When In-Network Processing Meets Time: Complexity and Effects of Joint Optimization in Wireless Sensor Networks

Qiao Xiang*, Jinhong Xu†, Xiaohui Liu*, Hongwei Zhang*, Loren J. Rittle‡

*Department of Computer Science, Wayne State University
†Department of Computer Science, Indiana University
‡Applied Research and Technology Center, Motorola
Introduction

- Wireless Sensor Networks
  - Highly resource-constrained

- In-Network Processing
  - Reduce traffic flow $\rightarrow$ resource efficient
  - End-to-end QoS are usually not considered

- Mission-Critical Real-Time CPS:
  - Close-loop control
  - More emphasis on end-to-end QoS, especially latency and reliability
Introduction

- Packet packing
  - Application independent INP
  - Simple yet useful INP in practice
    - UWB intra-vehicle control
    - IETF 6LowPAN: high header overhead

- Our focus:
  - Understanding problem complexity
  - Designing simple distributed online algorithm
  - Understanding systems benefits
Outline

- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion
System Model and Problem Formulation

- **System Model**
  - A directed collection tree \( T = (V,E) \)
  - Edge \((v_i, v_j) \in E\) with weight \( ETX_{v_i, v_j}(l) \)
  - A set of information elements \( X = \{x\} \)
  - Each element \( x \): \((v_x, l_x, r_x, d_x)\)

- **Problem (P):**
  - Schedule the transmission of \( X \) to \( R \)
  - Minimize the total number of transmissions
  - Satisfy the latency constraints of each \( x \in X \)
Outline

- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion
Complexity Analysis

- **Problem $P_0$**
  - Elements are of equal length
  - Each node has at most one element

- **Problem $P_1$**
  - Elements are of equal length
  - Each node generates elements periodically

- **Problem $P_2$**
  - Elements are of equal length
  - Arbitrary data generating pattern
Complexity Analysis

<table>
<thead>
<tr>
<th>$P_0, P_1, P_2, P$</th>
<th>$K \geq 3$</th>
<th>$K = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>strong NP-hard</td>
<td>strong NP-hard</td>
</tr>
<tr>
<td>Complexity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NP-hard to achieve approximation ratio</td>
<td>$1 + \frac{1}{200N}(1 - \frac{1}{\varepsilon})$</td>
<td>$1 + \frac{1}{120N}(1 - \frac{1}{\varepsilon})$</td>
</tr>
</tbody>
</table>

$K = \text{Maximal packet length}$  
$N = |X|$  
Re-aggregation: a packed packet can be dispatched for further packing.
K ≥ 3, \( P_0 \) is NP-hard in tree structures -- Reduction from SAT

Given a SAT instance with \( n \) clauses and \( m \) variables

For each clause \( j \)
For each variable occurred in clause $j$

Auxiliary elements related to the red ones
Complexity Analysis

- When $K \geq 3$ and $T$ is a tree, regardless of re-aggregation
  - $P_0$ is NP-hard $\rightarrow P_1$ is NP-hard $\rightarrow P_2$ is NP-hard $\rightarrow P$ is NP-hard

- When $K \geq 3$, and $T$ is a chain, regardless of re-aggregation
  - The reduction from SAT still holds*

- When $K = 2$ and re-aggregation is not prohibited
  - The reduction from SAT still holds in both tree and chain structures

- When $K = 2$ and re-aggregation is prohibited
  - Problem $P$ is equivalent to the maximum weighted matching problem in an interval graph.
  - Solvable in $O(N^3)$ by Edmonds’ Algorithm

* This solves an open problem in batch processing
Outline

- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion
When a node receives a packet $\text{pkt}$ with length $s_f$

- Decisions: to hold or to transmit immediately
- Utility of action: Reduced Amortized Cost
- One-hop locality

$$\text{AC} = \frac{\text{# of TX}}{\text{length of data}}$$
A Utility Based Online Algorithm

- Utility of holding a packet:
  \[ AC'_l = \frac{1}{L - s'_f} ETX_{jR}(L - s'_f) \]
  \[ AC_l = \frac{1}{L - s'_f + S_l} ETX_{jR}(L - s'_f + S_l) \]
  
  Cost without packing
  \[ U_l = AC'_l - AC_l \]

- Utility of transmitting a packet:
  \[ U'_p = \frac{t'_f}{t_p} \frac{ETX_{p_jR}(s_p)}{s_p} - \frac{t'_f}{t_p} \frac{ETX_{p_jR}(L)}{L} \]
  \[ U''_p = \frac{\frac{L-s'_f}{L-s_p} ETX_{p_jR}(s_p)}{\frac{L-s'_f}{L-s_p}} - \frac{\frac{L-s'_f}{L-s_p} ETX_{p_jR}(L)+I_{mod}ETX_{p_jR}(s_p+l_{mod})}{\frac{L-s'_f}{L-s_p}s_p+L-s'_f} \]

  \[ U_p = \begin{cases} 
  U'_p & \text{if } \frac{t'_f}{t_p}(L - s_p) \leq L - s'_f \\
  U''_p & \text{otherwise} 
\end{cases} \]

Every packet received by parent can get fully packed via \textbf{pkt}.
A Utility Based Online Algorithm

- **Decision Rule**
  - The packet should be immediately transmitted if $U_p > U_l$
  - The packet should be held if $U_p \leq U_l$

- **Competitive Ratio**
  - Problem $P'$
    - T is a complete tree
    - Leaf nodes generate elements at a common rate $R$
  - Theorem: For problem $P'$, tPack is $\min\{K, \max_{v_j \in V_{>1}} \frac{ETX_{v_j R}}{ETX_{p_j R}}\}$ -competitive, where $K$ is the maximum number of information elements that can be packed into a single packet, $V_{>1}$ is the set of nodes that are at least two hops away from the sink $R$.
  - Example: When ETX is the same for each link, tPack is 2-comptetive
Outline

- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion
Performance Evaluation

Experiment Setting Up

- Testbed: NetEye, a 130-sensor testbed
- Topology: 120 nodes, half are source nodes
- Protocols compared: noPacking, simplePacking, tPack
- Traffic patterns: 6 second periodic traffic and event traffic

Metrics:
- packing ratio
- delivery reliability
- delivery cost
- latency jitter
Packing Ratio

The diagram illustrates the packing ratio for different maximum allowable latencies (L1, L5, L9, L14) across three different packing methods: noPack, simplePack, and tPack. The packing ratio is represented on the y-axis, while the maximum allowable latency is on the x-axis. The error bars indicate the variability around the mean for each method at each latency level.
Delivery Reliability

The graph shows the delivery reliability for different maximum allowable latencies with three different packing methods: noPack, simplePack, and tPack. The reliability is higher for tPack compared to noPack and simplePack across all latency levels. The error bars indicate the variability in the data.
Delivery Cost

![Delivery Cost Chart]

- **noPack**
- **simplePack**
- **tPack**

**Maximum allowable latency vs Delivery cost**

- L1
- L5
- L9
- L14
Latency Jitter

![Graph showing latency jitter across different maximum allowable latencies (L1, L5, L9, L14). The graph compares latency jitter for different packet types: noPack, simplePack, and tPack.]
Outline

- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion
Conclusion and Future Work

- **Conclusion**
  - Impact of INP constraints on problem complexity
  - Feasibility of a simple, distributed online algorithm
  - Systems benefits in terms of efficiency and predictable latency

- **Future Work**
  - Complete competitive analysis on the utility based algorithm
  - Joint optimization of other INP and QoS constraints in WCPS