Computationally Efficient Decision Making Under Uncertainty in High-Dimensional State Spaces

Dmitry Kopitkov and Vadim Indelman

***Abstract*— We develop a novel approach for decision making under uncertainty in high-dimensional state spaces, considering both active unfocused and focused inference, where in the latter case reducing the uncertainty of only a subset of variables is of interest. State of the art approaches typically first calculate the posterior information (or covariance) matrix, followed by its determinant calculation, and do so separately for each candidate action. In contrast, using the generalized matrix determinant lemma, we avoid calculating these posteriors and determinants of large matrices. Furthermore, as our key contribution we introduce the concept of calculation re-use, performing a one-time computation that depends on state dimensionality and system sparsity, after which evaluating the impact of each candidate action no longer depends on state dimensionality. Such a concept is derived for both active focused and unfocused inference, leading to general, non-myopic and exact approaches that are faster by orders of magnitude compared to the state of the art. We verify our approach experimentally in two scenarios, sensor deployment (focused and unfocused) and measurement selection in visual SLAM, and show its superiority over standard techniques.**

I. Introduction

Decision making under uncertainty is a fundamental problem in robotics and artificial intelligence, with applications including autonomous driving, surveillance, sensor deployment, object manipulation and active SLAM. The goal is to autonomously determine best actions according to a specified objective function, given the current belief about random variables of interest that could represent, for example, robot poses, tracked target or mapped environment, while accounting for different sources of uncertainty. Such a problem is an instantiation of partially observable Markov decision process (POMDP), while calculating an optimal solution of POMDP was proven to be computationally intractable [17].

Decision making, also sometimes referred to as active inference, can be formulated as selecting optimal action from a set of candidates, based on some cost function. In information-based decision making the cost function typically contains terms that evaluate the expected posterior uncertainty upon action execution, with commonly used costs including (conditional) entropy and mutual information. Thus, the corresponding calculations typically involve calculating a determinant of a posteriori covariance (information) matrices, and moreover, these calculations are to be performed for *each* candidate action.

D. Kopitkov is with the Technion Autonomous Systems Program (TASP), Technion - Israel Institute of Technology, Haifa 32000, Israel, dimkak@tx.technion.ac.il . V. Indelman is with the Department of Aerospace Engineering, Technion - Israel Institute of Technology, Haifa 32000, Israel, vadim.indelman@technion.ac.il. This work was supported by the Israel Science Foundation.

Decision making under uncertainty becomes an even more challenging problem when considering *high* dimensional state spaces. Such a setup is common in robotics, for example in the context of belief space planning in uncertain environments, active SLAM, sensor deployment, graph reduction and graph sparsification. In particular, calculating a determinant of information (covariance) matrix for an n-dimensional state is in general $O(n^3)$, and is smaller for sparse matrices as in SLAM problems [1].

Moreover, state of the art approaches typically perform these calculations from scratch for *each* candidate action. For example, active SLAM and belief space planning approaches first calculate the posterior belief within the planning horizon, and then use that belief to evaluate the objective function, which typically includes an information-theoretic term [11], [20], [29]. These approaches then determine the best action by performing the mentioned calculations for each action from a given set of candidate actions. Alternatively, belief space planning approaches that consider continuous (state, action and observation) spaces typically refine a nominal action into a locally-optimal one using dynamic programming or gradient descent, with both cases also involving propagating belief and calculating an information-theoretic term [16], [27], [28], [30].

Sensor deployment is another example of decision making in high dimensional state spaces. The basic formulation of the problem is to determine locations to deploy the sensors in a field such that some metric can be measured most accurately through the entire area (e.g. temperature). The problem can also be viewed as selecting optimal action from the set of candidate actions (available locations) and the objective function usually contains a term of uncertainty, like the entropy of a posterior system [21]. Also here, state of the art approaches evaluate a determinant over large posterior covariance (information) matrices for each candidate action, and do so from scratch [32], [33].

A similar situation also arises in measurement selection [3], [6] and graph pruning [2], [10], [24], [31] in the context of long-term autonomy in SLAM. In the former case, the main idea is to determine the most informative measurements (e.g. image features) given measurements provided by robot sensors, thereby discarding uninformative and redundant information. Such a process typically involves reasoning about mutual information, see e.g. [5], [6], for each candidate selection. Similarly, graph pruning and sparsification can be considered as instances of decision making in high dimensional state spaces [2], [10], with decision corresponding to determining what nodes to marginalize out [12], [22], and avoiding the resulting fill-in in information matrix by resorting to sparse approximations of the latter [2], [10], [24], [31]. Also here, existing approaches typically involve calculation of determinant of large matrices for each candidate action.

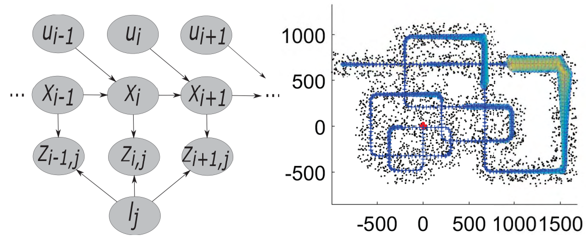
In this paper we develop a computationally efficient and exact approach for decision making in high-dimensional state spaces that addresses the aforementioned challenges. The *key idea* is to use the general matrix determinant lemma to calculate action impact with complexity *independent* of state dimensionality n, while *re-using* calculations between evaluating impact for different candidate actions. Our approach supports general observation and motion models, and nonmyopic planning, and is thus applicable to a wide range of applications such as those mentioned above, where fast decision making in high-dimensional state spaces is required.

Moreover, we show the proposed concept is applicable also to active *focused* inference. Unlike the *unfocused* case discussed thus far, active *focused* inference approaches aim to reduce the uncertainty over only a predefined set of the variables. The two problems can have significantly different optimal actions, with an optimal solution for the *unfocused* case potentially performing badly for the *focused* setup, and vice versa (see e.g. [23]). While the set of *focused* variables can be small, exact state of the art approaches calculate the marginal posterior covariance (information) matrix, for each action, which involves a computationally expensive Schur complement operation. For example, Mu et al. [25] calculate posterior covariance matrix per each measurement and then use the selection matrix in order to get marginal of *focused* set. Levine et al. [23] develop an approach that determines mutual information between *focused* and *unfocused* variables through message passing algorithms on Gaussian graphs but their approach is limited to only graphs with unique paths between the relevant variables.

