplifying constraint specification and implementation while providing additional run-time savings. The new decision model incorporates false alarm agent sensors, exploits anticipated action outcome feedback and relies on an abstract network representation, that can be coupled to parallel computing (e.g. using the CPLEX solver [11]) to rapidly solve practical size problems. A comparative performance analysis has been conducted to show the expected gain. Experimental results prove the proposed approach to significantly outperform a myopic search path planning heuristic by approximately 30% on average in terms of information gain, for a random sample of problem instances. The approach may further resort to Lagrangian relaxation to provably generate an exploitable optimal solution lower bound. Finally, long planning horizon problems may be dynamically handled by periodically solving new problem instances taking advantage of feedback information (from real observation outcomes), over short receding horizons. The intent is to promptly benefit from real action outcomes made available episodically to constructively refine path plans. This allows improving solution quality rather than averaging performance over long outcome sequences dictated by a distant horizon and ignoring past intermediate outcomes.

The remainder of the paper is structured as follows. Section 2 first defines the problem. It presents the main problem features characterizing open-loop centralized search path planning with anticipated feedback. The modeling solution concept to address the search path planning problem is then introduced in Section 3. It namely portrays a new information-theoretic –based integer linear programming network flow decision model to enabling near-optimal solution computation. Section 4 presents comparative performance results promoting the value of the advocated approach. Finally, the main findings and conclusive remarks are summarized in Section 5.

2 Problem definition

2.1 Description

The discrete open-loop with anticipated feedback version of the centralized search path planning problem (*SPP*) consists in a number of homogeneous sensing agents exploring a well-defined bounded environment over a specific period of time to successfully search and find stationary targets. Mainly inspired from reconnaissance mission domains, the objective of the team is aimed at maximizing information gain or equivalently minimizing target occupancy uncertainty or entropy within a given region. This situation typically occurs in military domains, such as land mine detection, enemy surface-to-air missile launcher or radar detection. Path planning is centralized and carried out by a base control station or an agent leader over a specific period (planning episode). The 'open-loop with anticipated feedback' characteristic of the problem refers to the offline nature of problem-solving, while capturing predicted feedback information (observation outcomes) explicitly in the decision model. Modeled as a regular grid, the search region describes a two-dimensional area defining a set N of cell elements, populated by non-cooperative stationary targets, assumed to individually occupy a single cell. The number of targets and their respective locations are initially unknown. Based on domain knowledge, a prior target location probability density distribution defines individual cell occupancy characterizing a grid cognitive map. The cognitive map is a knowledge base capturing local environment state representation, reflecting target occupancy belief distribution, agent positions and orientations, as well as sensor observations history and related sources. A typical cognitive map at a given point in time is illustrated in Figure 1.

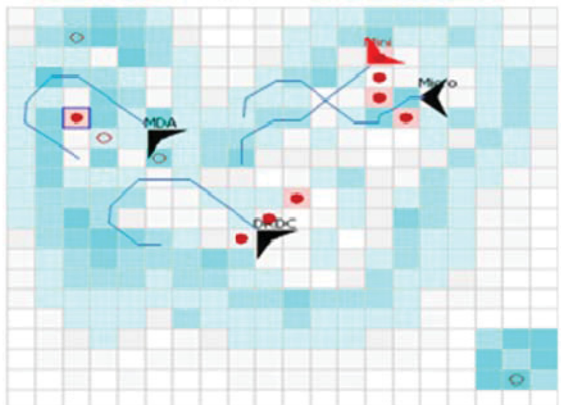


Figure 1. Uncertainty grid /cognitive map for a specific episode*.* The magnitude of a belief associated with target presence in a cell is represented through a proportional color tone. Agent path plans are depicted over a limited time horizon. Cell locations having dark (void) circle symbols correspond to discovered (to be discovered) targets.

The target occupancy probability is assumed to be independent between cells. Given an initial team configuration, centrally controlled agents synchronously explore the environment, acting as stand-off imperfect sensors gathering observations while episodically exchanging state and plan information with the base control station (or agent leader). Cell visit time defining episode duration is assumed to be constant. Vehicles are assumed to move at slightly different altitudes to avoid colliding with each other. Search solution consists in constructing agent path plans to minimize team uncertainty (entropy) over cell target occupancy.

2.2 Observation model

On completing a cell visit an agent may positively or negatively perceive the sought target. The observation model governing agent sensor's perception accounts for partial world state observability. During each time step t, an agent visits a cell searching for target occupancy. An agent observation or sensor reading z_t at time t may be positive (z_t=1) or negative (z_t=0) and is governed by an observation model, which accounts for uncertainty through conditional probability of detection and false alarm given cell target vacancy or occupancy state $X \in \{0,1\}$ respectively:

z_t: cell occupancy observation at the end of period *t.*

$p_c = p(z_t = 1 | X = 1)$ probability of correct observation

$p_f = p(z_t = 1 | X = 0)$ probability of false alarm (false positive)

The probability of false positive observation p_f is primarily domain-dependent and reflects sensor agent imperfect behaviour and sensitivity under environmental and observation conditions such as weather, light intensity and topography where a searched target may possibly be confused with an alternate object/subject. In the current setting, we assume p_f to be within 10% quite reasonable. The probability of a false-negative observation outcome refers to an incorrect reading and is naturally given by 1-p_c.

Agent sensor's range defining visibility or footprint (coverage of observable cells given the current sensor position) is limited to the cell being searched. The observation model is assumed to be known by the decision-maker.

2.3 Bayesian filtering

Based on real or anticipated agent sensor observation, local cell target occupancy belief (*p(X=1)*) (real or anticipated) can be modified using Bayesian filtering (see [12]) for data fusion:

$$p_t(X | z_t) = \frac{p(z_t | X) p_{t-1}(X)}{p(z_t)} \tag{1}$$

where

$$p(z_t) = \sum_{x \in \{0,1\}} p(z_t | X = x) p_{t-1}(X = x) \tag{2}$$

p_{t-1} and p_t ($t > 0$) refer to prior and posterior cell target occupancy probability (belief) respectively, defining the team-level cognitive map.

