Bottleneck Throughput Maximization for Correlated Data Routing: A Game Theoretic Approach

Engin Zeydan*, Didem Kivanc Tureli†, Cristina Comaniciu‡, Uf Tureli†

*Lane Dept. of CSEE, West Virginia University, Morgantown, WV, 26506.
†Dept. of ECE, West Virginia University Institute of Technology, Montgomery, WV, 25131.
‡Dept. of ECE, Stevens Institute of Technology, Hoboken, NJ, 07030.
Email : ezeydan@mix.wvu.edu, didemk@ieee.org, ccomanic@stevens.edu, tureli@ieee.org

***Abstract*— In this paper, we propose an efficient bottleneck throughput maximizing routing framework for correlated data gathering in wireless sensor networks. Our proposed routing metric exploits the the data correlation present in sensor networks. For throughput-maximizing correlation aware routing, a game theoretic framework is developed for a local solution of the NP-complete optimization problem. The proposed throughput maximization algorithm selects the best routes to increase the bottleneck throughput of each source in the network using best response dynamics. Numerical results corroborates predicted throughput gains.**

***Index Terms*— correlated data aggregation, game theory, throughput maximization, wireless sensor networks**

I. Introduction

In large scale wireless sensor networks (WSNs), nodes in a certain region have knowledge of only nodes in their neighborhoods, so efficient optimization of communication patterns, like resource allocation and network connectivity should be done in a distributed manner. This signifies the importance of efficient decentralized routing of data with minimum communication overhead. The main goal of most routing algorithms in WSNs is to minimize the total transmission cost of transporting the data collected by nodes. Many routing protocols have been proposed for WSNs emphasizing various metrics depending on the application and design specifications [1] [2] [3].

In WSNs, the data from different nodes in a region is highly correlated. For example, if the data are random variables, e.g. temperature measurements, the measured values at nodes are spatially and temporally correlated. Transmitting all sensor data can increase traffic and data redundancy at the destination nodes. This may result in inefficient resource usage of the overall network. Therefore, correlation aware routing can significantly affect routing decisions when data aggregation is involved [4]. In WSNs, routing with data aggregation aims to find the optimum network topology for maximum data gathering to reduce the cost function in resource-limited sensor networks. Recent work has looked at exploiting the data correlation by data aggregation along with multi-hop routing in WSNs [5][6].

The research was supported in part by National Science Foundation (NSF) Grant CNS-0520232.

In this paper, we propose an efficient throughput maximizing routing strategy for gathering correlated data using a multi-hop data aggregation model which relies on the source data correlation between neighbor nodes. We exploit the potential for collaboration among sensors in data gathering and processing. The proposed routing algorithm also combines interference avoidance and correlation awareness to maximize throughput. We develop a simple game-theoretic model with different utility functions that account for interference and data correlation. Numerical studies are used to evaluate the performance of these different routing strategies.

The rest of the paper is organized as follows: In Section II, we define the system and data aggregation models. An efficient routing framework for throughput maximization is proposed in Section III. In Section IV, we present different facility cost selection choices for the congestion game and also show the convergence of our proposed algorithms. In Section V, we present numerical analysis. Finally, Section VI is concluding remarks and observations.

II. System Model

In this paper, joint data aggregation and routing for data gathering in WSNs is combined with interference avoidance. We consider the problem of maximum correlated data gathering with a single sink, to which all the data has to be sent. Let the network graph $G = (V, E)$ consist of all nodes V, where $|V| = N + 1$, with E consisting of edges between nodes that can communicate with each other. We assume that a set $S \subset V$ of $\alpha = N\varphi$ nodes are data sources where φ is the ratio of source nodes to all nodes N (excluding sink node). The sources are labeled as Y_1 through Y_α and the relays are labeled as $Y_{\alpha+1}$ through Y_N. For the sake of simplicity, we assume there is only single sink labeled as D where the data from all the nodes has to arrive. In the following, throughput maximizing data gathering from source nodes to sink is examined. We consider synchronous direct-sequence CDMA (DS-CDMA) where all nodes use a variable spreading sequences of length L. The minimum spreading gain between the nodes Y_i and Y_j to reach a certain target SIR, γ^*, is [3]

$$L_{i,j} = \frac{\gamma^* \left[\sum_{k=1,k\neq i,j}^{N} h_{k,j} P_k\right]}{h_{i,j} P_i - \gamma^* \sigma^2}, \tag{1}$$

where the link gain $h_{i,j} = 1/d_{i,j}^2$ and $d_{i,j}$ is the distance of between the nodes Y_i and Y_j, σ^2 is the thermal noise. The transmission rate, or bit throughput between nodes Y_i and Y_j in bits-per-second (bps) is determined by the spreading rate $L_{i,j}$: $\Omega_{i,j} = W/L_{i,j}$ where W is the system bandwidth.

