Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Acoustic Monitoring using Wireless Sensor Networks

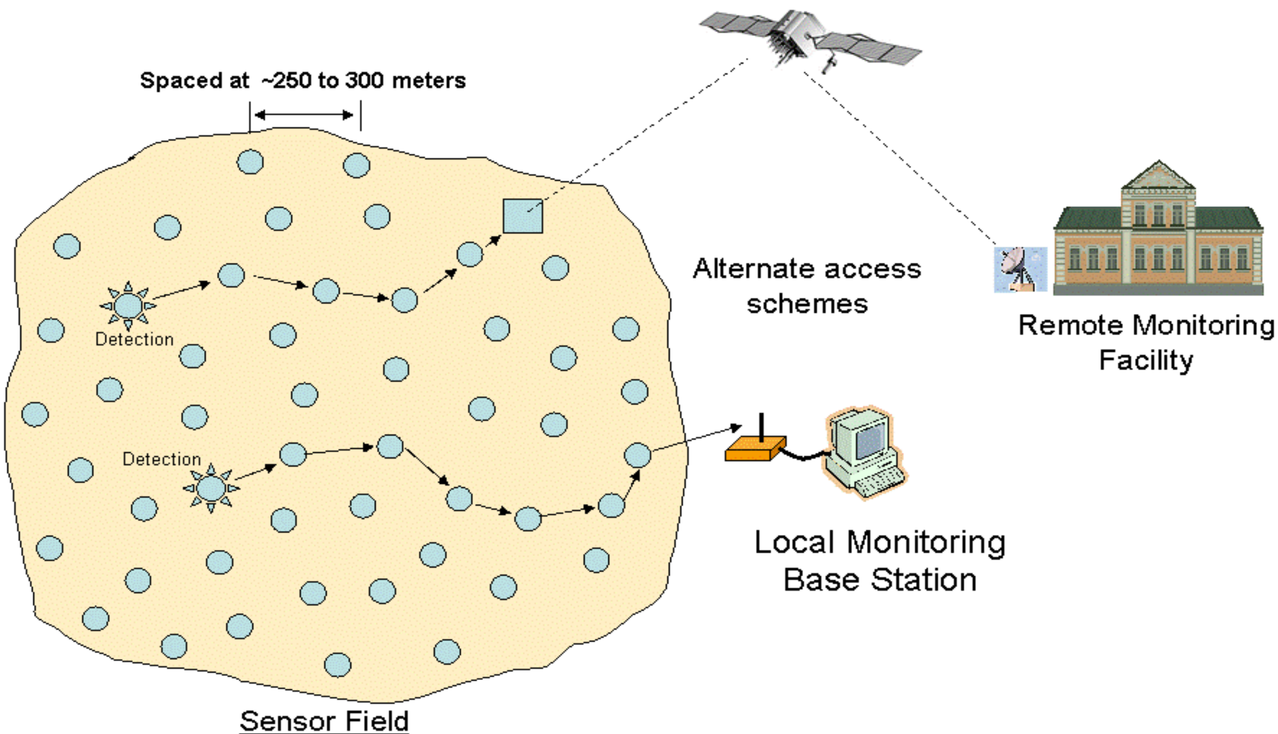
Presented by:

Farhan Imtiaz

Wireless Sensor Networks

- Wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations (Wikipedia).

Wireless Sensor Networks



What is acoustic source localisation?

Given a set of acoustic sensors at known positions and an acoustic source whose position is unknown, estimate its location

Applications

- Gunshot Localization
- Acoustic Intrusion Detection
- Biological Acoustic Studies
- Person Tracking
- Speaker Localization
- Smart Conference Rooms
- And many more

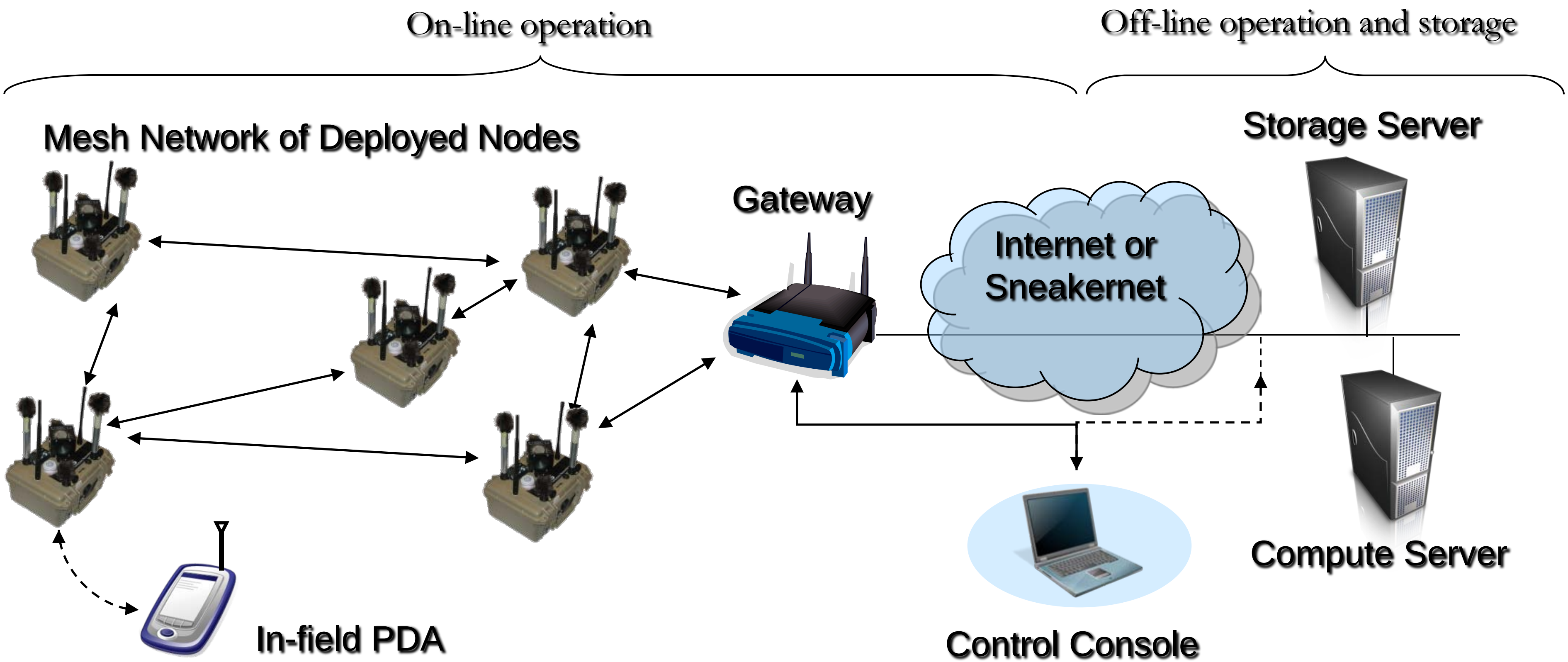
Challenges

- Acoustic sensing requires high sample rates
  - Cannot simply sense and send
  - Implies on-node, in-network processing
  - Indicative of generic high data rate applications
- A real-life application, with real motivation
  - Real life brings deployment and evaluation problems which must be resolved

Acoustic Monitoring using VoxNet

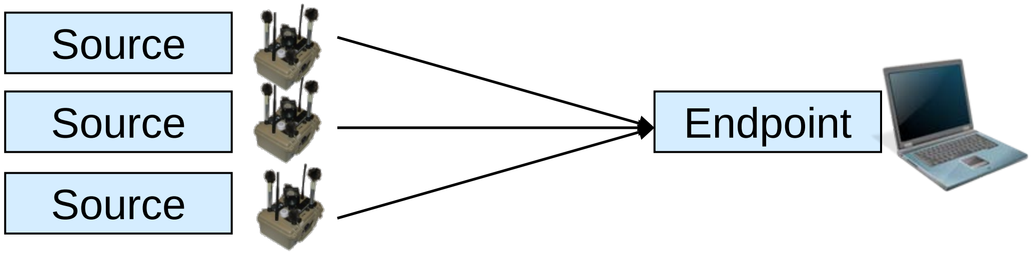
- VoxNet is a complete hardware and software platform for distributing acoustic monitoring applications.

