2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

An Outer-Approximation Algorithm for Generalized Maximum Entropy Sampling

Han-Lim Choi, Jonathan P. How, and Paul I. Barton

Abstract— **This paper presents an outer-approximation algorithm to address a generalized maximum entropy sampling (GMES) problem that determines a set of measurement locations providing the largest entropy reduction. A new mixed-integer semidefinite program (MISDP) formulation is proposed to handle a GMES problem with a jointly Gaussian distribution over the search space. This formulation employs binary variables to indicate if the corresponding measurement location is selected, and exploits the linear equivalent form of a bilinear term involving binary variables to ensure convexity of the objective function and linearity of the constraint functions. An outer-approximation algorithm is developed for this formulation that obtains the optimal solution by solving a sequence of mixed-integer linear programs. Numerical experiments are presented to verify the solution optimality and the computational effectiveness of the proposed algorithm by comparing it with an existing branch-and-bound method that utilizes nonlinear programming relaxation. Sensor selection for best tracking of a moving target under a communication budget constraint is specifically considered to validate the superiority of the suggested algorithm in handling quadratic constraints.**

I. INTRODUCTION

One frequently addressed objective in the context of sensor networks is to find a set of measurement points that leads to the largest reduction in entropy of certain random variables of interest, which is referred to as *maximum information gain sampling* (MIGS). Guestrin et al. [1]–[3] dealt with sensor placement in a finite gridspace to achieve the largest entropy reduction of a sensor-vacant region when the spatial distribution of the temperature is described by a Gaussian process or a graphical model. Williams et al. [4] addressed a sensor management problem that schedules the sensors to turn on in order to best track the motion of moving targets with considering communication cost as well. Zhang and Ji [5] handled facial expression understanding within the dynamic Bayesian network framework by treating each facial motion as a measurement and each expressional attribute as a quantity of interest. Recently, Choi et al. [6,7] addressed the targeting of mobile sensor platforms to improve the weather forecast at a specified verification site. In addition, communication- and power-aware multi-sensor cooperation can be addressed as a decentralized form of MIGS.

As a similar concept addressed in different contexts, *maximum entropy sampling* (MES) is decision making to select a set of design points representing the largest entropy, which was first introduced for design of experiments [8] and was named by Shewry and Wynn [9]. In the case that all of the random variables are jointly Gaussian, MES corresponds to finding the submatrix of the covariance matrix that has the largest determinant; however, it was shown to be NP-hard even if all the entries of the covariance matrix are rational, whether or not the cardinality of the selected set is predetermined [10].

MIGS is a different (and usually harder) problem than MES, since information gain, unlike entropy, is not submodular in general [1]; however, MIGS and MES are closely related. MIGS for which the posterior covariance matrix is diagonal can be reduced to a decision very similar to MES, which this work refers to as *generalized maximum entropy sampling* (GMES). Since submodularity holds for GMES, solution techniques for MES can incorporate GMES without extensive modification. A sensor selection problem for moving target tracking presented in Williams et al. [4] is a GMES problem. GMES is further reduced to MES, if the posterior covariance matrix is a scalar multiplier of the identity matrix and the number of selection points is given. Also, MES-type decision making has approximated MIGS when the computation of the conditional entropy is computationally intractable [11]. Thus, developing a good solution strategy for MES (or GMES) can be conducive to solving MIGS.

One approach to find the optimal solution of MES (or MIGS) for the Gaussian case is to formulate it as an optimization problem employing binary variables to indicate which rows and columns of the covariance matrix will be chosen. This type of approach has been quite successful, and all existing optimization-based methods have been based on the branch-and-bound (BB) algorithm with various upper bounding mechanisms: largest eigenvalues of the covariance matrix [10,12]–[14], nonlinear programming relaxation using rational and exponential mixture function of the binary variables [14,15], partition-based spectral bound [16], linear-integer programming bound for improving the spectral bound [17], and factored mask spectral bound [18] that generalizes the eigenvalue bound and the partition-based bound.

In contrast, this work addresses the generalized maximum entropy sampling problem within the *outer-approximation* (OA) framework. The OA algorithm, which was first developed by Duran and Grossmann [19], extended to a general mixed-integer convex program (MICP) [20] and to mixed-

H. -L. Choi, PhD Candidate, Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA 02139, USA, hanlimc@mit.edu

J. P. How, Professor, Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA 02139, USA, jhow@mit.edu

P. I. Barton, Professor, Dept. of Chemical Engineering, MIT, Cambridge, MA 02139, USA, pib@mit.edu

integer (nonconvex) nonlinear programs (MINLP) [21,22], alternately solves a primal problem and a relaxed master problem. The primal problem is a nonlinear program (NLP) with all the integer variables being fixed in value, and the relaxed master problem is a mixed-integer linear program (MILP) constructed by linearizing the objective function and the constraints around a solution point of the primal problem. At each iteration, the best primal solution so far provides a lower bound (LBD) (in case of maximization), while the relaxed master problem gives a upper bound (UBD) on the optimal solution value and determines the next integer value to visit. The algorithm terminates when UBD converges to LBD, thus guaranteeing global optimality. The comparison of OA and BB in terms of computation time is controversial and problem-specific; however, OA has the following nice properties, which are exploited in the algorithm presented herein. First, it is no longer a concern to devise algorithmic heuristics such as the branching order and node selection, which, if inappropriately devised, could cause computational inefficiency in BB algorithms, because the order of integer values to visit is automatically determined by the relaxed master problem. Second, for pure integer-convex programs (ICP), the primal problem becomes just a function evaluation and only a sequence of MILPs needs to be solved. Thus, with a reliable solver for MILP such as CPLEX [24], an ICP can be solved very efficiently by OA.

