On Energy Optimized Averaging in Wireless Sensor Networks

Binbin Li and Ioannis Ch. Paschalidis

Division of Systems Engineering and Department of Electrical and Computer Engineering

Abstract

Wireless Sensor NETworks (WSNETs) have found their way into a wide variety of applications and systems. They have also motivated the rekindling of interest into distributed averaging/consensus algorithms. Meanwhile, power consumption in WSNETs is a key issue and has received significant attention in recent research. In this work, we study the energy costs of running distributed averaging/consensus algorithms in WSNETs in static and dynamic topologies.

First, we consider the static, time-invariant networks. It has recently been shown that running a synchronous agreement algorithm over a bidirectional spanning tree is preferable in terms of convergence time. We formulate the combinatorial optimization problem of selecting a minimal energy bidirectional spanning tree as a mixed integer linear programming problem. This problem has been shown to be NP-complete and can only be solved for small instances. We devise a semi-definite relaxation and establish bounds on the optimal cost. We also develop a series of graph-based algorithms that yield energy efficient bidirectional spanning trees and establish associated bounds on the optimal cost. Some of our algorithms can be run distributedly and numerical results establish that they produce near-optimal solutions.

Second, we deal with the dynamic, time-varying networks. Recent work indicates that an asynchronous load-balancing algorithm can guarantee convergence in polynomial time, given a symmetric communication condition and bounded interconnectivity times. We formulate the problem of selecting a minimal energy interconnected network as a sequential decision problem and cast into a Dynamic Programming (DP) framework. This problem is hard to solve when incurring a penalty cost for not reaching interconnectivity within a pre-determined block of time. We first consider the scenario of a large enough time horizon and show that solving DP is equivalent to constructing a Minimum Spanning Tree (MST), which can be done in a distributed manner. We then consider the scenario of a limited time horizon and employ a rollout heuristic that leverage the MST solution and yields efficient solutions for the original DP. Numerical experiments verify the correctness, effectiveness and efficiency of our proposed algorithm.

Consensus and Averaging

- **Distributed Consensus/Averaging in Wireless Sensor Networks**

$$x_i(t+1) = \sum_{j=1}^{N} a_{ij} x_j(t), \qquad i = 1, \ldots, N$$

- Key issues: topology design and weight set
- Convergence guarantee: Bounded Interconnectivity Times

$\exists B, \forall k, (\mathcal{N}, \mathcal{E}(kB) \cup \mathcal{E}(kB+1) \cup \cdots \cup \mathcal{E}((k+1)B))$ is strongly connected.

- **State-Of-The-Art Result**
  - Static Networks
    - equal-neighbor bidirectional spanning tree with convergence time

$$T_N(\epsilon) = O(N^2 \log(N/\epsilon))$$

  - Dynamic Networks
    - load-balancing algorithm with worst-case convergence time

$$T_n(B, \epsilon) \le cBn^3 \log \frac{1}{\epsilon}$$

Part I: Static Networks

Problem Formulation

- **Network Model**
  - transmit power requirements: e.g., in a path loss model $P_{ij} = d_{ij}^{\alpha}$, where d_{ij} is Euclidean distance and α is the channel loss exponent.
  - bidirectional links
  - broadcasting benefits
- **Objective: minimize the total transmit energy consumption while maintaining a bidirectional spanning tree**
- **Mixed Integer Linear Programming based on flow conservation**

$$C^{MILP} = \min \sum_{i=1}^{N} Y_i$$

s.t.

$$\sum_{\{j|(1,j)\in\mathscr{E}\}} F_{1j} = N-1, \quad \sum_{\{j|(1,j)\in\mathscr{E}\}} F_{j1} = 0,$$
$$\sum_{\substack{\{j|(i,j)\in\mathscr{E} \\ \text{or } (j,i)\in\mathscr{E}\}}} F_{ji} - \sum_{\substack{\{j|(i,j)\in\mathscr{E} \\ \text{or } (j,i)\in\mathscr{E}\}}} F_{ij} = 1, \ \forall i \in \mathscr{N} \setminus \{1\},$$

Strong connectivity guaranteed by flow conservation

$$(N-1) \cdot X_{ij} - F_{ij} \ge 0, \quad \forall (i,j) \in \mathscr{E},$$
$$(N-1) \cdot X_{ij} - F_{ji} \ge 0, \quad \forall (i,j) \in \mathscr{E},$$