In contrast, we provide a novel way to calculate posterior entropy of *focused* variables, which is fast, simple and general, yet, it does not require calculation of a posterior covariance matrix. In combination with our *re-use* algorithm, it provides *focused* decision making solver which is significantly faster (and exact) compared to standard approaches.

Finally, there is also a relation to the recently introduced concept of decision making in a conservative sparse information space [14], [15]. In particular, considering unary observation models (involving only one variable) and greedy decision making, it was shown that appropriately dropping all correlation terms and remaining only with a diagonal covariance (information) matrix does not sacrifice performance while significantly reducing computational complexity. While the approach presented herein confirms this concept for the case of unary observation models, our approach addresses a general non-myopic decision making problem, with arbitrary observation and motion models.

To summarize, our main contributions in this paper are as follows: (a) we develop an approach for a nonmyopic decision making in high-dimensional state spaces that uses the matrix determinant lemma to avoid calculating determinants of large matrices, with per-candidate complexity independent of state dimensionality; (b) we show calculations can be re-used when evaluating impact of different candidate actions; (c) we develop a corresponding approach also for active *focused* inference.



(a) (b)

Fig. 1: (a) Graphical model describing motion and observation models (2). For simplicity, the figure shows observations of only a single landmark l_j , indicating it is observed at time instances t_{i-1}, t_i and t_{i+1}; (b) Simulated trajectory of robot.

II. Notations and Problem Definition

Consider the joint probability distribution function (pdf) $\mathbb{P}(X_k|Z_{0:k}, u_{0:k-1})$ at time t_k over a high-dimensional problem-dependent state vector $X_k \in \mathbb{R}^n$. For example, in a SLAM problem X_k could represent robot poses and mapped landmarks, while in a sensor deployment problem X_k would represent an uncertainty field to be measured or monitored by adequately deploying sensors. Here, $Z_{0:k}$ and $u_{0:k-1}$ represent, respectively, all the observations and controls until time t_k.

The joint pdf can be written as

$$\mathbb{P}(X_k|Z_{0:k}, u_{0:k-1}) = \eta \mathbb{P}(x_0) \prod_{i=1}^{k} \mathbb{P}(x_i|x_{i-1}, u_{i-1}) \mathbb{P}(Z_i|X_i^o), \tag{1}$$

where η is a normalization constant and $\mathbb{P}(x_0)$ is a prior on the first pose. The motion and observation models $\mathbb{P}(x_i|x_{i-1}, u_{i-1})$ and $\mathbb{P}(Z_i|X_i^o)$ are defined by

$$x_{i+1} = f(x_i, u_i) + \omega_i \quad , \quad Z_i = h(X_i^o) + \upsilon_i. \tag{2}$$

In this paper we consider the process and measurement noise to be Gaussian and thus denote $\omega_i \sim \mathcal{N}(0, \Sigma_{\omega,i})$ and $\upsilon_{i,j} \sim \mathcal{N}(0, \Sigma_{\upsilon,i})$ with corresponding covariance matrices $\Sigma_{\omega,i}$ and $\Sigma_{\upsilon,i}$. While the latter could vary with time, to avoid clutter we will drop from here on the time notation.

We use the notation $X_i^o \subseteq X_i$ to indicate the involved subset of states in the observation function $h(\cdot)$. In particular, such a formulation can be used to describe a SLAM problem: at any given time step t_i, the robot may acquire multiple landmark observations, denoted by $Z_i = \{z_{i,1}, ..., z_{i,n_i}\}$, where $z_{i,j}$ is a single observation of the jth landmark and n_i is the number of such observations. In such a case, the measurement likelihood term in Eq. (1) becomes $\mathbb{P}(Z_i|X_i^o) = \prod_{j=1}^{n_i} \mathbb{P}(z_{i,j}|x_i, l_j)$. Figure 1a depicts the corresponding generative graphical model for this case, considering for simplicity only a single landmark l_j, while Figure 1b shows a simulated aerial visual SLAM scenario.

Following standard maximum a posteriori (MAP) inference, it is possible to efficiently infer, while exploiting sparsity and re-using calculations, the mean $\hat{X}_k$ and covariance Σ_k of the multivariate Gaussian $b[X_k] \doteq \mathbb{P}(X_k|Z_{0:k}, u_{0:k-1}) = \mathcal{N}(\hat{X}_k, \Sigma_k)$, see e.g. [19].

In the context of decision making under uncertainty, one can now reason how the pdf (1), or the *belief* $b[X_k]$, will evolve as a result of some candidate action. Considering a planning horizon of L look ahead steps and a sequence of actions $u_{k+1:k+L-1}$, the belief $b[X_{k+L}] \doteq \mathbb{P}(X_{k+L}|Z_{0:k+L}, u_{0:k+L-1})$ can be written as :

$$b[X_{k+L}] = \eta b[X_k] \prod_{l=k+1}^{k+L} \mathbb{P}(x_l|x_{l-1}, u_{l-1})\mathbb{P}(Z_l|X_l^o) \quad (3)$$

Note that one can go further and model, within the belief, whether a future observation will be actually obtained (see e.g. [16], [4]).

It is not difficult to show (see e.g. [16]) that the posterior information matrix of the belief $b[X_{k+L}]$ is given by:

$$\Lambda_{k+L} = \Lambda_k + \sum_{l=k+1}^{k+L} F_l^T \cdot \Sigma_{\omega,l}^{-1} \cdot F_l + \sum_{l=k+1}^{k+L} H_l^T \cdot \Sigma_{\upsilon,l}^{-1} \cdot H_l \quad (4)$$

where $F_l \doteq \bigtriangledown_x f$ and $H_l \doteq \bigtriangledown_x h$ are the Jacobian matrices of all the new factor terms in Eq. (3) (i.e. motion and observation factors).