This work presents a mixed-integer semidefinite program (MISDP) formulation for generalized maximum entropy sampling, in which binary variables indicate selection of the corresponding rows and columns, and continuous variables enable a convex reformulation of the objective function and the constraint functions. It will be shown that this formulation does not require the solution of any primal semidefinite program (SDP), since the primal feasible set is reduced to a singleton; therefore, only MILP relaxed master problems need to be solved. Algorithmic details of the proposed approach are described with highlighting the relative convenience of the computation of gradient and Hessian information in contrast to the case of the nonlinear programming-based BB (BB-NLP) algorithm. Numerical experiments validate the suggested method and compare its computation time with the BB-NLP method. In particular, the sensor management problem, which addresses measurement selection under a limited communication budget in order to minimize tracking uncertainty of a moving target, is presented to distinguish the performance of the proposed algorithm from that of the BB-NLP algorithm.

II. Problem Formulation

A. Generalized Maximum Entropy Sampling

Maximum entropy sampling determines the set of sensing points from a given search space that represents the largest entropy amongst them. If the joint probability distribution for any subset of the search space is Gaussian, MES corresponds to picking a principal submatrix of the covariance matrix for the search space $P \in \mathbb{R}^{N\times N}$ that provides the largest determinant:

$$\max_{\mathbf{s}\subset\mathcal{S}:|\mathbf{s}|=n} \log\det P[\mathbf{s},\mathbf{s}] \qquad (\mathbf{MES})$$

where $\mathcal{S} \triangleq [1,N]\cap\mathbb{Z}$ and $P[\mathbf{s},\mathbf{s}]$ denotes the $n\times n$ principal submatrix of P consisting of rows and columns indicated by index set $\mathbf{s}$. For the sake of well-posedness of the problem, P should be symmetric positive definite. The cardinality of $\mathbf{s}$ is usually specified, as otherwise the solution of (**MES**) is trivially $\mathcal{S}$ by the principle of "information never hurts." [23] The existence of other constraints may allow for removal of the cardinality restriction, although most existing algorithms for MES have assumed specified cardinality.

This paper considers the following constrained decision called generalized MES:

$$\begin{aligned}\max_{\mathbf{s}\subset\mathcal{S}} \;& \log\det P[\mathbf{s},\mathbf{s}] - \log\det Q[\mathbf{s},\mathbf{s}]\\ \text{s.t.}\;& A_{eq}\mathbf{y} = \mathbf{b}_{eq}, \quad A\mathbf{y}\le\mathbf{b}\\ & \mathbf{y}^TG_k\mathbf{y}\le g_k, \quad k=1,\cdots,m,\end{aligned} \qquad (\mathbf{GMES})$$

with $Q \succ 0$ being diagonal. The i-th element of the binary vector $\mathbf{y}\in\{0,1\}^N$ is related to $\mathbf{s}$ such that $y_i = 1$ if $i\in\mathbf{s}$, and 0 otherwise. In the context of sensor networks, Q may represent the posterior covariance matrix for the search space by the backward selection formulation described in Choi et al. [6]. Since Q is diagonal, $\log\det Q[\mathbf{s},\mathbf{s}] = \sum_{i=1}^N y_i \log q_{ii}$ where q_{ii} is the (i,i) element of Q. Regarding the constraints, a linear equality constraint can represent the cardinality constraint, while a linear inequality constraint can model power (or economic) budget limitation. The quadratic constraints can be used to represent restriction of communication capability. Note that the quadratic constraints defined by $G_k\in\mathbb{R}^{N\times N}$ and $g_k\in\mathbb{R}$ are in general *nonconvex*. To the authors' best knowledge, no optimization algorithm for MES has taken into account quadratic constraints, although information maximization with consideration of communication budget has been one of the most important issues in sensor network applications.

