DAO²: Overcoming Overall Storage Overflow in Intermittently Connected Sensor Networks

Bin Tang

Department of Computer Science
California State University Dominguez Hills
Outline

- ICSN and its overall storage overflow problem
- DAO\(^2\) (Data Aggregation for Overall Storage Overflow)
- MTSW (Multiple Traveling Salesman Walks)
- Distributed DAO\(^2\) algorithm
- Simulation results
- Conclusion and future work
Intermittently Connected Sensor Networks (ICSN)

- Deepwater sensor networks for tactical surveillance, volcano eruption/glacial melting monitoring
- Not feasible to install base station in field
- Data generated and stored in the network, periodically uploaded via data mules or satellite links
- Data uploading opportunities are intermittently available

Source: http://fiji.eecs.harvard.edu/Volcano
Data Preservation in ICSN

- Non-uniform data generation and limited storage capacity
- **Data nodes**
  - Storage-depleted
  - Overflow data
- **Storage nodes**
  - Available storage spaces
- **Data preservation**: offload overflow data from data node to storage nodes
Overall Storage Overflow In Data Preservation

- When total size of overflow data exceeds the total available storage in the network
Solution - DAO\textsuperscript{2} (Data Aggregation for Overall Storage Overflow)

- Spatial correlation among sensory data
- Two stages: 1) data aggregation, 2) data offloading

- Challenge: energy efficient data aggregation
Problem Formulation of DAO²

- Undirected weighted graph $G(V, E)$
- $p$ data nodes, each has $R$ bits of overflow data
  - when a data node receives data information from at least another data node, it becomes an aggregator, and
  - it reduces the size of its overflow data from $R$ to $r$, $0 \leq r < R$
- $|V| - p$ storage nodes, each has $m$ bits of storage space
- Due to overall storage overflow: $p \times R > (|V| - p) \times m$

$$\frac{|V|m}{m + R} < p \leq \left\lfloor \frac{|V|m - R + r}{m + r} \right\rfloor$$
Problem Formulation of DАО²

- **q**: number of aggregators,

\[ q = \left\lceil \frac{p \times (R + m) - |V| \times m}{R - r} \right\rceil \]

- Therefore, at most \( p-q \) sensor nodes can be initiators

- Energy model - \( u \) sends \( R \)-bit data to \( v \) over \( l_{u,v} \)

\[ E_t(R, l_{u,v}) = E_{elec} \times \hat{R} + \epsilon_{amp} \times R \times l_{u,v}^2 \]
\[ E_r(R) = E_{elec} \times R \]

- **Goal**: select a set of \( b \) (\( 1 \leq b \leq p-q \)) initiators, each visiting some data nodes (aggregators) following a walk, s.t. total \( q \) distinct aggregators are visited with minimum total energy cost
An Example of DAO$^2$

- Data nodes: $B, D, E, G, I$
- Storage nodes: $A, C, F, H$
- $R = m = 1$ → overall storage overflow!
- $r = \frac{3}{4}$, number of aggregators $q = 4$, leaving one data node to be initiator
- Edge weight = 1
- Decide: which data node to be initiator and which path it visits?
Two Optimal Solutions: cost=5

(a) B is initiator

(b) I is initiator
Next, data offloading...

Equivalent to minimum cost flow problem (Tang et al. TOSN 2013)
Multiple Traveling Salesman Walks (MTSW)

- **Input:** An undirected weighted graph $G(V,E)$ and a number $q$
- **Output:** Finds at most $|V| - q$ starting nodes, each visiting some other nodes following a walk, s.t.
  - **Goal:** total $q$ nodes are visited while the total cost of the walks is minimized
MTSW is NP-hard

- Traveling salesman path problem (TSPP) is a special case of MTSW, with $q = |V| - 1$
- Prove TSPP is NP-hard, reduced from traveling salesman problem (TSP)
Algorithm 1: Approximation Algorithm for MTSW.
Input: $G(V, E)$ and number of nodes to visit $q$;
Output: a walks: $W_1, W_2, ..., W_a$, and $\sum_{1 \leq j \leq a} c(W_j)$;

0. Notations:
   $E_q$: set of $q$ cycleless edges;
   $G[E_q]$: a $q$-edge forest;
   $C(G[E_q])$: set of connected components in $G[E_q]$;
   $C_j$: the $j^{th}$ connected component in $C(G[E_q])$;

