This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

Minimum Cost Data Aggregation with Localized Processing for Statistical Inference

Animashree Anandkumar*, Lang Tong*, Ananthram Swami† and Anthony Ephremides‡

*ECE Dept.,Cornell University, Ithaca, NY 14853, Email: {aa332@,ltong@ece}.cornell.edu
†Army Research Laboratory, Adelphi, MD 20783, Email: a.swami@ieee.org
‡ EE Dept., University of Maryland College Park, MD 20742, Email: tony@eng.umd.edu

***Abstract*—The problem of minimum cost in-network fusion of measurements, collected from distributed sensors via multihop routing is considered. A designated fusion center performs an optimal statistical-inference test on the correlated measurements, drawn from a Markov random field. Conditioned on the delivery of a sufficient statistic for inference to the fusion center, the structure of optimal routing and fusion is shown to be a Steiner tree on a transformed graph. This Steiner-tree reduction preserves the approximation ratio, which implies that any Steiner-tree approximation can be employed for minimum cost fusion with the same approximation ratio. The proposed fusion scheme involves routing packets of two types viz., raw measurements sent for local processing, and aggregates obtained on combining these processed values. The performance of heuristics for minimum cost fusion are evaluated through theory and simulations, showing a significant saving in routing costs, when compared to routing all the raw measurements to the fusion center.**

Index Terms— **Sensor networks, in-network processing and aggregation, statistical inference, cost minimization**

I. Introduction

Classical routing in a general data network aims at delivering data from source(s) to destination in some optimal manner, for example, by minimizing the total routing cost. Traditionally, the content of a data packet is unchanged en-route to the destination; in general, packets from different sources are not combined at intermediate nodes.

A sensor network deployed for specific applications, however, may not require all the raw data at the destination; some form of summary statistic may suffice. For example, to detect the occurrence of certain events, only a *sufficient statistic* of the data, based on a statistical model, is needed at the decision node (the so-called *fusion center*). Importantly, a sufficient statistic does not destroy any information about the underlying phenomenon and often enables a significant reduction of the data dimension. Therefore, the classical approach of routing all the raw data to the fusion center is inefficient. Instead, a data-centric approach of delivering only a sufficient statistic to the fusion center could be an economic alternative.

The maximum reduction of data dimension through the sufficient statistic is when the sensor measurements are statistically independent (conditioned on the specific physical phenomenon). In this extreme scenario, the sufficient statistic (likelihood function) is a sum function over components involving individual node values. Such a sum function can be obtained by aggregating the partial sums along a tree. The other extreme, of course, is when the sufficient statistic does not permit any dimension reduction and hence, all the measurements are needed at the fusion center. What then about the cases between these extremes?

We examine optimal in-network processing strategies for multihop sensor networks, when the sensor measurements are drawn from a structured statistical model. In any realistic scenario, the sensor measurements are spatially correlated, and our framework takes this into account. Specifically, we assume that the measurements are drawn from a *Markov random field* (MRF), (a detailed description of the model is given in section III-B). How can such spatial dependence be exploited for minimum cost fusion? Can the minimum cost fusion be reduced to a known optimization problem to enable the characterization of its complexity? Are there approximations that are simple and yet have guaranteed performance? How much saving can one expect over the conventional approach of forwarding all the raw data to the fusion center?

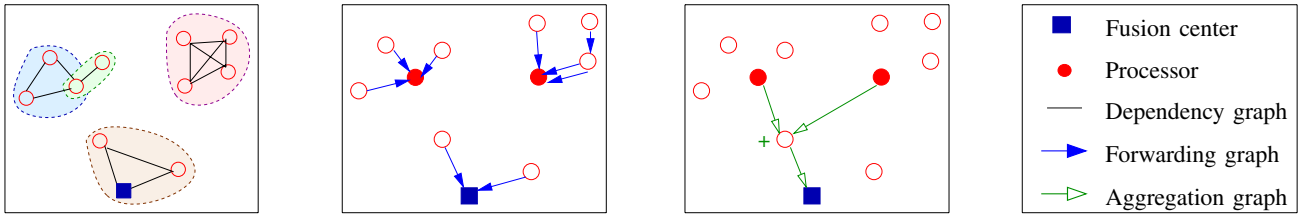
By optimal data fusion we mean the following. First, the (minimal) sufficient statistic is delivered to the fusion center and optimal statistical inference is undertaken at the fusion center. Second, the cost (e.g. energy) of delivering the sufficient statistic through multihop routing is minimized. Departing from the classical-routing paradigm, we allow raw data at individual sensors to be aggregated into some economic form as they propagate through the network, contributing to the sufficient statistic at the fusion center. The idea of combining data packets at routers to form new economic representations is of course not new (see a brief review below); however, doing so while guaranteeing optimality at the destination is nontrivial and so far an open problem.

II. Related work

An overview of routing for mobile-wireless networks can be found [1], [2].Correlated data gathering has been considered

This research was conducted using the resources of the Cornell University Center for Advanced Computing, which receives funding from Cornell University, New York State, the National Science Foundation, and other leading public agencies, foundations, and corporations. This work was supported in part through the collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011 and by the National Science Foundation under Contract CNS-0435190. The third author was partially supported by the DARPA ITMANET program. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

978-1-4244-2026-1/08/$25.00 © 2008 IEEE



(a) Maximal cliques of dependency graph (b) Forwarding subgraph to compute clique potentials (c) Aggregation subgraph to add the potentials (d) Legend

Fig. 1. Schematic of dependency graph of Markov random field and stages of data aggregation. The cliques of Markov random field arise due to spatial dependence of data. The set of all links used for aggregation is known as the packet-operation digraph. Its forwarding and aggregation subgraphs consist of links transporting raw data and aggregated values. The delivery of likelihood function to the fusion center needs to be ensured.

in [3]–[5]. But these schemes focus on compression, with the aim of routing all the measurements to a designated sink. Efficient aggregation schemes have been studied in [6]–[10], but without taking into account the spatial correlation among the measurements. For example, in [9], it is assumed that multiple incoming packets at a node can be processed to a single outgoing packet; this holds only for some special functions such as sum, maximum etc. A survey of in-network processing of various functions may be found in [11], [12].

The model for spatially-correlated data crucially affects the in-network processing schemes. However, since only few real systems have been deployed, varying assumptions have been made in the literature. Joint-Gaussian distributions and distance-based correlation function have been widely assumed due to their simplicity [13]–[16]. The model proposed in [17] is a special case of a Markov random field (MRF). The use of the MRF model for spatial data in sensor networks is relatively new (e.g., [18]), although it is widely used in image processing [19] and geo-statistics [20]. This could be due to the complexity of the model for arbitrarily-placed nodes.