To quantify the amount of data generated by each sensor node and data aggregation along the route, sources are associated with data rates or weights [6]. Each source node Y_i generates a certain amount of data at rate $\Psi(Y_i)$, where $\Psi(Y_i)$ is the weight (or data rate) of source Y_i. The data rate $\Psi_i(Y_i)$ shows the average number of bits per symbol used to encode the data source and has units bits/symbol. It is assumed that data collected by the sensor nodes is correlated over geographical regions. Therefore, depending on the density of the WSNs in the field the readings from nearby nodes maybe highly correlated and hence contain data redundancies.

A. Data Aggregation Model

For data aggregation, we adopt the lossless step-by-step multi-hop aggregation model introduced in [6]. In this approach, each source node aggregates its data with that of the children nodes in a step-by-step fashion. We assume that relay nodes only convey the data and does not perform data aggregation.

Since the data aggregation is performed at parent node to aggregate its own data with that of its children, in order to avoid confusion, we denote $\Psi(Y_i)$ as the data rate of node Y_i before aggregation, and $\breve{\Psi}(Y_i)$ as the temporary stored data rate of node Y_i after data aggregation of parent node with one of its child node, and $\Psi_i(\mathbf{Y}_i)$ as sum of all aggregated data rate at node Y_i after the aggregation process is done.

Assume that source node Y_i has q source nodes (i.e. parent node Y_i has q children nodes) denoted as $\{Y_1, Y_2, ..., Y_q\}$ with data rates $\{\Psi(Y_1), \Psi(Y_2), ..., \Psi_k(\mathbf{Y}_k), ..., \Psi(Y_q)\}$ (note that some of the children nodes, like node Y_k here, may also have already aggregated data from their "own" sub-tree). The new temporary aggregated data content at node Y_i, i.e. $\breve{\Psi}(Y_i)$, after aggregation of node Y_i and one of its children node Y_j is calculated as [6]

$$\begin{aligned}\breve{\Psi}(Y_i) = \max\left(\Psi(Y_i), \Psi(Y_j)\right) \\ + (1-\rho_{i,j}) \min\left(\Psi(Y_i), \Psi(Y_j)\right),\end{aligned} \tag{2}$$

where $\rho_{i,j}$ is the correlation coefficient between nodes Y_i and Y_j.

In this approach, the aggregation of multiple inputs $\mathbf{Y}_i = \{Y_i, Y_1, Y_2, ..., Y_q\}$ is performed step by step with each node taking their turns depending on their arrival times. At the end of the aggregation process, the total data content of node Y_i becomes $\Psi_i(\mathbf{Y}_i)$. The reason for step by step aggregation of multiple inputs rather than joint aggregation schemes like differential entropy model [7] is that storing multiple sources' data and aggregating them at once requires large memory and power for sensors. Moreover, the data reported with different nodes will arrive at the parent node at different times due to intermediate nodes signal processing, distance between nodes, wireless medium characteristics, error-control schemes, interference, noise or transmitted powers.

III. Efficient routing framework for throughput maximization

A. Symbol Throughput

Two components of the network will determine the symbol throughput of a node, namely the data rate of sources (or the weight of sources) and bit throughput. Given the correlation models in Section II.A, the symbol throughput of a link between nodes Y_i and Y_j is defined as

$$\zeta_{i,j}(\Psi_i(\mathbf{Y}_i)) = \frac{W}{L_{i,j}\Psi_i(\mathbf{Y}_i)}, \tag{3}$$

where $\mathbf{Y}_i$ is the set of all q sources using node Y_i including Y_i, i.e. $\mathbf{Y}_i = \{Y_i, Y_1, Y_2, \ldots, Y_q\}$, W is the system bandwidth and $L_{i,j}$ is defined in (1). The symbol throughput has units symbol/second (sps) and symbolizes the total amount of symbol transmitted per second to the destination.