Combining all the Jacobian matrices in Eq. (4) into matrix $\widetilde{A}$, and all the noise covariances into matrix Φ, yields

$$\Lambda_{k+L} = \Lambda_k + \widetilde{A}^T \cdot \Phi^{-1} \cdot \widetilde{A} = \Lambda_k + A^T \cdot A \quad (5)$$

where $A \doteq \Phi^{-\frac{1}{2}} \cdot \widetilde{A}$ is an $m \times n$ matrix that represents both Jacobians and noise covariances of all new factor terms in Eq. (3). The above equation can be considered as a single iteration of Gauss-Newton optimization and, similar to prior work [16], [20], [30], we assume it sufficiently captures the impact of action $u_{k:k+L-1}$. Under this assumption, the posterior information matrix Λ_{k+L} is independent of the (unknown) future observations $Z_{k+1:k+L}$ [16].

Each block row in A represents a single factor from the new terms in Eq. (3) and thus has a sparse structure. Only sub-blocks that correspond to the involved variables in the relevant factor are actually non-zero.

For notational convenience, we define the set of candidate actions by $\mathcal{A} = \{a_1, a_2, .., a_k\}$ with appropriate Jacobian matrices $\Phi_A = \{A_1, A_2, .., A_k\}$. While the planning horizon is not explicitly shown, each $a \in \mathcal{A}$ can represent a sequence of actions, e.g. $a = u_{k:k+L-1}$ for L look ahead steps.

In this paper we focus on information-theoretic decision-making and consider differential entropy $\mathcal{H}$ (further referred to just as entropy) as the cost function. Thus, we re-define the objective function as

$$J_{\mathcal{H}}(a) \doteq \mathcal{H}\left(b[X_{k+L}]\right), \quad (6)$$

where the belief $b[X_{k+L}]$ is a function of the controls $a = u_{k:k+L-1}$, see Eq. (3).

In particular, for Gaussian distributions, entropy is a function of the determinant of a posterior information (covariance) matrix, i.e. $\mathcal{H}\left(b[X_{k+L}]\right) \equiv \mathcal{H}\left(\Lambda_{k+L}\right)$ and can be expressed as

$$\mathcal{H}(b[X_{k+L}]) = \frac{n}{2} \cdot (1 + \ln(2\pi)) - \frac{1}{2} \ln \left|\Lambda_{k+L}\right|, \quad (7)$$

where $\Lambda_{k+L} = \Lambda_k + A^T A$, according to Eq. (5). Thus, evaluating $J_{\mathcal{H}}(a)$ requires determinant calculation of an $n \times n$ matrix, which is in general $O(n^3)$, per candidate action $a \in \mathcal{A}$. The optimal action is then given by $a^* = \arg\min_{a \in \mathcal{A}} J_{\mathcal{H}}(a)$.

Information gain (IG) is another common information-theoretic cost (e.g. [12], [25]) that we will use in this paper:

$$J_{IG}(a) \doteq \mathcal{H}(b[X_k]) - \mathcal{H}(b[X_{k+L}]). \quad (8)$$

The optimal action is defined for this cost as $a^* = \arg\max_{a \in \mathcal{A}} J_{IG}(a)$. Note that both objective functions (6) and (8) yield the same result, yet the latter will be computationally beneficial (see Section III).

Thus far, the exposition referred to active *unfocused* inference, where the action impact is calculated by considering all the random variables in the system, i.e. the entire state vector. However, as will be shown in the sequel, our approach is applicable also to active *focused* inference.

Active *focused* inference is another important problem, where in contrast to the former case, only a subset of variables is of interest (see, e.g., [21], [23], [25]). The complexity of such problem is much higher and proposed techniques succeeded to solve it in $O(kn^3)$ [21], [23] with k being size candidate set, and in $O(\tilde{n}^4)$ [25] with $\tilde{n}$ being size of the involved clique. Considering posterior entropy over the *focused* variables $X_{k+L}^F \subseteq X_{k+L}$ we can write:

$$J_{\mathcal{H}}^F(a) = \mathcal{H}(X_{k+L}^F) = \frac{n_F}{2} \cdot (1 + \ln(2\pi)) + \frac{1}{2} \ln \left|\Sigma_{k+L}^{M,F}\right|, \quad (9)$$

where n_F is the dimension of X_{k+L}^F, and $\Sigma_{k+L}^{M,F}$ is the posterior marginal covariance of X_{k+L}^F (suffix M for marginal), calculated by simply retrieving appropriate parts of posterior covariance matrix $\Sigma_{k+L} = \Lambda_{k+L}^{-1}$.

In the following section we develop a computationally efficient approach that addresses both active *unfocused* and *focused* inference. As will be seen, this approach naturally supports non-myopic planning with arbitrary motion and observation models, and it is in particular attractive for decision making in high-dimensional state spaces.

III. Approach

Our approach utilizes the well-known matrix determinant lemma [9] and re-use of calculations to significantly reduce computational complexity of the active inference problem as defined in Section II. In Section III-A we develop our approach for active *unfocused* inference, and then discuss in Section III-B how to re-use calculations when considering different candidate actions. Section III-C then extends the approach to active *focused* case.

A. Active Inference via Matrix Determinant Lemma

Information theoretic decision making involves evaluating the cost (7) or (8), an operation that requires calculating the determinant of a large $n \times n$ matrix (posterior information matrix), with n being the dimensionality of the state X. State of the art approaches typically perform these calculations from scratch for each candidate action.

In contrast, our approach contains a one-time calculation that will be reused afterwards to calculate impact of each candidate action (see Section III-B). As will be seen below, the latter depends only on the number of new factor terms in the Jacobian matrix $A \in \mathbb{R}^{m\times n}$, which is a function of the planning horizon L.

We first consider the IG as the utility function. It is not difficult to show that Eq. (8) can be written as $J_{IG}(a) = \frac{1}{2}\ln\frac{|\Lambda_k + A^T A|}{|\Lambda_k|}$. Using the generalized matrix determinant lemma [9], this equation can be written as

$$J_{IG}(a) = \frac{1}{2}\ln\left|I_m + A\cdot\Sigma_k\cdot A^T\right|, \quad \Sigma_k \equiv \Lambda_k^{-1} \tag{10}$$

as previously suggested in [12], [25] in the context of compact pose-SLAM and *focused* active inference.

Eq. (10) provides an exact and general solution for information-based decision making, where each action candidate can produce any number of new factors (nonmyopic planning) and where factors themselves can be of any motion or measurement model (unary, pairwise, etc.).