In this paper, we employ the Markov random field model, taking into account only its graphical dependency structure and no parametric function is assumed for spatial correlation. The use of a Markov random field model leads to the formation of "clusters" that are based on the statistical dependence, rather than other considerations such as residual energy [10]. Also, these clusters in general contain common nodes and there is the issue of aggregation of the processed values rather than simple forwarding to the destination. The dependency structure and model parameters of the Markov random field model can be estimated by incorporating a training phase. Recently, learning graphical models from data, specifically for binary hypothesis testing, has been considered in [21].

Aggregation for inference in resource-constrained sensor networks are fewer. In [22], sensor collaboration issue in target tracking is addressed. In [23], the local vote decision fusion rule fuses binary decisions locally by a majority rule before transmitting to a fusion center. *Chernoff routing*, with a link measure for detection, has been proposed in [24] and assumes an one-dimensional Gauss-Markov random process, not applicable when the nodes are on a plane. In [25], a dynamic-programming approach to resource management for object tracking, based on a graphical model, is proposed. However, the possibility of aggregation, en-route, is not considered. In [26], a decision-theoretic approach to inference with single-bit communication is considered and the network topology is predefined by a directed acyclic graph. In [27], we analyzed the optimal sensor density in an energy-constrained random network, and measurements are i.i.d. Gaussian under the null hypothesis and under the alternative, form a Gauss-Markov random field with nearest-neighbor dependency.

A. Our Approach and Contributions

The main contribution of this work is twofold. First, conditioned on the requirement that the sufficient statistic for statistical inference (i.e., the likelihood function) is delivered to the fusion center, we obtain the minimum cost routing and aggregation scheme, when sensor measurements are drawn from a Markov random field. Such a scheme involves computing the likelihood function consisting of components, each of which depends on a subset of the measurements. See Fig.1. These components can be computed independently at various nodes. Therefore, an aggregation scheme involves the following considerations, viz., each component is assigned a computation site or a processor; measurements of the component members are then transported to its processor to enable computation of the component values. These values are then combined and delivered to the fusion center.

We show that the Steiner tree on an expanded communication graph minimizes the sum costs of routing for the above tasks. The specific Steiner-tree reduction preserves the *approximation factor*. The approximation factor ρ of polynomial-time algorithm guarantees that its performance is no worse than ρ times the optimal value. Hence, our approximation-factor preserving reduction implies that any Steiner-tree approximation algorithm can be used for the problem of optimal fusion with the same approximation ratio. The expansion of the communication graph involves adding component-representative nodes, as selectors of the processors for each component of the likelihood function and connecting them to the component members through edges incorporating the local routing costs.

In contrast to the Steiner-tree approach, we propose a simpler heuristic based on the minimum spanning tree (MST)

that ensures optimal inference at the fusion center and has an approximation ratio of two for the special case of the nearest-neighbor dependency graph. Our simulations show that in-network processing achieves significant savings compared to forwarding all the raw data to the fusion center, especially for sparse spatial dependencies.

The substantial reduction in the routing costs comes from the exploitation of the Markovian correlation structure, the use of which is both a contribution and a limitation. To the best of our knowledge, there has been no study of strategies that guarantee optimal statistical inference at the fusion center, while minimizing multihop routing costs. We are able to address this fundamental problem analytically by exploiting the Markov random field structure. On the other hand, the assumed structure raises the practical issues of accuracy and overhead in learning the dependency structure. Within the limitations of our model-based assumptions, we hope to provide insights applicable to more general structures.

Our paper is organized as follows. The system model and problem formulation are explained in sections III and IV. The MST-based heuristic and Steiner-tree reduction for optimal fusion are in sections V and VI. The experimental results are in section VII and section VIII concludes the paper.

III. System Model

A. Notations and Definitions

An undirected graph G is a tuple $G = (V, E)$, where V is the vertex set and $E = \{(i, j)\}, i, j \in V$ is the edge set. We allow graphs to have multiple or parallel edges, but no loops. The neighborhood function $\mathcal{N}_u(i; G)$ of a node i is the set of all other nodes having an edge with it in G. The set of nodes with a single neighbor are known as the leaves, denoted by $Leaf(G)$, otherwise they are internal. A subgraph induced by $V' \subset V$ on G is denoted by $G(V')$ and a clique is a complete subgraph having edges between any two nodes in V'. A maximal clique is one that is not contained in any other clique. Henceforth, a clique will refer to a maximal clique, unless otherwise mentioned.

For a directed graph (digraph), we denote the edges (arcs) by $< i, j >$, where the direction is from i to j, and j is an immediate successor of i, denoted by $\mathcal{N}_s(i)$, and i an immediate predecessor of j, denoted by $\mathcal{N}_p(j)$. The above graph functions f are extended to sets, defined by $f(A) := \bigcup_{i \in A} f(i)$. For example, (i, A) denotes the set of edges between i and members of A. For sets A and B, let $A \backslash B = \{i : i \in A, i \notin B\}$ and let $|\cdot|$ denote cardinality.

B. Statistical Model for Sensor Data

We assume that the sensor measurements are drawn from a Markov random field (MRF). The MRF falls under the framework of acausal graphical models and satisfies conditional-independence properties, based on its dependency graph. A simple example is the first order auto-regressive process. A general spatial random field is defined below.

Definition 1 (Markov random field): Let $\mathbf{Y}_V = [Y_i, i \in V]^T$ denote the random vector of measurements in set V. $\mathbf{Y}_V$ is a Markov random field with an (undirected) dependency graph $G = (V, E)$, if $\forall\, i \in V$,

$$Y_i \perp \mathbf{Y}_{V \backslash \{i, \mathcal{N}_u(i)\}} | \mathbf{Y}_{\mathcal{N}_u(i)}, \tag{1}$$

where $\perp$ denotes conditional independence.
In words, the above definition states that the value at any node, given the values at its neighbors, is conditionally independent of the rest of the network.

The Hammersley-Clifford theorem [28] states that for a MRF $\mathbf{Y}_V$ with dependency graph $G = (V, E)$, the joint PDF f, under the positivity condition, can be expressed as

$$-\log f(\mathbf{Y}_V; \Upsilon) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{Y}_c), \tag{2}$$

where $\mathcal{C}$ is a set of (maximal) cliques in G, the functions ψ_c, known as the normalized[1] clique potentials, are real valued, non-negative and not zero everywhere on the support of $\mathbf{Y}$ and the tuple $\Upsilon = \{G, \mathcal{C}, \psi\}$ specifies the MRF in (2).

