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 → 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 $ETX_{vi, vj}(l)$
  - 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

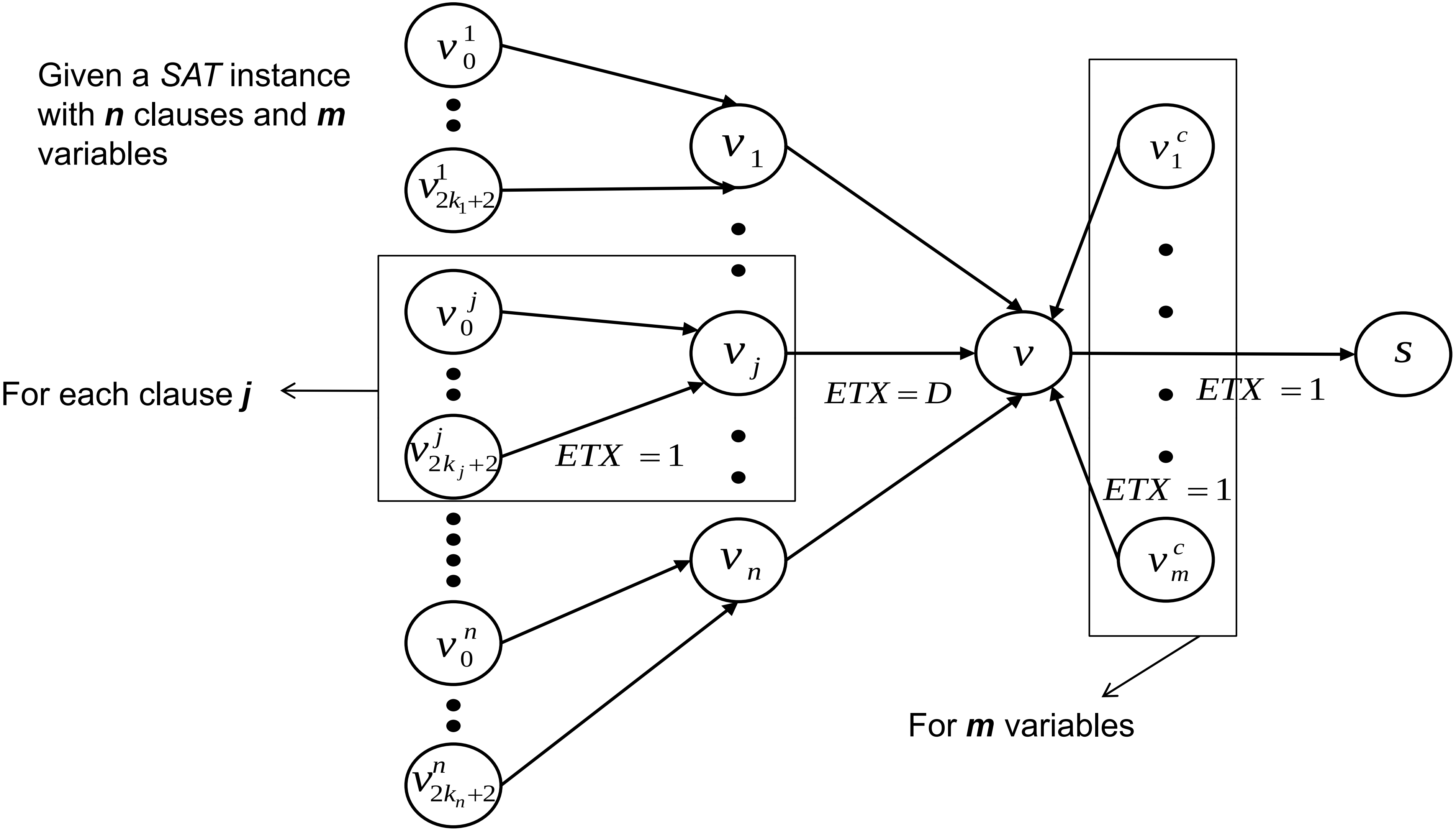
P_0, P_1, P_2, P	$K \geq 3$	K = 2	
		re-aggregation is not prohibited	re-aggregation is prohibited
Complexity	strong NP-hard	strong NP-hard	$O(N^3)$
NP-hard to achieve approximation ratio	$1+\frac{1}{200N}(1-\frac{1}{\varepsilon})$	$1+\frac{1}{120N}(1-\frac{1}{\varepsilon})$	

K = Maximal packet length $N = \|X\|$

Re-aggregation: a packed packet can be dispatched for further packing.