VoxNet architecture

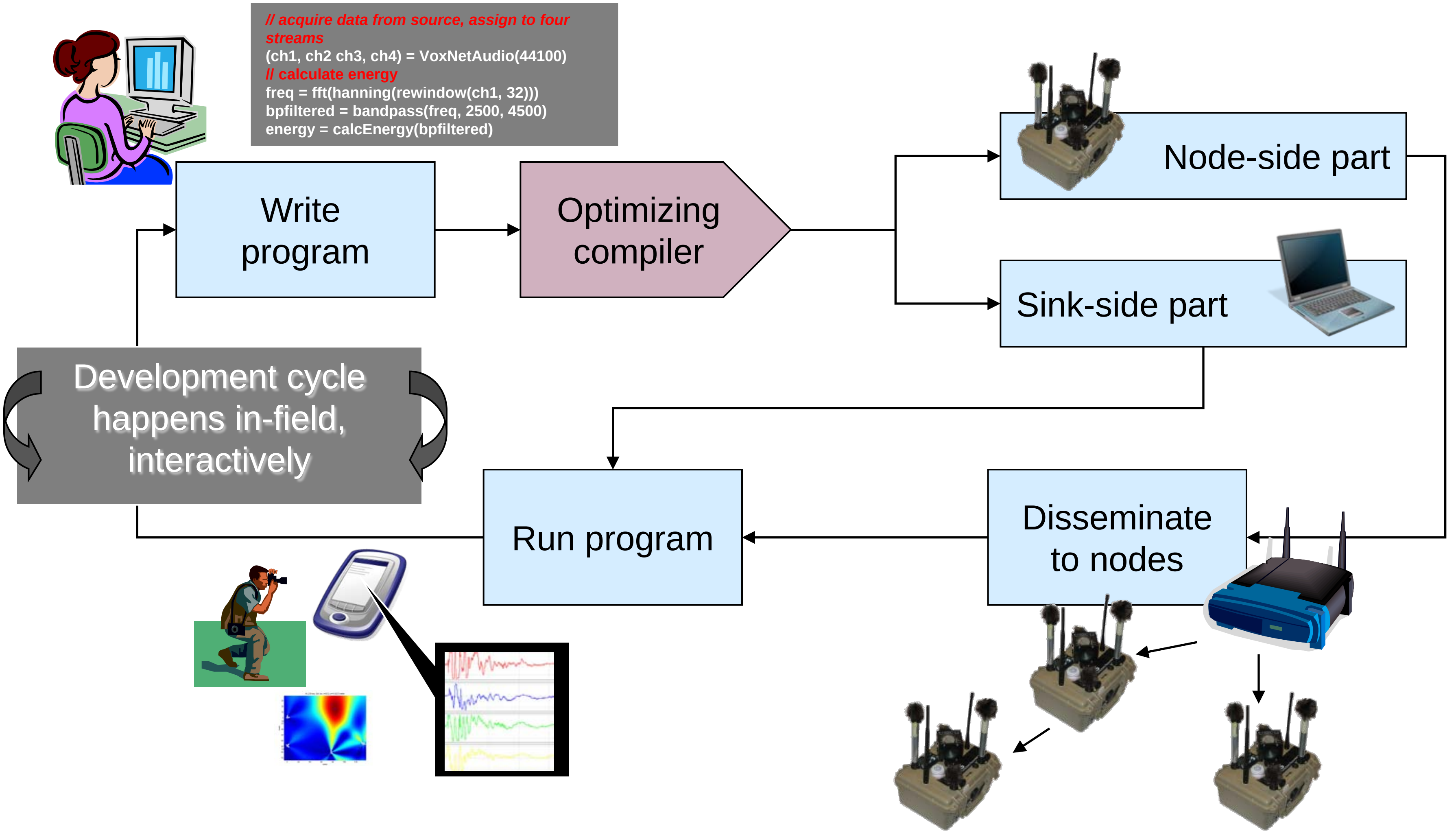


Programming VoxNet

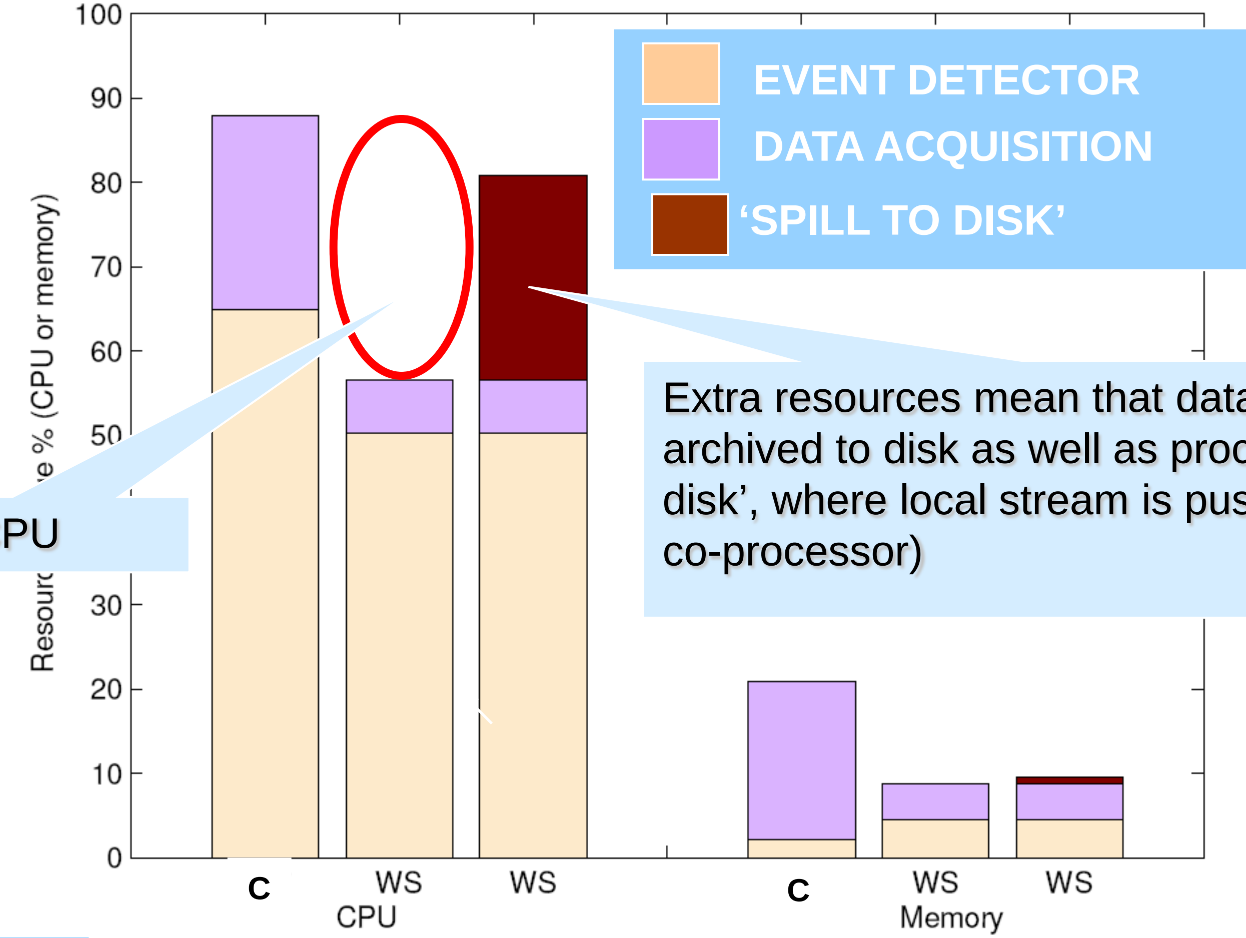
- Programming language: Wavescript
  - High level, stream-oriented macroprogramming language
  - Operates on stored OR streaming data
- User decides where processing occurs (node, sink)
  - Explicit, not automated processing partitioning



VoxNet on-line usage model



Hand-coded C vs. Wavescript



30% less CPU

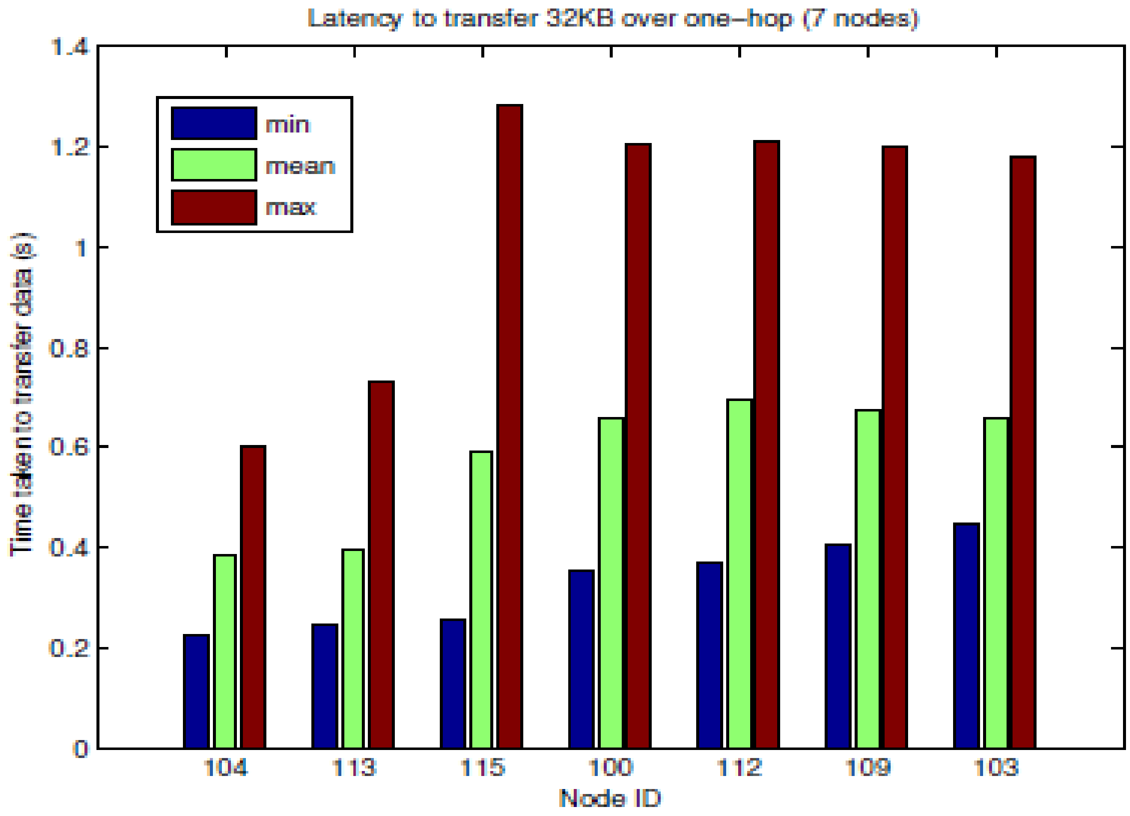
Extra resources mean that data can be archived to disk as well as processed ('spill to disk', where local stream is pushed to storage co-processor)