2.4 Path planning

A centralized decision-making process episodically makes path planning move decisions based on key sensing agent's state variables, namely, its position (cell location) and orientation $\{N,S,E,W,NE,SE,SW,NW\}$. Decision-making is subject to limited computational resources imposed by episode duration. Basic agent decisions are restricted to three possible moving directions, that is, *ahead, right* or *left* respectively, as shown in Figure 2. These limited moves account for the physical acceleration associated with agent's kinematics.

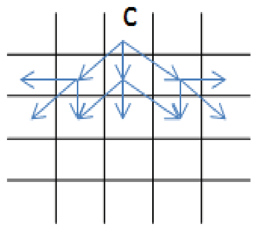


Figure 2. Possible agent move projection based on current position and orientation over a two–step time horizon.

Base-level agent action moves are planned to minimize overall entropy or uncertainty related to target occupancy over an area of interest defining the search region (grid). The entropy function E is borrowed from information theory [5]:

$$E = -\sum_{x \in \{0,1\}} p(x) \log_2 p(x) \tag{3}$$

where $p(x)$ refers to the current probability/belief of cell target occupancy. A cell zero entropy means absolute certainty about occupancy or vacancy whereas a maximum entropy (1) refers to complete uncertainty. Target occupancy uncertainty over the the grid simply consists in summing up entropy values over individual cells. The entropy function is pictured in Figure 3. Initial system entropy can then be initially computed as a function of prior probability of target cell occupancy p_0 given as a problem input.

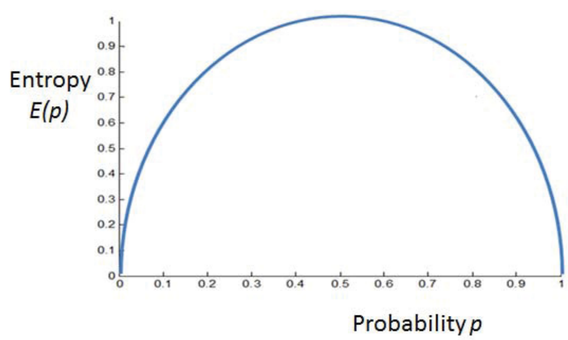


Figure 3. Entropy function reflecting uncertainty level associated with a given probability p characterizing a state variable. Entropy vanishes under certainty (p = 0 or 1) and culminates when $p = ½$.

2.5 Long-term planning

Long-term planning is dynamically managed by periodically solving new problem instances incorporating actual observation outcomes from previous episodes over receding horizons, while committing a computed agent path plan for the next single move execution one step at a time. The value of that strategy consists in opportunistically improving path plan quality conditioned by real episodic feedback information. A time horizon is selected such that computing a revisited plan over a single time step (period) remains feasible. Typical horizon ranging into {12, …, 20} can be easily handled under a minute cell visit timescale. In that setting, the centralized decision-making and distributed plan execution scheme assume fast and perfectly reliable communication between base control station and subordinate searching/sensing agents. Through a star topology agent/physical communication network, information-sharing is ensured periodically between agents and the central base control station. The control station first gathers agent state and action observations, computes a revised team path solution plan, and then disseminates revisited action plans (next sequence of moves) to subordinate agents to be executed over the next time horizon.

3 Integer Linear Programming Model Formulation

3.1 Expected entropy objective function

The proposed information-theoretic –based discrete ILP decision model captures expected system entropy for a given path solution. It measures the uncertainty of target occupancy over the grid, averaging projected entropy that may result from solution plan execution. As target occupancy over cells are independent, and the observation model strictly depends on current agent position, decision variables may be partitioned in subsets with separate contributions to the average entropy objective function. In other words, as cell entropy uniquely depends on local visits being conducted, the expected entropy objective function proves to be separable. That separability property makes feasible precomputing expected cell entropy values $\overline{E}_{cl}$ in advance, since these contributions essentially rely on homogeneous sensing characteristics and visit multiplicity l on cell c. The 'expected' nature of entropy relates to partial observability characterizing imperfect sensing agent capabilities, inducing variability on anticipated feedback information over possible scenarios (sequence of observations) when executing path plan solution. Expected cell entropy is computed by averaging projected entropy values simulating path plan execution over all possible histories (sequence of events) for a given path solution. As team visit ordering on a given cell has no impact on overall expected cell entropy, symmetry on observation outcomes steams may be exploited. Accordingly, scenarios are divided in equivalence classes based upon the number of success (s anticipated positive observations) and failure (l-s anticipated negative observation outcomes) events, defining for a given number of visits l, a binomial probability distribution over possible observation outcomes. Cell c entropy for a particular s success/ l-s failure scenario resulting in a probability of target containment $p_{cT}(s,l)$ at the end of time horizon T is expressed by $E_c(p_{cT}(s,l))$ using equation (3) . Therefore, expected cell c entropy for a path solution involving l visits to c is defined as follows:

$$\overline{E}_{cl} = \sum_{s=0}^{l} p(s \mid l)\, E_c(p_{cT}(s,l)) \quad l \geq 1 \tag{4}$$

Where

$$p_{cT}(s,l) = \frac{1}{1 + \alpha_c^s \beta_c^{l-s} (\frac{1}{p_{c0}} - 1)} \tag{5}$$

$$p(s \mid l) = \sum_{x \in X} p(s \mid l, x)\, p(x) \tag{6}$$

$$= p(s \mid l, x=1)\, p_{c0} + p(s \mid l, x=0)(1 - p_{c0})$$

$$p(s \mid l, x=1) = \binom{l}{s} p_{cc}^s (1 - p_{cc})^{l-s} \tag{7}$$

$$p(s \mid l, x=0) = \binom{l}{s} p_{fc}^s (1 - p_{fc})^{l-s} \tag{8}$$

$$\alpha_c = \frac{p_{fc}}{p_{cc}}; \quad \beta_c = \frac{1 - p_{fc}}{1 - p_{cc}} \tag{9}$$

$$\overline{E}_{c0} = E_{c0} \tag{10}$$

Equation (5) describes posterior cell c target occupancy belief at the end of horizon T. It derives from recursive expressions (1) and (2), where p_{c0} refers to cell c target occupancy initial belief. The probability $p(s|l)$ for a sensing agent to read s positive observations out of l visits is reflected in (6). The expression $p(s|l,x)$ is a binomial probability distribution of positive observations, giving the probability to obtain s success out of l visits, conditional on occupancy state x. Inspired from section 2.2, p_{cc} ($p(z_t=1|X_c=1)$) and p_{fc} ($p(z_t=1|X_c=0)$) refer to probability of correct observation and false alarm rate on cell c respectively. Agent homogeneity and symmetry over equivalent sequence of success/failure events allows considering a linear rather than an exponential number of scenarios, and therefore a reduction of contributing terms from 2^l to l.