From (2), we see that the complexity of the likelihood function is vastly reduced for sparse dependency graphs; here, the conditional-independence relations in (1) results in the factorization of the joint likelihood into a product of components, each of which depends on a small set of variables.

Remark: In (2), $\mathcal{C}$ contains only those cliques over which the potentials are non-zero. For example, for independent measurements, $\mathcal{C}$ is the vertex set; for Besag's auto-model [30], generated by exponential-family distributions, $\mathcal{C}$ is the edge set. In this paper, we assume that $|\mathcal{C}|$ is polynomial in the number of nodes. This is satisfied by graph families such as bounded-degree graphs [31].

C. Network Model

We assume the presence of a medium-access control that eliminates collisions or interferences among the nodes. All real numbers are quantized with sufficiently high precision so that the quantization error can be ignored. All nodes can function as both sensors and routers. The network is connected via a communication graph containing set of feasible bidirectional communication links. Note that this communication graph is different from the dependency graph of the MRF. We consider the unicast mode of routing, where a packet from a node is routed to a single destination and the intermediate nodes do not perform any processing or store the packet for future use.

D. Cost Model

In our formulation, the processing costs are assumed constant, and thus ignored in the optimization. Usually the routing costs reflect transmission energy, but it could also represent, for example, delay, bandwidth, or a combination of these considerations. We represent the routing of a real number by a packet. A symmetric routing cost function is assumed, and is denoted by $C_{i,j} > 0$. For a set of communication links G

[1]For general potentials, finding the normalizing constant (partition function) is NP-hard, but approximate algorithms have been proposed in [29].

(some of which may be parallel edges), let $\mathrm{C}(G)$ denote the total cost of routing using these links,

$$\mathrm{C}(G) := \sum_{e \in E} C_e, \tag{3}$$

where C_e is the cost of the link e and E is the edge set of G. If the cost function is not a metric, we consider its metric closure[2], and denote the metric costs by $\bar{C}_{i,j}$. There is no loss of generality, since the edges of the metric closure can be replaced with the corresponding shortest paths. The metric closure can also be approximated with localized spanners [33].

IV. Problem Formulation

A. Statistical Inference

We consider the distributed-inference setup, where a number of sensors measure the signal field and the designated fusion center makes a final decision on the underlying phenomenon. We specify the class of inference problems addressed in this paper. We consider binary hypothesis testing, with null hypothesis $\mathcal{H}_0$ and alternative hypothesis $\mathcal{H}_1$. Let $f(\mathbf{Y}_V; \mathcal{H}_j)$ be the PDF of the measurements $\mathbf{Y}_V$ of sensors in set V under hypothesis $\mathcal{H}_j$. The optimal decision rule is a threshold test based on the log-likelihood ratio (LLR),

$$\mathrm{LLR}(\mathbf{Y}_V) := \log \frac{f(\mathbf{Y}_V; \mathcal{H}_0)}{f(\mathbf{Y}_V; \mathcal{H}_1)}. \tag{4}$$

A *minimal sufficient statistic* for inference represents the maximum possible reduction in dimensionality of the sensor data, without destroying information about the underlying phenomenon [34]. The log-likelihood ratio (LLR) is the minimal sufficient statistic for hypothesis testing [35].

We assume that the measurement samples are drawn from distributions specified by distinct Markov random fields,

$$\mathcal{H}_0 : \Upsilon_0 = \{G_0(V), \mathcal{C}_0, \psi_0\} \text{ vs. } \mathcal{H}_1 : \Upsilon_1 = \{G_1(V), \mathcal{C}_1, \psi_1\}. \tag{5}$$

From (2), the LLR is given by

$$\mathrm{LLR}(\mathbf{Y}_V) = \sum_{a \in \mathcal{C}_1} \psi_{1,a}(\mathbf{Y}_a) - \sum_{b \in \mathcal{C}_0} \psi_{0,b}(\mathbf{Y}_b) \tag{6}$$

It is easily seen that the LLR is expressed as the sum of potentials of an "effective" Markov random field $\Upsilon = \{G_\Upsilon, \mathcal{C}, \phi\}$ specified as follows: the effective dependency graph $G_\Upsilon = (V, E_\Upsilon)$, with edge set $E_\Upsilon := E_0 \cup E_1$; the effective clique set $\mathcal{C}$ is $\mathcal{C} := \mathcal{C}_0 \cup \mathcal{C}_1$, with only the resulting maximal cliques retained; the effective potential functions ϕ_c are given by

$$\phi_c(\mathbf{Y}_c) := \sum_{a \in \mathcal{C}_1, a \subset c} \psi_1(\mathbf{Y}_a) - \sum_{b \in \mathcal{C}_0, b \subset c} \psi_0(\mathbf{Y}_b), \ \forall\, c \in \mathcal{C}. \tag{7}$$

[2]The metric closure on graph G, is defined as the complete graph where the cost of each edge is the cost of its shortest path in G. [32, p. 58].

Therefore, the LLR has a succinct form

$$\mathrm{LLR}(\mathbf{Y}_V; \Upsilon) = \sum_{c \in \mathcal{C}} \phi_c(\mathbf{Y}_c). \tag{8}$$

B. Localized Fusion Schemes

The succinct form of the LLR in (8) consists of clique potential functions ϕ and hence, is amenable to localized processing within the cliques of the Markov random field. In order to compute a potential function ϕ_c of clique c, access to measurements of all the clique members is needed. Therefore, each potential function ϕ_c is assigned a unique computation site, known as its *processor*, denoted by $Proc(c)$. We assume that the clique potential functions are processed "locally", at one of its members, i.e., $Proc(c) \subset c$. In practice, the information about the potential functions can be sent to the nodes by the fusion center after empirical joint-density estimation. Hence, such localized processing can significantly reduce the overhead involved in communication and storage of the function parameters. Localized processing can be especially efficient for proximity graphs, where the edges are included based on local point configuration [36].

The set of communication links G used by any fusion scheme fall into two categories, viz., those transporting raw measurements to the processor to compute the specified potential function, known as the *forwarding subgraph* $\mathrm{FG}(G)$ and the set of links that transport/aggregate these processed values, known as the *aggregation subgraph* $\mathrm{AG}(G)$. The tuple consisting of the forwarding and the aggregation subgraphs $\{\mathrm{FG}(G), \mathrm{AG}(G)\}$ of a fusion scheme is known as the *packet-operation digraph*. A schematic of a fusion scheme is shown in Fig.1.