WS = Wavescript

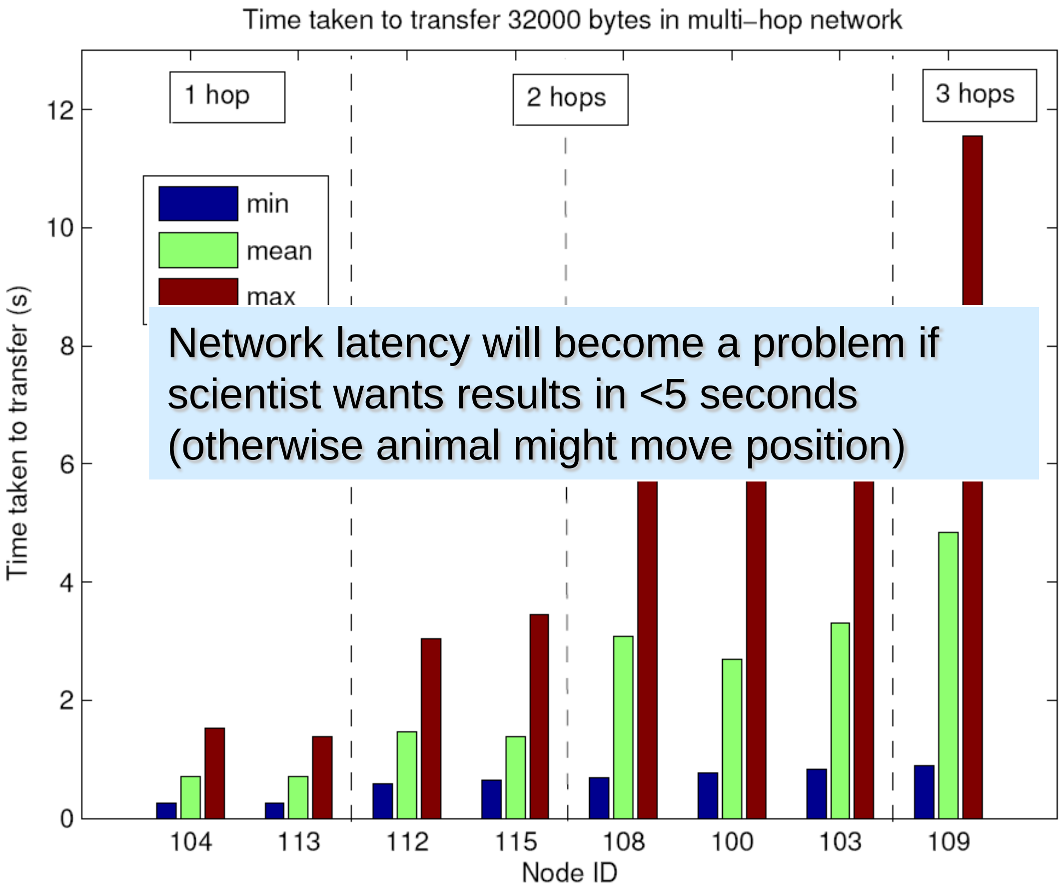
In-situ Application test

- One-hop network-> extended size antenna on the gateway
- Multi-hop network -> standard size antenna on the gateway

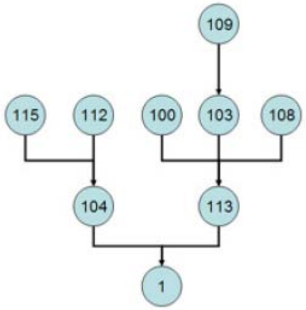
Detection data transfer latency for one-hop network



Detection data transfer latency for multi-hop network



Network latency will become a problem if scientist wants results in <5 seconds (otherwise animal might move position)

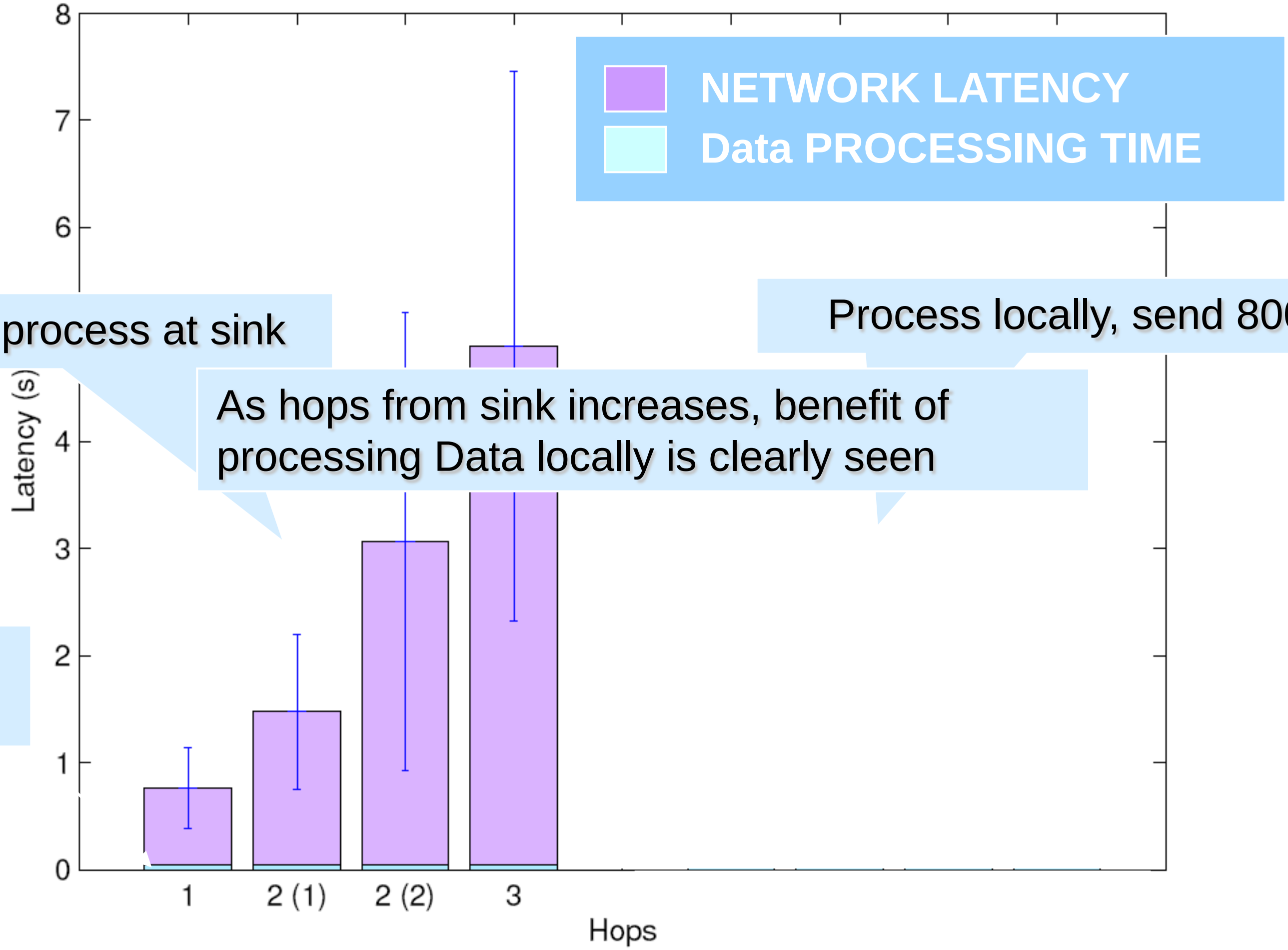


General Operating Performance

- To examine regular application performance -> run application for 2 hours
  - 683 events by mermot vocalization
  - 5 out of 683 detections dropped(99.3% success rate)
  - Failure due to overflow of 512K network buffer