Complexity Analysis

- K ≥ 3, P_0 is NP-hard in tree structures -- Reduction from *SAT*

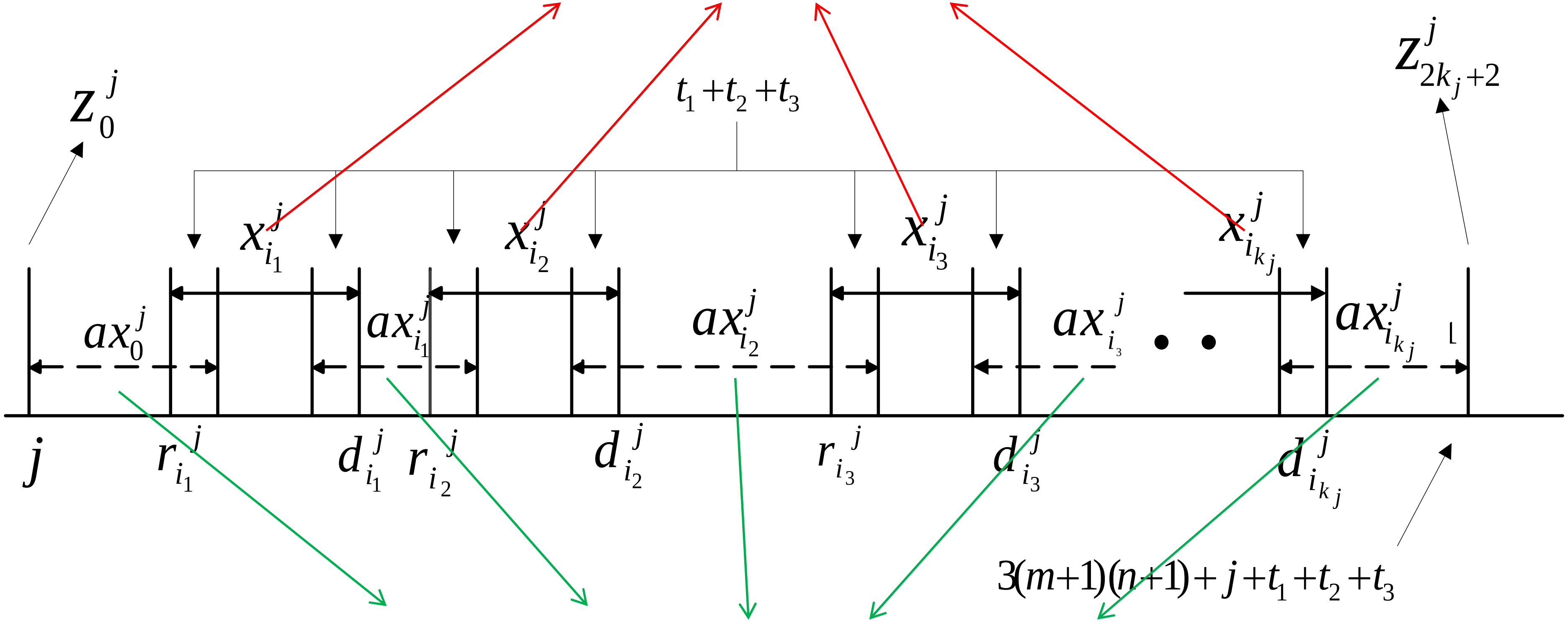


Given a *SAT* instance with ***n*** clauses and ***m*** variables

For each clause ***j***

For ***m*** variables

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 ≥ 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

A Utility Based Online Algorithm

- When a node receives a packet ***pkt*** with length s_f
  - Decisions: to hold or to transmit immediately
  - Utility of action: Reduced Amortized Cost
  - One-hop locality

$$\mathrm{AC} = \frac{\text{\# of TX}}{\text{length of data}}$$

A Utility Based Online Algorithm

- Utility of holding a packet:

Cost with packing

$$AC_l' = \frac{1}{L - s_f'} ETX_{jR}(L - s_f') \qquad AC_l = \frac{1}{L - s_f' + \mathcal{S}_l} ETX_{jR}(L - s_f' + \mathcal{S}_l)$$