B. Optimization Problems

Our joint optimization problem in the network and physical layers for throughput maximization can be formulated as follows

$$\begin{aligned}\text{Maximize} \quad & \sum_{i=1}^{\alpha} \sum_{k,l \in S_i} \zeta_{k,l}(\Psi_k(\mathbf{Y}_k)) \\ \text{subject to} \quad & SIR_{k,l} \geq \gamma^*, \\ & P_k = C \ \text{ and } \ S_i \in X_i,\end{aligned} \tag{4}$$

where C is the constant transmission power, $\mathbf{Y}_k$ is the set of all sources using node k, i.e. $\mathbf{Y}_k = \{Y_1, Y_2, \ldots, Y_q : k \in S_i\}$, S_i is the set of relaying or aggregating nodes used for source Y_i, and X_i is the set of all possible relaying nodes for a route of source Y_i and $\zeta_{k,l}(\Psi_k(\mathbf{Y}_k))$ is the symbol throughput between nodes k and l defined in (3), α is the number of data sources.

We redefine the throughput of each source Y_i as the minimum of the throughput of all the links in a route, since the link with the least throughput determines the throughput of each source. This link is the bottleneck link for that particular source node. Hence, we rewrite the optimization problem as

$$\begin{aligned}\text{Maximize} \quad & \sum_{i=1}^{\alpha} \zeta_i \\ \text{subject to} \quad & SIR_{k,l} \geq \gamma^*, \\ & P_k = C,\end{aligned} \tag{5}$$

where

$$\zeta_i = \min_{\forall k,l \in S_i} \ \zeta_{k,l} \ \text{ and } \ S_i \in X_i. \tag{6}$$

Note that $\zeta_{k,l}$ changes when the source uses one of the relaying nodes, since the interference level on the overall network will change due to power transmission of each added relay node.

The bottleneck throughput maximization problem defined above i.e. the *max-min* optimization problem is analogous to the *min-max* network lifetime maximization problem [8] [9]. Furthermore, the algorithms proposed in this paper for bottleneck throughput maximization can also be used to decrease the load over heavily used nodes to increase the network lifetime or to decrease the *delay* in the network. As the *min* or *max* function is not differentiable, the distributed solutions based on the gradient algorithms are not directly applicable to the optimization problem in (5). One solution is to use sub-gradient algorithms to solve the *min-max* problem in a distributed manner [9]. However, number of iterations required for convergence is substantial. Another solution is to approximate the *min-max* function using some smoothing functions [8]. However, approximation of *min-max* problem with the smoothing function needs special structure of network and can't be easily extended to general network structures.

As each sensor node in the network aggregates all inputs with its own data to form an outgoing aggregated data, the optimum solution will depend on the constructed tree rooted at sink D. The optimum solution is hard if each sensor uses the data aggregation model in Section II.A since at each route multi-hop aggregation is employed. Moreover, finding the optimal solution is harder if the interference level in the network changes dynamically. Note that $\zeta_{k,l}(\Psi_k(\mathbf{Y}_k))$ between nodes Y_k and Y_l will change if the source node uses one of the relaying nodes. This will change the interference level on the overall network due to addition/subtraction of each relay node. The joint optimization of transmission cost and data aggregation is known to be NP-complete even for simplistic assumptions of data aggregation models [10]. Therefore, finding the optimal bottleneck throughput maximization algorithm is a hard optimization problem. Next, we will introduce some decentralized throughput maximization algorithms using correlation structure of the network. We propose a game theoretic formulation which can be shown to converge to a local optimal solution with relatively low complexity and in a distributed fashion.

1) Game theoretic interpretation: The above problems can be formulated as a congestion game model which can be shown to be isomorphic with a potential game. In this game, the players are the source nodes in quest for routes, the relaying nodes are the shared facilities, the action of the players is the selection of a group of facilities that form a route, and costs can be associated with various route selections.