- Deployment during rain storm
  - Over 436 seconds -> 2894 false detections
  - Successful transmission -> 10% of data generated
  - Ability to deal with overloading in a graceful manner

Local vs. sink processing trade-off



Send raw data, process at sink

Process locally, send 800B

As hops from sink increases, benefit of processing Data locally is clearly seen

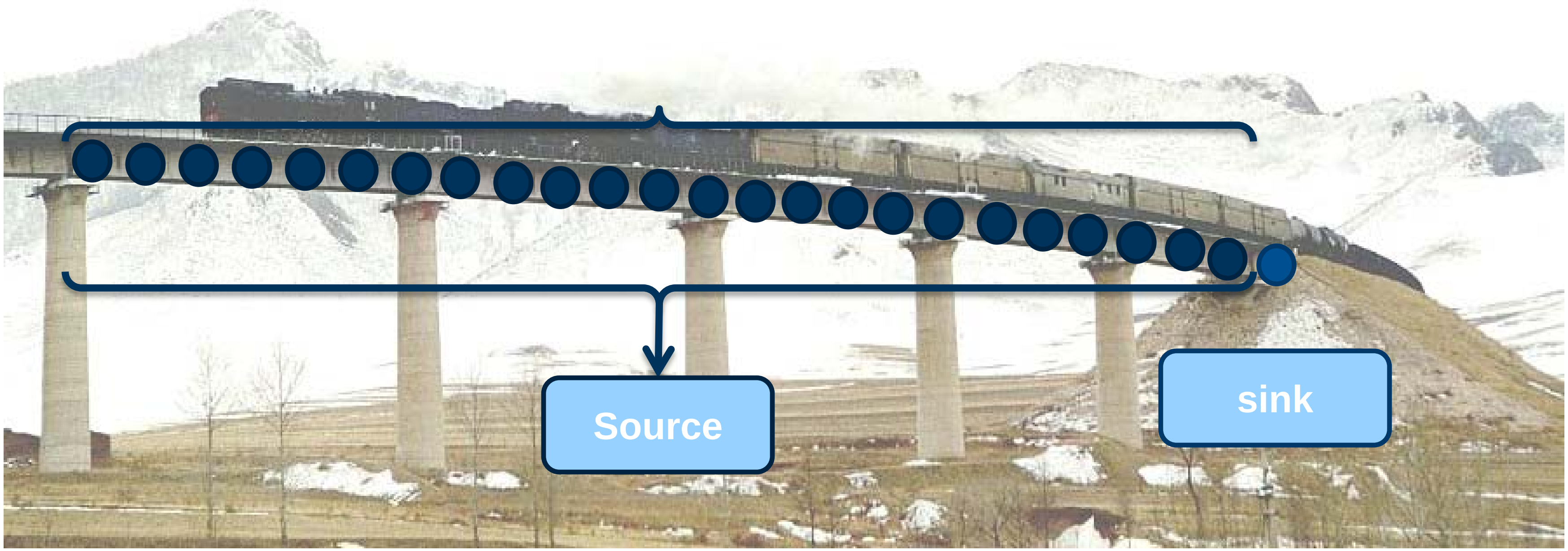
Data processing

Motivation for Reliable Bulk data Transport Protocol

Power Efficiency

Interference

Bulky Data

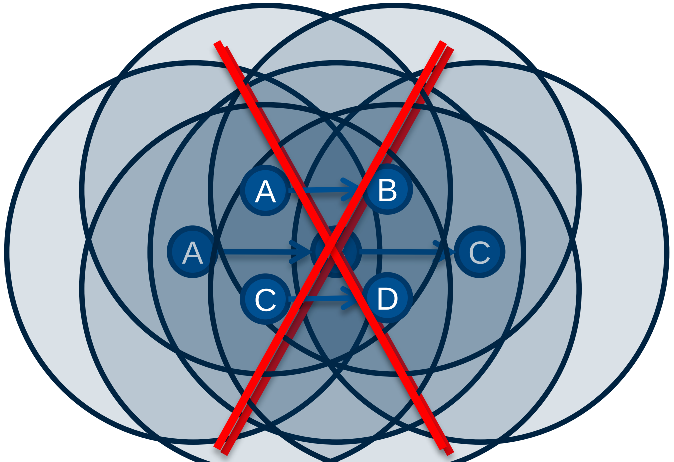


Goals of Flush

- Reliable delivery
  - End-to-End NACK
- Minimize transfer time
  - Dynamic Rate Control Algo.
- Take care of Rate miss-match
  - Snooping mechanism

Challenges

- Links – Lossy
- Interference
  - Interpath (more flows)
  - Intra-path (same flow)
- Overflow of queue of intermediate nodes

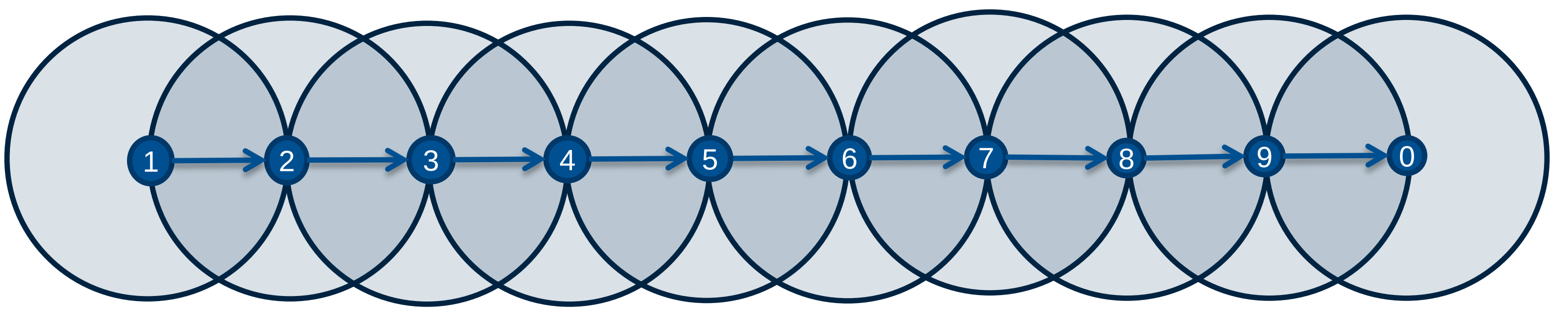


Intra-path interference

Interpath interference

Assumptions

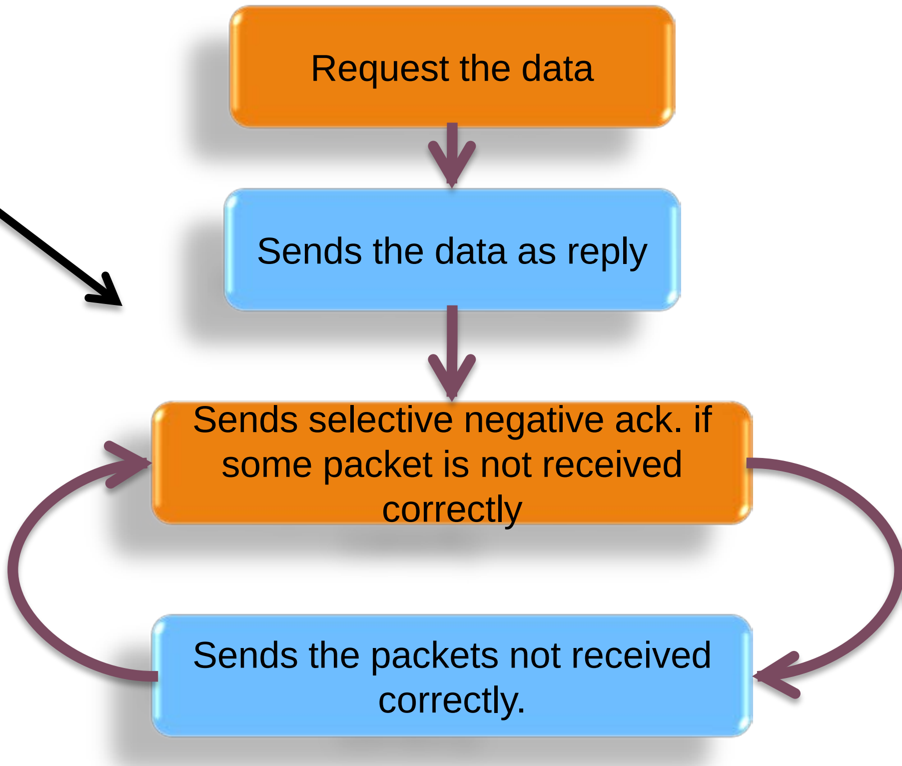
- Isolation — The sink schedule is implemented so inter-path interference is not present. Slot mechanism.
- Snooping
- Acknowledgements — Link layer ack. are efficient
- Forward routing — Routing mechanism is efficient
- Reverse Delivery — For End-to-End acknowledgements.