Cost without packing

$$U_l = AC_l' - AC_l$$

- Utility of transmitting a packet:

$$\begin{aligned} U_p' &= \frac{\frac{t_f'}{t_p} ETX_{p_j R}(s_p)}{\frac{t_f'}{t_p} s_p} - \frac{\frac{t_f'}{t_p} ETX_{p_j R}(L)}{\frac{t_f'}{t_p} L} \\ &= \frac{ETX_{p_j R}(s_p)}{s_p} - \frac{ETX_{p_j R}(L)}{L} \end{aligned}$$

$$U_p'' = \frac{\lceil \frac{L - s_f'}{L - s_p} \rceil ETX_{p_j R}(s_p)}{\lceil \frac{L - s_f'}{L - s_p} \rceil s_p} - \frac{\lfloor \frac{L - s_f'}{L - s_p} \rfloor ETX_{p_j R}(L) + I_{mod} ETX_{p_j R}(s_p + l_{mod})}{\lceil \frac{L - s_f'}{L - s_p} \rceil s_p + L - s_f'}$$

$$U_p = \begin{cases} U_p' & \text{if } \frac{t_f'}{t_p}(L - s_p) \leq L - s_f' \\ U_p'' & \text{otherwise} \end{cases}$$

Every packet received by parent can get fully packed via ***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
  - Theorem: For problem *P'*, tPack is $\min\{K, \max_{v_j \in V_{>1}} \frac{ETX_{v_jR}}{ETX_{p_jR}}\}$

    -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

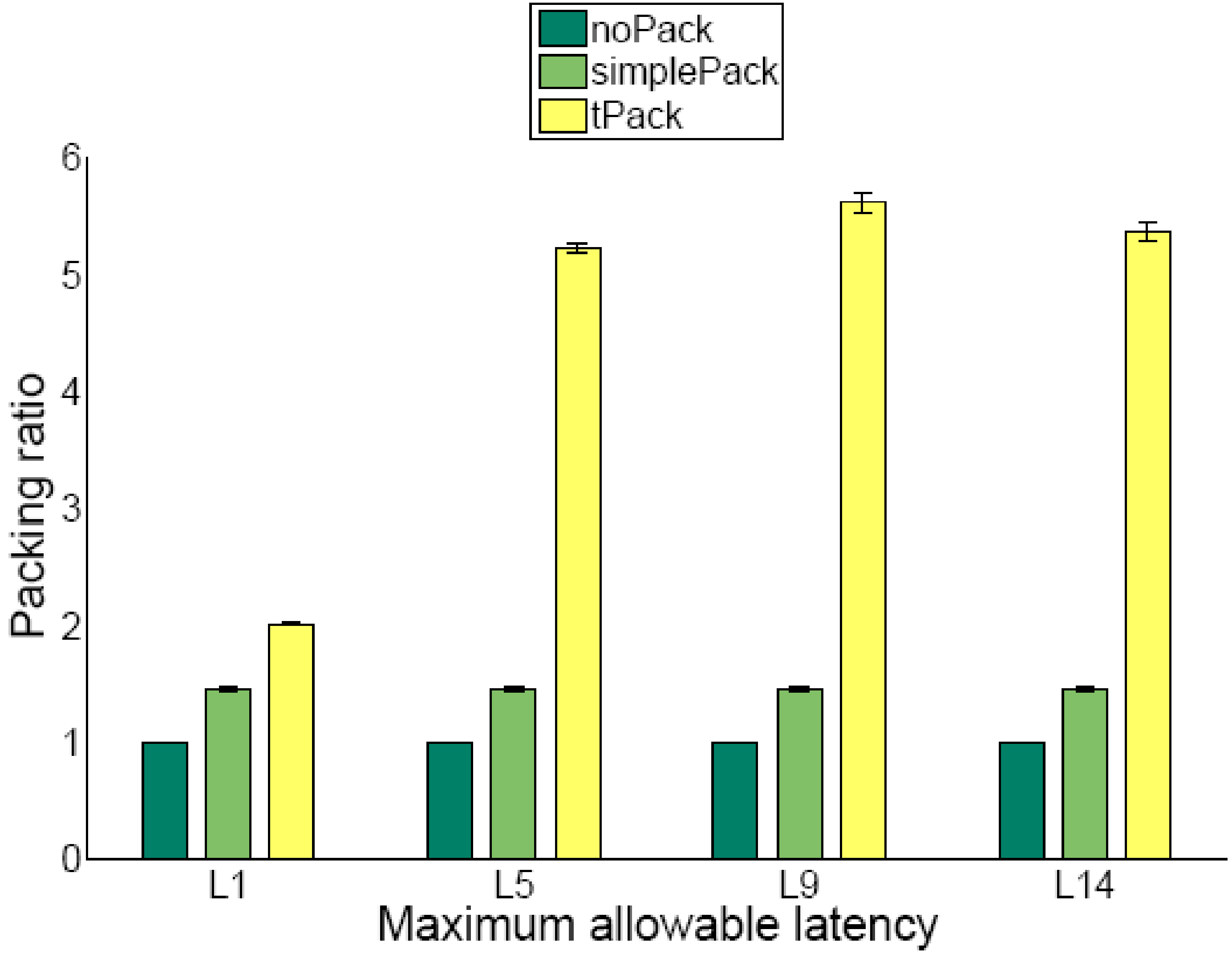
- System Model and Problem Formulation
- Complexity Analysis
- A Utility Based Online Algorithm
- Performance Evaluation
- Conclusion