In many problem domains, such as SLAM, inference is typically performed in the information space and as such, the joint covariance matrix Σ_k is not readily available and needs to be calculated upon demand, which is expensive in general. While in first sight, it might seem the entire joint covariance matrix needs to be recovered, in practice this is not the case due to *sparsity* of the Jacobian matrix A.

Indeed, recalling Eq. (4), it is evident that A is a *sparse* $m\times n$ matrix, see Figure 2. More precisely, the only non-zero blocks in each factor will be of variables that are involved in this factor. For example, the Jacobian matrix that corresponds to a motion model factor $p(x_{k+l}|x_{k+l-1}, u_{k+l-1})$ will involve only two non-zero block entries for the state variables x_{k+l} and x_{k+l-1}. Factors for most of the measurement models like projection and range model, will also have only two non-zero blocks (see Figure 2).

Consequently, only specific entries from the covariance matrix Σ_k are really required, and sparse matrix techniques exist to calculate them efficiently [8], [18]. More formally, denote by C the set of all variables that are involved in A, i.e. these are the variables that are involved in at least one factor among the new factors generated due to the currently considered candidate action $a \equiv u_{k:k+L-1}$, see Eq. (3). Clearly, the columns of A that correspond to the rest of the variables, $X \setminus C$, are entirely filled with zeros (see Figure 2). Thus, Eq. (10) can be re-written as

$$J_{IG}(a) = \frac{1}{2}\ln\left|I_m + A_C\cdot\Sigma_k^{M,C}\cdot A_C^T\right| \tag{11}$$

where A_C is constructed from A by removing all zero columns, and $\Sigma_k^{M,C}$ is a prior joint marginal covariance of variables in C, which should be calculated from the (square root) information matrix Λ_k.

Intuitively, the posterior uncertainty that corresponds to action a is only a function of the prior marginal covariance over variables involved in A (i.e. $\Sigma_k^{M,C}$) and the new information introduced by the Jacobian A, with latter also involving the same variables C. Thus, uncertainty reduction in the posterior will be significant for large entries in A and high prior uncertainty over the variables C.

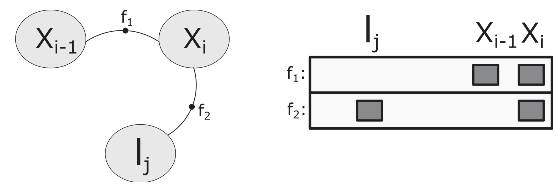


Fig. 2: Concept illustration of A's structure. Each column represents some variable from state vector. Each row represents some factor from Eq. (1). Factor f_1 of motion model that involves two poses x_i and x_{i-1} will have non-zero values only at columns of x_i and x_{i-1}. Factor f_2 of observation model that involves together variables x_i and l_j will have non-zero values only at columns of x_i and l_j.

In particular, in case of myopic decision making with unary observation models (that involve only a single state variable), calculation of $IG(a)$ for different candidate actions only requires recovering the diagonal entries of Σ_k, regardless of the actual correlations between the states, as recently shown in [14], [15]. However, while in the mentioned papers the per-action calculation takes $O(n)$, the $IG(a)$ calculation is $O(1)$ as will be shown in Section III-B.

In problems where inference is performed in the covariance form (e.g., in sensor deployment problems [21]), the covariance Σ_k is given. Thus, the calculation in Eq. (11) is bounded by calculating determinant of an $m \times m$ matrix which is in general $O(m^3)$, where m is the number of constraints due to new factors (for a given candidate action a). This calculation should be performed for each candidate action in the set $\mathcal{A}$. Furthermore, in many problems it is logical to assume that $m << n$, as m depends mostly on the planning horizon L, which is typically defined and constant, while n (state dimensionality) can be huge and grow with time in real systems (e.g. SLAM). Consequently, given the prior covariance our complexity for selecting best action is $O(|\mathcal{A}|)$, i.e. *independent* of state dimensionality n.

To conclude this section, we showed that calculation of impact for single candidate action does not depend on n. While this result is interesting by itself in the context of active inference, in the next section we go a step further and present an approach to calculate covariance entries, required by all candidates, with one-time calculation which can be re-used afterwards.

B. Re-use of Calculations

Calculating IG for different actions in $\mathcal{A}$ involves evaluating Eq. (11) for the corresponding Jacobian matrices from the set Φ_A. As mentioned in Section III-A, these calculations are computationally cheap *given* the covariance matrix as they do not depend on state dimensionality. However, retrieving covariance separately for each action is expensive and will most probably not suit real-time applications.

In this section we make the *key observation* that covariance calculation for all candidate actions can be united into one computational block, calculated at the beginning of decision making phase and then later be *re-used* upon calculating IG for each candidate action.

In more detail, recalling that only the joint covariance of involved variables is needed in order to calculate IG (11), we define the mutual set C_{All} as the state variables that are involved in at least one candidate action in $\mathcal{A}$. Before evaluating Eq. (11) for each candidate, we perform a *one-time calculation* to retrieve the joint covariance for variables C_{All}, i.e. $\Sigma_k^{M,C_{All}}$. Then Eq. (11) can be evaluated for each candidate action $a \in \mathcal{A}$ by simply retrieving appropriate blocks from $\Sigma_k^{M,C_{All}}$ in order to get $\Sigma_k^{M,C}$ for action a.

Intuitively, in most cases the candidates will have many mutual variables as they all are related to the current robot's location, in one way or another, making it even more reasonable to compute C_{All} in one calculation.

The complexity of this one-time calculation is different in different applications. When we use information filter, the system is represented by information matrix Λ_k, and in general the inverse of Schur complement of C_{All} variables should be calculated. Yet, there are techniques that use sparse matrix nature of SLAM in order to efficiently recover marginal covariances [18] or to keep and update them through the whole SLAM process [13].

In particular, in iSAM [19] the (linearized) system is represented by a squared root information matrix R_k, which is encoded, while exploiting sparsity, by the Bayes tree data structure. Decision making then can be performed by calculating, for each candidate action, the posterior matrix R_{k+L} (e.g. via Givens rotations [19] or another incremental factorization update method), and then calculating the determinant $|\Lambda_{k+L}| = \prod_{i=1}^{n} r_{ii}^2$, with r_{ii} being the ith entry on the diagonal of R_{k+L}. Yet, calculating R_{k+L} for each action can be expensive, particularly in loop closures, and requires copy/clone of the original matrix R_k. In contrast, per candidate calculation in Eq. (11) is constant in general.