B. Mixed-Integer Semidefinite Program Formulation

This work poses the following mixed-integer semidefinite program (MISDP), which is a mixed-integer convex program (MICP), to address (**GMES**) described in the previous section:

$$\max_{\mathbf{y},\mathbf{x}} \; f(\mathbf{y},\mathbf{x}) \equiv \log\det S(\mathbf{y},\mathbf{x}) + \mathbf{c}^T\mathbf{y} + \mathbf{d}^T\mathbf{x} \qquad (\mathbf{P})$$

s.t.

$$S(\mathbf{y},\mathbf{x}) \equiv I + \textstyle\sum_{i=1}^N y_iY_i + \sum_{i=1}^{N-1}\sum_{j>i} x_{ij}X_{ij} \succ 0 \quad (1)$$

$$x_{ij}\le y_i, \quad x_{ij}\le y_j, \quad x_{ij}\ge y_i+y_j-1, \quad \forall i,\ \forall j>i \quad (2)$$

$$A_{eq}\mathbf{y} + B_{eq}\mathbf{x} = \mathbf{b}_{eq}, \quad A\mathbf{y}+B\mathbf{x}\le\mathbf{b} \quad (3)$$

$$\mathbf{y}\in\{0,1\}^N, \qquad \mathbf{x}\in[0,1]^{N(N-1)/2} \quad (4)$$

where Y_i and X_{ij} are defined as

$$Y_i = (p_{ii}-1)[\mathbf{e}_i\mathbf{e}_i^T], \quad X_{ij} = p_{ij}[\mathbf{e}_i\mathbf{e}_j^T + \mathbf{e}_j\mathbf{e}_i^T]. \quad (5)$$

p_{ij} is the (i,j) element of the matrix P, and $\mathbf{e}_i$ is the i-th unit vector. The set of linear constraints in (2) equivalently represent the bilinear relation $x_{ij} = y_i y_j$ when y_i and y_j are integers. Thus, (2) being satisfied, $S(\mathbf{y}, \mathbf{x})$ is related to the original covariance as follows

$$[S(\mathbf{y},\mathbf{x})]_{ij} = \begin{cases} p_{ij}, & \text{if } y_i = y_j = 1 \\ \delta_{ij}, & \text{otherwise} \end{cases} \tag{6}$$

where δ_{ij} is a Kronecker delta. Thus, the determinant of $S(\mathbf{y}, \mathbf{x})$ is equal to that of $P[\mathbf{s}, \mathbf{s}]$. The linear matrix inequality (LMI) constraint in (1) maintains the positive definiteness of $S(\mathbf{y}, \mathbf{x})$; note that positive definiteness is always satisfied with binary $\mathbf{y}$ and corresponding $\mathbf{x}$ that satisfies (2). Also note that a nonconvex quadratic constraint in (**GMES**) can be written as a linear constraint in terms of both $\mathbf{y}$ and $\mathbf{x}$ by replacing y_i^2 by y_i and $y_i y_j$ by x_{ij} for binary y_i's. Similarly, the linear term $\mathbf{d}^T\mathbf{x}$ in the objective function enables consideration of a bilinear cost function, although it is not involved in (**GMES**).

Observe that (**P**) becomes a convex program if integrality of $\mathbf{y}$ is relaxed, since LMI and linear constraints comprise a convex feasible set and $\log\det(\cdot)$ is a concave function in the space of symmetric positive definite matrices [26]. It should be pointed out that (**P**) is not the only possible way to formulate a MICP for GMES; however, the authors have found that the rational-exponential mixture formulation given in [15] is not suitable for the purpose of applying the outer-approximation algorithm because the gradient and Hessian are not defined everywhere for that formulation, while (**P**) might not be suitable for implementing branch-and-bound because of the computational burden of solving a large SDP relaxation.

III. Algorithm

A. Primal Problem

The primal problem for the k-th iteration of the OA algorithm is, in general, a convex program finding a best real decision vector $\mathbf{x}^\star(\mathbf{y}^k)$ for a given integer decision vector $\mathbf{y}^k$. In the case of pure integer programming, this reduces to a function evaluation using $\mathbf{y}^k$. It is noticed that the latter is the case for (**P**), although continuous decision variables x_{ij}'s are apparently involved. For any integer $\mathbf{y}^k$, constraint (2) restricts the feasible set for $\mathbf{x}$ to a singleton; the primal optimal solution is

$$x_{ij}^\star(\mathbf{y}^k) = y_i^k y_j^k \tag{7}$$

for a feasible $\mathbf{y}^k$. Then, the primal optimal objective value $f(\mathbf{y}^k, \mathbf{x}^\star(\mathbf{y}^k))$ becomes an underestimate of the optimal value of (**P**); if it is larger than the tightest lower bound LBD, it replaces LBD.

The integer vector $\mathbf{y}^k$ is the optimal solution to $(k-1)$-th relaxed master problem (section III-B) for $k > 1$; such $\mathbf{y}^k$ is always a feasible solution to (**P**), if (**P**) itself is a feasible problem. In order to generate the initial binary vector $\mathbf{y}^1$, this work proposes a MILP feasibility problem:

$$\max_{\mathbf{y},\mathbf{x}} \ \textstyle\sum_{i=1}^N y_i \log p_{ii} + \mathbf{c}^T\mathbf{y} + \mathbf{d}^T\mathbf{x}$$

subject to the same linear constraints as (**P**). This MILP provides an upper bounding solution to feasible (**P**) [7]; its infeasibility means (**P**) is an infeasible problem. In case the only constraint is cardinality restriction, the greedy solution [6] is a good feasible candidate for $\mathbf{y}^1$.