Performance Evaluation

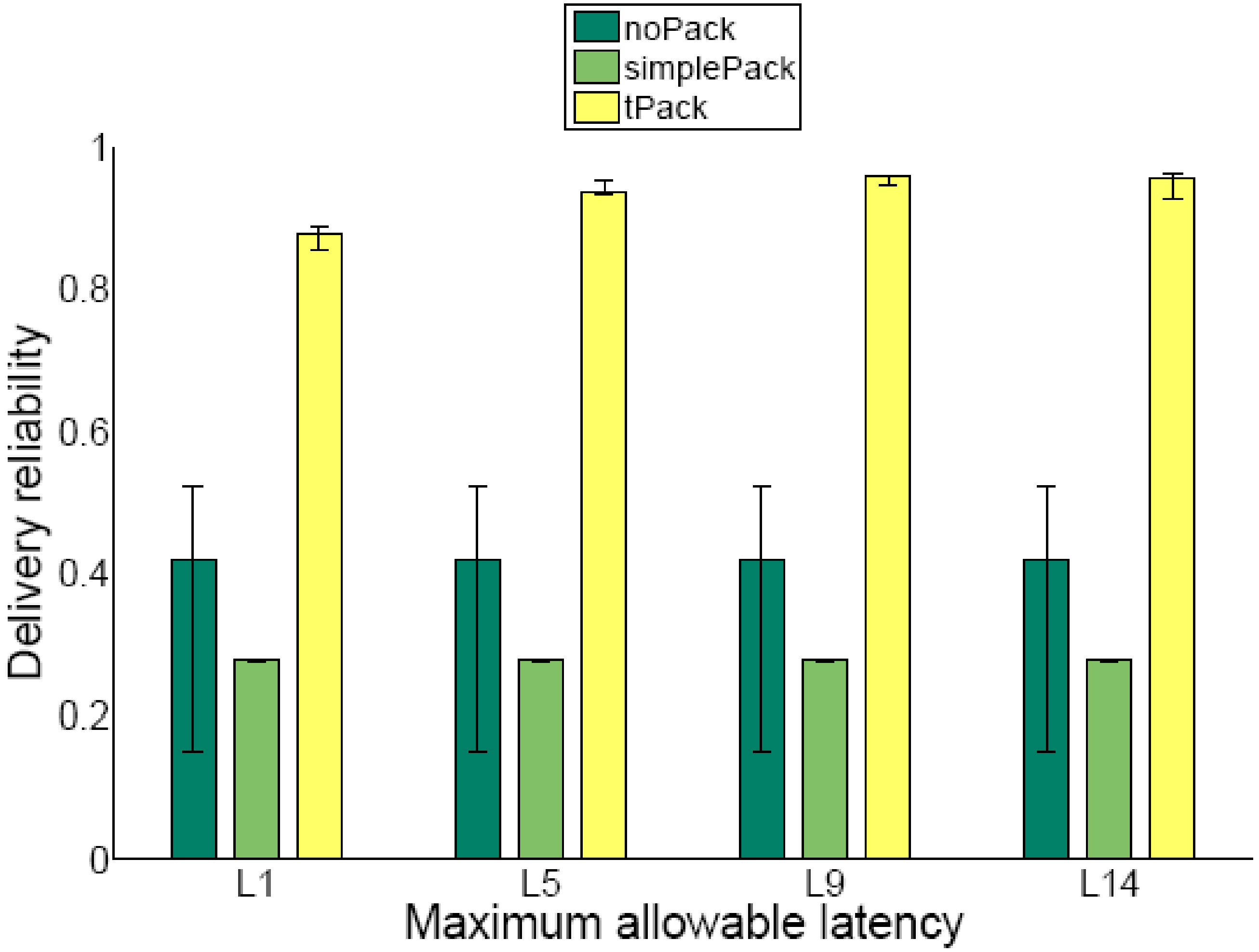
- Experiment Setting Up
  - Testbed: NetEye, a 130-sensor testbed
  - Topology: 120 nodes, half are source nodes
  - Protocols compared: noPacking, simplePacking, tPack
  - Traffic patterns: 6 second periodic traffic and event traffic