C. Extension to Active Focused Inference

In this section we present a novel approach to calculate entropy of a *focused* set of variables, and then combine it with the ideas from the previous sections (generalized matrix determinant lemma, IG cost function and calculation re-use) to develop a computationally efficient algorithm for *focused* information-based decision making.

First we recall definitions from Section II and introduce additional notations: $X_k^F \in \mathbb{R}^{n_F}$ denotes the set of *focused* variables, $X_k^R \doteq X_k / X_k^F \in \mathbb{R}^{n_R}$ is a set of the remaining variables, with $n = n_F + n_R$. The $n_F \times n_F$ marginal covariance and information matrices of X_k^F are denoted, respectively, by $\Sigma_k^{M,F}$ (suffix M for marginal) and $\Lambda_k^{M,F} \equiv (\Sigma_k^{M,F})^{-1}$. Furthermore, we partition the joint information matrix Λ_k as

$$\Sigma_k = \begin{bmatrix} \Sigma_k^{M,R} & \Sigma_k^{M,RF} \\ (\Sigma_k^{M,RF})^T & \Sigma_k^{M,F} \end{bmatrix}, \; \Lambda_k = \begin{bmatrix} \Lambda_k^{R} & \Lambda_k^{R,F} \\ (\Lambda_k^{R,F})^T & \Lambda_k^{F} \end{bmatrix}. \tag{12}$$

where $\Lambda_k^F \in \mathbb{R}^{n_F \times n_F}$ is constructed by retrieving from Λ_k only the rows and the columns related to F (it is actually conditional information matrix of X_k^F, conditioned on rest of variables X_k^R), $\Lambda_k^R \in \mathbb{R}^{n_R \times n_R}$ is defined similarly for X_k^R, and $\Lambda_k^{R,F} \in \mathbb{R}^{n_R \times n_F}$ contains remaining blocks of Λ_k as shown in Eq. (12).

The marginal information matrix of X_k^F, i.e. $\Lambda_k^{M,F}$, can be calculated via Schur complement $\Lambda_k^{M,F} = \Lambda_k^F - (\Lambda_k^{RF})^T \cdot (\Lambda_k^R)^{-1} \cdot \Lambda_k^{RF}$. However, one of Schur complement's properties [26] is $|\Lambda_k| = |\Lambda_k^{M,F}| \cdot |\Lambda_k^R|$, from which we can conclude that

$$|\Lambda_k^{M,F}| = \frac{1}{|\Sigma_k^{M,F}|} = \frac{|\Lambda_k|}{|\Lambda_k^R|}. \tag{13}$$

Therefore, the posterior entropy of X_{k+L}^F (see Eq. (9)) is a function of the posterior Λ_{k+L} and its partition Λ_{k+L}^R:

$$J_{\mathcal{H}}^F(a) = \mathcal{H}(X_{k+L}^F) = \frac{n_F}{2} \cdot (1 + \ln(2\pi)) - \frac{1}{2} \ln \frac{|\Lambda_{k+L}|}{|\Lambda_{k+L}^R|}. \tag{14}$$

From Eq. (5) one can observe that $\Lambda_{k+L}^R = \Lambda_k^R + A_R^T A_R$, where $A_R \in \mathbb{R}^{m \times n_R}$ is constructed from Jacobian A by taking only the columns that are related to variables in X_k^R.

The next step is to use IG instead of entropy, with the same motivation and benefits as in the *unfocused* case (Sections III-A and III-B). The optimal action $a^* = \arg\max_{a \in \mathcal{A}} J_{IG}^F(a)$ will maximize $J_{IG}^F(a) = \mathcal{H}(X_k^F) - \mathcal{H}(X_{k+L}^F)$, and using the generalized matrix determinant lemma we can write:

$$J_{IG}^F(a) = \frac{1}{2} \ln \frac{|I_m + A \cdot \Sigma_k \cdot A^T|}{|I_m + A_R \cdot \Sigma_k^{R|F} \cdot A_R^T|}, \tag{15}$$

where $\Sigma_k^{R|F} \in \mathbb{R}^{n_R \times n_R}$ is a prior covariance matrix of X_k^R conditioned on X_k^F, and it is actually the inverse of Λ_k^R.

We can see that the *focused* and *unfocused* information gains have actually simple relation between them $J_{IG}^F(a) = J_{IG}(a) - \frac{1}{2} \ln |I_m + A_R \cdot \Sigma_k^{R|F} \cdot A_R^T|$. The second term is negative and reduces the action's impact on posterior entropy of X_{k+L}^F.

As we saw in Eq. (11), sparsity of A provides easier calculations through $A \cdot \Sigma_k \cdot A^T = A_C \cdot \Sigma_k^{M,C} \cdot A_C^T$, where C is set of involved variables. The same idea exactly can be applied for the term $A_R \cdot \Sigma_k^{R|F} \cdot A_R^T$. Also here we will end up with the necessity of calculating only those entries of $\Sigma_k^{R|F}$ that correspond to the involved variables from subset X_k^R, which can lead to additional time complexity reduction.

The idea of calculation re-use is also applicable in *focused* case in exactly the same way. The one-time calculation here will contain calculation of $\Sigma_k^{M,C_{All}}$ from *unfocused* case and calculation of entries from $\Sigma_k^{R|F}$ that are related to variables in subset union $X_k^R \cup C_{All}$. Then the per-action calculation of *focused* IG (15) will contain calculation of two determinants of $m \times m$ matrices, which is almost constant as was explained in *unfocused* case.

To conclude, the presented technique solves *focused* decision making in exact way, and by applying calculation re-use, the evaluation of each candidate action does not depend anymore on dimensionality of the state but only on the planning horizon L.

D. Application to different problem domains

In this section we briefly discuss various problem domains of decision making and show how our approach can be applied for each case.

Sensor Deployment: We formulate the problem as follows. Let $X \in \mathbb{R}^n$ denote some metric at n locations (e.g. temperature), and assume its prior distribution is a known multivariate Gaussian, $X \sim N(\mu, \Sigma) = N^{-1}(\eta, \Lambda)$. In its most basic form, the objective is to find the optimal locations set $S \subseteq X$ for deploying k sensors, i.e. $S^\star = \arg\min_{S \subseteq X} \mathcal{H}(U|S)$, s.t. $|S| = k$, where $U \doteq X$ for *unfocused* case, and $U \subset X$ in the *focused* case.

