



Aeolus: An Optimizer for Distributed Intra-Node-Parallel Streaming Systems

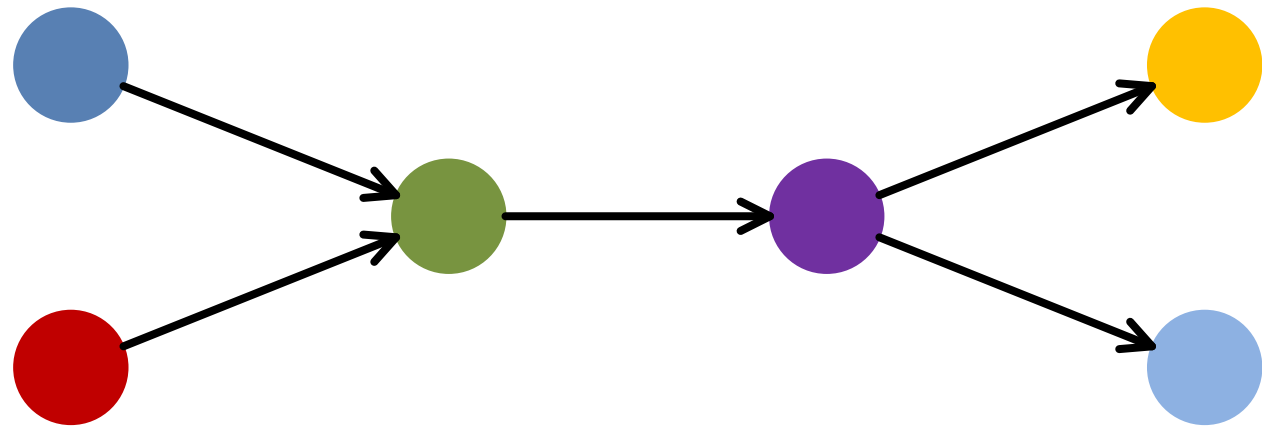


Matthias J. Sax^{#12}, Malu Castellanos⁺², Qiming Chen⁺², Meichun Hsu⁺²

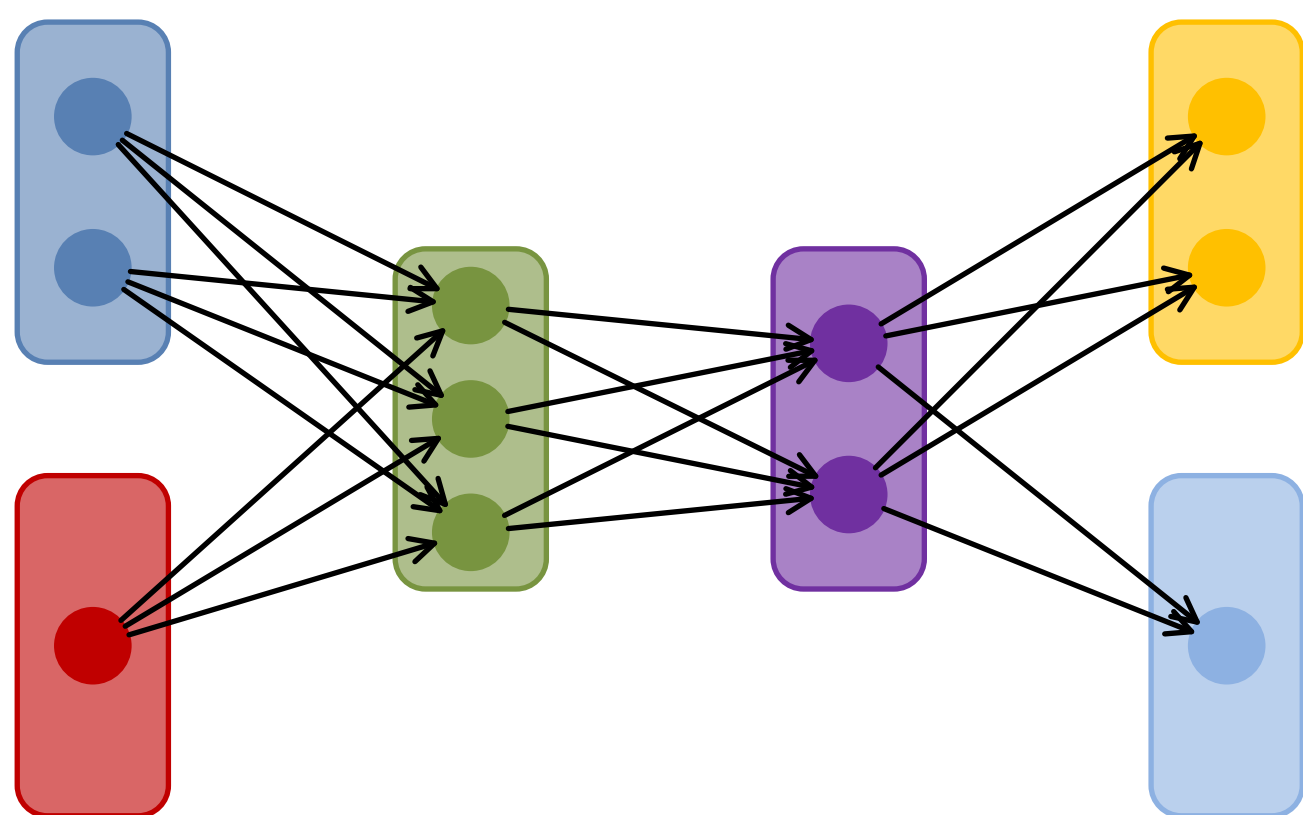
Streaming and Parallel Dataflow Execution