B. Relaxed Master Problem

The relaxed master problem is, in general, a mixed-integer linear program that optimizes the linear outer approximation of the objective function linearized at primal solution points over the feasible set. The relaxed master problem of (**P**) for the k-th iteration is written as follows:

$$\max_{\eta_k,\,\mathbf{y},\mathbf{x}} \ \eta_k \tag{$\mathbf{M}^k$}$$

s.t.

$$\eta_k \le f(\mathbf{y}^m, \mathbf{x}^\star(\mathbf{y}^m)) + \nabla f(\mathbf{y}^m, \mathbf{x}^\star(\mathbf{y}^m))^T \begin{pmatrix} \mathbf{y} - \mathbf{y}^m \\ \mathbf{x} - \mathbf{x}^\star(\mathbf{y}^m) \end{pmatrix}, \ \forall m \le k \tag{8}$$

$$x_{ij} \le y_i, \ x_{ij} \le y_j, \ x_{ij} \ge y_i + y_j - 1, \ \forall i, \ \forall j > i \tag{9}$$

$$A_{eq}\mathbf{y} + B_{eq}\mathbf{x} = \mathbf{b}_{eq}, \ A\mathbf{y} + B\mathbf{x} \le \mathbf{b} \tag{10}$$

$$\eta_k \in \mathbb{R}, \ \mathbf{y} \in \{0,1\}^N, \ \mathbf{x} \in [0,1]^{N(N-1)/2}. \tag{11}$$

The outer approximation of the LMI constraint in (1) can be neglected because (9) defines a subset of the feasible set of the LMI. Note that η_k is non-increasing in k because one constraint is added at every iteration, and it provides an upper bound on the optimal value $f^\star$. Thus, at every iteration η_k represents the tightest upper bound UBD. The algorithm terminates when UBD = LBD at a global optimum; every ($\mathbf{M}^k$) is feasible before termination, if (**P**) is feasible.

The gradient of the objective function $\nabla f(\mathbf{y}, \mathbf{x}^\star(\mathbf{y}))$ can be computed as

$$\tfrac{\partial f}{\partial y_i} = \left[S(\mathbf{y},\mathbf{x})^{-1}\right]_{ii}(p_{ii} - 1) + c_i \tag{12}$$

$$\tfrac{\partial f}{\partial x_{ij}}\big|_{\mathbf{x}^\star(\mathbf{y})} = 2\left[S(\mathbf{y},\mathbf{x})^{-1}\right]_{ij} p_{ij} + d_{ij} \tag{13}$$

by exploiting the self-concordance of the $\log\det$ function [25] where c_i and d_{ij} are corresponding elements in the linear objective term. It is noted that computation of the above gradient does not require inversion of a (possibly) large matrix $S(\mathbf{y}, \mathbf{x})$, which was often required for the case for the NLP-based branch-and-bound algorithm [15], since $S(\mathbf{y}, \mathbf{x})^{-1}$ is a sparse matrix with a very special form. It can be shown that

$$\left[S(\mathbf{y},\mathbf{x})^{-1}\right]_{ii} = 1, \qquad \text{if } y_i = 0 \tag{14}$$

$$\left[S(\mathbf{y},\mathbf{x})^{-1}\right]_{ij:\,i\ne j} = 0, \qquad \text{unless } y_i = y_j = 1. \tag{15}$$

Therefore, $S(\mathbf{y}, \mathbf{x})^{-1}$ can be computed effectively by inverting the submatrix corresponding to those $y_i = 1$.

IV. Numerical Experiments with MES

For validation of the proposed method, unconstrained MES problems that involve only the cardinality condition are first considered. Monte-Carlo experiments are performed using MATLAB 7.1 with TOMLAB/CPLEX 10.0 [27] to

TABLE I
AVERAGE COMPUTATION TIME (SEC.) [# OF UBD COMPUTATIONS]

N	n	OA		BB-NLP		# of cand.
20	10	3.5	[19.4]	11.1	[3.4]	184756
30	10	10.7	[20.8]	124.9	[8.6]	30045015
30	15	288.9	[122.8]	103.6	[10.6]	155117520
40	10	46.4	[36.0]	>1000	[N/A]	847660528

solve the MILP relaxed master problems. The covariance matrix is randomly generated as:

$$P = \frac{1}{M-1}\Pi\Pi^T, \quad \Pi \in \mathbb{R}^{N\times M} \tag{16}$$

where each entry Π_{ij} is i.i.d with $\mathcal{N}(0,1)$. For the purpose of comparison, NLP-based branch-and-bound (BB-NLP) [15] is also implemented for the same setting, with TOMLAB/BARNLP [28] being used to solve the associated NLP relaxations. The greedy rule [10] is adopted to determine the branching order, and a node corresponding to the largest upper bound is selected first. Optimality of the solutions by OA and BB-NLP is verified by comparing them with the solution from explicit enumeration with small-size problems.

Table I represents the average computation time and the average number of upper-bounding problems – MILPs for OA and NLPs for BB-NLP – of both algorithms. Five different P matrices for each (N,n) setting are generated with $M(=10N)$ sample vectors. The initial binary value $\mathbf{y}^1$ is selected in a greedy way. It is found that both algorithms perform comparably in general; but, for a certain size of problem OA performs much faster than BB-NLP.