Formally, the proposed game-theoretic routing model for correlation aware routing considers the route selection of each sensor node as a congestion game Γ. The game Γ is defined as a tuple $(\mathcal{N}, \mathcal{F}, (X_i)_{i\in\mathcal{N}}, (w_f)_{f\in\mathcal{F}})$ where $\mathcal{N} = \{Y_1, \ldots, Y_\alpha\}$ denotes the set of players, i.e. the source nodes in our game, $\mathcal{F} = \{1, \ldots, m_f\}$ is the set of facilities, $X_i \subseteq 2^{\mathcal{F}}$ is the strategy space of player (or source) Y_i, and $w_f : \mathbf{S} \to \mathbb{R}$ is a cost function associated with using the facility f. $\mathbf{S} = (S_1, \ldots, S_\alpha)$ is a *state of the game* in which player Y_i chooses strategy $S_i \in X_i$. We define $\theta_f(\mathbf{S})$ as the subset of *sources* directly connected to facility f including the source node at facility f, that is $\theta_f(\mathbf{S}) = \{Y_i | f \in S_i\}$. The players aim at choosing strategies $S_i \in X_i$ minimizing their individual cost, where the cost $\delta_i(\mathbf{S})$ of player Y_i is given by $\delta_i(\mathbf{S}) = \sum_{f\in S_i} w_f(\theta_f(\mathbf{S}))$.

We define utility function for source Y_i in our congestion game as

$$\begin{aligned} u_i : \mathbf{S} \to \mathbb{R},\ \ u_i(S_i, S_{-i}) &= -\delta_i(\mathbf{S}) \\ &= -\sum_{f\in S_i} w_f(\theta_f(\mathbf{S})), \end{aligned} \tag{7}$$

where $S_{-i} = (S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_\alpha)$ is the strategy space of player Y_i's opponents. The game performance is influenced by the selection of cost functions $w_f(\theta_f(\mathbf{S}))$ for facilities.

IV. T-CAR AND T-ICAR FOR THROUGHPUT MAXIMIZATION

To solve the effective bottleneck throughput maximization problem described in (5), we propose throughput maximizing correlation aware routing (T-CAR). We define the branches of the network as the distinct clusters of nodes rooted at facility f that are one hop distance into the sink D (see Fig. 1). The branch is denoted as $B_k,\ k = \{1, 2, 3, ..., M\}$ where M is the total number of branches in the network. The total throughput in the network is the sum of bottleneck throughput of all sources in all branches. Assume there are n_{B_k} source nodes in a branch B_k. The bottleneck throughput maximization problem is the selection of the best path or strategy S_i in any branch $B_k, k = \{1, 2, 3, ..., M\}$ that gives maximum utility. Clearly, every branch in the tree may have different number of sources and communication structures between those sources and may have different bottleneck throughput. However, the bottleneck throughput of all sources in a branch are same. Define $\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))$ as the bottleneck throughput of branch B_k over all facilities f in branch B_k for the set of state $\mathbf{S}$. In other words,

$$\begin{aligned} \xi_{B_k}(\Psi_f(\theta_f(\mathbf{S}))) &= \min_{\forall f\in B_k} \left(\zeta_f\left(\Psi_f(\theta_f(\mathbf{S}))\right)\right), \\ &= \min_{\forall f\in B_k} \left(\frac{W}{L_f \Psi_f(\theta_f(\mathbf{S}))}\right), \end{aligned} \tag{8}$$

where L_f is the minimum spreading gain for transmitting on the link used by facility f. The cost of using the branch B_k is

$$w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))) = -n_{B_k}\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S}))). \tag{9}$$

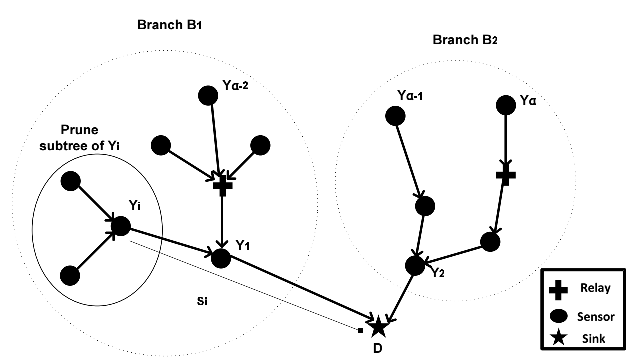

Fig. 1. Pruning of the subtree rooted at Y_i to calculate $\Psi_f(\theta_f^{-i}(\mathbf{S}))$ over facilities $\{Y_i, Y_1\} \in S_i$.