- *Data Intensive Computing*.
- Requires low response time, up to (near) real-time analysis.
- MapReduce does not fit (batch system).
- New class of intra-node parallel streaming systems address this problem: e.g., Storm, S4, Muppet.

Input is a dataflow specified as directed acyclic graph (DAG):

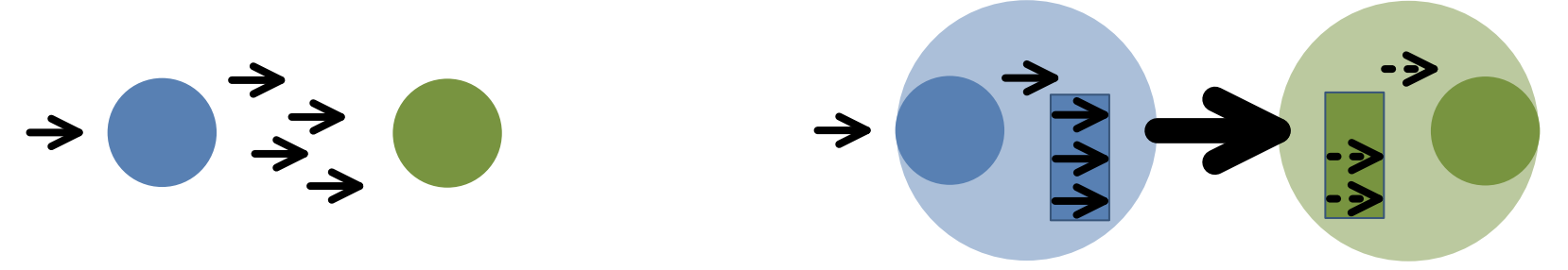


Dataflow (called Topology in Storm) is executed in a parallel manner.