Bidirectional link selection

$$Y_i - X_{ij} P_{ij} \ge 0, \quad \forall (i,j) \in \mathscr{E},$$
$$Y_j - X_{ij} P_{ji} \ge 0, \quad \forall (i,j) \in \mathscr{E},$$

Transmit power requirement

$$\sum_{(i,j)\in\mathscr{E}} X_{ij} = N-1,$$

Redundant link trimming and tree guarantee

$$X_{ij} \in \{0,1\}, \quad \forall (i,j) \in \mathscr{E},$$
$$F_{ij}, F_{ji} \ge 0, \quad \forall (i,j) \in \mathscr{E},$$
$$Y_i \ge 0, \quad \forall i.$$

where $\mathscr{E} = \{(i,j) \mid i,j \in \mathscr{N}, i<j, P_{ij} \le Y_i^{max}, P_{ji} \le Y_j^{max}\}$ contains all possible links.

A Centralized Algorithm Based on Semi-Definite Relaxation

- **Semi-Definite Programming**

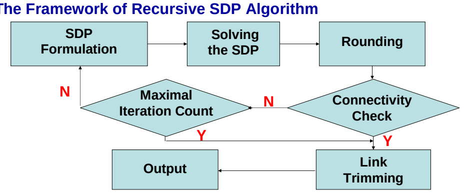
$$\begin{aligned} \max \quad & \mathbf{M}_0 \bullet \mathbf{Z} \\ \text{s.t.} \quad & \mathbf{M}_i \bullet \mathbf{Z} = c_i, \quad i = 1, \ldots, n, \\ & \mathbf{Z} \succeq 0, \end{aligned}$$

where $\mathbf{Z}$ is a symmetric matrix and "$\succeq 0$" denotes semi-definiteness.

- **Recursive SDP from MILP**
  - Relaxation $\quad X_{ij} \in \{0,1\} \longrightarrow X_{ij}^2 - X_{ij} = 0$
  - Sampling and Rounding

$$\mathbf{X} = \begin{bmatrix} \mathbf{R_x} & \mathbf{m_x} \\ \mathbf{m_x'} & 1 \end{bmatrix}, \qquad \mathbf{\Sigma_x} = \mathbf{R_x} - \mathbf{m_x m_x'} \succeq 0$$

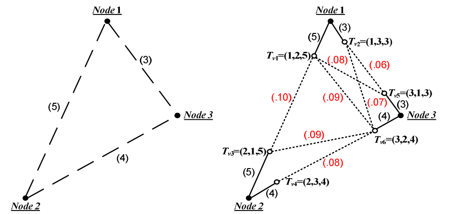
- **The Framework of Recursive SDP Algorithm**



•Bounds $\quad C^{SDP} \le C^{MILP} \le C^{RSDP}$

Distributed Graph-Based Algorithms

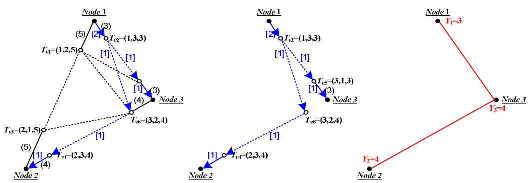
- **Augmented Graph Construction**
  - adding artificial nodes, corresponding to different power levels the real nodes may work on
  - generating arcs, exploiting all the possible topologies of the network
  - assigning weights, transferring energy costs from nodes to arcs



Denote the corresponding undirected and directed augmented graphs as $\mathscr{G}_a$ and $\vec{\mathscr{G}}_a$, respectively.

- **Minimum Cost Flow Based Algorithm**

•Working on $\vec{\mathscr{G}}_a$
  - Obtaining MILP feasible solution based on MCF solution

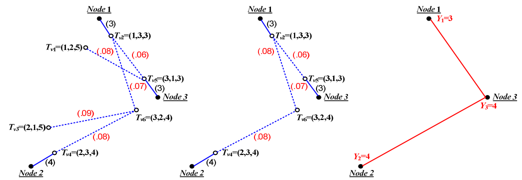


- Bounds

$$\frac{1}{2(N-1)} \sum_{s=1}^{N} C_s^{MCF} \le C^{MILP} \le \min_s C_{BST,s}^{MCF}$$