1. Let $w(e_1) \leq w(e_2) \leq ... \leq w(e_{|E|})$;
2. $E_q = \emptyset$ (empty set), $i = j = k = 1$;
3. while ($k \leq q$)
   4. if ($e_i$ is a cycleless edge w.r.t. $E_q$)
      5. $E_q = E_q \cup \{e_i\}$;
      6. $k += 1$;
   7. end if;
   8. $i += 1$;
9. end while;
10. Let $|C(G[E_q])| = a$; /* Connected components */
11. for ($1 \leq j \leq a$)
12. if ($C_j$ is linear) Start from one end node of $C_j$ and
    visit the rest nodes in $C_j$ once;
13. if ($C_j$ is a tree) Do a B-walk on $C_j$;
14. Let the resulted walk (or path) be $W_j$;
15. end for;
16. RETURN $W_1, W_2, ..., W_a$, and $\sum_{1 \leq j \leq a} c(W_j)$.

- Find the $q$ smallest cycleless edges, referred to as $q$-edge forest
- Traverse each tree in the forest using a B-walk
- Works alike and generalizes Kruskal’s MST algo.
- $O(|E| \log |E|)$
Binary Walk (B-Walk) and Longest-Path Walk (LP-Walk)

\[ c(W) \leq \left( 2 - \frac{1}{|T|} \right) \times c(T) \]
An Example of Algo. 1 (q=5)

- Edge weight = 1
- How to visit 5 nodes energy efficiently?
Many solutions...

- B, D are starting nodes
- C, F, E, H, I are visited
- Cost = 5

- E is starting node
- B, C, D, H, I are visited
- Cost = 8
Analysis of Algo. 1

- **Lemma 1:** Its resulted $q$-edge forest is a *minimum* $q$-edge forest

- **Lemma 2:** The cost of the minimum $q$-edge forest is a lower bound of the optimal MTSW cost

- **Theorem:** Algorithm 1 is a $(2-1/q)$ approximation algorithm
Equivalency b/t MTSW and DAO²

**Theorem:** DAO² in sensor network $G(V,E)$ is equivalent to MTSW in aggregation network $G'(V',E')$
Distributed DAO² Algorithm

Based on distributed minimum spanning tree algorithm by Gallager, Humblet, and Spira

1. Starts with each node being considered as a fragment, with level value 0

2. Each level 0 node
   1) Chooses its minimum-weight incident edge and mark that edge as a branch edge.
   2) Sends a message via the branch edge to notify the node on the other side.
   3) Waits for a message from the other end of the edge.

3. The edge chosen by both nodes it connects becomes the core with level 1.

4. For a non-zero level fragment, the execution takes three stages: broadcast, convergecast, and change core

5. while (number of branch edges < q)
   1) Each fragment finds its minimum weight outgoing edge
   2) Uses it to combine with other fragments, using two operations: merge and absorb

end while

It runs in O(NlogN) time and uses O(NlogN+E) messages.
Performance Evaluation

- Visual Performance Comparison
  - 50 nodes in 1000m×1000m network, Tr = 250m
  - R=m=512MB, ρ = 1 - r/R

Fig. 6. Valid range of p while varying ρ (R = m).
One initiator

(a) Sensor network graph. (b) Aggregation graph. (c) B-Walk (cost=381.2J). (d) LP-Walk (cost=290.6J).

Four initiators

(a) Sensor network graph. (b) Aggregation graph. (c) B-Walk (cost=255.9J). (d) LP-Walk (cost=203.0J).
Comparing B-Walk and LP-Walk

(a) Total aggregation cost (KJ).  
(b) Performance improvement.

Fig. 9. Comparing B-Walk with LP-Walk by varying $p$ and $\rho$. 
Distributed DAO² Algorithm

- Implemented in DistAlgo (Liu et al. OOPSLA 2012)
- 100 nodes in 2000m × 2000m sensor network, Tr = 250m
- R = m = 512MB, overhead message = 20B
Conclusions and Future Works

- DAO\(^2\) is an architectural and algorithmic framework to tackle overall storage overflow
- A new multiple traveling salesman walk problem
- Energy-efficient optimal, approximation, heuristic, and distributed algorithm
- Techniques applicable for any application where data correlation and resource constraints coexist
- Varying overflow data size and storage capacity
- Integrate data aggregation and data offloading