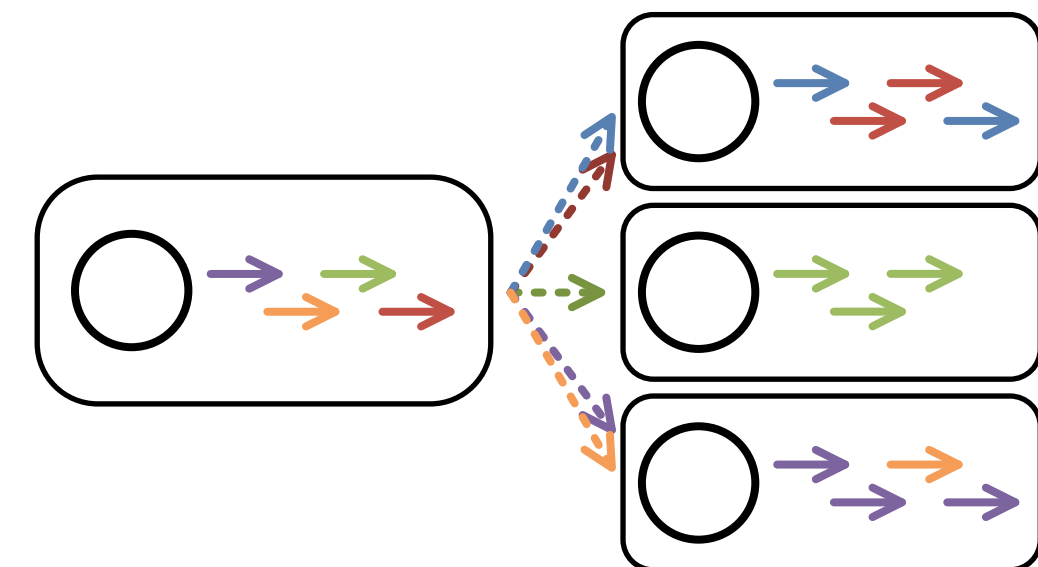


Batching in Streaming Systems

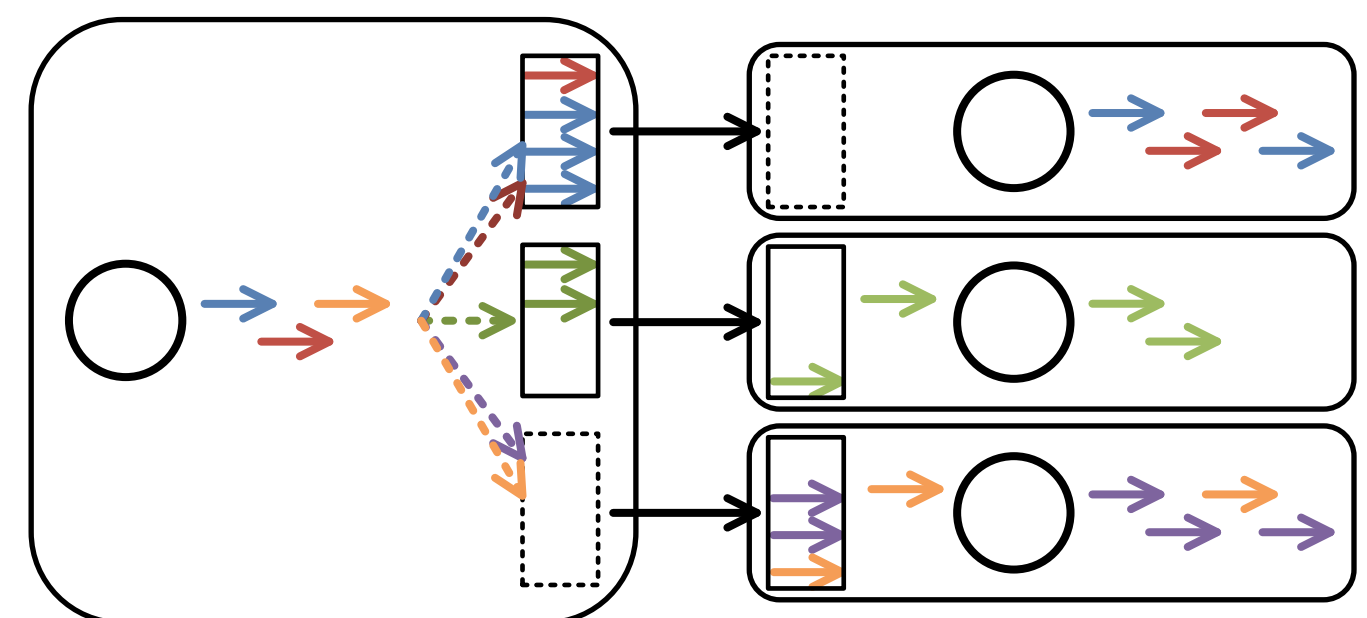
- Sending data tuple-by-tuple results in high network overhead.
- Tuple batching can increase throughput.