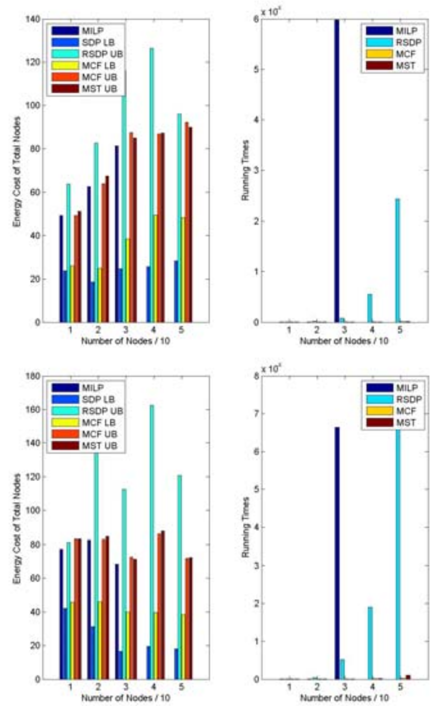
- **A distributed approach based on shortest paths**
  - One-source-multi-sink min cost flow problem is equivalent to $(N-1)$ shortest path problem
  - Solve shortest path problem distributedly via Bellman-Ford algorithm

- **A distributed approach based on minimum spanning trees**
  - working on $\mathscr{G}_a$
  - Apply distributed minimum spanning tree algorithm (*Gallager, Humblet and Spira '83*) and recursively trim leaf artificial nodes



Numerical Results

- **Sparse vs. Dense Network**



Part II: Dynamic Networks

Problem Formulation

- **Network Model**

link trials until success follows (memoryless property)

$$P(Y_k = y) = \begin{cases} \frac{p_k(1-p_k)^{y-1}}{1-(1-p_k)^{M_k}}, & y = 1, 2, \ldots, M_k; \\ 0, & \text{otherwise.} \end{cases}$$

- **A Dynamic Programming Formulation**
  - Network State

$\mathscr{S}(t) = (\mathscr{G}_c(t), \mathscr{L}(t))$, where $\mathscr{G}_c(t) = (\mathscr{N}, \mathscr{E}_c(t))$ (graph information) and $\mathscr{L}(t) = \{k(m_k)\}_{k=1, k \notin \mathscr{E}_c(t)}^{|\mathscr{A}|}$ (link information)

  - System dynamics $\mathscr{S}(t+1) = (\mathscr{G}_c(t+1), \mathscr{L}(t+1)) =$

$$\begin{cases} ((\mathscr{N}, \mathscr{E}_c(t) \cup \{k\}), \mathscr{L}(t) \setminus k(m_k)), & \text{w.p. } \frac{p_k}{1-(1-p_k)^{M_k - m_k}}, \\ (\mathscr{G}_c(t), \mathscr{L}(t) \setminus k(m_k) \cup k(m_k+1)), & \text{w.p. } 1 - \frac{p_k}{1-(1-p_k)^{M_k - m_k}}. \end{cases}$$

  - DP iteration $J_t(\mathscr{S}(t)) = \min_{k \in \{\mathscr{A}, \emptyset\},\ k \notin \mathscr{E}_c(t)} E\left[c_k + J_{t+1}(\mathscr{S}(t+1))\right]$
  - Terminal cost

$$J_B(\mathscr{S}(B)) = \begin{cases} W \gg 1, & \text{if } \mathscr{G}_c(B) \text{ is not strongly connected,} \\ 0, & \text{otherwise.} \end{cases}$$

  - Monotonicity Property: It holds that $J_t(\mathscr{S}^{\alpha}(t)) \ge J_t(\mathscr{S}^{\beta}(t))$ for all t and $\mathscr{S}^{\alpha}(t) = (\mathscr{G}_c^{\alpha}(t), \mathscr{L}^{\alpha}(t))$, $\mathscr{S}^{\beta}(t) = (\mathscr{G}_c^{\beta}(t), \mathscr{L}^{\beta}(t))$ such that $\mathscr{G}_c^{\alpha}(t) \subseteq \mathscr{G}_c^{\beta}(t)$ and $\mathscr{L}^{\alpha}(t)$ coincides with $\mathscr{L}^{\beta}(t)$ for all links $k \notin \mathscr{E}_c^{\beta}(t)$