V. SENSOR MANAGEMENT PROBLEM

A. Problem Description

The sensor management problem [4] addresses decision making on which sensors located at fixed positions to turn on under a limited communication budget, to reduce the uncertainty in the position and velocity estimate of a moving target over a specified time horizon. The motion of the target in two-dimensional space is assumed to be modeled by the following linear state-space model:

$$\mathbf{x}_{t+1} = F\mathbf{x}_t + \mathbf{w}_t \tag{17}$$

where $\mathbf{x} = [p_x\ v_x\ p_y\ v_y]^T$ and $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, W)$ is a white Gaussian noise. F and W are given as

$$F = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix},\ W = w\begin{bmatrix} \frac{\tau^3}{3} & \frac{\tau^2}{2} & 0 & 0 \\ \frac{\tau^2}{2} & \tau & 0 & 0 \\ 0 & 0 & \frac{\tau^3}{3} & \frac{\tau^2}{2} \\ 0 & 0 & \frac{\tau^2}{2} & \tau \end{bmatrix}. \tag{18}$$

The target is supposed to move along a straight line with constant speed, but the process noise represented by a random walk in acceleration perturbs the trajectory.

Denoting the measurement taken by s-th sensor at time t as z_t^s, a nonlinear measurement model is assumed:

$$z_t^s = h(\mathbf{x}_t, s) + v_t^s \tag{19}$$

where $v_t^s \sim \mathcal{N}(0, R_i)$ that is independent of process noises and sensing noises for other sensors. Each sensor measures the quasi-distance $h(\mathbf{x}_t, s) = \alpha/(||L\mathbf{x}_k - l^s||_2^2 + \beta)$ where L is the matrix that extracts the position components from the state, and l^s is the location of s-th sensor. The constants α and β are selected to model the signal-to-noise ratio (SNR) of the sensor.

The one-step lookahead sensor management decision at time t considers the following maximum information gain sensor selection:

$$\max_{\mathbf{s}_t}\ \mathcal{H}(\mathbf{x}_t|\mathbf{z}_{0:t-1}) - \mathcal{H}(\mathbf{x}_t|\mathbf{z}_t^{\mathbf{s}_t}, \mathbf{z}_{0:t-1}) \tag{20}$$

where $\mathbf{z}_{0:t-1}$ and $\mathbf{z}_t^{\mathbf{s}_t}$ denote the measurement sequence up until time $t-1$, and the current measurement taken by sensors $s \in \mathbf{s}_t$, respectively; $\mathcal{H}(X_1|X_2)$ represents the conditional entropy of a random vector X_1 conditioned on X_2. Note that formulation (20) is equivalent to the following *backward* formulation:

$$\max_{\mathbf{s}_t}\ \mathcal{H}(\mathbf{z}_t^{\mathbf{s}_t}|\mathbf{z}_{0:t-1}) - \mathcal{H}(\mathbf{z}_t^{\mathbf{s}_t}|\mathbf{x}_t, \mathbf{z}_{0:t-1}) \tag{21}$$

as a result of the commutativity of the mutual information [23]. The backward formulation provides better computational efficiency [6] and leads to a (**GMES**) problem under the linear Gaussian assumption. The linear Gaussian assumption means the entropy values of $\mathbf{z}_t$ can be well approximated by the $\log\det$ of its covariance matrix that can be derived by linearly propagating the covariance of the state estimate. Then, the prior entropy of $\mathbf{z}_t$ is presented as

$$\mathcal{H}(\mathbf{z}_t^{\mathbf{s}_t}|\mathbf{z}_{0:t-1}) = \frac{1}{2}\log\det\left(H_t^{\mathbf{s}_t}P_{\mathbf{x}_t|\mathbf{z}_{0:t-1}}(H_t^{\mathbf{s}_t})^T + R_{\mathbf{s}_t}\right) + \lambda$$

where $\lambda = \frac{1}{2}\log(2\pi e)|\mathbf{s}_t|$ and $H_t^{\mathbf{s}_t}$ is Jacobian of the observation function each row of which is represented as

$$H_t^s = \frac{-2\alpha}{||L\mathbf{x}_t - l^s||_2^2 + \beta}(L\mathbf{x}_t - l^s)^T L,\ \forall s \in \mathbf{s}_t. \tag{22}$$

Since the current measurement is conditionally independent of previous measurements for a given current state, the posterior entropy term is very simple:

$$\mathcal{H}(\mathbf{z}_t^{\mathbf{s}_t}|\mathbf{x}_t, \mathbf{z}_{0:t-1}) = \frac{1}{2}\sum_{s\in\mathbf{s}_t}\log R_s + \lambda. \tag{23}$$

The selection decision incurs communication cost depending on the communication topology. This work assumes that direct communication between two sensors incurs a cost proportional to the squared distance between them: $\tilde{B}_{ij} = \gamma||l^i - l^j||_2^2$ with an appropriate scaling coefficient γ, and that distant sensors can communicate each other using a multi-hop scheme. Thus, the communication cost between two arbitrary sensors is the accumulated cost along the shortest (in a squared distance sense) path: $B_{ij} = \sum_{k=1}^{n_{ij}}\tilde{B}_{i_{k-1}i_k}$ where $\{i_0, \cdots, i_{n_{ij}}\}$ is the shortest path from the sensor $i = i_0$ to $j = i_{n_{ij}}$. This work considers a particular worst case scenario in which every sensor must communicate to every other sensor in the selected set. The communication budget constraint in this case is written as $\sum_{i,j\in\mathbf{s}_t} B_{ij} \le B_{\max}$.