Key-based data distribution (w/o batching):

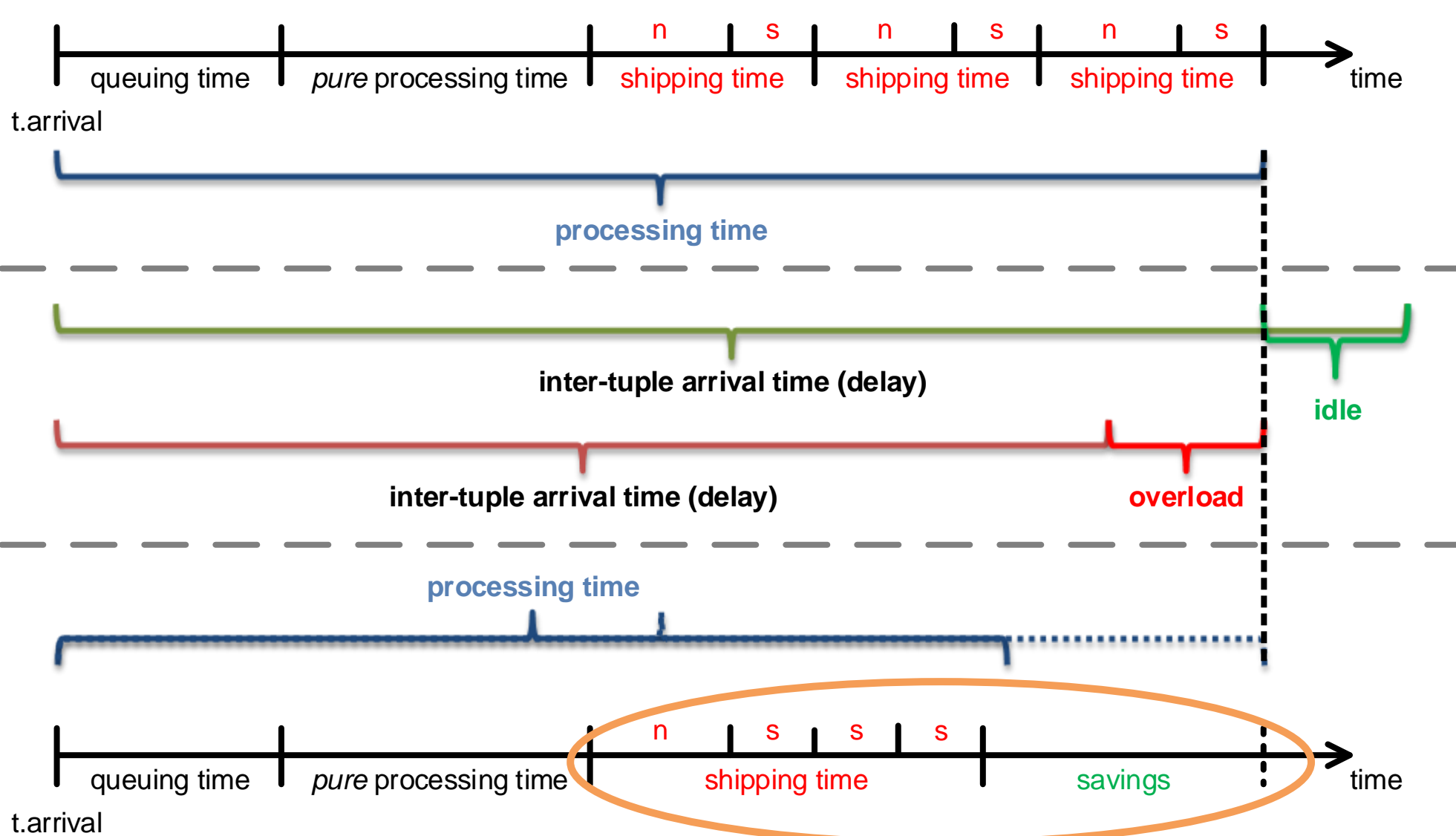


Novel batching schemas for intra-node parallelism:

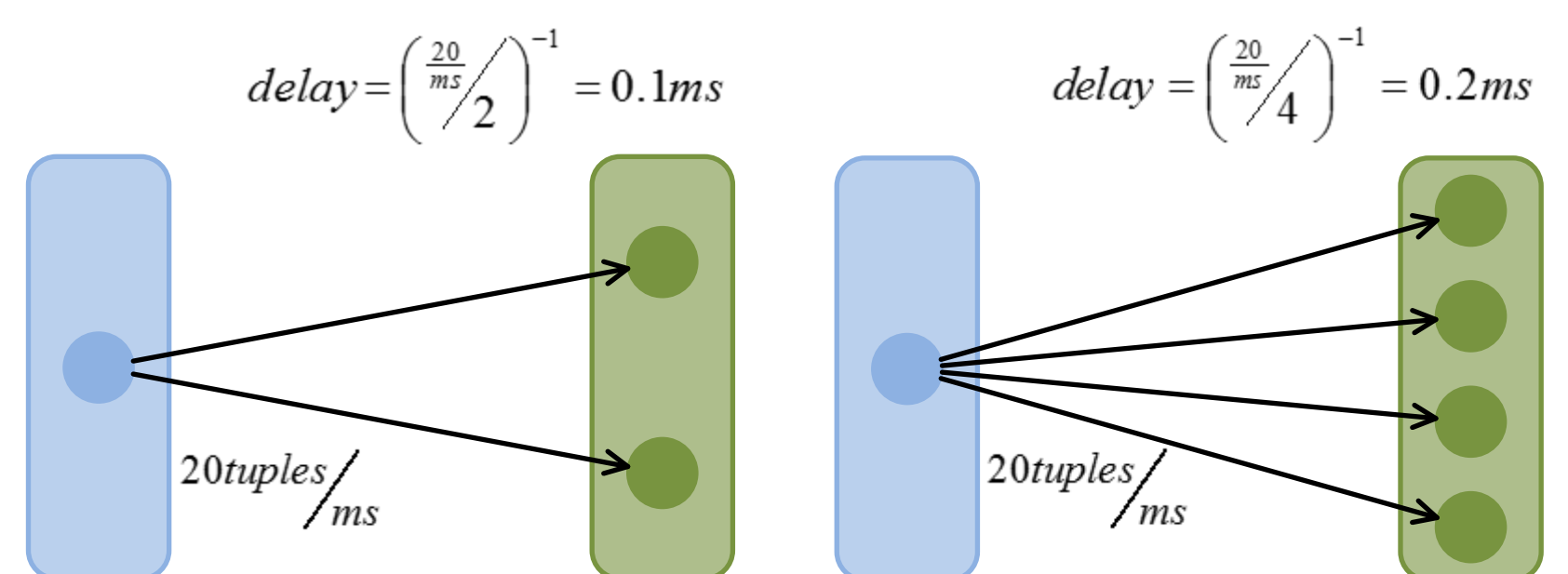


Cost Model for Batch Size and Degree of Parallelism

Optimizing *Batch Size*: (batching reduces network overhead n ; n is shared over multiple payloads s)