3.2 Model formulation

In this section we present a search-theoretic -based network flow formulation with resource constraints. A network representation mainly inspired from a previous work [29] is reused to simplify problem and constraint modeling ultimately reducing run-time, as it implicitly captures key constraints by construction. The basic approach reported in [29] is summarized in the next paragraph.

As referenced in [29], let $\mathcal{G}_k=(\mathcal{V}_k, \mathcal{A}_k)$ be a directed acyclic graph (the grid network) coupled to agent $k \in \eta=\{1,\ldots,n\}$. $\mathcal{V}_k = \bigcup_{t\in\mathcal{T}} \mathcal{V}_{kt}$ is a set of vertices defining agent states. A state is expressed in terms of agent's position and orientation during a given episode $t \in \mathcal{T} =\{1,2,\ldots,T\}$. $\mathcal{A}_k$ is a set of arcs (i,j) where $i,j \in \mathcal{V}_k$, connecting a prior state i to a posterior state j. It captures episodic state transition describing possible agent moves resulting from the action set A=$\{left, ahead, right\}$. $N_{kt} = N$ and $O_{kt} = O$ refer to admissible agent cell locations $\{1,\ldots,|N|\}$ and orientations/headings $\{E,NE,N,NW,W,SW,S,SE\}$ respectively during episode t. It turns out that $\mathcal{V}_k = \bigcup_{t\in\mathcal{T}} \mathcal{V}_{kt} = \bigcup_{t\in\mathcal{T}} (N_{kt} \times O_{kt})$. Fictitious origin ($o$) and destination ($d$) location nodes are artificially introduced to conveniently define legitimate path in the graph. An agent sub-network representation over two consecutive episodes is depicted in Figure 4. An integer binary flow decision variable x_{ijk} linked to an arc $(i,j)\in\mathcal{A}_k$ defines a basic agent path's construct. Accordingly, agent k's path solution includes 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_{cl} reflecting that l visits on cell $c(j)$ (via posterior state j transitions) are part of the agent path solution when $v_{cl} = 1$. 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.

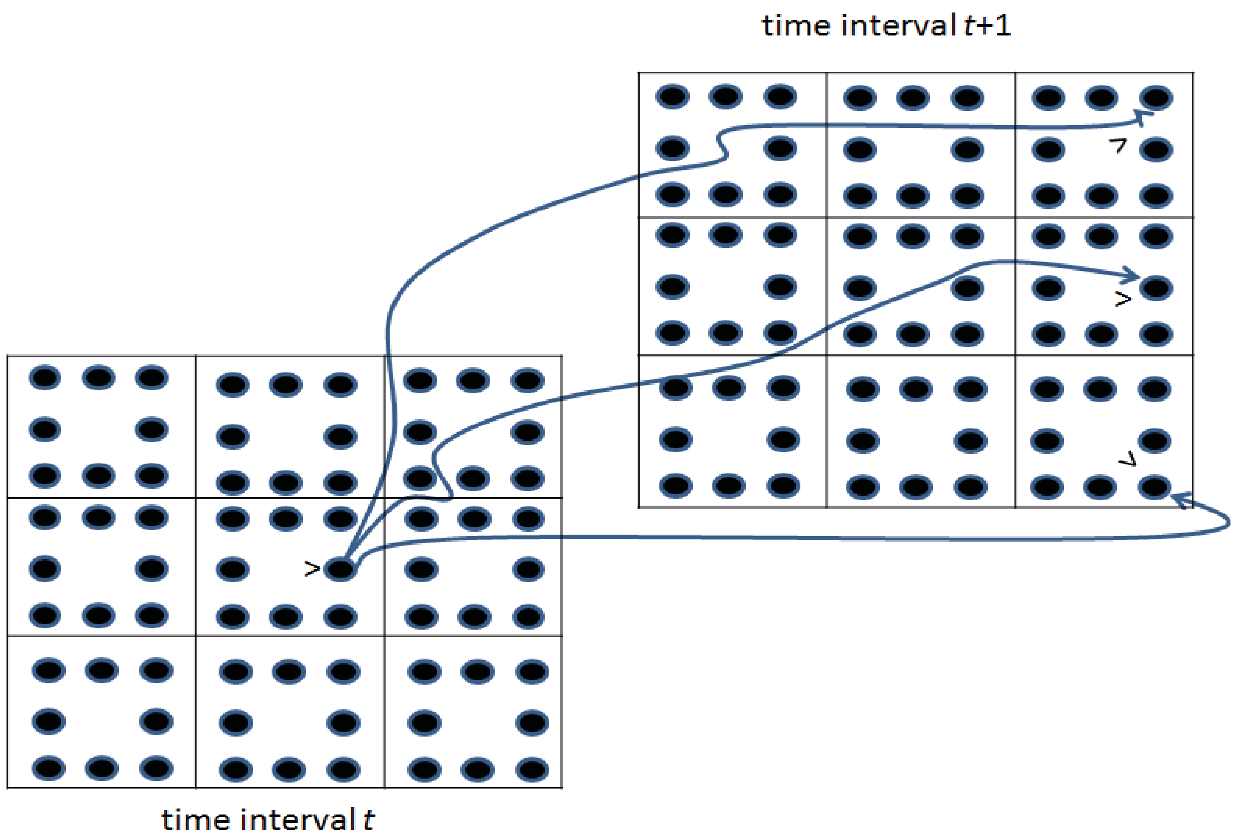


Figure 4. Agent grid network reflecting agent state transition over a 3x3 -cell grid. An agent characterized by a specific position and orientation during episode t (prior state) is moving to a legal neighbor cell modifying position (square/cell) and orientation (black dots) which stand over time interval t+1 (posterior state). A path solution may be built from the network moving along legal arcs connecting state nodes.