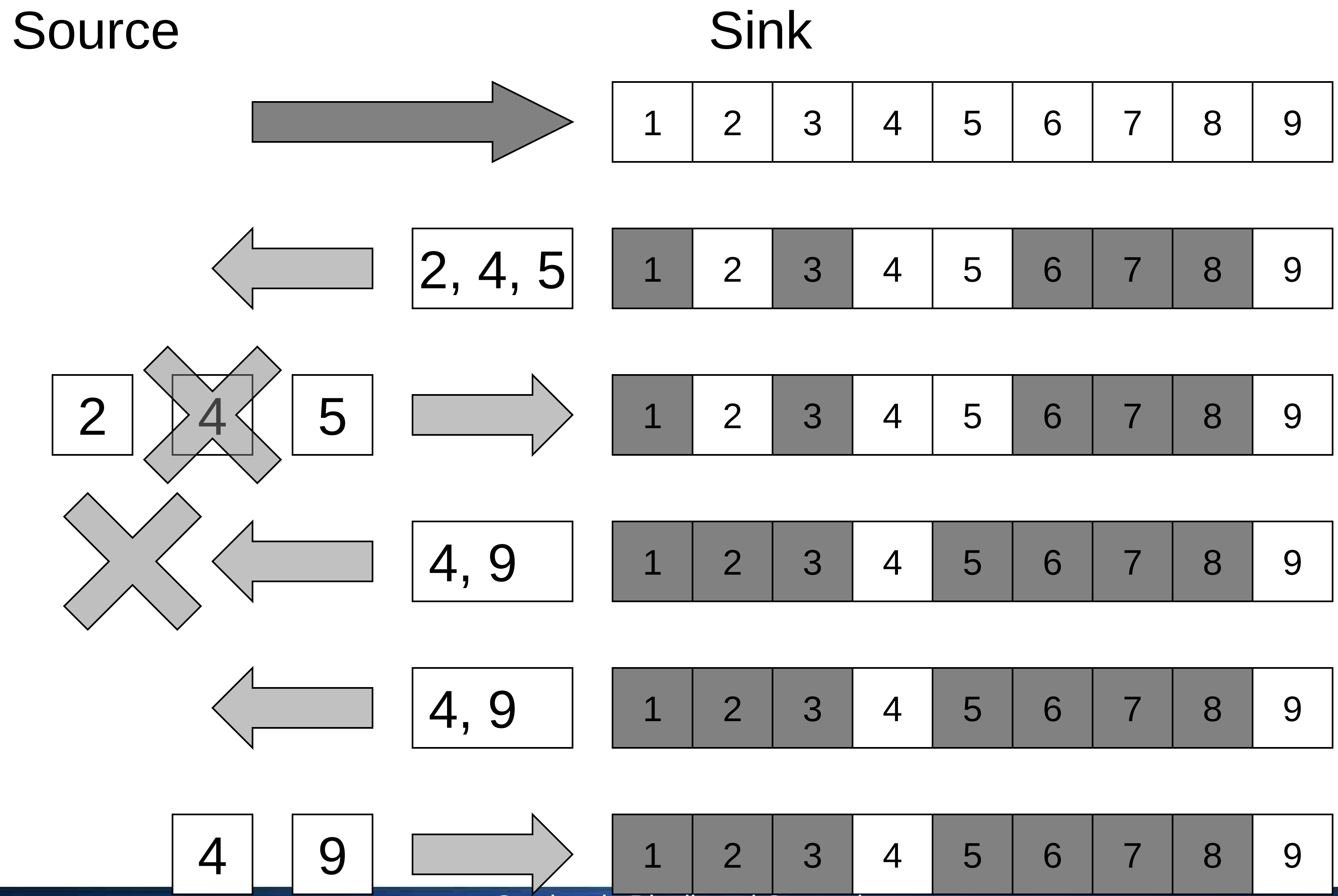
How it works

- Red – Sink (receiver)
- Blue – Source (sensor)

- 4 Phases
  1. Topology query
  2. Data transfer
  3. Acknowledgement
  4. Integrity check



Reliability



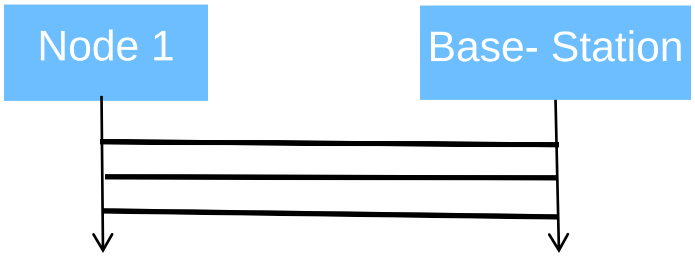
Rate control: Conceptual Model Assumptions

- Nodes can send exactly one packet per time slot
- Nodes can not send and receive at same time
- Nodes can only send and receive packets from nodes one hop away
- Variable range of Interference I may exist