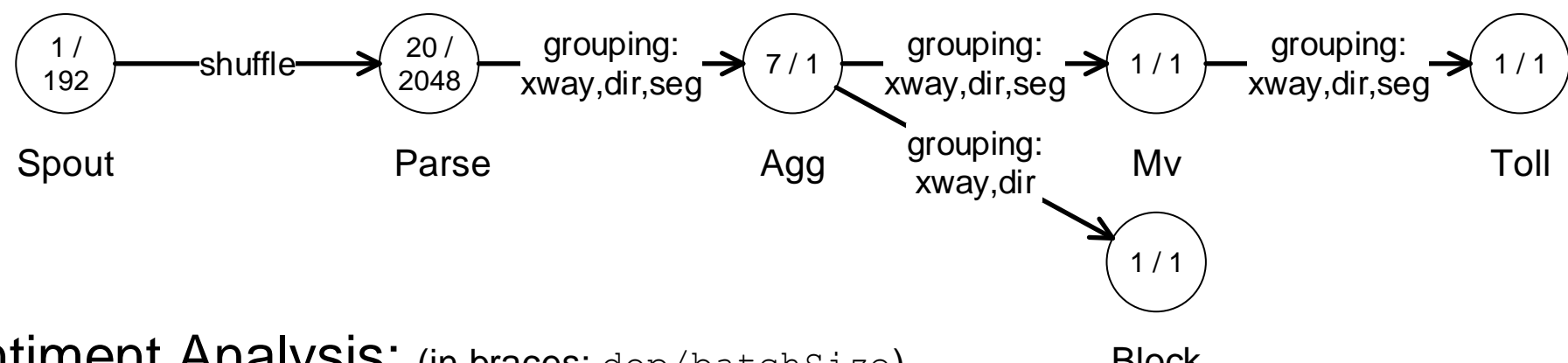


Optimizing *Degree of Parallelism (dop)*: (increasing dop reduces load on single node)