Let $\xi_{B_k}(\Psi_f(\theta_f^{-i}(\mathbf{S})))$ denote the bottleneck throughput of branch B_k when source node Y_i and its sub-tree nodes are not present for the same set of state $\mathbf{S}$. Then, the utility function $u_i(S_i, S_{-i})$ of source Y_i is defined as

$$\begin{aligned} u_i(S_i, S_{-i}) &= -\Big(w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))) \\ &\quad - w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f^{-i}(\mathbf{S}))))\Big), \\ &= \Big(n_{B_k}\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S}))) \\ &\quad - (n_{B_k} - n_i)\xi_{B_k}(\Psi_f(\theta_f^{-i}(\mathbf{S})))\Big), \end{aligned} \tag{10}$$

where n_i is the number of source nodes on the subtree rooted at source node Y_i including Y_i. Hence, each player Y_i in the network wants to find the best strategy S_i searching over all branches B_k $k = \{1, 2, ..., M\}$ that gives maximum utility defined in (10).

An alternative approach which considers both opportunities for aggregation and interference impact, is throughput maximizing interference and correlation aware routing (T-ICAR). T-ICAR is a combination of T-CAR and Interference Aware Routing (IAR) [3]. The only difference between T-ICAR and T-CAR algorithms is the starting trees in the first iteration. T-ICAR starts with IAR and T-CAR algorithm starts with Minimum Energy Routing (MER). MER and IAR are constructed based on the utility function definitions in [4].

A. Potential Game Formulations

An exact potential function $\mathcal{P}(.)$ is defined as

$$\begin{aligned} \mathcal{P} : \mathbf{S} \to \mathbb{R},\ \ \forall i \in \mathcal{N} \ \text{ and } \ S_i, S_i' \in \mathbf{S}, \\ u_i(S_i, S_{-i}) - u_i(S_i', S_{-i}) = \mathcal{P}(S_i, S_{-i}) - \mathcal{P}(S_i', S_{-i}). \end{aligned} \tag{11}$$

A game that has an exact potential function is called an exact potential game. We will demonstrate that throughput maximizing correlation aware routing with utility function given by (10) is an exact potential game (EPG).

Theorem 1 : T-CAR with utility function defined by (10) is an EPG with potential function,

$$\mathcal{P}(S_i, S_{-i}) = -\sum_{k=1}^{M} w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))), \tag{12}$$

where $w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))) = -n_{B_k}\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S})))$.

Proof: The proof follows a similar approach as in Appendix of [4] with the new definition of cost function $w_{B_k}(.)$. ■

With the help of a exact potential function, congestion games always admit at least one pure Nash equilibrium (NE) [11] [12].

Corollary 1 : T-CAR (T-ICAR) always have NE and converge to NE strategies by using a best response adaptive strategy.

Corollary 2 : Any NE of the T-CAR (T-ICAR) is locally optimal, i.e. in a NE, all the players can't increase the throughput unilaterally and thus the game reaches a local maximum throughput point.

The procedure for sequential updates of routes with best response dynamics of T-CAR can be summarized as following:

$\mathbf{Initialization}$: Construct minimum energy routes using the distributed algorithms like Dijsktra's Algorithm [13]. Then, use the following iterations for throughput maximization (T-CAR).

$\mathbf{Repeat}$: At each iteration n.

1) For each of the source node, Y_i, $i \in \{1, 2, ..., \alpha\}$,
   a) Select $S_i \in X_i$ that maximizes the utilities in (10) for T-CAR.
   b) Update routing strategy S_i for source Y_i

$\mathbf{Until}$: The stopping criteria Δ is met.

The stopping criterion Δ is the minimum number of iteration steps κ for the algorithm to converge, where κ is a counter which adds one after each updating process.

The total throughput in the network for T-CAR (or T-ICAR) is defined as the total bottleneck throughput of each branch on the tree,

$$\zeta^{total} = -\left(\sum_{k=1}^{M} w_{B_k}(\xi_{B_k}(\Psi_f(\theta_f(\mathbf{S}))))\right). \tag{13}$$

V. Simulation Results

In this section, we present an extensive set of numerical results to evaluate the performance of our proposed routing algorithms with the other classical approaches.