Rate control: Conceptual Model

For N = 1

Rate = 1



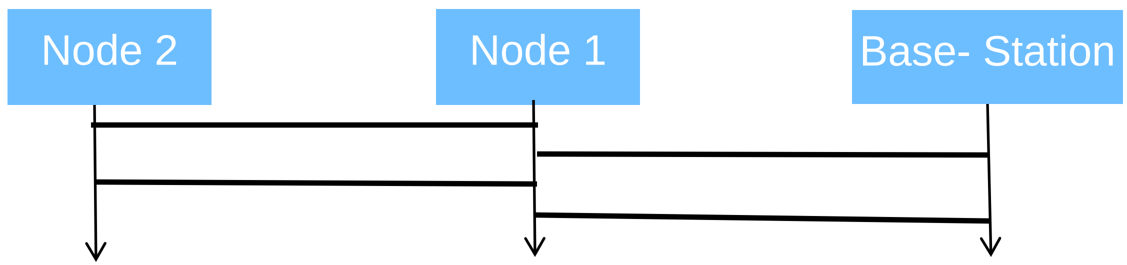
Packet Transmission

Interference

Rate control: Conceptual Model

For N = 2

Rate = 1/2



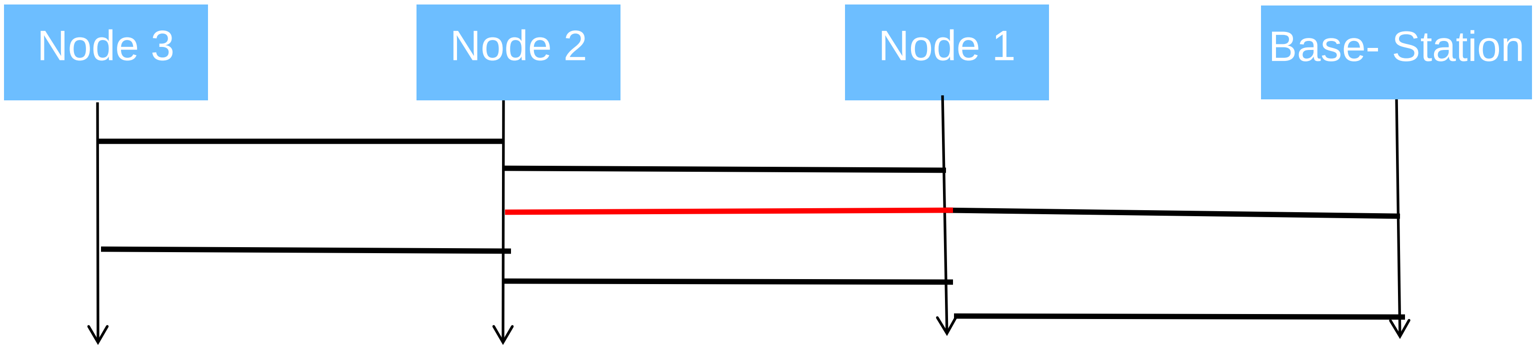
Packet Transmission

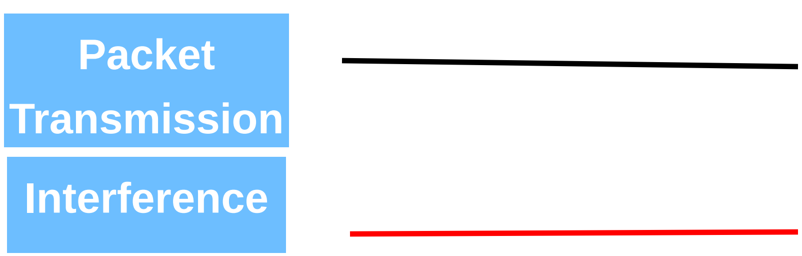
Interference

Rate control: Conceptual Model

For N >= 3 Interference = 1

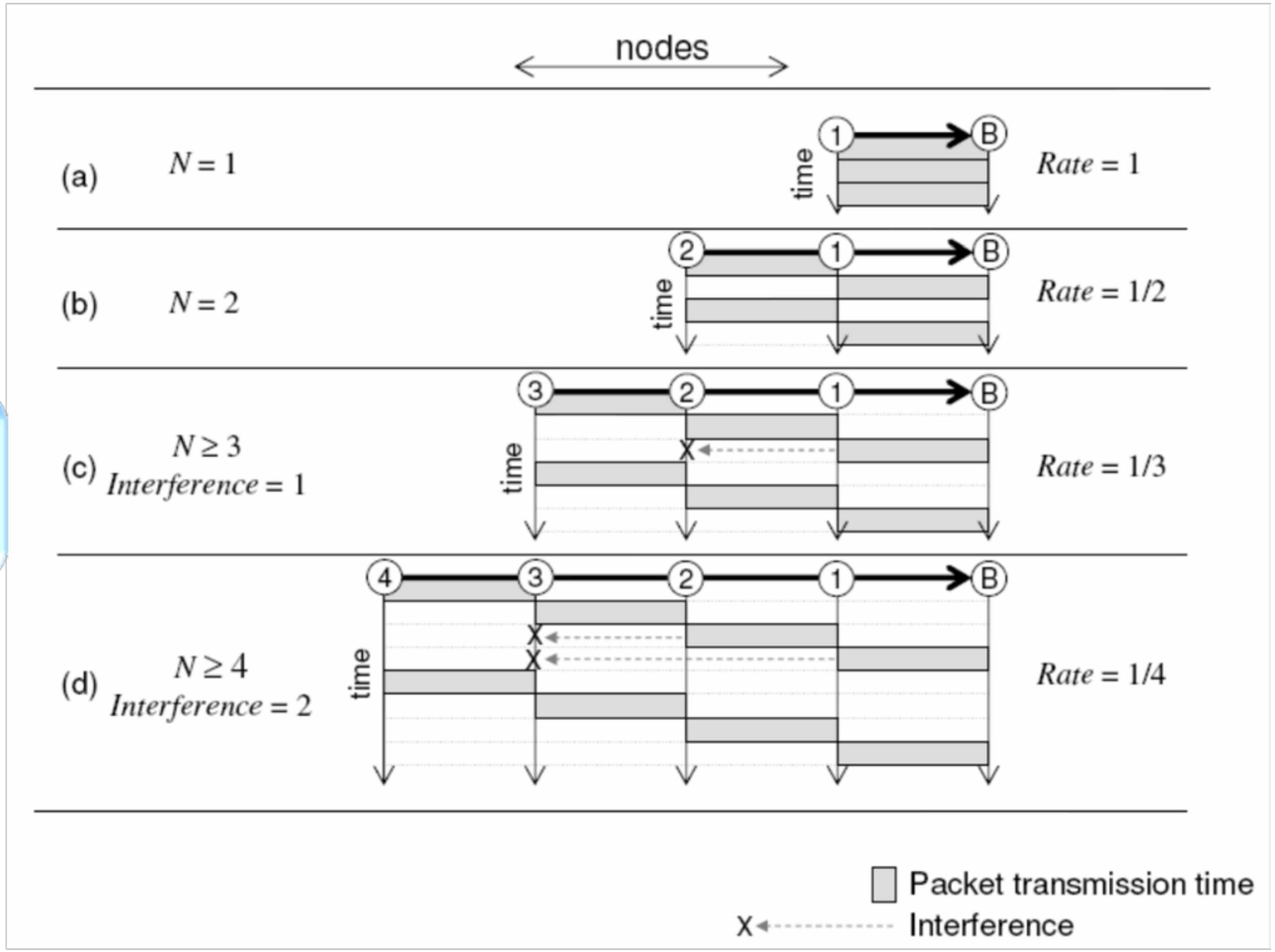
Rate = 1/3



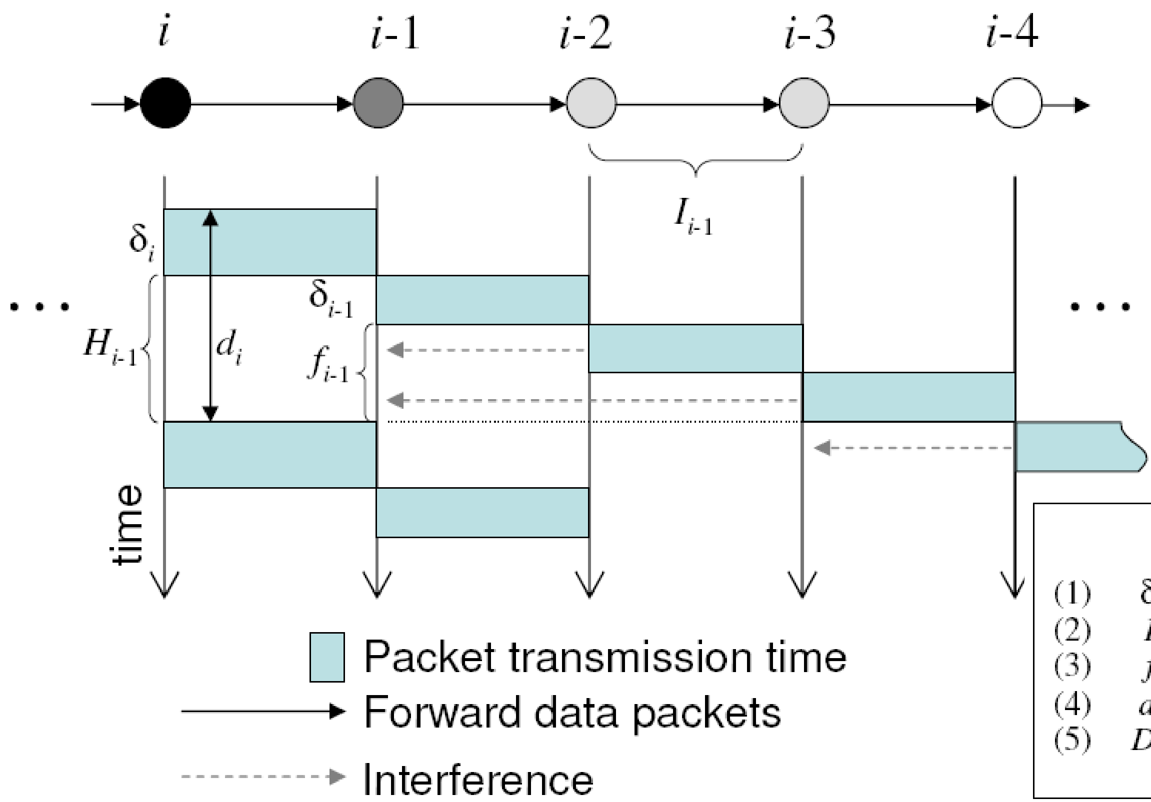


Rate control: Conceptual Model

$$r(N,I) = \frac{1}{\min(N, 2+I)}$$



Dynamic Rate Control



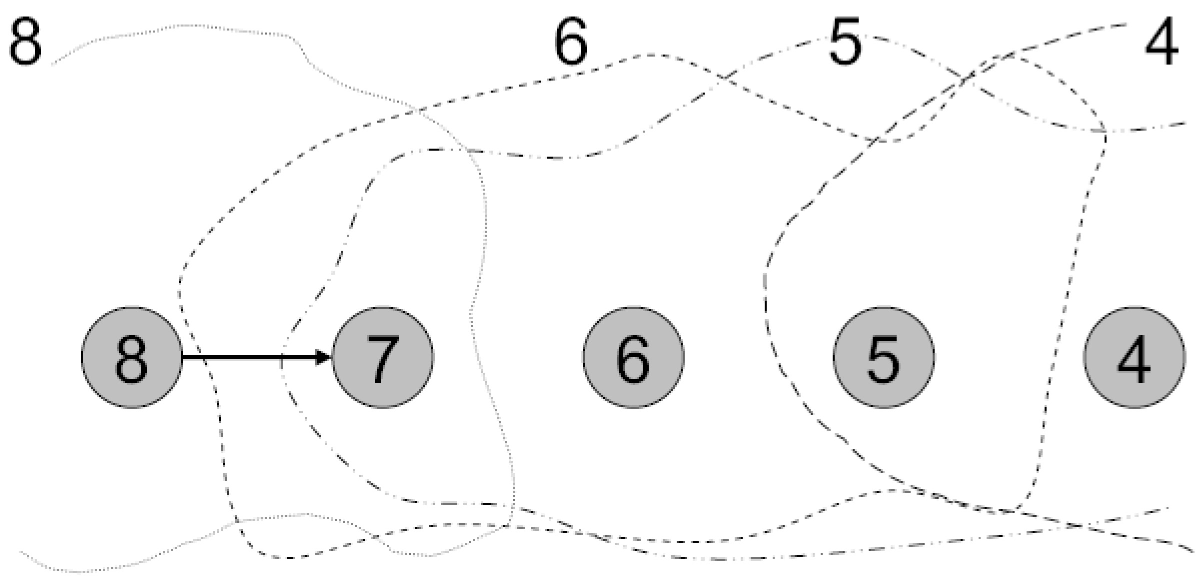
Rule – 1
A node should only transmit when its successor is free from interference.

Rule – 2
A node's sending rate cannot exceed the sending rate of its successor.