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

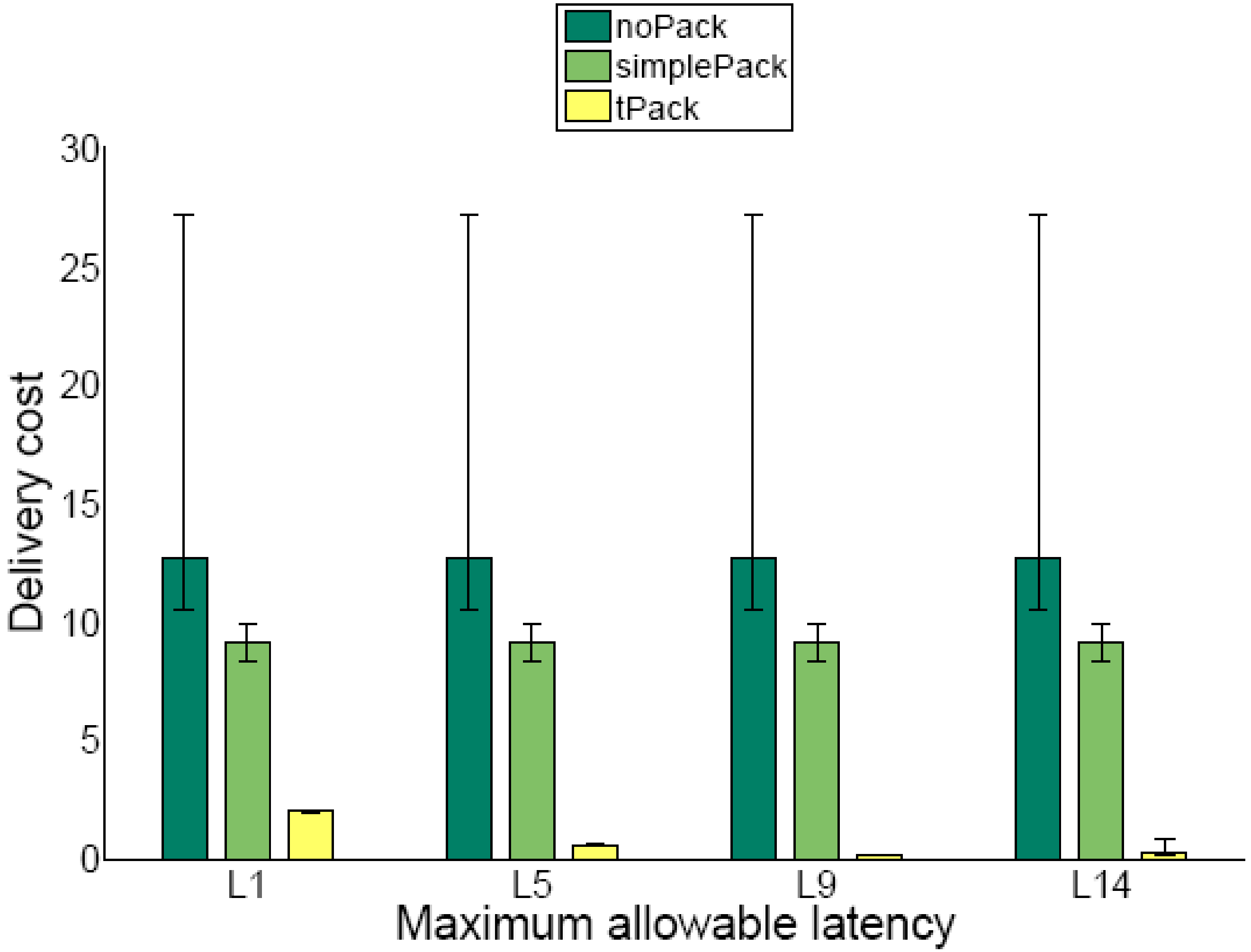
Packing Ratio