Thus, sensor selection with a communication constraint can be written as a generalized maximum entropy sampling problem:

$$\begin{aligned}&\max_{\mathbf{s}_t}\log\det P_{\mathcal{S}}[\mathbf{s}_t,\mathbf{s}_t] - \textstyle\sum_{i=1}^N y_i\log R_i \\ &\text{s.t. } \textstyle\sum_{i=1}^N\sum_{j>i} B_{ij}y_iy_j \le B_{max}\end{aligned} \tag{GMES-S}$$

where the covariance matrix of the search space $P_{\mathcal{S}} \in \mathbb{R}^{N \times N}$ is defined as $P_{\mathcal{S}} \equiv H_t^{\mathcal{S}} P_{\mathbf{x}_t|\mathbf{z}_{0:t-1}} (H_t^{\mathcal{S}})^T + R_{\mathcal{S}}$. Note that the cardinality of $\mathbf{s}_t$ is not specified in advance. In this work, the state covariance estimate $P_{\mathbf{x}_t|\mathbf{z}_{0:t-1}}$ is provided by an extended Kalman filter (EKF). Given this information, the presented outer-approximation algorithm can be implemented to (**GMES-S**) straightforwardly.

B. Modification of BB-NLP

A modified version of BB-NLP method is considered for comparison with the proposed outer-approximation algorithm; modification is needed because the original BB-NLP cannot handle quadratic constraints and unspecified cardinality. Introducing additional real variables $x_{ij} = y_i y_j$ with the set of constraints in (2) enables BB-NLP to deal with quadratic constraints. The original BB-NLP explicitly utilizes cardinality information to effectively construct the branch-and-bound tree. Two types of modification can be conceived regarding unspecified cardinality. One way is solving (**GMES-S**) with an additional cardinality constraint $\mathbf{1}^T\mathbf{y} = n$ for reasonably chosen values of n – call this way BB-NLP(1). The other way is modifying the branch-and-bound tree in such a way that lower bounds are computed for intermediate nodes as well as the leaf nodes, and leaf nodes are determined by infeasibility of the communication constraint rather than by the cardinality – denote this as BB-NLP(2). It was found empirically that the first way is usually faster than the second for small-size problems, while the opposite is the case for large-size problems.

C. Numerical Results

For numerical experiments, the following parameter values are set to be the same as in [4]:

$$\tau = 0.25,\ w = 0.01,\ \alpha = 2000,\ \beta = 100,\ R_i = 1. \quad (24)$$

A total of N sensors are located at fixed locations determined randomly on a 20×20 two-dimensional space; the pairwise communication cost values B_{ij}'s are computed by solving a shortest-path problem using dynamic programming [29]. The initial state value is $\mathbf{x}_0 = [0,\ 2,\ 0,\ 2]^T$, which results in the nominal position at t-th time step $(0.5t, 0.5t)$. The (**GMES-S**) sensor selection is addressed at time $t = 20$, before which an EKF has used randomly selected $n_0 = 10$ sensor measurements for state estimation every time step.

$N = 30, 40$ are used; five randomly generated sets of sensor deployments are considered for each N, while three different values of $B_{\max} = 100, 200, 300$ are taken into account for each deployment. The modified branch-and-bound method, BB-NLP(2) is used, as it performs faster than BB-NLP(1) for most problems of this size. Every MILP relaxed master problem in OA is solved using TOMLAB/CPLEX 10.0; TOMLAB/KNITRO [30] is utilized to solve NLP upper bounding subproblems for BB-NLP(2).

Table II shows average computation times and numbers of upper bounding problems for OA and BB-NLP(2) for various $(N, B_{\max})$ settings. The maximum cardinality of feasible sensor selection, $n_{\max}$, is also tabulated as an indicator of the problem complexity. Optimality of the solutions from OA and BB-NLP(2) are verified by crosscheck. First, it is noticeable that OA performs an order-of-magnitude faster than BB-NLP(2) with less than 100 subproblem calculations being needed for all the cases. BB-NLP requires a much larger number of subproblem computations than OA, while it solved less subproblems than OA for unconstrained MES cases. Seeing as unit computation time per UBD computation for BB-NLP is small, it can be inferred that the main cause of large computation time for BB-NLP is not merely introduction of additional variables x_{ij}'s but weakness of upper bounds from its NLP relaxations. The linear representation in (2) is equivalent to the bilinear relation $x_{ij} = y_i y_j$ for integral $\mathbf{y}$; however, such x_{ij} can be far from $y_i y_j$ if integrality of $\mathbf{y}$ is relaxed.