Such a problem cannot be efficiently handled due to exponential number of candidates, and usually the greedy approach is applied, where sequence of sub-decisions is made instead, selecting at each the optimal subset S', s.t. $|S| = k'$ with k' being small enough to be solvable. Given an observation model of sensor at each location x_i as $z_i = h(x_i) + \nu_i$ with Gaussian white noise, one can see that after deploying sensors at $S' = \{x_1, .., x_{k'}\}$ locations, the posterior information matrix is $\Lambda^+ = \Lambda + A_{S'}^T \cdot A_{S'}$, where $A_{S'}$ is a Jacobian matrix of measurements at S'. Applying our approach and in particular, re-using calculations, each decision turns to have an already familiar form:

$$S'^* = \underset{S' \subseteq X, |S'| = k'}{\arg\max} \; J_{IG}(S') = \frac{1}{2} \ln \left| I_m + A_{S'} \cdot \Sigma \cdot A_{S'}^T \right| \quad (16)$$

where Σ is calculated only once for all candidates by inverting matrix Λ.

Measurement Selection: Due to desired sparsity of the system only the informative factors should be added (or on the opposite, uninformative factors should be removed). To reasonably decide which factor to add/remove the informativeness and change in sparsity of each one of them need to be computed. The former can be computed through Eq. (11), where C will be the set of variables involved in the measurement model, and A_C will contain only the relevant Jacobians of C (see Section III-A), while the latter can be computed, e.g., by counting the number of non-zero entries introduced into Λ by including the measurement.

Graph Reduction: Additionally, sometimes it is vital to reduce number of system variables (e.g. for long-term SLAM). In such cases, having set of candidate nodes $X = \{x_1, .., x_n\}$, it would be logical to remove the most uninformative nodes - the ones that marginalizing them out would leave the system with the smallest entropy $\mathcal{H}(X/x_i)$. Using Eq. (14) and assuming that all nodes have same dimension, it is possible to show that $\mathcal{H}(X/x_i)$ will be proportional to $\ln \left| \Lambda^{x_i} \right|$, where Λ^{x_i} is a partition of the information matrix Λ related to variable x_i. Calculating log-determinant of every Λ^{x_i} is relatively light calculation, linear in n, and will allow to intelligently decide which node to expel.

IV. Results

In this section we present simulation results of applying our approach to sensor deployment (both *unfocused* and *focused* cases), and to measurement selection problems. In the former case, each candidate action represents possible locations for sensor deployment, with each candidate location corresponding to a unary factor. We consider a nonmyopic setting and let each candidate action represent 2 sensor locations. In the measurement selection problem, we consider greedy decision making in the context of visual SLAM with pairwise factors. The code is implemented in Matlab; for measurement selection we use the GTSAM library [7], [19]. All scenarios were executed on a Linux machine with i7 2.40 GHz processor and 32 Gb of memory.

A. Sensor Deployment (unfocused and focused)

In this section we apply our approach to the sensor deployment problem, considering both *focused* and *unfocused* instantiations of this problem (see Section III-D). The prior of sensor field is represented by information matrix Λ and it is dense as usual in problem of sensor deployment.

We compare our re-use approach against the Standard incremental technique where first posterior squared-root matrix R^+ is calculated through Givens rotations, and then the posterior entropy is computed from its diagonal values (see Section III-B).

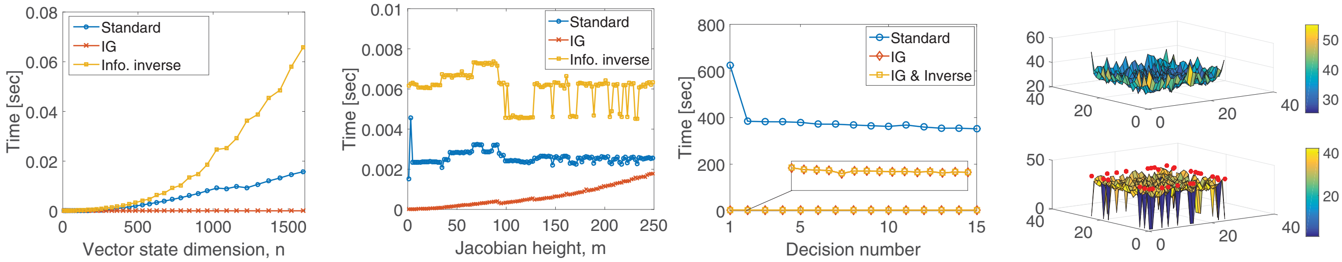
While decision making involves evaluating action impact for all candidate actions $\mathcal{A}$, we first analyze action impact calculation ($J_{IG}(a)$) for a single candidate $a \in \mathcal{A}$, comparing our approach to the Standard approach for the *unfocused* case. Figure 3 shows these timing results as a function of state dimension n (Figure 3a) and as function of Jacobian A's height m (Figure 3b). As expected, n effects running time of both the Standard technique and calculation of Σ_k (inverse of Λ_k which is dense in case of sensor deployment), while m only effects calculation of IG (red line).

One might think, based on Figures 3a-3b, that the proposed approach is slower than Standard alternatives because of the time needed for inverse calculation to get Σ_k. Yet, it is exactly here that our calculation re-use paradigm comes into play (see Section III-B): this calculation is performed only *once* for all candidate actions $\mathcal{A}$, while, given Σ_k, calculating IG for each action is no longer a function of n.

The substantial reduction in running time of our approach, compared to the Standard approach, can be clearly seen in Figure 3c, which considers the entire decision making problem, i.e. evaluation of all candidate actions $\mathcal{A}$. The figure shows running time for sequential decision making, where at each time instant we choose the best locations of 2 sensors, with around $|\mathcal{A}| = 10^5$ candidate actions. The number of all sensor locations is $n = 625$ in this example. Overall, 15 sequential decisions were made. As seen, decision making using our approach requires only about 3 seconds, while the the Standard approach requires about 400 seconds.