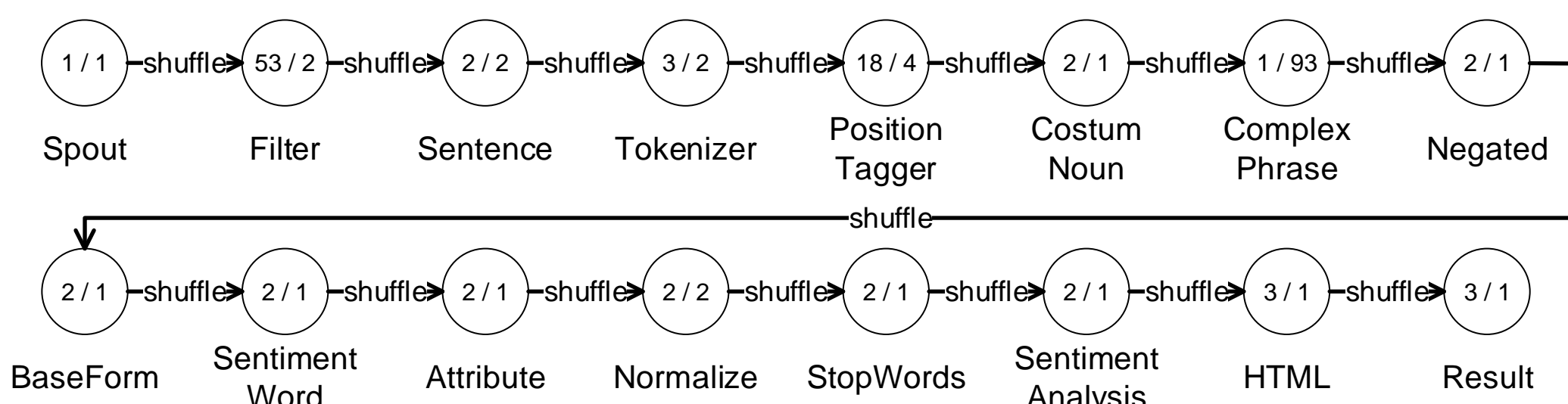


Example Topologies

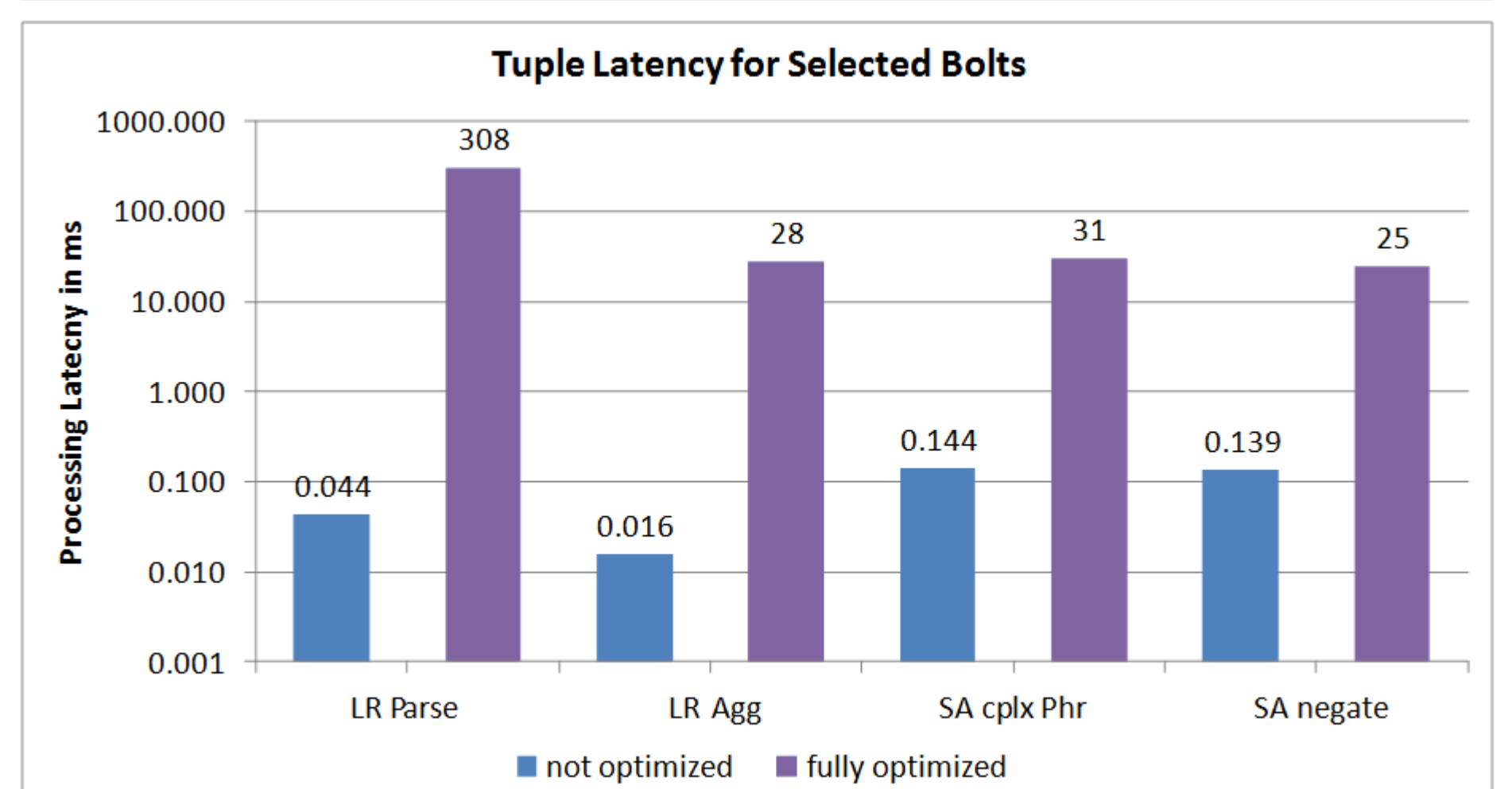
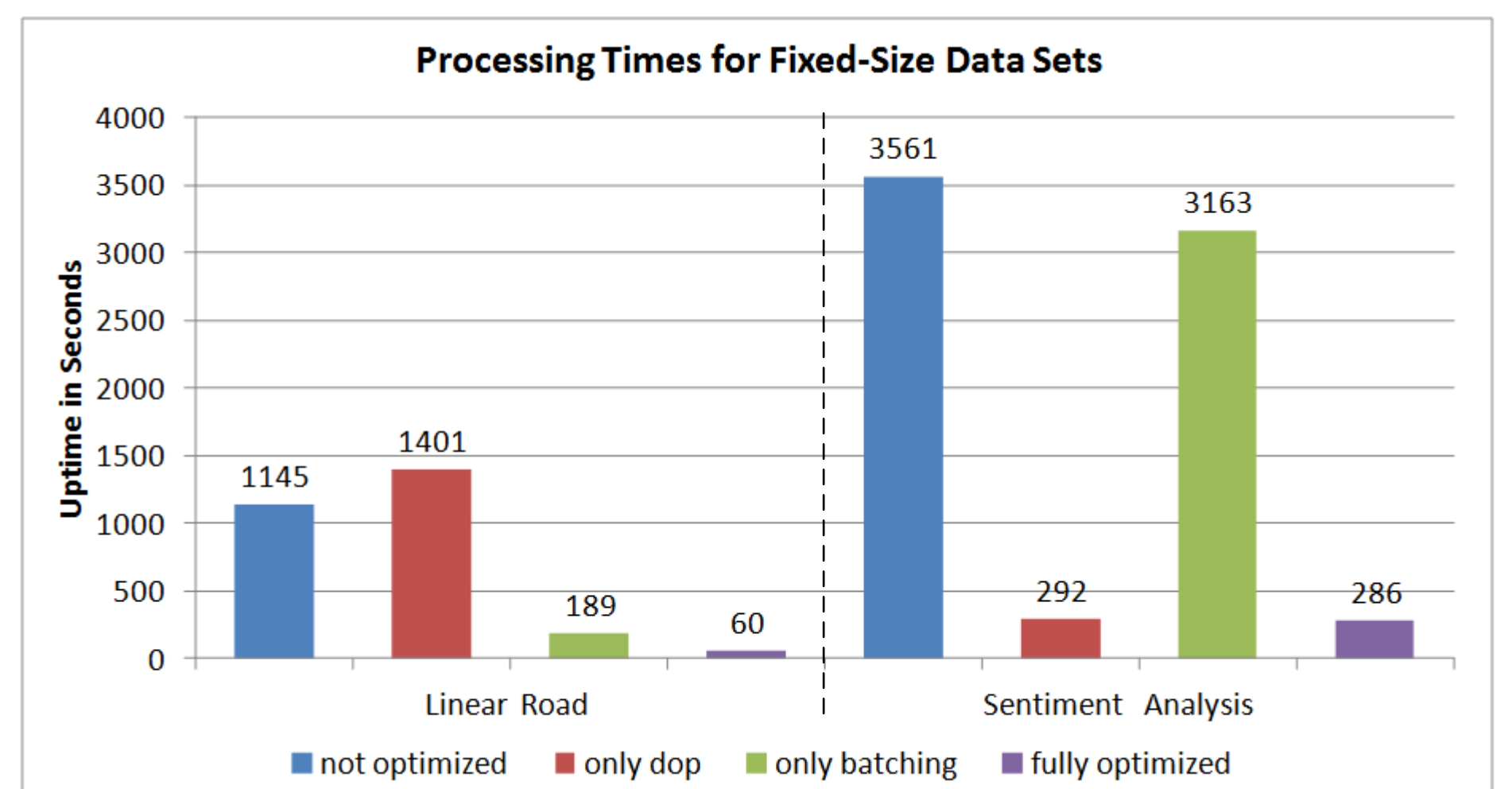
Linear Road: (in braces: dop/batchSize)



Sentiment Analysis: (in braces: dop/batchSize)



Evaluation



[#]Humboldt-Universität zu Berlin
⁺Hewlett-Packard Laboratories

¹mjsax@informatik.hu-berlin.de
²{firstname.lastname}@hp.com