The aim of any feasible fusion scheme for inference is to deliver the LLR in (8) to the designated fusion center v_0. Such a scheme is specified by a processor-assignment mapping $Proc$, a packet-operation digraph $\{\mathrm{FG}(G), \mathrm{AG}(G)\}$ and a sequence in which data is transported and processed. Hence, a fusion scheme is represented by tuple $\Gamma := \{Proc, \mathrm{FG}(G), \mathrm{AG}(G)\}$. Let $AggVal(i; \Gamma)$ be the value at node i at the end of fusion. Formally, the constraints on any feasible localized fusion scheme Γ are specified as follows:

- the LLR is delivered to the fusion center,

$$AggVal(v_0; \Gamma) = \mathrm{LLR}(\mathbf{Y}_V; \Upsilon), \tag{9}$$

- local processor assignment,

$$Proc(c; \Gamma) \subset c, \quad \forall\, c \in \mathcal{C}. \tag{10}$$

C. Cost Minimization

The minimum-energy fusion scheme for inference delivers the LLR to the fusion center, while minimizing the total cost of routing. Formally, it is formulated as finding a scheme Γ^*

with the optimal processor assignment and packet-operation digraph, such that the total routing cost is minimized

$$\Gamma^* := \arg\min_{\Gamma} \mathrm{C}(G), \tag{11}$$

subject to the constraints (9) and (10).

A special case of (11) addressed in the literature (e.g., [9]) is when the measurements are conditionally-independent. In this case, the minimum cost scheme is given by the directed minimum spanning tree (DMST), with the directions toward the fusion center, and the sum is calculated hierarchically, starting at the leaves and ending at the fusion center. This in fact turns out to be a lower bound for the cost of optimal fusion.

Lemma 1 (Lower bound on $C(G^*)$): The total routing cost for optimal fusion in (11) is no less than that of the minimum spanning tree (MST), based on the routing cost function, i.e.,

$$\mathrm{C}(\mathrm{MST}(V)) \leq \mathrm{C}(G^*(V)). \tag{12}$$

Proof: The constraints satisfied by an instance in (11) include the following: there is at least one link going out of every node other than possibly, the fusion center and the packet-operation digraph is weakly connected. MST is the minimum cost graph satisfying these constraints and in general, it does not deliver the LLR to the fusion center. □

Under our setup, any scheme delivering the LLR to the fusion center consists of a processor-assignment mapping and a packet-operation digraph, consisting of two types of links viz., forwarding and aggregation links. Given such an input, a simple algorithm that specifies a sequence of operations and transmissions to compute the LLR and deliver it to the fusion center is provided in [37, Appendix A]. It can be easily verified that all the schemes proposed in this paper deliver the LLR.

V. MST-BASED AGGREGATION

In this section, we propose a simple heuristic (AggMST), based on the minimum spanning tree. The heuristic is based on the fact that the LLR in (8) is the sum of potentials, and these potentials once computed, can be aggregated along the MST. However, note that unlike the case of independent data, the potentials depend on the data of a clique and therefore, additional transmissions are required.

We specify the AggMST scheme in Fig.2. For a clique c, the processor is assigned arbitrarily to the clique member with the lowest index (line 5). Other suitable factors such as residual energy can instead be used for the assignment. The shortest-path routes from other members of c to the processor are added to the forwarding subgraph FG (line 7), and the raw data is routed along these links to enable the computation of the clique potentials. Note that the construction of the FG can be implemented in a localized manner whenever the dependency graph is local (e.g., k nearest-neighbor graph, disk graph). The aggregation subgraph AG is DMST(V), the minimum spanning tree, directed towards the fusion center (line 11) and potentials are added hierarchically along AG.

Input: $V = \{v_0, \ldots, v_{|V|-1}\}$, v_0= Fusion center,
1: $\mathcal{C} = \{c_0, \ldots, c_{|\mathcal{C}|-1}\}$= maximal clique set of the MRF,
2: DMST(V) = Minimum spanning tree, direct toward v_0
3: SP(i, j)= (Directed) shortest path from i to j
4: **for** $j \leftarrow 0, |\mathcal{C}| - 1$ **do**
5: $Proc(c_j) \leftarrow \min_{v_i \in c_j} v_i$ // Arbitrary processor assignment
6: **if** $|c_j| > 1$ **then**
7: Add SP($c_j \backslash Proc(c_j), Proc(c_j)$) to FG
8: **end if**
9: **end for**
10: AG $\leftarrow$ DMST(V), $\Gamma \leftarrow \{Proc, \mathrm{FG}, \mathrm{AG}\}$
11: **return** Γ

Fig. 2. Heuristic for aggregation in a Markov random field (AggMST).

A. Performance bounds

In this section, we quantify the performance of the AggMST scheme for a special scenario that allows us to utilize the lower-bound result of Lemma 1.

Theorem 1 (Approximation): For the case when the routing costs are Euclidean and the dependency graph is a subgraph of the Euclidean MST, the AggMST scheme has an approximation ratio of 2.

Proof: The MST in the lower bound (Lemma 1) is Euclidean, since the transmission costs are Euclidean. Since the dependency graph is a subgraph of the Euclidean MST, all the links in AggMST are contained in the Euclidean MST. Hence, we have the approximation ratio of 2. To show that the bound is tight, we note that the case of extended equilateral triangles on the Euclidean plane achieves this bound. □

An important dependency graph that is a subgraph of the MST is the nearest-neighbor graph. It is the simplest proximity graph. We evaluate the performance of AggMST for other proximity graphs, based on simulations in section VII.

VI. STEINER-TREE REDUCTION FOR GENERAL MRF

In this section, we show that optimal fusion has a Steiner-tree reduction. We specify the graph transformations required for such a reduction and obtain the optimal processor assignment and packet-operation digraph. The *Steiner minimal tree* on graphs [38, p. 27] is defined as the tree of minimum total edge weight containing a specified set of vertices, known as *terminals*. We first show that a simplified version of the minimum cost aggregation problem, where the processor assignment is predetermined, is a Steiner tree.