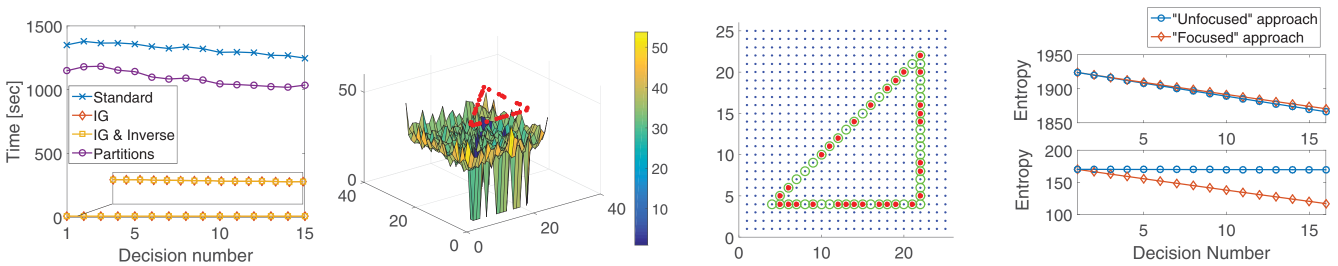
We now consider the *focused* version of the sensor deployment problem (Eq. 9). In other words, the goal is to find sensor locations that maximally reduce uncertainty about chosen *focused* variables X^F. We have 54 such variables, which are shown in Figure 4c, while the rest of the problem setup remains identical to the *unfocused* case.

In Figure 4 we show the corresponding results of our approach, compared to the Standard approach. The latter first calculates, for each candidate action, the posterior $\Lambda^+ = \Lambda + A^T A$, followed by calculation of Schur complement $\Lambda^{M,F}$ of the *focused* set X^F, and its determinant $\left|\Lambda^{M,F}\right|$ in order to get $J_{\mathcal{H}}^F(a)$ (Eq. 9). We also compare to an



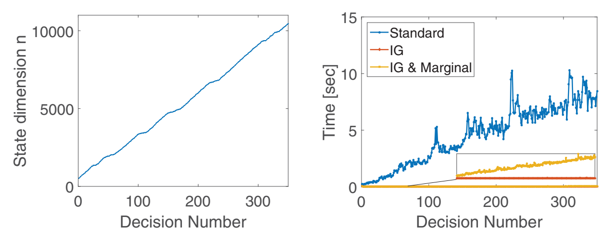
(a) (b) (c) (d)

Fig. 3: *Unfocused* sensor deployment scenario. Running time for calculating impact of a single action as a function of state dimension n (a) and as a function of Jacobian A's height m (b). In (a), $m = 2$, while in (b) $n = 625$. (c) Running time for sequential decision making, i.e. evaluating impact of all candidate actions, each representing candidate locations of 2 sensors. (d) prior and final uncertainty of the field, with red dots marking selected locations.



(a) (b) (c) (d)

Fig. 4: *Focused* sensor deployment scenario, (a) overall time it took to make decision with different approaches, (b) final uncertainty of the field, with red dots marking selected locations, (c) *focused* set of variables (green circles) and locations selected by algorithm (red dots), (d) overall system entropy (above) and entropy of *focused* set (bottom) after each decision, with blue line representing *unfocused* algorithm, and red line - *focused* algorithm.



(a) (b)

Fig. 5: Measurement selection scenario, (a) state's dimension n per decision, (b) overall time it took to evaluate impacts of pose's all measurements, with different approaches.

additional approach, termed "Partitions", which uses Givens rotations to compute R^+ and instead of performing Schur complement, calculates the posterior entropy of the *focused* set via Eq. (14). This equation is one of our main contributions, being an essential step in the derivation of our approach, and we show here that comparing to Standard technique, the Partitions approach is considerably faster. Our *focused* approach applies the matrix determinant lemma, transforming Eq. (14) to Eq. (15), which, together with the *re-use* concept (Section III-B), makes it possible to drastically reduce running time as shown in Figure 4a (9 seconds versus about 1000 in Partitions and 1400 in Standard approach).

B. Measurement Selection in SLAM

In this section we consider a measurement selection problem (see Section III-D) within a visual aerial SLAM framework, where one has to choose the most informative image feature observations from the numerous image features typically calculated for each incoming new image. We demonstrate application of our approach in this problem, which in contrast to sensor selection problem, involves pairwise factors of the type $p(z_{i,j}|x_i, l_j)$, relating between an image observation $z_{i,j}$, camera pose x_i and landmark l_j.

A top view of the considered aerial scenario is shown in Figure 1b: an aerial vehicle performs visual SLAM, mapping the environment and at the same time localizing itself. The figure shows the landmarks and the estimated trajectory, along with the uncertainty covariance for each time instant. One can clearly see the impact of loop closure observations on the latter. In the considered scenario there are about 25000 landmarks and roughly 500 image features in each view.

The number of image features that correspond to previously seen landmarks is relatively small (30-50), which corresponds to a much smaller set of actions $\mathcal{A}$ compared to the sensor deployment problem (Section IV-A) where the cardinality of $\mathcal{A}$ was huge (10^5). Such a dataset was chosen on purpose in order to show the behavior of the proposed algorithm in domains with small number of candidates. Also, in this scenario the actions are myopic since the measurements are greedily selected.

Additionally, as opposed to sensor deployment problem, in the current problem, state dimensionality n grows with time as more poses and landmarks are added into inference (see Figure 5a) and the information matrix is sparse.

Figure 5b shows the timing results for choosing 10 most informative image observations comparing the proposed approach to the Standard approach (computing posterior square root information matrix using iSAM, and then calculating determinant, see Section III-B). This decision making problem is solved sequentially, each time a new image is acquired. As seen, our approach is substantially faster than the Standard approach, while providing identical results (the same decisions). In particular, running time of the Standard approach for the last time index with $n = 10000$ state dimensionality, is around 7 seconds. In contrast, our approach takes about 0.05 seconds: calculation time of action impacts via calculation re-use is negligible (red line), while the one-time calculation of marginal covariance $\Sigma_k^{M,C}$ (yellow line) is efficiently performed, in the current implementation, via sparse factorization techniques using GTSAM [7], [19].