A. Simulation Setup

The number of sensor nodes in the network is varied between $N = 10$ to $N = 34$, which is uniformly distributed over a square area of dimension $100m \times 100m$. The ratio of all nodes to source nodes is selected to be $\varphi = 1$ (i.e. all nodes are sources) or $\varphi = 0.5$ (i.e. the number of relay nodes and sensor nodes are same). We adopt *Gaussian random field* data correlation model that is frequently encountered in practice [7]. In this model, correlation coefficient $\rho_{i,j}$ between nodes Y_i and Y_j decreases exponentially with the increase of the distance between nodes $d_{i,j}$, i.e. $\rho_{i,j} = \exp(-d_{i,j}^2/c)$ where c is the correlation constant where $c = 0$ corresponds to no aggregation, $c = 100$ corresponds to low correlation and $c = 1000$ corresponds to high correlation environment. Similar to [6], in order to distinguish the correlation between the raw data and aggregated data, we use a "forgetting" factor for aggregated data. The correlation between aggregated data at two parent nodes is only a fraction of their own data correlation calculated according to their distance. Throughout the simulation, we use a factor of 0.8. The noise power is $\sigma^2 = 10^{-13}$ Watts, which corresponds to thermal noise power for a bandwidth of $W = 1$ Mhz. We choose the equal transmit powers of all nodes to be 110 dB above the noise floor ($P_i = 10^{-2}$ Watts). The target SIR is selected to be $\gamma^* = 5$ (7 dB). We assume that the packet length is $M = 80$. The generated raw data rate of each source, $\Psi(Y_i)$, is assumed to be constant for all $Y_i \in \mathcal{N}$ and without loss of generality, each symbol is represented with 1 bit of information, i.e. $\Psi(Y_i) = 1$ bits/symbol. The results are simulated and averaged over 100 different network configurations for each routing algorithm.

B. Throughput Improvements

Fig. 2 shows the total throughput of T-CAR and MER versus number of nodes for $\varphi = 1$, $c = 100$ and $c = 1000$. We see that the throughput of T-CAR is higher for high correlation condition ($c = 1000$). The reason is that when the data correlation between neighboring nodes becomes higher, more redundant data can be removed with data aggregation.

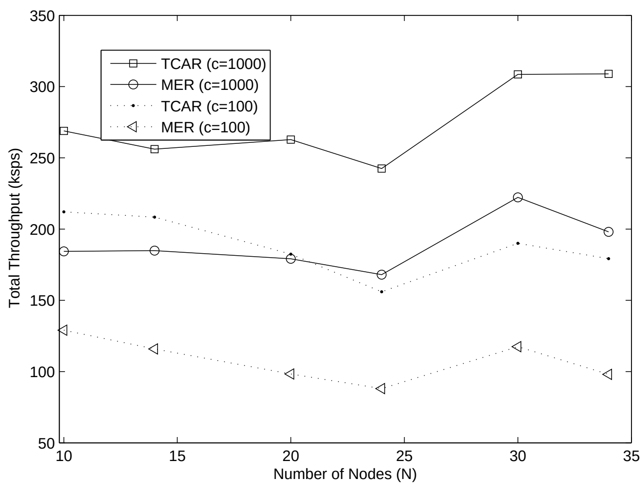

Fig. 2. Total throughput versus number of nodes for $\varphi = 1$.

The total throughput is not linearly increasing, but may fluctuate as number of node increases. This is due to the fact that MER algorithm, which is the starting tree of T-CAR algorithm, is optimized for energy, not for throughput. The increased number of nodes has mainly two affects on the network throughput of T-CAR and MER algorithms: First, it increases the simultaneous transmissions which also increases the interference in the network. As a result, the bottleneck throughput of each node decreases. Second, increasing the number of nodes clearly increases the total throughput. These results have reverse effect on network throughput. However, from Fig. 2, we can see that in general the total bottleneck throughput of the network tends to increase which shows that the increasing number of nodes dominates the interference effect for the bottleneck throughput. As an example, for $N = 10$ and $c = 1000$, the throughput of MER is 184.34 ksps, while the throughput of T-CAR is 268.83 ksps and the percentage improvement of T-CAR over MER is approximately 31.43 percent. On the other hand, for $N = 30$, the throughput of MER is 222.21 ksps, the throughput of T-CAR is 308.51 ksps which corresponds to the improvement of 35.88 percent.