Lemma 2 (Fixed processor assignment): If the assignment of the processors computing the clique potential functions is fixed, prior to cost minimization, then minimum cost aggregation in (11) is given by the packet operation digraph, with the forwarding subgraph consisting of the shortest-path routes from the clique members to the corresponding processor and the aggregation subgraph is the Steiner tree, with the set of processors and the fusion center as the terminals and the links directed toward the fusion center.

Input: $V = \{v_0, \ldots, v_{|V|-1}\}$, v_0= Fusion center,
$\mathcal{C} = \{c_0, \ldots, c_{|\mathcal{C}|-1}\}$= maximal clique set of the MRF,
G_t= Metric closure of comm. graph, $\bar{C}$ = Link costs in G_t,
ST$(G, \mathcal{L})$ = δ-approx. Steiner tree on G, terminal set $\mathcal{L}$
$G', V_c \leftarrow Map(G_t; \bar{C}, \mathcal{C})$
DST = ST$(G', V_c \cup v_0)$ and directed towards v_0
$\Gamma \leftarrow RevMap(\text{DST}; V_c, V, \mathcal{C})$
return Γ

Fig. 3. δ-approx. min. cost aggregation scheme Γ with processor assignment and packet-operation digraph via Steiner-tree reduction (AggApprox).

Proof: Once the processor assignment is fixed, in order to compute the potential functions, measurements from other nodes are routed to the corresponding processors through shortest-path routing. Now, the sum of the potential functions at the processors has to be delivered at the fusion center and is done optimally by aggregating along the Steiner tree. □

Such a fixed processor assignment could be due to different processing capabilities or considerations such as residual energy. Alternatively, one of the clique members can be randomly selected as a processor. We compare this scheme with optimal fusion through simulations in section VII.

The goal of this paper is to find the optimal processor assignment, since it affects the total cost of aggregation. In Lemma 3, we consider a simpler version of the problem and ignore the routing costs incurred in transporting the raw measurements to a processor and show a group Steiner tree reduction. A group Steiner tree is defined as the tree with minimum total edge weight, such that it includes at least one vertex from each specified group [39].

Lemma 3 (Minimum aggregation subgraph): If the routing costs of the forwarding subgraph or any other considerations for processor assignment are ignored, then the minimum cost aggregation subgraph is given by the group Steiner tree, with the cliques of the MRF as the groups.

Proof: At least one member of every clique of the MRF has to be in the aggregation subgraph, since its potential function needs to be processed locally. Since all the clique members have equal weights prior to selection, the optimal set of the processors are those that minimize the total cost of the aggregation subgraph, given by the group Steiner tree. □

In general, the processor assignment is not only dependent on the cost of the aggregation subgraph, but also on costs of forwarding subgraph. In Fig.4, we define a graph transformation $Map(G_t)$ on the metric communication graph G_t to incorporate these raw-data routing costs. After constructing a feasible (not necessarily the optimal) solution to the Steiner tree on the transformed graph $Map(G_t)$, we map it back to a feasible fusion scheme using the operation $RevMap$ in Fig.5. The complete procedure is summarized in Fig.3 (AggApprox).

The $Map(G_t)$ in Fig.4 operation involves adding new clique-representative nodes for each non-trivial clique (size greater than one) and connecting it to all its corresponding clique members (line 6). In line 9, the edge cost from a representative node to a clique member incorporates the raw-data routing costs, which is the initial cost incurred in assigning a member as the processor for the clique potential function. Note that for the group Steiner problem, such a transformation would require assigning artificial costs to such edges, whereas here, they are part of the minimization. We then find a feasible solution to the Steiner-tree on $Map(G_t)$, with the clique-representative nodes and the fusion center as the terminals. It is then directed towards the fusion center and denoted by DST. The reverse mapping $RevMap$(DST) assigns the unique immediate successor of every clique-representative node in DST as the clique processor. The edges from the

1: **function** $Map(G_t(V); \bar{C}, \mathcal{C})$
2: $\mathcal{N}_u(v; G)$ = Neighborhood of v in undirected G
3: Initialize $G' \leftarrow G_t$, $V_c \leftarrow \emptyset$, $n \leftarrow |V|$
4: **for** $j \leftarrow 0, |\mathcal{C}| - 1$ **do**// Let V and $\mathcal{C}$ be ordered
5: **if** $|c_j| > 1$ **then**
6: $V_c \leftarrow v_{n-1+j}$, Add new node v_{n-1+j} to G',
7: **for each** $v_i \subset c_j$ **do**
8: Add node v_i to $\mathcal{N}_u(v_{n-1+j}; G')$
9: $\bar{C}(v_{n-1+j}, v_i; G') \leftarrow \sum_{v_k \subset c_j, k \neq i} \bar{C}(v_i, v_k; G_t)$
10: **end for**
11: **else**
12: $V_c \leftarrow v_i$, for $v_i \subset c_j$// For trivial cliques
13: **end if**
14: **end for**
15: **return** G', V_c
16: **end function**

Fig. 4. $Map(G_t; \bar{C}, \mathcal{C})$ adds nodes corresponding to each non-trivial clique and returns the expanded graph G' and node set representing cliques V_c.

function $RevMap(G'; V_c, V, \mathcal{C})$
$\mathcal{N}_s(v; G)$, $\mathcal{N}_p(v; G)$ = Imm. successor, predecessor of v
Initialize $G \leftarrow G'$, $n \leftarrow |V|$
for each $v_j \in V_c$ **do**
if $j > n - 1$ **then**
$k \leftarrow j - n + 1$,
$Proc(c_k) \leftarrow \mathcal{N}_s(v_j; G')$, for $c_k \in \mathcal{C}$,
$V_j \leftarrow c_k \backslash Proc(c_k)$, Replace $< v_j, Proc(c_k) >$ in G with edges $< V_j, Proc(c_k) >$, mark them
if $\mathcal{N}_p(v_j; G) \neq \emptyset$ **then** Replace $< \mathcal{N}_p(v_j), v_j >$ in G with edges $< \mathcal{N}_p(v_j), Proc(c_k) >$
end if
else
$Proc(c_l) \leftarrow v_j$, for $v_j \subset c_l$ // For trivial cliques
end if
end for
FG $\leftarrow$ Marked edges of G, AG $\leftarrow G \backslash$FG
$\Gamma \leftarrow \{Proc, \text{FG}, \text{AG}\}$
return Γ
end function

Fig. 5. $RevMap(G; V_c, V, \mathcal{C})$ maps tree G' to fusion scheme Γ with processor assignment $Proc$, forwarding and aggregation subgraphs FG, AG.

representative nodes in DST are replaced by shortest paths from other clique members to the processor and added to the forwarding subgraph of the fusion scheme. All other edges, not belonging to representative nodes in in DST, are assigned as the aggregation subgraph. In the theorem below, we prove that this reduction is approximation-factor preserving.