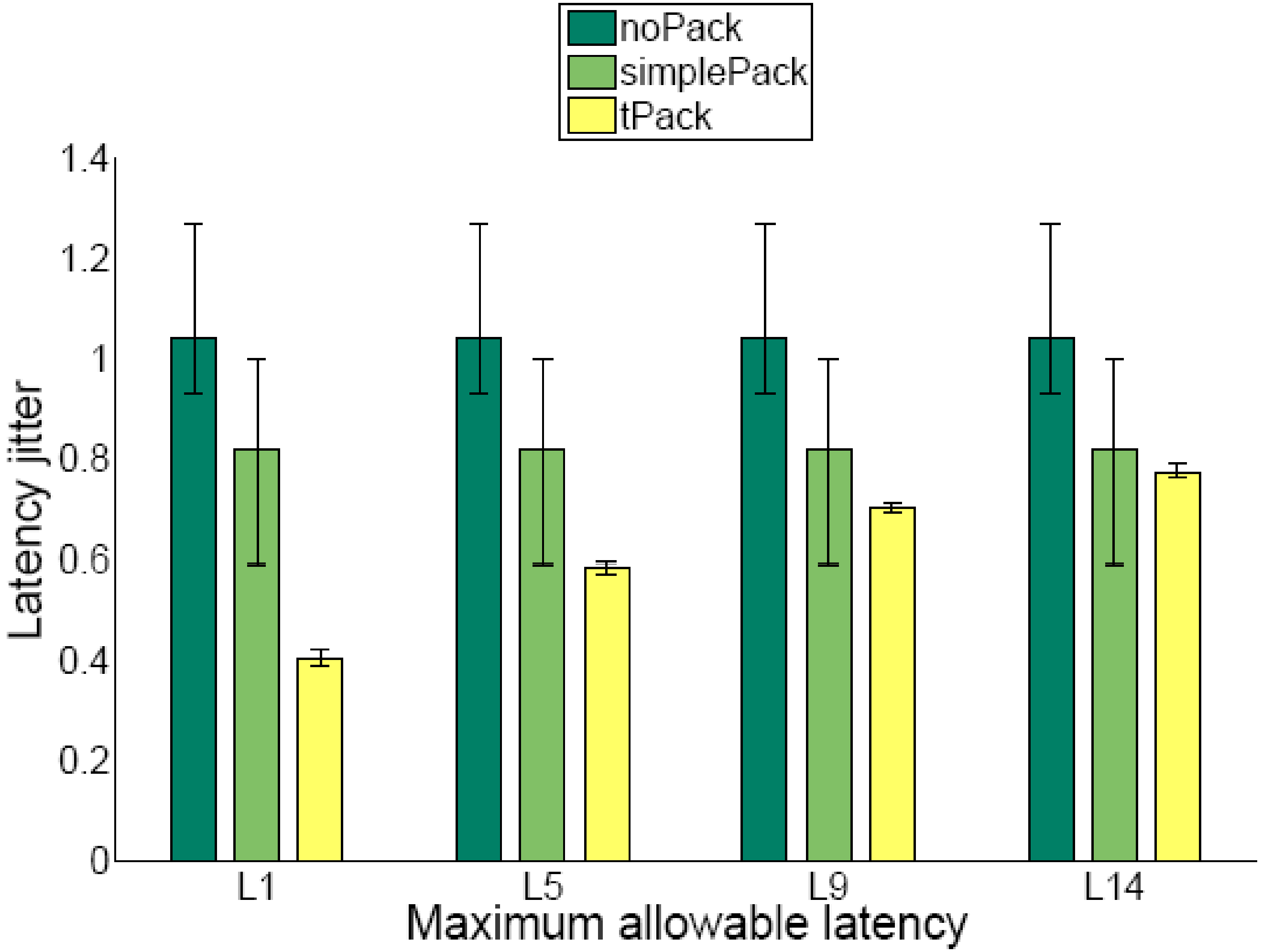
Delivery Reliability



Delivery Cost



Latency Jitter



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