Numerical studies also show important total throughput improvements of T-ICAR algorithm over T-CAR, IAR, and MER algorithms as shown in Fig. 3 for $N = 24$, $\varphi = 0.5$ and $c = 100$. In terms of total throughput, T-ICAR and T-CAR outperform IAR and MER, because their utility functions are designed for throughput maximization. The total throughput improvement of T-ICAR is on the order of 13.71 percent over T-CAR, 115.47 percent over IAR and 136.63 percent over MER. From Fig. 3, we also see that T-CAR performs 89.50 percent better than IAR at this moderate interference environment. However, it has been shown in [3], that the gains of IAR can also diminish for very low, or very high interference environments.

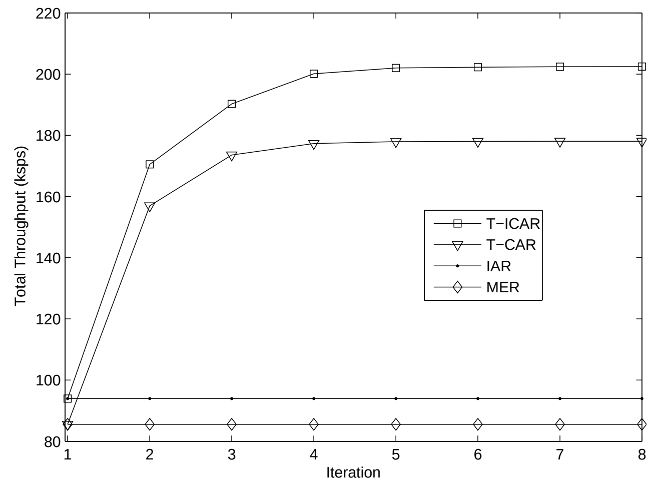

Fig. 3. Total throughput of MER, IAR, CAR and ICAR throughout the iteration process for $N = 24$, $\varphi = 0.5$ and $c = 100$.

C. Impact of Correlation Coefficient

Fig. 4 shows the total throughput of T-CAR and MER algorithms as the correlation constant c increases from 0 to 1000 for different network sizes. T-CAR algorithm gives significant throughput improvements compared to MER algorithm for all network size and under all the correlation constants. The throughput of both algorithms increases with correlation constant. However the improvements of T-CAR compared to MER decrease with increasing correlation constant. This is due to the fact that MER algorithm's optimization objective is energy, not throughput. At low correlation levels and small network sizes, T-CAR has more potential and opportunity to give higher throughput improvements. Hence, the gain of throughput improvements is higher.

As the correlation constant increases, the throughput increase of MER algorithm is higher than the throughput increase of T-CAR algorithm.

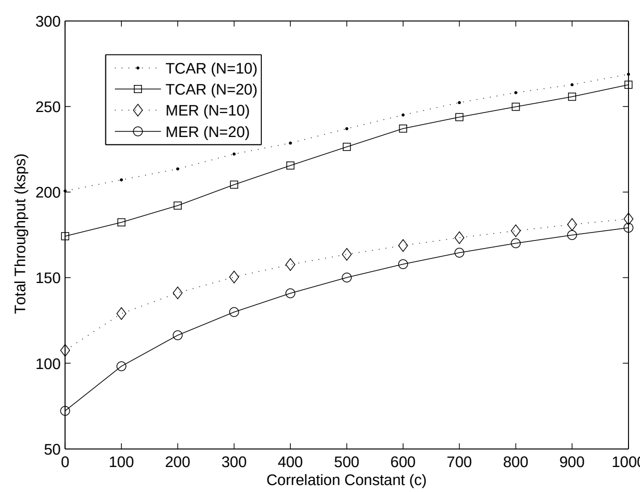

Fig. 4. Total throughput versus correlation constant (c) for $\varphi = 1$.

For example, for correlation constant of $c = 200$, the percentage throughput improvements of T-CAR over MER are 46.12 and 37.7 for network sizes of $N = 20$ and $N = 10$ respectively, whereas, for correlation constant of $c = 800$, the percentage improvements are 31.92 and 31.26 for $N = 20$ and $N = 10$ respectively. Similar conclusions may also be drawn for T-ICAR algorithm's throughput as the correlation coefficient changes.

D. Convergence of the Algorithm

In addition to the effectiveness of T-CAR, the number of iterations for the convergence of the distributed algorithm is also important. In this section, we show the number of required iterations for the convergence of the T-CAR algorithms for different network sizes.