Theorem 2 (Optimal Fusion): Given a hypothesis-testing problem with the log-likelihood ratio in (8), whose form has an effective Markov random field $\Upsilon = \{G_\Upsilon, \mathcal{C}, \phi\}$, with dependency graph G_Υ, clique set $\mathcal{C}$ of polynomial cardinality, and potential functions ϕ over cliques in $\mathcal{C}$, optimal fusion in (11) can be approximated via AggApprox in Fig.3 and has the same approximation factor as the Steiner tree on graphs.

Proof: AggApprox results in a feasible fusion and runs in polynomial time since there are polynomial number of cliques. For any feasible solution to Steiner tree, replacement of links in line 9 of $RevMap$ in Fig.5 reduces the sum cost. In $Map(G_t)$, the clique-representative nodes satisfy $\bar{C}_{v_{n-1+k},i}, \bar{C}_{v_{n-1+k},j} \geq \bar{C}_{i,j}$, $\forall i,j \subset c_k \in \mathcal{C}_\Upsilon$. Hence, representative nodes are leaves in the optimal Steiner tree, and the cost of optimal fusion and Steiner tree are equal. Hence, by definition [38, A.3.1], the reduction from minimum cost fusion to Steiner tree preserves the approximation factor. □

A. Comparison with Shortest-Path Routing

Since the optimal fusion scheme has a constraint of local processing, the shortest-path routing to the fusion center is not an instance in the optimization. Hence, it is not always possible to guarantee if optimal fusion in (11) performs better than shortest-path routing. For a special class of MRF, such a guarantee is given in the lemma below.

Lemma 4 (Advantage over Shortest-path Routing): The cost of optimal fusion is no greater than the cost of shortest-path routing of all the data to the fusion center, when all the cliques of the effective MRF of the LLR in (8) contain the fusion center as a member.

Proof: If all cliques contain the fusion center, then it is a possible processor for every clique potential function. Since shortest-path routing assigns the fusion center as the sole processor, it is one of the candidates in the optimization. □

Any random field without special properties has a complete graph as the dependency graph and falls into the above-mentioned category. Therefore, even without special structure, savings are possible, since only the LLR is needed at the fusion center and not the raw data. Optimal fusion involves finding an efficient processing site for the LLR in this case and then transporting it to the fusion center. In this special case, the optimal Steiner tree on the clique-expanded graph $Map(G_t)$ reduces to the shortest path between the group vertex v_c and the fusion center v_0, computable in polynomial time.

VII. Experimental Study

The conventional shortest-path routing is independent of the dependency graph. In this section, for different dependency graphs, we compare it with the schemes proposed in this paper:

1) AggMST heuristic (Fig.2),
2) Steiner-tree heuristic for optimal fusion (Fig.3),
3) random processor selection (Lemma 2).

We also plot the lower bound for the costs, stated in Lemma 1. Finding the Steiner tree is NP-hard and there has been extensive work on finding approximation algorithms. A simple MST heuristic for Steiner tree approximates the metric Steiner tree with the minimum spanning tree MST(L) over the set of terminals L, and has an approximation ratio of 2. The best known approximation bound for Steiner tree on graphs is 1.55, derived in [40]. In this section, the MST heuristic is used for Steiner trees. Note that this is different from the MST over the set of nodes, since the Steiner tree is over a new graph with added vertices. For dependency-graph models, we focus on two classes of proximity graphs: the k-nearest neighbor graphs and the ρ-constrained disk graphs. In the former, the number of neighbors is fixed, whereas the latter has a bounded neighborhood region.

A. Simulation Environment

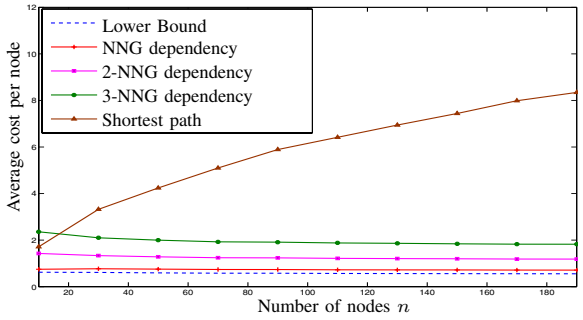
We assume that the set of feasible direct transmissions between the nodes is given by a connected disk graph. Power control is used to adjust the transmission power to a receiver's position and the routing cost is given by the minimum energy required for successful transmission from node i to node j, given by $C_{i,j} = |\text{dist}(i,j)|^\nu$. We find that under different radii for the set of feasible links and path loss ν, similar trends are observed. Hence, only the results for the complete graph (i.e., for a sufficiently large radius) and $\nu = 2$ are plotted.

Although, the approximation guarantee is valid for any node configuration, we employ a random placement. In our setup, n nodes are uniformly distributed in a square of area n (constant density scaling), since typically, nodes are added to new regions to enable sensing of new information. For any other scaling, the plots can be suitably modified. We randomly fix a node as the fusion center.

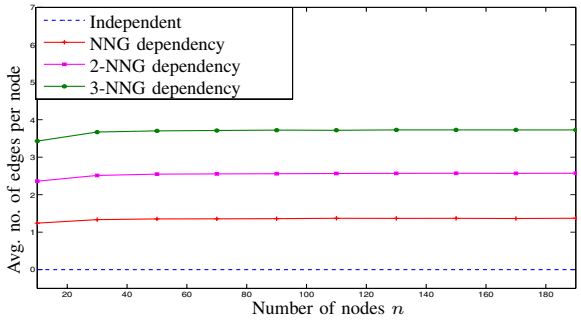
B. Besag's Model with k-Nearest Neighbor Graphs

We employ Besag's model [30], where the clique set is limited to set the edges of the dependency graph. Exponential family of conditional probabilities can generate such pairwise dependencies. The dependency graph is assumed to be the k-nearest neighbor graph (k-NNG). It has edges (i,j), whenever i is one of the k nearest neighbors of j or viceversa. We improve the AggMST heuristic for k-NNG graphs by modifying the arbitrary processor assignment (line 5 in Fig.2), to instead not include the leaves of the aggregation subgraph. This results in energy savings since the leaves do not participate during the phase when the potentials are combined.