The Flush rate control algorithm

(1)	δ_i	:	actual transmission time at node i	
(2)	I_i	:	set of forward interferers at node i	
(3)	f_i	=	$\sum_{k \in I_i} \delta_k$	
(4)	d_i	=	$\delta_i + (\delta_{i-1} + f_{i-1})$	(Rule 1)
(5)	D_i	=	$\max(d_i, D_{i-1})$	(Rule 2)

Dynamic rate Control (cont…)



$$d_8 = \delta_8 + H_7 = \delta_8 + \delta_7 + f_7 = \delta_8 + \delta_7 + \delta_6 + \delta_5$$

$$D_i = \max(d_i, D_{i-1})$$

Performance Comparison

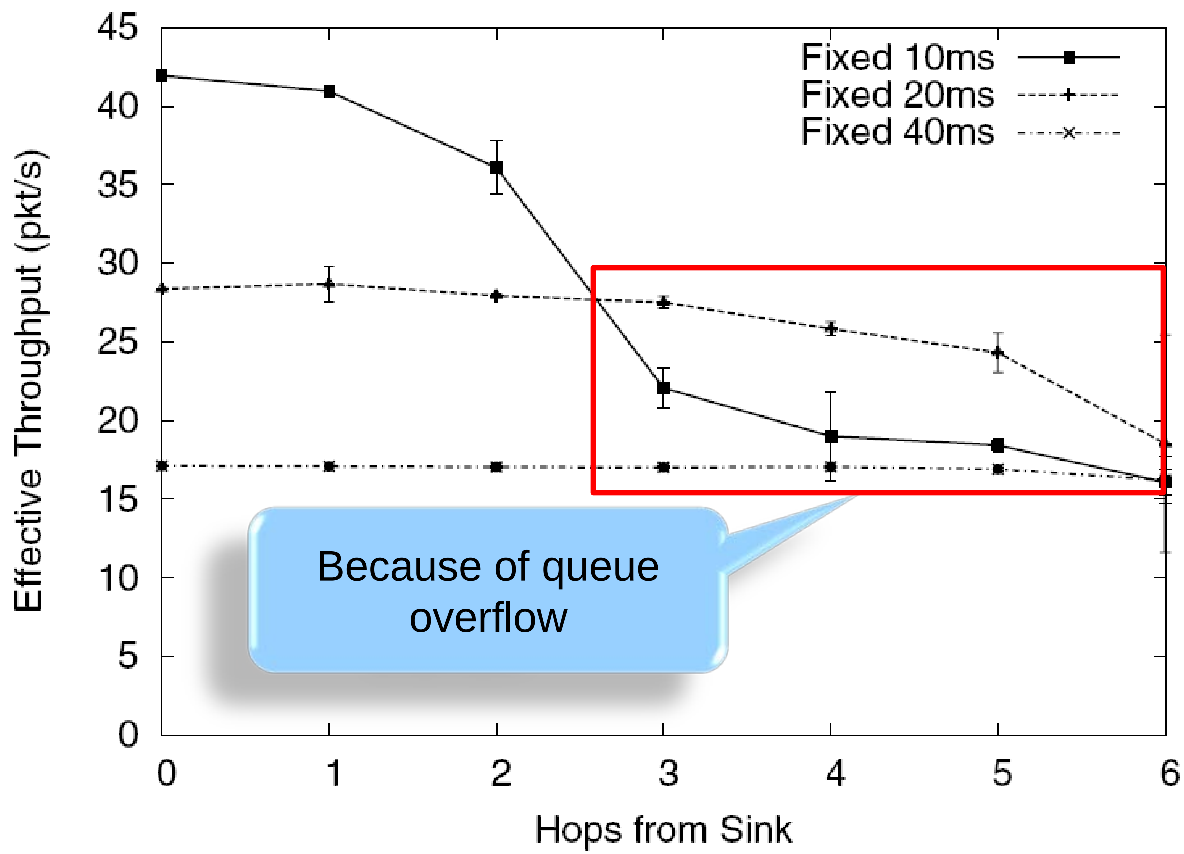
- Fixed-rate Algorithm: In such an algorithm data is sent after a fixed interval.
- ASAP(As soon as Possible): It's a naïve transfer algorithm that sends a packet as soon as last packet transmission is done.

Preliminary experiment

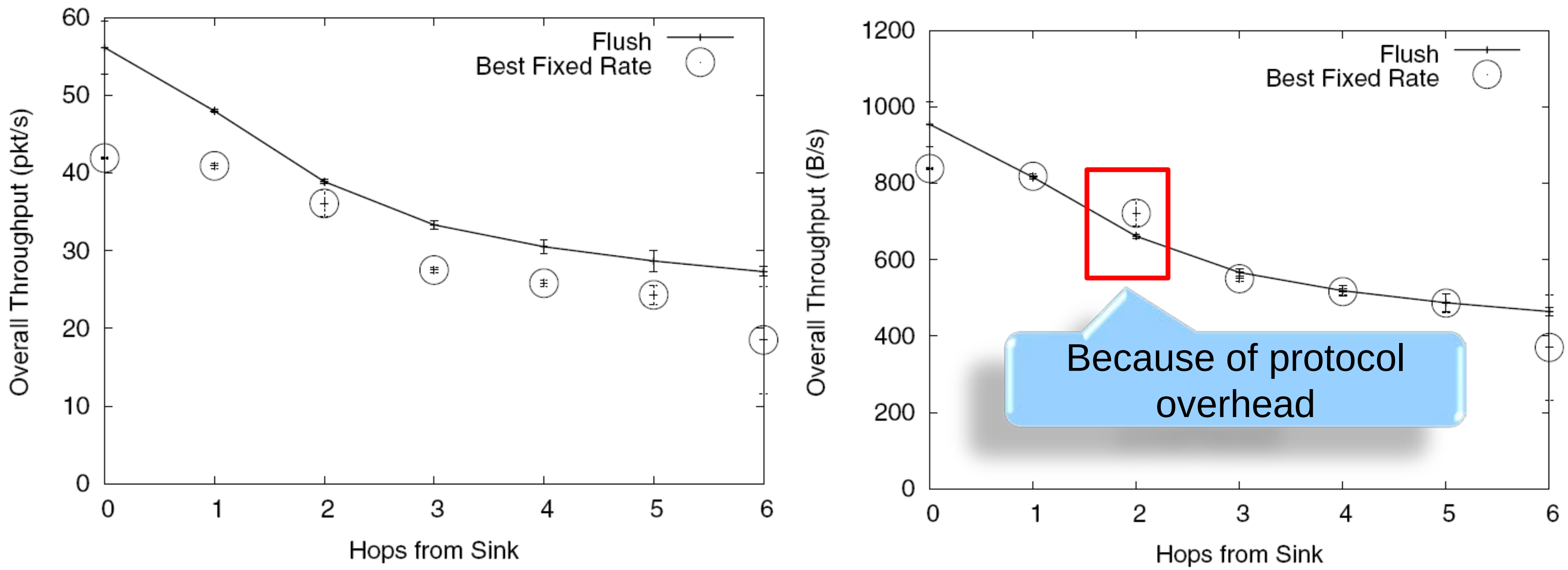
Throughput with different data collection periods.

Observation

There is tradeoff between the throughput achieved to the period at which the data is sent.