The parameters and variables defining the decision model are given as follows:

Parameters:

η : set of homogeneous agents $\{1,2,\ldots,n\}$

N: set of cells defining the grid search area $\{1,2,..,|N|\}$

$\mathcal{T}$: set of time intervals defining the time horizon $\{1,..., T\}$

V_c: maximum number of visits on cell $c \in N$

$\overline{E}_{cl}$: expected entropy (target occupancy uncertainty) over cell $c \in N$ after l visits ($l \in \{0,1,..., V_c\}$)

E_{c0} : initial entropy over cell c

E_{thr} : threshold entropy value under which certainty is assumed for a given cell

p_{c0}: initial belief of target occupancy over cell c

Decision variables:

v_{cl}: binary decision variable corresponding to the number of visits l ($l \in \{0,1,\dots, V_c\}$) on cell $c \in N$ by the end of the time horizon – $v_{cl}=1$ (otherwise 0)

x_{ijk}: state transition binary variable. $x_{ijk} = 1$ reflects agent k network state transition from state i to j between consecutive episodes. Agent k's path solution includes arcs $(i,j) \in A_k$ for which $x_{ijk} = 1$

The corresponding search-theoretic -based ILP decision model of the problem *SPP* may be defined as an expected system entropy minimization problem formulation:

$$\underset{\{v_{cl}\},\{x_{ijk}\}}{Min} \quad \sum_{c \in N} \sum_{l=0}^{V_c} \overline{E}_{cl} v_{cl} \tag{11}$$

Subject to the constraint set:

$$\sum_{i \in \mathcal{V}_k} \sum_{k \in \eta} x_{ijk} - \sum_{l=0}^{V_c} l v_{cl} = 0 \qquad \forall c \in N, (i, j(c)) \in \mathcal{A}_k \tag{12}$$

$$\sum_{l=0}^{V_c} v_{cl} = 1 \qquad \forall c \in N \tag{13}$$

$$v_{c0} > E_{thr} - E_{c0} \qquad \forall c \in N \tag{14}$$

Initial agent position

$$x_{o\, i_0(k)\, k} = 1 \qquad \forall k \in \eta, i_0(k) \in \mathcal{V}_k \tag{15}$$

Initial/final path condition

$$\sum_{i \in \mathcal{V}_k} x_{oik} = 1 \qquad \forall k \in \eta \tag{16}$$

$$\sum_{i \in \mathcal{V}_k} x_{idk} = 1 \qquad \forall k \in \eta \tag{17}$$

Flow conservation:

$$\sum_{i \in \mathbb{V}_k^{\circ} \cup \{o\}} x_{ijk} - \sum_{i \in \mathbb{V}_k^{\circ} \cup \{d\}} x_{jik} = 0 \quad \forall k \in \eta, \forall j \in \mathbb{V}_k^{\circ}, (i,j) \in \mathcal{A}_k \tag{18}$$

Maximum path length

$$\sum \sum_{(i,j) \in \mathcal{A}_k} x_{ijk} = T \qquad \forall k \in \eta \tag{19}$$

Binary decision variables

$$v_{cl} \in \{0,1\} \quad \forall c \in N, \forall l \in \{0,1,\ldots,V_c\} \tag{20}$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in \eta, (i,j) \in \mathcal{A}_k \tag{21}$$

The objective function shown in (11) refers to overall expected entropy summing up average entropy contributions over cell c assuming at most V_c visits at the site. The bound V_c can be pre-computed or selected arbitrarily large such that (11) may safely capture the optimal solution. Constraints are modeled through equations (12)-(21). Coupling constraints (12) first map number of visits and incoming arcs to a site c. The arc $(i,j(c))$ relates to any agent state transition terminating in position c. Constraints (13) simply represent the number of visits to be ultimately paid on site c. Constraints (14) capture entropy threshold E_{thr} under which target occupancy certainty may be initially assumed (e.g. $E_{thr} = E(0.99)$). Initial cell c entropy is captured by E_{c0}. Network flow conservation and itinerary constraints (15)-(19) are based on the formulation exposed in [29] . Constraints (15)-(17) ensure path solution departure and end points to be uniquely defined. Flow conservation dictated by constraints (18) aims at balancing the number of incoming and outgoing arcs respectively for a given node. Constraints (19) guarantee a T-move path solution for an agent, but turn out to be unnecessary as solution constraints are implicitly satisfied by agent graph construction. Finally, v_{cl} and x_{ijk} refer to binary decision variables for the number of visits l on cell c and agent state transition along arcs (i,j) at each move respectively. The visit decision variable assignment $v_{cl} = 1$ corresponds to a path solution including l visits to cell c. The assignment $x_{ijk} = 1$ reflects an agent k legally transitioning from state i to j.

3.3 Single Team Network Simplification

Given agent homogeneity, a single 'team' T-stage network $\mathcal{G}=(\mathcal{V},\mathcal{A})$ to represent all team paths at once may be used. Single network utilization instead of multiple network-agent mapping provides additional speed-up, number of decision variable reduction and significant computer space and management savings (by a factor n). The proposed team directed acyclic graph $\mathcal{G}=(\mathcal{V},\mathcal{A})$ alternatively captures agent multiplicity substituting binary integer variables x_{ijk} by integer flow decision variables x_{ij} ranging over $\{1,2,\ldots,n\}$ to account for possible concurrent arc traversal by multiple agents. The change simply requires to slightly revisit some flow constraints to reflect agent multiplicity. The computational gain anticipated simply requires the modest extra cost of reconstructing individual agent paths from the overall team solution. As agents share an identical observation model assumed to be strictly occupancy state-dependent, any resulting path solutions converge to the same overall solution quality for similar cell coverage.