The results are plotted in Fig.6. Only AggMST is shown, since all the heuristics performed similarly. In Fig.6a, we observe that the cost increases with k, but performs better than shortest-path routing, especially for large networks. In Fig.6b, we plot the average number of edges of k-NNG. and observe a correspondence between the two plots: the increase in routing cost for a larger k is due to more edges in the k-NNG, leading

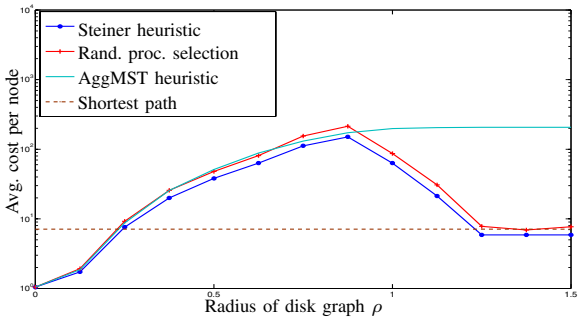


(a) Avg. routing cost per node

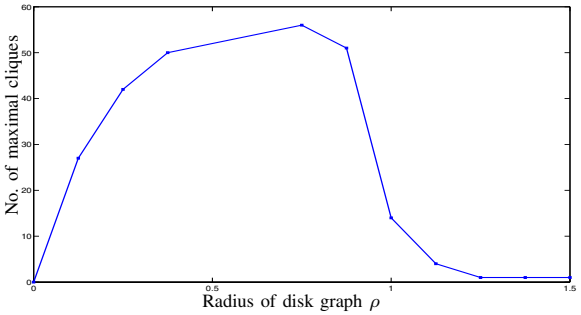


(b) Avg. no. of Edge potentials calculated

Fig. 6. Results for k nearest-neighbor dependency graphs. Uniform random placement of nodes. 500 runs. Constant density scaling for avg. cost.



(a) Avg. routing cost per node



(b) No. of maximal cliques of size > 1

Fig. 7. Results for disk dependency graph. 50 nodes with uniform random placement.

to more potential functions to be calculated. Also, note that the cost of the heuristics and number of edges per node in the two plots converge to constants. This is due to the law of large numbers for edge functionals of k-NNG [41].

C. ρ-Constrained Disk Graphs

A disk graph has edges between nodes within a specified threshold ρ and is used to model short-range spatial dependence [42]. We assume that all the (maximal) cliques of the disk graph have non-zero potential functions. Unlike the previous case with pairwise dependencies, here, both the number of cliques and their sizes depend on ρ. As the threshold ρ is increased, the size of the cliques increases; however, the number of maximal cliques initially increases and then decreases. We use the clique-enumeration algorithm for dynamic graphs in [43], to update the clique list for different values of ρ.

In Fig.7, we plot the routing cost per node and the number of non-trivial cliques (of size greater than one), as functions of ρ. Again, there is correspondence between the two plots. For low values of ρ, both the cost and the number of cliques are low. The maxima in both the plots occur for similar values of ρ. We also note the presence of a critical radius below which there is advantage over shortest-path routing. In this regime, the different heuristics have similar performance. For large values of ρ, the dependency graph is complete and hence, by Lemma 4, the Steiner-tree heuristics have similar performance as shortest-path routing. But the AggMST heuristic performs worse since it uses the entire MST to combine the potentials.

D. Implications

We see that savings due to aggregation are considerable compared to shortest-path routing for k-NNG and ρ-disk graphs, at low values of k and ρ. These graphs are probably the best candidates, after the independent-data case, for in-network processing of the likelihood function. For such sparse dependencies, the AggMST-heuristic has performance comparable to that of the Steiner-tree approximations. However, its implementation is much simpler. Also, we observe that there is direct correspondence between the number of cliques and the aggregation cost. Hence, the number of cliques is a good measure for judging the effectiveness in-network processing. The gap between the heuristics and the lower bound, represents the overhead arising due to correlation. A dense dependency graph has high aggregation costs due to the complexity of its likelihood function. This is unlike the case of compression with the aim of routing all the raw data to a destination, where a dense dependency graph (more correlation) implies redundancy and hence, reduction in routing costs.

VIII. Conclusion

We considered data aggregation for an inference application. Using the Markov random field model, we exploited the spatial dependencies to specify efficient localized processing of the data, without loss in global inference performance. We proposed a simple heuristic, based on the minimum spanning

tree. We proved a Steiner-tree reduction of optimal fusion enabling the use of approximation schemes for the Steiner tree with the same performance guarantee. Simulations show a significant saving in cost due to in-network processing for proximity-based sparse dependency graphs, compared to routing all the data to the fusion center.

We have considered a single round of data aggregation without explicitly addressing the issue of network lifetime. Our approach can also be adapted to multiple rounds, incorporating considerations such as residual energy. We have made a number of simplifying assumptions in this paper. Possible extensions are considering probabilistic reception of data, balancing the routing costs in the network and exploiting the broadcast nature of the wireless medium. Extension to other inference problems such as m-ary hypothesis testing and optimal quantization of correlated measurements would be of interest. We have also not considered the interplay between the cost and the time required to fuse all the data and the quality of the resulting decision at the fusion center.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for detailed comments. The first author would like to thank Prof. D.P. Williamson, Prof. A. Wagner, Dr. C. Bisdikian and Y. Sharma for extensive discussions.

REFERENCES

[1] K. Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks," *Elsevier Adhoc Networks*, vol. 3, pp. 325–349, 2005.

[2] S. Misra, L. Tong, and A. Ephremides, "Application dependent shortest path routing in ad-hoc sensor networks," in *Wireless sensor networks: signal processing & comm.* J. Wiley & Sons, 2007, ch. 11, pp. 277–310.

[3] R. Cristescu, B. Beferull-Lozano, M. Vetterli, and R. Wattenhofer, "Network correlated data gathering with explicit communication: NP-completeness and algorithms," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. 1, pp. 41–54, 2006.

[4] A. Goel and D. Estrin, "Simultaneous Optimization for Concave Costs: Single Sink Aggregation or Single Source Buy-at-Bulk," *Algorithmica*, vol. 43, no. 1, pp. 5–15, 2005.

[5] A. Scaglione and S. Servetto, "On the Interdependence of Routing and Data Compression in Multi-Hop Sensor Networks," in *Proc. MobiCom 2002*, Atlanta, Georgia, September 2002.

[6] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, "TinyDB: an acquisitional query processing system for sensor networks," *ACM Transactions on Database Systems*, vol. 30, no. 1, pp. 122–173, 2005.