V. Conclusions

We developed a novel non-myopic and exact approach for information theoretic decision making in high dimensional state spaces, considering both *unfocused* and *focused* active inference problems. The key idea is to use the generalized matrix determinant lemma and re-use of calculations to efficiently evaluate the impact of each candidate action on posterior entropy. Our approach drastically reduces running time compared to the state of the art, especially when set of candidate actions is large, with running time being independent of state dimensionality. The approach was examined in two problems, sensor deployment and measurement selection in visual SLAM, exhibiting in each superior performance compared to the state of the art, and reducing running time by several orders of magnitude (e.g. 3 versus 400 seconds in sensor deployment). Possible directions for future research include experiments with real data and extension to belief space planning.

References

[1] Zhaojun Bai, Gark Fahey, and Gene Golub. Some large-scale matrix computation problems. *Journal of Computational and Applied Mathematics*, 74(1):71–89, 1996.

[2] Nicholas Carlevaris-Bianco, Michael Kaess, and Ryan M Eustice. Generic node removal for factor-graph slam. *IEEE Trans. Robotics*, 30(6):1371–1385, 2014.

[3] L. Carlone, A. Censi, and F. Dellaert. Selecting good measurements via l1 relaxation: A convex approach for robust estimation over graphs. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 2667–2674, 2014.

[4] Stephen M Chaves, Jeffrey M Walls, Enric Galceran, and Ryan M Eustice. Risk aversion in belief-space planning under measurement acquisition uncertainty. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2015.

[5] M. Chli and A. J. Davison. Active matching for visual tracking. *Robotics and Autonomous Systems*, 57(12):1173–1187, December 2009.

[6] A.J. Davison. Active search for real-time vision. In *Intl. Conf. on Computer Vision (ICCV)*, Oct 2005.

[7] F. Dellaert. Factor graphs and GTSAM: A hands-on introduction. Technical Report GT-RIM-CP&R-2012-002, Georgia Institute of Technology, September 2012.

[8] G.H. Golub and R.J. Plemmons. Large-scale geodetic least-squares adjustment by dissection and orthogonal decomposition. *Linear Algebra and Its Applications*, 34:3–28, Dec 1980.

[9] David A Harville. Matrix algebra from a statistician's perspective. *Technometrics*, 40(2):164–164, 1998.

[10] G. Huang, M. Kaess, and J.J. Leonard. Consistent sparsification for graph optimization. In *Proc. of the European Conference on Mobile Robots (ECMR)*, pages 150 – 157, 2012.

[11] S. Huang, N. Kwok, G. Dissanayake, Q. Ha, and G. Fang. Multi-step look-ahead trajectory planning in SLAM: Possibility and necessity. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1091–1096, 2005.

[12] V. Ila, J. M. Porta, and J. Andrade-Cetto. Information-based compact Pose SLAM. *IEEE Trans. Robotics*, 26(1), 2010. In press.

[13] Viorela Ila, Lukas Polok, Marek Solony, Pavel Smrz, and Pavel Zemcik. Fast covariance recovery in incremental nonlinear least square solvers. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 4636–4643. IEEE, 2015.

[14] V. Indelman. Towards information-theoretic decision making in a conservative information space. In *American Control Conference*, pages 2420–2426, July 2015.

[15] V. Indelman. No correlations involved: Decision making under uncertainty in a conservative sparse information space. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):407–414, 2016.

[16] V. Indelman, L. Carlone, and F. Dellaert. Planning in the continuous domain: a generalized belief space approach for autonomous navigation in unknown environments. *Intl. J. of Robotics Research*, 34(7):849–882, 2015.

[17] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.

[18] M. Kaess and F. Dellaert. Covariance recovery from a square root information matrix for data association. *Robotics and Autonomous Systems*, 57(12):1198–1210, 2009.

[19] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Intl. J. of Robotics Research*, 31:217–236, Feb 2012.

[20] A. Kim and R. M. Eustice. Active visual slam for robotic area coverage: Theory and experiment. *Intl. J. of Robotics Research*, 2014.

[21] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *J. of Machine Learning Research*, 9:235–284, 2008.

[22] H. Kretzschmar and C. Stachniss. Information-theoretic compression of pose graphs for laser-based slam. *Intl. J. of Robotics Research*, 31(11):1219–1230, 2012.

[23] D. Levine and J. P. How. Sensor selection in high-dimensional gaussian trees with nuisances. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2211–2219, 2013.

[24] Mladen Mazuran, Gian Diego Tipaldi, Luciano Spinello, and Wolfram Burgard. Nonlinear graph sparsification for slam. In *Robotics: Science and Systems (RSS)*, pages 1–8, 2014.

[25] Beipeng Mu, Ali-akbar Agha-mohammadi, Liam Paull, Mathew Graham, Jonathan How, and John Leonard. Two-stage focused inference for resource-constrained collision-free navigation. In *Robotics: Science and Systems (RSS)*, 2015.

[26] Diane Valerie Ouellette. Schur complements and statistics. *Linear Algebra and its Applications*, 36:187–295, 1981.

[27] S. Patil, G. Kahn, M. Laskey, J. Schulman, K. Goldberg, and P. Abbeel. Scaling up gaussian belief space planning through covariance-free trajectory optimization and automatic differentiation. In *Intl. Workshop on the Algorithmic Foundations of Robotics*, 2014.

[28] R. Platt, R. Tedrake, L.P. Kaelbling, and T. Lozano-Pérez. Belief space planning assuming maximum likelihood observations. In *Robotics: Science and Systems (RSS)*, pages 587–593, Zaragoza, Spain, 2010.

[29] R. Valencia, M. Morta, J. Andrade-Cetto, and J.M. Porta. Planning reliable paths with pose SLAM. *IEEE Trans. Robotics*, 2013.

[30] J. Van Den Berg, S. Patil, and R. Alterovitz. Motion planning under uncertainty using iterative local optimization in belief space. *Intl. J. of Robotics Research*, 31(11):1263–1278, 2012.

[31] J. Vial, H. Durrant-Whyte, and T. Bailey. Conservative sparsification for efficient and consistent approximate estimation. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 886–893. IEEE, 2011.

[32] Zhengyuan Zhu and Michael L Stein. Spatial sampling design for prediction with estimated parameters. *Journal of Agricultural, Biological, and Environmental Statistics*, 11(1):24–44, 2006.

[33] Dale L Zimmerman. Optimal network design for spatial prediction, covariance parameter estimation, and empirical prediction. *Environmetrics*, 17(6):635–652, 2006.