Large Enough Horizon Length

- **No terminal cost:** $J_B(\mathscr{S}(B)) = 0$
  - Monotonicity of fail trials: Suppose we are at some state $\mathscr{S}^{\alpha} = (\mathscr{G}_c^{\alpha}, \mathscr{L}^{\alpha})$ and link $k \notin \mathscr{E}_c^{\alpha}$. Assume that there is a positive probability that link k participates in the connected graph at the end of the horizon, i.e., $k \in \mathscr{E}_c^{\alpha}$ at time B. Consider some other state $\mathscr{S}^{\beta} = (\mathscr{G}_c^{\beta}, \mathscr{L}^{\beta})$ such that $\mathscr{G}_c^{\alpha} = \mathscr{G}_c^{\beta}$ and $\mathscr{L}^{\beta} = \mathscr{L}^{\alpha} \setminus k(m_k) \cup k(m_k+1)$. Then, $J(\mathscr{S}^{\alpha}) > J(\mathscr{S}^{\beta})$.
  - MST Algorithm is optimal: For every link k present in $\mathscr{A}$, assign $c_k E[Y_k]$ as its weight and compute the Minimum Spanning Tree.
  - *Minimum expected interconnectivity times* vs. *Minimum expected cost*
- **Numerical Results**

	MST-based Algorithm		DP Algorithm	
n	Running Time	Result	Running Time	Result
3	< 1 sec	47.99	< 1 sec	47.99
4	< 1 sec	55.93	10.01 secs	55.93
5	< 1 sec	127.84	2.45×10^3 secs	127.84
6	< 1 sec	149.42	6.21×10^4 secs	NA

Limited Horizon Length

- Rollout Algorithm: employ a given heuristic in the construction of an optimal cost-to-go function approximation, which is then used in the spirit of reinforcement learning methodology.

$$l = \arg \min_{k \in \{\mathscr{A}, \emptyset\},\ k \notin \mathscr{E}_c(t)} E\left[c_k + H_{t+1}(\mathscr{S}(t+1))\right]$$

  - Cost-to-go function approximation:
    - Inter connectivity cost: approximated by MST and the associated cost
    - Penalty cost: calculated based on MST selection and z-transform

$$H_{t+1}^{cost}(\mathscr{S}(t+1)) = \begin{cases} 0, & \text{if } \mathscr{G}_c(t+1) \text{ is connected,} \\ W, & \text{if } t+1 = B \text{ and } \mathscr{G}_c(B) \text{ is not connected,} \\ \tilde{J}_{cost}(\mathscr{S}(t+1)) + W p_{t+1}^{F}, & \text{otherwise.} \end{cases}$$

- **Numerical Results**
  - Correctness

	DP Algorithm		Rollout Algorithm		
Horizon Length	Optimal Cost-to-go	RT	Rollout AC	RT	SC
3 Node Case					
5	166.6921	0.2272 secs	1628.1	0.1769 sec	49
10	46.6824	0.4397 secs	48.3993	0.1921 sec	50
15	46.6787	0.6513 secs	47.2706	0.1959 sec	50
4 Node Case					
5	2792.5	341.4915 secs	6390.7	0.7886 sec	46
10	87.5363	668.9488 secs	89.2265	0.7556 sec	50
15	84.1002	994.1547 secs	86.0340	0.7723 sec	50

  - Effectiveness

B	25			30			35		
n	SC	AS	AE	SC	AS	AE	SC	AS	AE
10	10	11.1	1139.1	10	10.8	977.8	10	12.4	747.6
12	10	11.8	1033.1	10	11.0	888.7	10	12.0	603.7
14	10	16.7	1354.9	10	15.5	1317.3	10	14.8	871.1
16	10	18.0	1717.6	10	16.3	1642.5	10	16.4	1448.0

  - Efficiency

n	Link Density	Decision Time (sec)
10	68 / 90	0.32
20	242 / 380	2.97
30	474 / 870	11.66
40	928 / 1560	50.01
50	1296 / 2450	104.21
60	2276 / 3540	447.75

Acknowledgement

We acknowledge the NSF and the DOE for partial support.

References

- Ioannis Ch. Paschalidis and Binbin Li, "*On Energy Optimized Averaging in Wireless Sensor Networks*", Invited, Proceedings of the 48th IEEE Conference on Decision and Control, pages 3763-3768, December 2009, Shanghai, China.
- Ioannis Ch. Paschalidis and Binbin Li, "*On Energy Optimized Averaging in Wireless Sensor Networks in Dynamic Topologies*", submitted to Proceedings of the 49th IEEE Conference on Decision and Control.