[7] J. Gehrke and S. Madden, "Query processing in sensor networks," *IEEE Pervasive Computing*, vol. 03, no. 1, pp. 46–55, 2004.

[8] C. Intanagonwiwat, R. Govindan, and D. Esterin, "Directed Diffusion : A Scalable and Robust Paradigm for Sensor Networks," in *Proc. 6th ACM/Mobicom Conference*, Boston,MA, 2000, pp. pp 56–67.

[9] B. Krishnamachari, D. Estrin, and S. Wicker, "Modeling data centric routing in wireless sensor networks," in *INFOCOM*, New York, 2002.

[10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Tran. on W. Comm.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[11] A. Giridhar and P. Kumar, "In-network information processing in wireless sensor networks," in *Wireless sensor networks: signal processing & comm.* J. Wiley & Sons, 2007, ch. 3, pp. 43–68.

[12] R. Rajagopalan and P. Varshney, "Data aggregation techniques in sensor networks: A survey," *Comm. Surveys & Tutorials, IEEE*, vol. 8, no. 4, pp. 48–63, 2006.

[13] A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong, "Model-driven data acquisition in sensor networks," in *VLDB*, 2004.

[14] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On Network Correlated Data Gathering," in *IEEE Infocom*, Hong-Kong, March 2004.

[15] D. Marco, E. Duarte-Melo, M. Liu, and D. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *Proc. IPSN*, 2003, pp. 1–16.

[16] S. Yoon and C. Shahabi, "The Clustered Aggregation technique leveraging spatial and temporal correlations in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 3, no. 1, 2007.

[17] A. Jindal and K. Psounis, "Modeling spatially correlated data in sensor networks," *ACM Tran. on Sensor Net.*, vol. 2, no. 4, pp. 466–499, 2006.

[18] M. Cetin, L. Chen, J. Fisher, A. Ihler, R. Moses, and A. Willsky, "Distributed fusion in sensor networks: A graphical models perspective," in *Wireless sensor networks: signal processing & comm.* J. Wiley & Sons, 2007, ch. 9, pp. 215–250.

[19] S. Li, *Markov random field modeling in computer vision*. Springer-Verlag London, 1995.

[20] N. Cressie, *Statistics for spatial data*. J. Wiley, 1993.

[21] S. Sanghavi, V. Tan, and A. Willsky, "Learning Graphical Models for Hypothesis Testing," in *IEEE Workshop on Stat. Signal Proc.*, 2007.

[22] W. Zhang and G. Cao, "Optimizing tree reconfiguration for mobile target tracking in sensor networks," in *INFOCOM*, Hong Kong, 2004.

[23] N. Katenka, E. Levina, and G. Michailidis, "Local vote decision fusion for target detection in wireless sensor networks," in *Joint Research Conf. on Statistics in Quality Industry and Tech.*, Knoxville, USA, June 2006.

[24] Y. Sung, S. Misra, L. Tong, and A. Emphremides, "Cooperative Routing for Signal Detection in Large Sensor Networks," *IEEE J. Select. Area Comm.*, vol. 25, no. 2, pp. 471–483, 2007.

[25] J. Williams, J. Fisher III, and A. Willsky, "An Approximate Dynamic Programming Approach to a Communication Constrained Sensor Management Problem," in *Intl. Conf. on Information Fusion*, vol. 1, 2005.

[26] O. Kreidl and A. Willsky, "Inference with Minimal Communication: a Decision-Theoretic Variational Approach," in *Advances in Neural Information Processing Systems*, 2006.

[27] A. Anandkumar, L. Tong, and A. Swami, "Detection of Gauss-Markov Random Fields under Routing Energy Constraint," in *Proc. of 45-th Allerton Conf. on Communication, Control and Computing*, Sept. 2007.

[28] J. Hammersley and P. Clifford, "Markov fields on finite graphs and lattices," *Unpublished manuscript*, 1971.

[29] K. Jung and D. Shah, "Approximate message-passing inference algorithm," in *IEEE ITW*, 2007, pp. 224–229.

[30] J. Besag, "Statistical analysis of non-lattice data," *The Statistician*, vol. 24, no. 3, pp. 179–195, 1975.

[31] D. Eppstein, "All maximal independent sets and dynamic dominance for sparse graphs," *Proc. of ACM-SIAM symp. on discrete algorithms*, pp. 451–459, 2005.

[32] B. Wu and K. Chao, *Spanning Trees and Optimization Problems*. Chapman & Hall, 2004.

[33] X. Li, "Algorithmic, geometric and graphs issues in wireless networks," *Wireless Comm. and Mobile Computing*, vol. 3, no. 2, March 2003.

[34] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1994.

[35] E. Dynkin, "Necessary and sufficient statistics for a family of probability distributions," *Tran. Math, Stat. and Prob.*, vol. 1, pp. 23–41, 1961.

[36] L. Devroye, "The expected size of some graphs in computational geometry." *Comp. & math. with app.*, vol. 15, no. 1, pp. 53–64, 1988.

[37] A. Anandkumar, L. Tong, and A. Swami, "Energy-efficient data fusion in Gauss-Markov random field," Cornell University, Tech. Rep. ACSP TR 07-07-01, July 2007. [Online]. Available: http://acsp.ece.cornell.edu/papers/AnandkumarInfocomReport.pdf

[38] V. Vazirani, *Approximation Algorithms*. Springer, 2001.

[39] G. Reich and P. Widmayer, "Beyond steiner's problem: a vlsi oriented generalization," in *Proc. of Intl. workshop on Graph-theoretic concepts in computer science*, 1990, pp. 196–210.

[40] G. Robins and A. Zelikovsky, "Tighter Bounds for Graph Steiner Tree Approximation," *SIAM Journal on Discrete Mathematics*, vol. 19, no. 1, pp. 122–134, 2005.

[41] M. Penrose and J. Yukich, "Limit Theory for Random Sequential Packing and Deposition," *Annals of Applied probability*, vol. 12, no. 1, pp. 272–301, 2002.

[42] A. Pettitt, I. Weir, and A. Hart, "A Conditional Autoregressive Gaussian Process for Irregularly Spaced Multivariate Data with Application to Modelling Large Sets of Binary Data," *Statistics and Computing*, vol. 12, no. 4, pp. 353–367, 2002.

[43] V. Stix, "Finding All Maximal Cliques in Dynamic Graphs," *Comp. Optimization and Applications*, vol. 27, no. 2, pp. 173–186, 2004.