TABLE II

AVG. COMP. TIME (SEC.) [# OF UBD COMPUTATIONS] FOR SMP

N	$B_{\max}$	OA		BB-NLP		$n_{\max}$
30	100	8.9	[7.0]	633.6	[4216]	6
30	200	20.2	[14.3]	870.6	[6794]	7
30	300	69.9	[27.8]	1419.8	[12017]	7.75
40	100	101.8	[38.0]	>1hr	[N/A]	7
40	200	216.7	[37.3]	>1hr	[N/A]	7.67
40	300	934.9	[83.5]	>1hr	[N/A]	8.33

Regarding scalability of OA, bigger N leads to longer computation time in two aspects: first, it increases the number of decision variables and constraints, and second, it results in a larger total number of feasible candidates for a given $B_{\max}$. For the same value of $B_{\max}$, computation time for $N = 40$ is about ten times longer than for $N = 30$. It is also found that bigger $B_{\max}$ leads to longer computation time for given N; however, the total number of UBD computations does not increase as fast as the computation time in this case. This implies that the computation time grows mainly because unit computation time for solving each MILP increases rather than because upper bounds provided by the MILPs weaken. Note that the feasible set becomes larger as $B_{\max}$ increases; thus, each MILP has to consider a bigger branch-and-cut tree (CPLEX utilizes branch-and-cut algorithms for solving a MILP).

The optimal selection for larger $B_{\max}$ usually consists of more sensors than that for smaller communication budget – on average, 5 and 6.75 sensors for $B_{\max} = 100$ and 300 for both $N = 30$ and 40. On the other hand, Fig. 1 illustrates the case for which both the solutions for $B_{\max} = 200$ and 300 consist of 7 sensors, to effectively represent the trade-off between information and communication. The solid and dashdotted lines depict the actual and estimated trajectories of the target until $t = 20$ at which the one-step lookahead sensor management decision is made. The optimal solution for $B_{\max} = 300$ (blue diamonds) turns out to be the optimal solution for unconstrained MES with fixed cardinality of 7; thus, it is the best way choosing 7 sensors if an infinite amount of communication is allowed. Under the limitation of the communication budget $B_{\max} = 200$, the optimal solution (red squares) selects two nearby sensors instead of two sensors far from the other five.

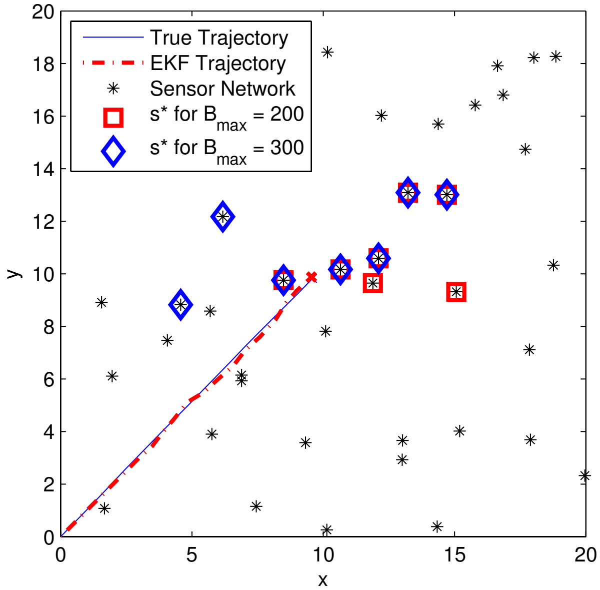

Fig. 1. An illustrative solution representing trade-off between information and communication ($N = 40$)

VI. Concluding Remarks

This work presented the outer-approximation approach to a generalized maximum entropy sampling problem. The mixed-integer semidefinite programming formulation was newly proposed; the outer-approximation algorithm resulting in a sequence of mixed-integer linear programs is presented. Numerical experiments verified that the performance of the suggested method is superior to the existing nonlinear programming-based branch-and-bound method especially in solving quadratically constrained problems such as communication-constrained sensor management. Future work will extend the presented outer-approximation algorithm to more general maximum information gain sampling. Also, other outer-approximation-based algorithms such as LP/NLP-based branch-and-bound [32] and branch-and-cut [31] can be adopted within the same MISDP framework.

Acknowledgment

This work is funded by NSF CNS-0540331 as part of the DDDAS program with Dr. Frederica Darema as the overall program manager. This work is based on the class project for "10.557 Mixed-integer Nonconvex Optimization," at MIT in SP'07 semester. The authors thank Dr. Jon Lee for answering the inquiries about his branch-and-bound algorithms, and Dr. Sven Leyffer for pointing out the references for other outer-approximation-based algorithms.

References

[1] A. Krause and C. Guestrin, "Near-Optimal Nonmyopic Value of Information in Graphical Methods," *21st Conference on Uncertainty in Artificial Intelligence*, Edinburgh, Jul. 2005.

[2] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Near-Optimal Sensor Placements: Maximizing Information while Minimizing Communication Cost," *5th International Conference on Information Processing in Sensor Networks*, Apr. 2006.