3.4 Discussion

It is worth noticing that the proposed problem formulation turns out to be very attractive over alternate existing modeling approaches, as the linear model provides a lower bound on solution quality (MIP problem model) using Lagrangian relaxation. The computed bound offers a sound basis to achieve performance comparison with other techniques or support performance gap analysis to trade-off solution quality and run-time when working under stringent temporal constraints. Constraint handling effort may be further reduced through network construction and node duplication strategies whenever required. The proposed linearization approach can also be easily adapted to different objective functions, substituting expected entropy by any measure of performance (e.g. target discovery maximization). An optimal solution can be computed using the IBM CPLEX software [11] package, a well-recognized problem-solving tool exploiting a variety of powerful techniques efficiently implemented.

4 Computational Experiments

A computational experiment has been conducted to show the value of the proposed framework. Advocated methodology, tested algorithms and computational results are presented next.

4.1 Methodology

The experiment aims at comparing performance of the proposed ILP approach to an alternate heuristic. Computed solutions from respective methods are reported against two measures of performance, namely, differential relative information gain ($[g_T^* - g_T]/g_T^*$) and computational run-time. Differential relative information gain *RIG(T)* shown at the end of the horizon *T* is defined as follows:

$$RIG(T) = \frac{g_T^* - g_T}{g_T^*} = \frac{(E_0 - \overline{E}_T^*) - (E_0 - \overline{E}_T)}{E_0 - \overline{E}_T^*} = \frac{\overline{E}_T - \overline{E}_T^*}{E_0 - \overline{E}_T^*} \tag{22}$$

where E_0, $\overline{E}_T$ and $\overline{E}_T^*$ are the initial system entropy, the expected entropy computed by the heuristic at the end of episode *T*, and the corresponding optimal expected entropy (or a lower bound if the optimal value can't be computed) from the ILP model respectively. The larger the final entropy gap, the better the relative ILP performance. Run-time is measured in seconds and must not significantly exceed a specific deadline (typically 1 to 2 minutes) in order to account for physical movements between regions (cells) and support feasible cell exploration (visit duration).

Computer simulations were conducted under the following conditions.

- The search environment is a regular grid involving $|N|$ regular cells or regions. $|N|$ runs over $\{15^2, 20^2, 25^2, 30^2\}$.
- $V_c = 10$
- Prior cell occupancy belief magnitude probability density function:

  exponential ($f_\lambda(p_0) = \frac{\lambda}{1-e^{-\lambda}} e^{-\lambda p_0}$) $\lambda \in \Re^+$

  $$\overline{p}_0 = \frac{1 - (1+\lambda)e^{-\lambda}}{\lambda(1-e^{-\lambda})}, \quad (\lambda = 2, 5)$$

  quasi-uniform: λ=0.1, $\overline{p}_0 \approx 1/2$
- Prior cell occupancy belief spatial distribution: uniform
- Agent

  Homogeneous sensors (same sensor characteristics)

  team size *n* (5,10)

  Actions: 3 moves possible: right, ahead, left

  Sensor parameters: $p_c = 0.8$, $p_f = 0.1$

  Initial position/orientation: uniform over the grid

- Planning time horizon range: {12,14,16,18,20}
- Scenario data set: 84 problems, 10 instances/problem, one simulation run per instance
- Hardware Platform:

  Intel (R) Xeon (R) CPU X5670

  Shared-memory multi-processing: 8 processors, 2.93 GHz

  Random Access Memory: 16 Gb

  64 bits binary representation (double precision)

Key scenario parameters were selected to ensure acceptable grid coverage ($nT/|N|$) by the team to support meaningful comparative performance analysis (e.g. $0.1 < nT/|N| < 0.5$ approximately).

4.2 Algorithms

4.2.1 ILP - CPLEX Solver

The IBM ILOG CPLEX parallel Optimizer version 12.2.0.0 [11] presenting the best software tool commercially available to solve large size problems has been exploited. Three separate implementations of the proposed ILP approach were considered. These include classical mixed integer programming (MIP) advocated by CPLEX, an hybrid combination of Lagrangian relaxation and classical (LP+MIP), and, pure Lagrangian relaxation (LP). The rationale for the hybrid approach (LP+MIP) relies on its potential to further prune the solution space and significantly reduce computational time in solving practical size problems. It relies on a two-step process. A problem in which all integer decision variables are relaxed (now ranging in a continuous domain) is first optimized (the LP phase). As a result, decision variables naturally emerging with null values are further exploited in the next step. Those specific decision variables are fixed to their corresponding values (0) in the original MIP decision problem to be solved in a second step (the MIP phase). The revisited constrained MIP problem then consists in instantiating the remaining integer decision variables. The two-step approach aims at eliminating low-payoff subpaths in the first place (therefore reducing the number of decision variables) while focusing on highest payoffs contentious subroute candidates (expressed through residual integer decision variables) in order to build the final path solution. The proposed near-optimal strategy is basically designed to bring additional speedup. Many variants of the idea might be explored to suitably trade-off solution quality and run-time. The LP+MIP procedure may be summarized as follows:

1. Compute solution S_L from the Lagrangian relaxation version of the *SPP* problem.

2. Compute solution to the original *SPP* problem subject to an additional constraint set: the constraint consists in setting to zero *SPP* decision variables for which corresponding computed S_L decision variables are null, significantly reducing combinatorial complexity.

Single team network and multiple agent network implementations have also been carried out for run-time comparison purposes.