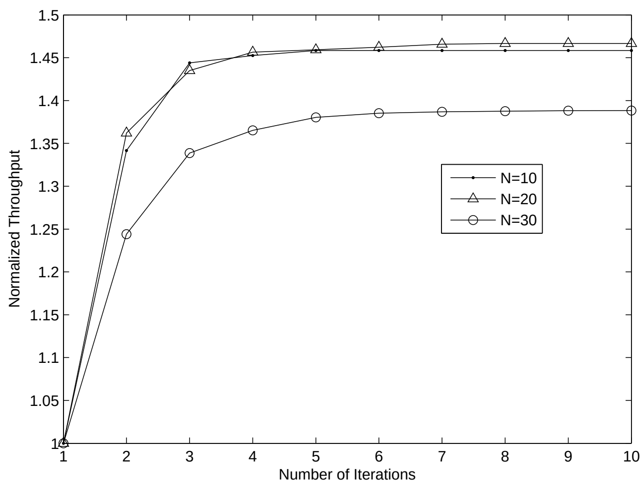

Fig. 5. Normalized throughput of T-CAR with respect to MER versus number of iterations for $\varphi = 1$ and $c = 1000$.

The convergence of the T-CAR can be observed from Fig. 5. The percentage improvements in terms of throughput of the T-CAR algorithm converge in $\kappa = 6$ to 9 iterations for network sizes of $N = 10$ to 30. The existence of Nash equilibrium in T-CAR are supported by the convergence of curves from Fig. 5. Furthermore, we can see that T-CAR algorithm involve small number of iterations after MER is established, and it can be implemented efficiently in a distributed fashion. Similar convergence results may also be shown for T-ICAR algorithms after IAR is established.

VI. Conclusions and Future Work

In this paper, we addressed the problem of efficient transmission structure in wireless sensor networks where each source transmits and aggregates the correlated data over intermediate nodes to the sink. We have investigated the impact of interference, as well as efficient data aggregation in establishing routing paths towards the sink for throughput maximization problem. For throughput-maximizing correlation aware routing, we have proposed a distributed iterative protocol based on a game theoretic framework, which is shown to converge within a couple of iterations. We have also shown that, by accounting for both correlation structure and interference impact in constructing routes, significant throughput gains over classic approaches can be achieved. Although we have looked into the total throughput improvements in the network, the advantages of proposed algorithms can be incorporated into other metrics like decreasing the load over overwhelmed bottleneck nodes to increase the network lifetime or to decrease the delay in the network.

References

[1] V. Rodoplu and T. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1443, 1999.

[2] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-efficient broadcast and multicast trees in wireless networks," *Mob. Netw. Appl.*, vol. 7, no. 6, pp. 481–492, 2002.

[3] H. Mahmood and C. Comaniciu, "Interference aware cooperative routing for wireless ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 1, pp. 248–263, 2009.

[4] E. Zeydan, D. Kivanc, and C. Comaniciu, "Efficient routing for correlated data in wireless sensor networks," in *IEEE MILCOM'08*, November, 2008.

[5] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," in *ACM Wireless Networks*, vol. 11, pp. 149–160, 2005.

[6] H. Luo, Y. Liu, and S. K. Das, "Routing correlated data with fusion cost in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, pp. 1620–1632, November 2006.

[7] R. Cristescu, B. B. Lozano, and M. Vetterli, "Networked slepian-wolf theory: Theory, algorithms and scaling laws," *IEEE Transactions on Information Theory*, vol. 51, pp. 4057–4073, December 2005.

[8] C. Hua and T. P. Yum, "Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, pp. 892–903, August 2008.

[9] R. Madan and S. Lall, "Distibuted algorithms for maximum lifetime routing in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, pp. 2185–2193, August 2006.

[10] R. Cristescu, B. B. Lozano, M. Vetterli, and R. Wattenhofer, "Network correlated data gathering with explicit communication: NP completeness and algorithms," *IEEE/ACM Transactions on Networking*, vol. 14, pp. 41–54, February 2006.

[11] D. Monderer and L. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, pp. 124–143, May 1996.

[12] R. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *International Journal of Game Theory*, vol. 2, pp. 65–67, 1973.

[13] D. Bertsekas and R. Gallager, *Data Networks*. Upper Saddle River, NJ: Prentica Hall, 1992.