[3] C. Guestrin, A. Krause, and A. Singh, "Near-Optimal Sensor Placements in Gaussican Processes," *22nd International Conference on Machine Learning*, Bonn, Aug. 2005.

[4] J.L. Williams, J.W. Fisher III, and A.S. Willsky, "An Approximate Dynamic Programming Approach to a Communication Constrained Sensor Management," *8th Conference of Information Fusion*, 2005.

[5] Y. Zhang and Q. Ji, "Active and Dynamic Information Fusion for Facial Expression Understanding from Image Sequences," *IEEE Tran. on Pattern Analysis and Machine Intelligence*, Vol.27, No.5, 2005.

[6] H.-L. Choi, J.P. How, and J.A. Hansen, "Ensemble-Based Adaptive Targeting of Mobile Sensor Networks,"*American Control Conference*, July 2007.

[7] H.-L. Choi and J.P. How, "A Multi-UAV Targeting Algorithm for Ensemble Forecast Improvement,"*to appear in AIAA Guidance, Navigation, and Control Conference*, Aug. 2007.

[8] T.J. Mitchell, "An Algorithm for the Construction of 'D_k-Optimal' Experimental Designs," *Technometrics*, Vol.16, pp.203-210, 1974.

[9] M.C. Shewry and H.P. Wynn, "Maximum Entropy Sampling," *Journal of Applied Statistics*, Vol.46, pp.165-170, 1987.

[10] C.-W. Ko, J. Lee, and M. Queyranne, "An Exact Algorithm for Maximum Entropy Sampling," *Operations Research*, Vol.43, No.4, 1995.

[11] H. Wang, K. Yao, G. Pottie, and D. Estrin, "Entropy-Based Sensor Selection Heuristic for Target Localization," *3rd International Conference on Information Processing in Sensor Networks*, Apr. 2004.

[12] J. Lee, "Constrained Maximum-Entropy Sampling," *Operations Research*, Vol.46, No.5, 1998.

[13] M.C. Bueso, J.M. Angulo, and F.J. Alonso, "A State-Space Model Approach to Optimum Spatial Sampling Design Based on Entropy," *Environmental and Ecological Statistics*, Vol.5, pp.29-44, 1998.

[14] K.M. Anstreicher, M. Fampa, J. Lee, and J. Williams, "Maximum-Entropy Remote Sampling," *Discrete Applied Mathematics*, Vol.108, pp.211-226, 2001.

[15] K.M. Anstreicher, M. Fampa, J. Lee, and J. Williams, "Using Continuous Nonlinear Relaxations to Solve Constrained Maximum-Entropy Sampling Problems," *Mathematical Programming*, Vol.85, pp.221-240, 1999.

[16] A. Hoffman, J. Lee, and J. Williams, "New Upper Bounds for Maximum-Entropy Sampling," *MODA 6 – Advances in Model-Oriented Design and Analysis*, pp.143-153, Springer-Verlag, 2001.

[17] J. Lee and J. Williams, "A Linear Integer Programming Bound for Maximum-Entropy Sampling," *Mathematical Programming, Series B*, Vol.94, pp.247-256, 2003.

[18] S. Burer and J. Lee, "Solving Maximum-Entropy Samping Problems Using Factored Masks," *Mathematical Programming*, Vol.109, No.2-3, pp.263-281, 2007.

[19] M.A. Duran and I.E. Grossmann, "An Outer-Approximation Algorithm for a Class of Mixed-Integer Nonlinear Programs," *Mathematical Programming*, Vol.36, pp.307-339, 1986.

[20] R. Fletcher and S. Leyffer, "Solving Mixed Integer Nonlinear Programs by Outer Approximation," *Mathematical Programming*, Vol.66, pp.327-349, 1994.

[21] P. Kesavan and P.I. Barton, "Decomposition Algorithms for Nonconvex Mixed-Integer Nonlinear Programs," *AIChE Symposium Sereies*, Vol.96, No.323, pp.458-461, 1999.

[22] P. Kesavan, R.J. Allgor, E.P. Gatzke, and P.I. Barton, "Outer Approximation Algorithms for Separable Nonconvex Mixed-Integer Nonlinear Programs," *Mathematical Programming, Series A*, Vol.100, pp.517-535, 2004.

[23] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley Interscience, 1991.

[24] `http://www.ilog.com/products/cplex/`

[25] S. Boyd and L. Vandeberghe, *Convex Optimization*, Cambridge University Press, 2004.

[26] R.A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985.

[27] `http://tomopt.com/tomlab/products/cplex/`

[28] `http://tomopt.com/tomlab/products/barnlp/`

[29] D. Bertsekas, *Dynamic Programming and Optimal Control: Vol. I*, 2nd Edition, Athena Scientific, 2000.

[30] `http://tomopt.com/tomlab/products/knitro/`

[31] P. Bonami, T.B. Lorenz, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter, "An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs," *IBM Research Report RC23771(W0511-023)*, 2005.

[32] K. Abhishek, S. Leyffer, and J.T. Linderoth, "FilMINT: An Outer-Approximation-Based Solver for Nonlinear Mixed Integer Programs," *Argonne National Lab Report ANL/MCS-P1374-0306*, 2007.