4.2.2 Myopic algorithm

The greedy 1-step limited lookahead method consists in myopically planning moves one step ahead at a time, progressively visiting the closest neighbor cell with highest expected information gain (differential entropy reward). At each time step, the agent with the highest reward is selected first, and cell reward updated accordingly. Should a dead-end occur (e.g. physical boundary impeding any legal moves), the agent backtracks as much as required, exploring alternate directions to finally resynchronize with the current episode. The process is then repeated for the remaining agents, one at a time, until all team agents have been covered. The overall procedure is then reiterated for each episode over time horizon *T*. Run-time $\in$ O(nT). The basics of the algorithm are highlighted as follows:

$path_k = \phi \quad \forall k \in \eta = \{1,..,.n\}$

$for\ t \in \{1,...,T\}\ do$

$\quad \eta' = \eta$

$\quad While\ \eta' \neq \phi\ do$

$$cell_k = \underset{cell \in Neigh(k'),\ k' \in \eta'}{\arg\max} \text{ (expected information gain over a single time step)}$$

$\qquad path_k.cell(t) = cell_k$

$\qquad \eta' = \eta' / \{k\}$

$\quad end\ while$

$end\ for$

Deadlock detection condition verification has been deliberately omitted from the above algorithm description to simply expose the underlying idea, and for readability purposes. On such circumstance, necessary backtracking to a past suitable state is carried out to determine alternate valid agent legal move and resynchronize with other team members. This condition may occur for an agent moving close to grid boundaries.

4.3 Results and analysis

The experiment was conducted for a data set including 840 problem configurations, involving one trial (simulation run) per problem instance. The results show the extent and the related conditions under which the ILP CPLEX solver dominates the myopic search path planner.

4.3.1 Relative information gain

The overall average comparative performance gap for the examined data set is more than 27% as shown in Table 1.

Table 1 - Relative Information gain of LP+MIP over Myopic heuristic

LP+MIP Vs. Myopic Relative Information Gain	
Average : 27.1% Max: 47%	Standard Deviation: 6%

The relative information gain distribution is displayed in Figure 5. 99% of the simulation configurations show a minimal RIG of at least 12.5%. Similarly, 90%, 62% and 30% of the simulation runs reveal minimal RIG of at least 20%, 25%, and 30% for the respective proportions.

An instance even exhibits a 47% gain. The differential relative entropy clearly demonstrates the value of predictive/advance planning (ILP) over a limited lookahead myopic attitude. The reported relative information gains are

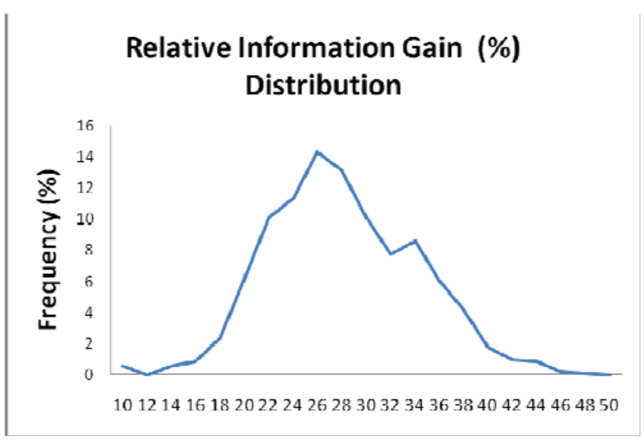


Figure 5. Relative information gain distribution, mapping frequency of simulation runs against performance (RIG).

expected to increase even further with sensor quality (smaller false alarm rate and larger probability of correct detection). Accordingly, long-term planning in near-perfect sensing environment is likely to outperform myopic planning by a larger margin due to a smaller frequency of false-alarm event occurrences in building a path solution over a longer time horizon. The same argument may hold for sparsely clustered entropy distributions.

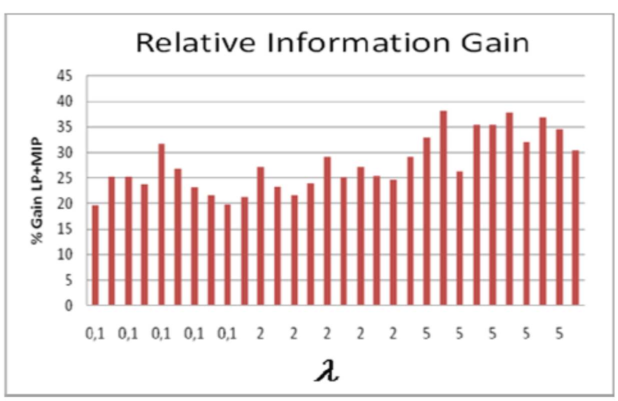
Performance gap with respect to the lower bound computed from the Lagrangian relaxation decision model noticeably exhibits a tight bound to optimality, as presented in Table 2. On average, LP+MIP obtains path solution quality within at most 0.75% of the optimal path, with a standard deviation of 0.79%, qualifying what is meant by near-optimality. The worst case indicates a gap less than 4% from the under-estimated lower bound. It would be possible to even further reduce this gap by leaving open, additional decision variables during the post-Lagrangian relaxation phase of the algorithm. This would come at the expense of modest additional run-time.

Table 2 - Relative Information Gain of LP (Lagrangian relaxation) over LP+MIP

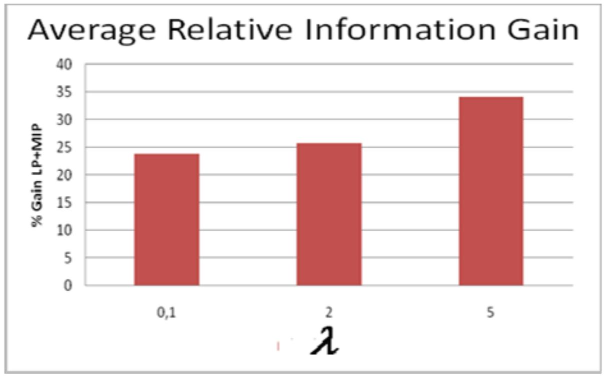
LP (Lagrangian relaxation) – lower bound Vs. LP+MIP Relative Information Gain	
Average : 0.75% Max: < 4%	Standard Deviation: 0.79%

It is worth mentioning that path solution quality computed by the exact MIP model is practically similar to the LP+MIP approach.

Simulation results describing performance gain (LP+MIP over Myopic) against prior belief λ-defined exponential distribution for a typical problem instance are given in Figure 6.



6a)



6b)

Figure 6. Performance gap against prior belief λ-defined exponential distribution.

Fig. 6(a) and 6(b) respectively present detailed and average results, exhibiting proportionality between performance gap and λ. Given the slow progression of the entropy curve around p=1/2, and the negative correlation governing λ and $\overline{p}_0$, problem configurations with smaller $\overline{p}_0$ values are likely to lead to higher long-term relative gain. In the case λ = 0.1 ($\overline{p}_0 \rightarrow 1/2$), prior belief evolves from an exponential to a quasi-uniform distribution, meaning that on average, an agent may often arbitrarily move in a suitable direction with a high payoff (maximum uncertainty reduction), consequently mitigating the value of advance (forward-looking) planning over a myopic view.

Figure 7 shows average performance gap against time horizon for a particular problem sample. Despite a weak proportionality, the small comparative gap between T=12 and T=20 (about 3%) suggests that small time horizon may be satisfactory in practice when run-time consumption need to be considered.

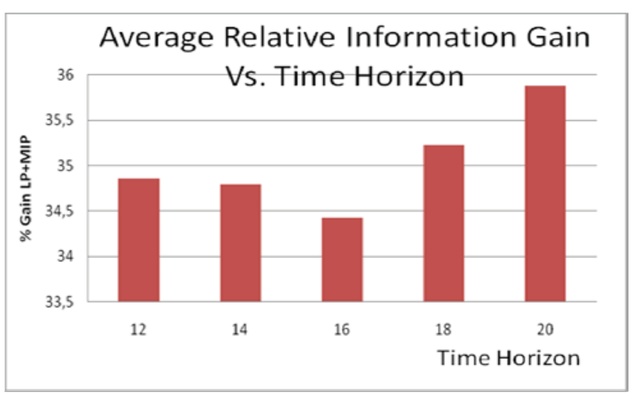


Figure 7. Average performance gap against planning time horizon.

In other respect, partial results have shown the proposed approach to be robust regarding false alarm rate p_f. Accordingly, computed path solution remains invariant when p_f slightly fluctuates around 10%.

4.3.2 Run-time

Comparative run-time performances are presented in Table 3 for the various ILP CPLEX optimizers and the myopic heuristic. Path solutions were computed to optimality for LP+MIP, and within 1% to optimality for the original MIP problem model respectively. Reported run-time is expressed in seconds (s). LP+MIP turns out to run an order of magnitude faster than the classical MIP procedure for equivalent quality solution.

Table 3 – Comparative Run-time Performance

Run-time (s)			
LP+MIP Single Net	LP+MIP Multiple Nets	MIP Single Net	Myopic heuristic
Average/Std Dev	Average/Std Dev	Average/Std Dev	Average/Std Dev
10.8 s	25.5 s	180.0 s	0.36 s
13.1 s	28.6 s	416.4 s	0.06 s
Max : 78 s			

The LP+MIP single network representation also shows a computational advantage over its multiple network model counterpart as run-time gain amounts to an average speedup of at least 2. The proportion of computer simulation exceeding a minute execution time is less than 1.7% as maximum run-time amounted to 78 seconds. A graphical representation of comparative run-time performances on a logarithmic scale is pictured in Figure 8.

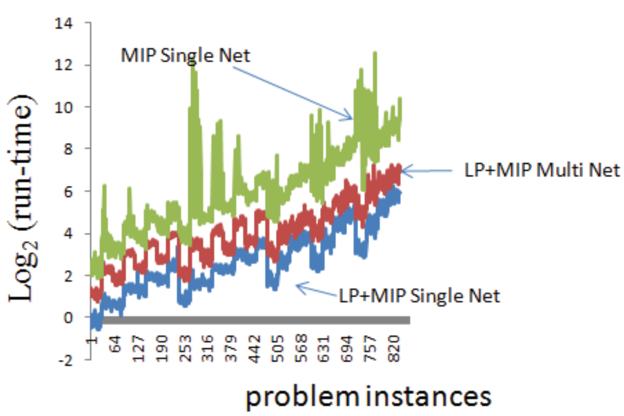


Figure 8. Comparative run-times (logarithmic scale) against problem instances.

Figure 9 displays simulation results over all data sets relating run-time to planning time horizon. In counterpart, Figure 10 maps average run-time against team size for a typical problem sample. As expected in both cases, run-time increases with time horizon and team size. Graphical results are self-explanatory.

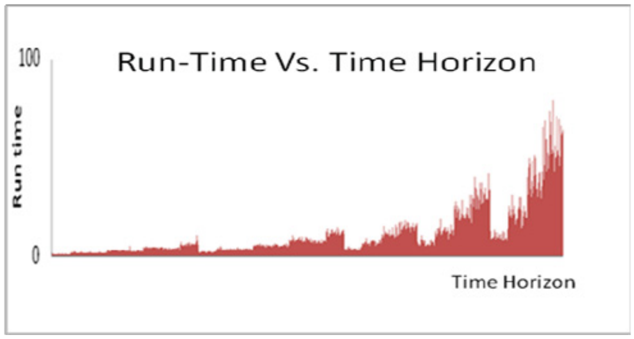


Figure 9. Run-time against time horizon, over all data sets.

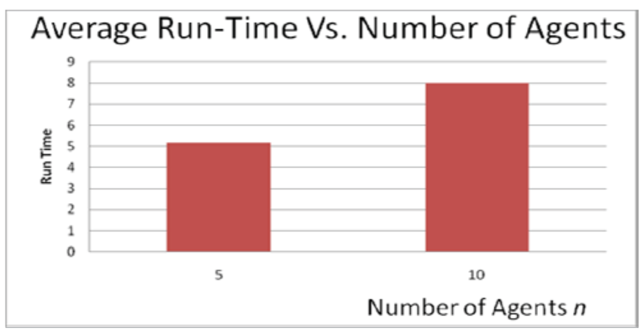


Figure 10. Run-time against agent team size for a typical problem sample, assuming other parameters identical.

Classes of scenarios initially depicting high team member competition for same site visits (high contention problem instances) finally show comparable problem-solving run-time to less contentious configurations (given similar simulation conditions for alternate parameters).

The overall run-time results clearly demonstrate that problem sizes involving 5/10 agents over a 20/18 -step planning time horizon, may be repeatedly solved very rapidly (on a minute time scale) over consecutive episodes (receding horizons). To a lesser extent, run-time performance does not scale with grid size either. However, execution time shown for 30x30 grid instances and 10-agent problems over horizon T=12 appears nonetheless quite acceptable. The impact of grid dimension on the performance of the approach remains therefore largely marginal in practice, as opposed to time horizon.

Additional gain can be contemplated starting from a good feasible initial solution, improving implementation, and resorting to the use of a large scale multi-processing environment. Besides speedup, the ILP approach is likely to show measurable gain in comparison to alternate heuristic methods, while further delivering a tight lower bound on the optimal path solution.

5 Conclusion

A new information-theoretic –based integer linear programming (ILP) formulation has been presented to solve the discrete search path planning problem in its open-loop with anticipated feedback form involving false-alarms. Exploiting system entropy function separability, conditional observation probability independence, innovative network representation, and parallel processing CPLEX technology, the proposed approach departs from traditional heuristic-based perspectives and techniques to cost-effectively compute near-optimal solution. Optimality gap obtainable from linear model relaxation allows conducting comparative performance analysis and/or balancing expected gain in path solution quality against run-time. Problems having large time horizons may be adapted to a dynamic setting by repeatedly solving new static problem instances over a rolling horizon, injecting observation outcomes from the last episode. Computational results shown for practical problem instances comprising up to ten agents prove the value of the proposed approach.

Future work aims at naturally extending the current decision model to capture a heterogeneous agent team, and investigate its practical limitations. Alternate directions consist in adapting the approach to different search objectives such as proportion of target discovery or expected detection time optimization. Other challenges lie in modeling search problem variants incorporating more complex observation models and various target occupancy dependency and domain constraints, possibly infringing separability and symmetry assumptions. Multi-dimensional search problems involving complex domain knowledge modeled as belief networks represent another challenge as well.

References

[1] N. Agmon, et al., The giving tree: constructing trees for efficient offline and online multi-robot coverage, Ann. Math. Artif. Intell. 52, 2008, 109–142.

[2] S. Benkoski, M. Monticino, and J. Weisinger, A Survey of the Search Theory Literature, Naval Research Logistics 38, 1991, 469–494.

[3] T. Chung, On Probabilistic Search Decisions under Searcher Motion Constraints. Workshop on Algorithmic Foundation of Robotics VIII, Guanajuato, Mexico. 2009, 501-516.

[4] 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.

[5] T. Cover, and J. Thomas, Elements of Information-Theory, 2nd edition, Wiley, 2006.

[6] A. Finn, et al., Design Challenges for an Autonomous Cooperative of UAVs, In Proceedings of the International Conference on Information, Decision and Control, 2007, pp. 160-169.

[7] F. E. Flint, E. Fernandez-Gaucherand, and M. Polycarpou, Efficient Bayesian Methods for Updating and Storing Uncertain Search Information for UAV's, Proc. 43rd IEEE Conf on Decision and Control, 2004, pp. 1093-1098.

[8] R. Hohzaki, and K. Iida, Optimal Search Plan for a Moving Target When a Search Path is Given. Math. Japonica, 41(1), 1995, 175-184.

[9] G.A. Hollinger, Search in the Physical World. CMU-RI-TR-10-20, Robotics Institute, PhD thesis, Carnegie Mellon University, 2010.

[10] G.A. Hollinger, and S. Singh, GSST: Anytime Guaranteed Search with Spanning Trees. Autonomous Robots, 29(1), 99-118, 2010.

[11] IBM ILOG CPLEX Optimizer. http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/cplex-optimizer/cplex-performance/

[12] 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, 571-587.

[13] 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.

[14] H. Lau, Optimal Search in Structured Environments, PhD Thesis, University of Technology, Sydney 2007.

[15] 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.

[16] 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.

[17] G. Martins, A New Branch-and-Bound Procedure for Computing Optimal Search Paths. Master's Thesis. Naval Postgraduate School. 1993.

[18] G. Mathews, and H. Durrant-Whyte, Scalable Decentralised Control for Multi-Platform Reconnaissance and Information Gathering Tasks. In 9th International Conference on Information Fusion, 2006, pp. 1-8.

[19] I. Rekleitis, et al., Efficient Boustrophedon Multi-Robot Coverage: an Algorithmic Approach, Ann. Math. Artif. Intell. 52, 2008, 109–142.

[20] 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.

[21] J. Svennebring, and S. Koenig, Building Terrain-covering Ant Robots: A Feasibility Study, Auton. Robots, 16 (3), 2004, 313–332.

[22] L. Stone, What's Happened in Search Theory Since the 1975 Lanchester Prize? *Operations Research*, **37** (3), 1989.

[23] P. Vincent, and I. Rubin, A Framework and Analysis for Cooperative Search Using UAV Swarms. ACM Symposium on Applied Computing, 2004, pp. 79-86.

[24] W. Wagner, et al., Distributed Covering by Ant-Robots Using Evaporating Traces, IEEE Trans. Robot. Autom., 15(5), 1999, 918-933.

[25] A. R.Washburn, Branch and Bound Methods for a Search Problem. Naval Research Logistics, 45, 1998, 243-257.

[26] 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, 1685–1690.

[27] S. Yang, and C. Luo, A Neural Network Approach to Complete Coverage Path Planning, IEEE Trans. Syst.,Man, Cybern. B, Cybern., 34(1), 2004, 718–724.

[28] Y. Yang, A. Minai, and M. Polycarpou, Evidential Map Building Approaches for Multi-UAV Cooperative Search. Proceedings of the American Control Conference, 2005, pp. 116–121.

[29] J.Berger and N. Lo, An innovative multi-agent search-and-rescue path planning approach, Computers & Operations Research, 53, 2015, 24 -31.