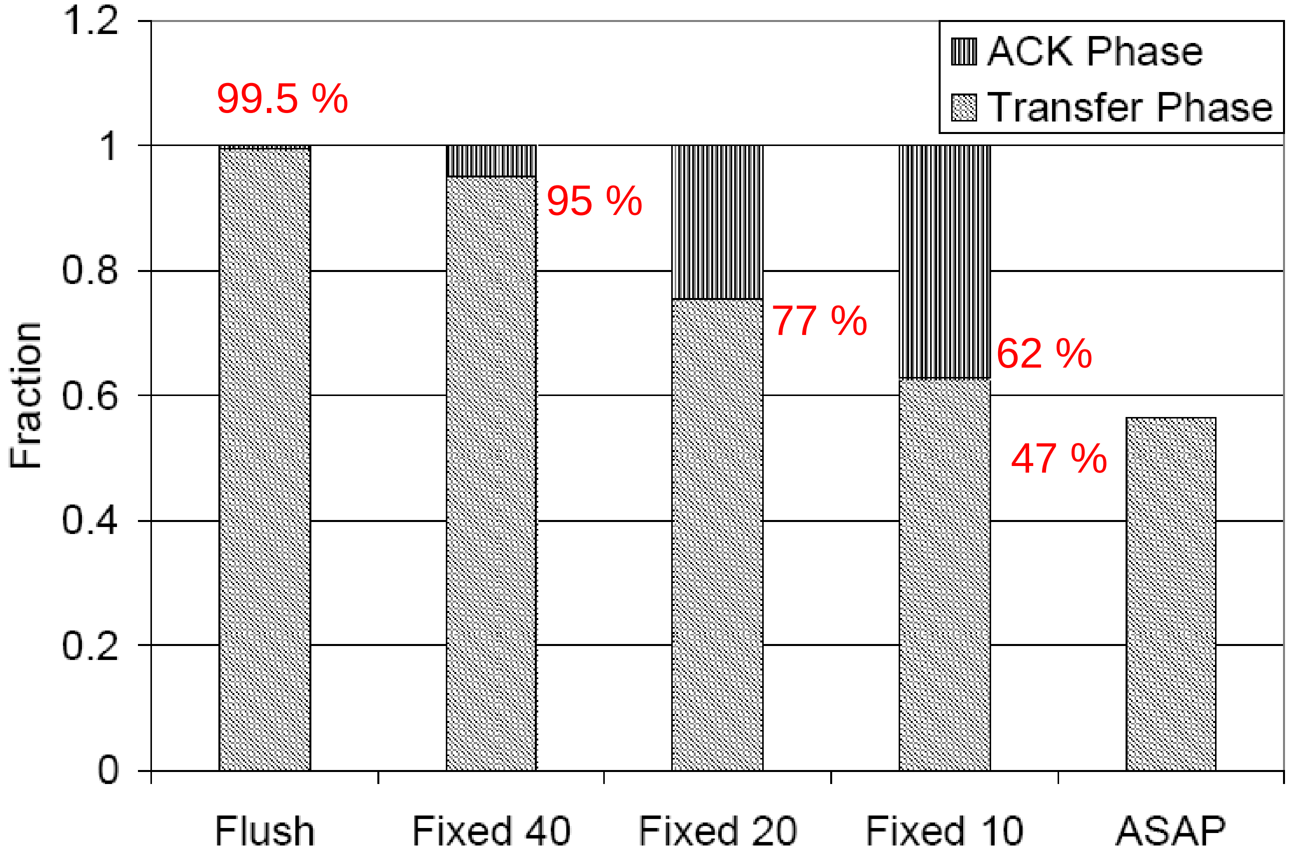


Flush Vs Best Fixed Rate



The delivery of the packet is better then fixed rate, but because of the protocol overhead some times the byte throughput suffers.

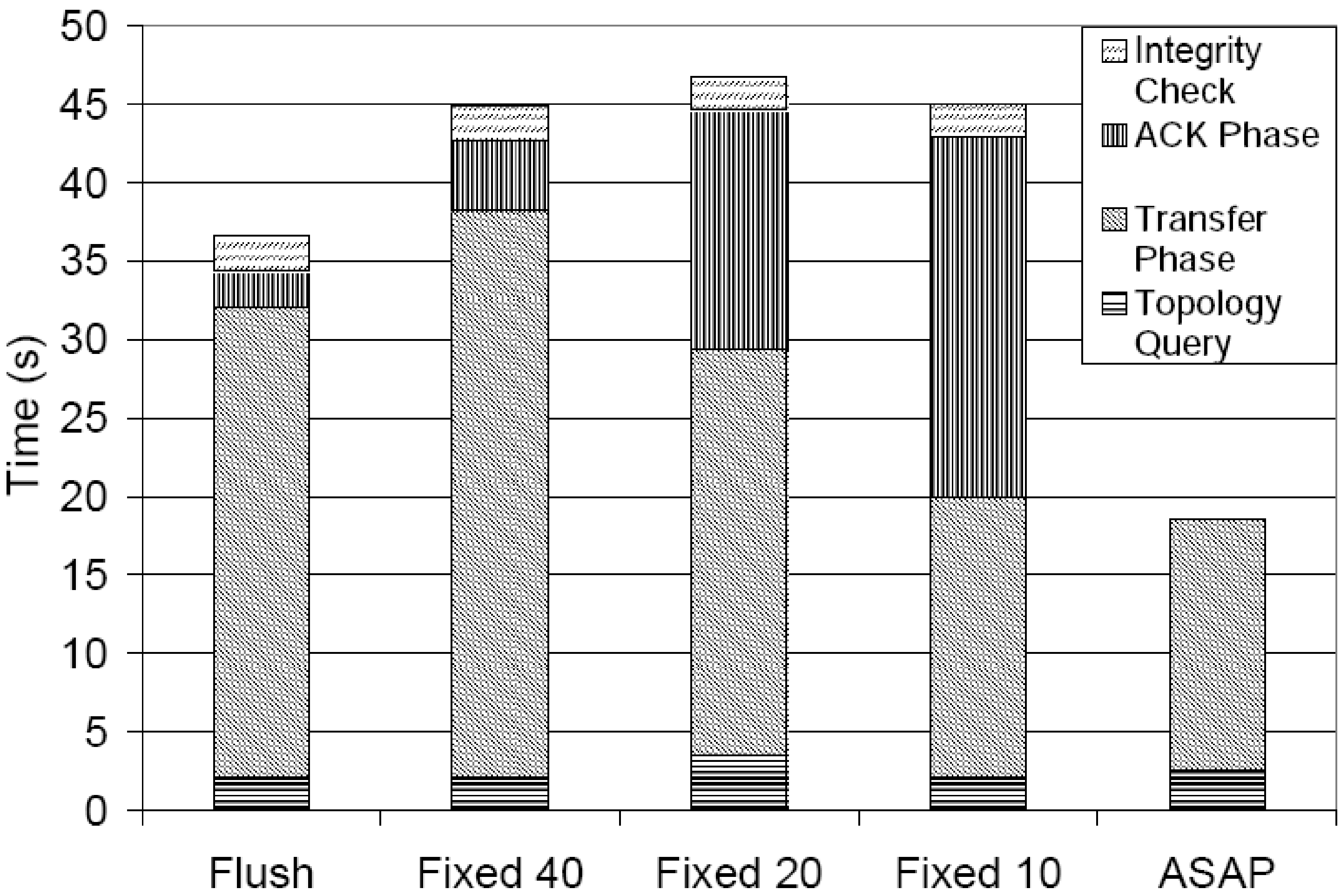
Reliability check



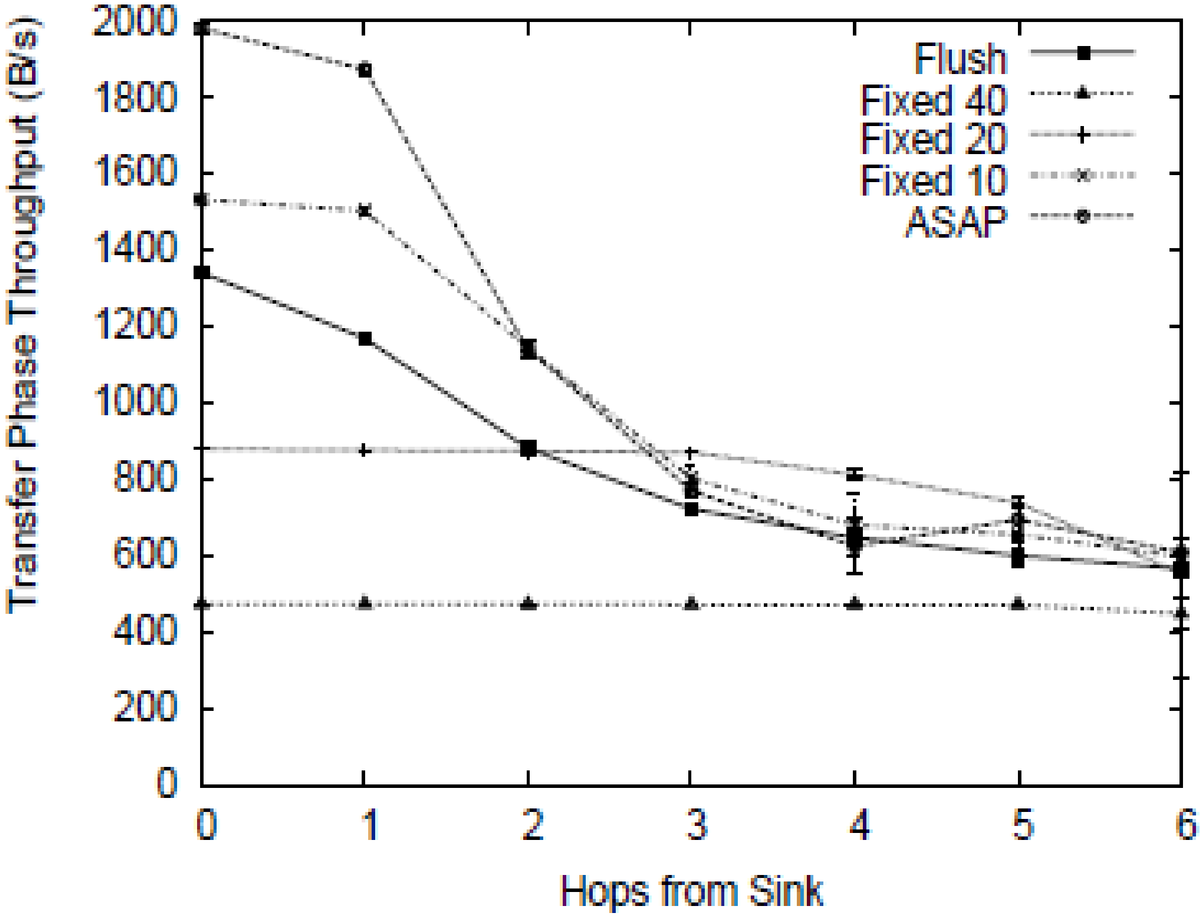
Hop – 6th

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Timing of phases

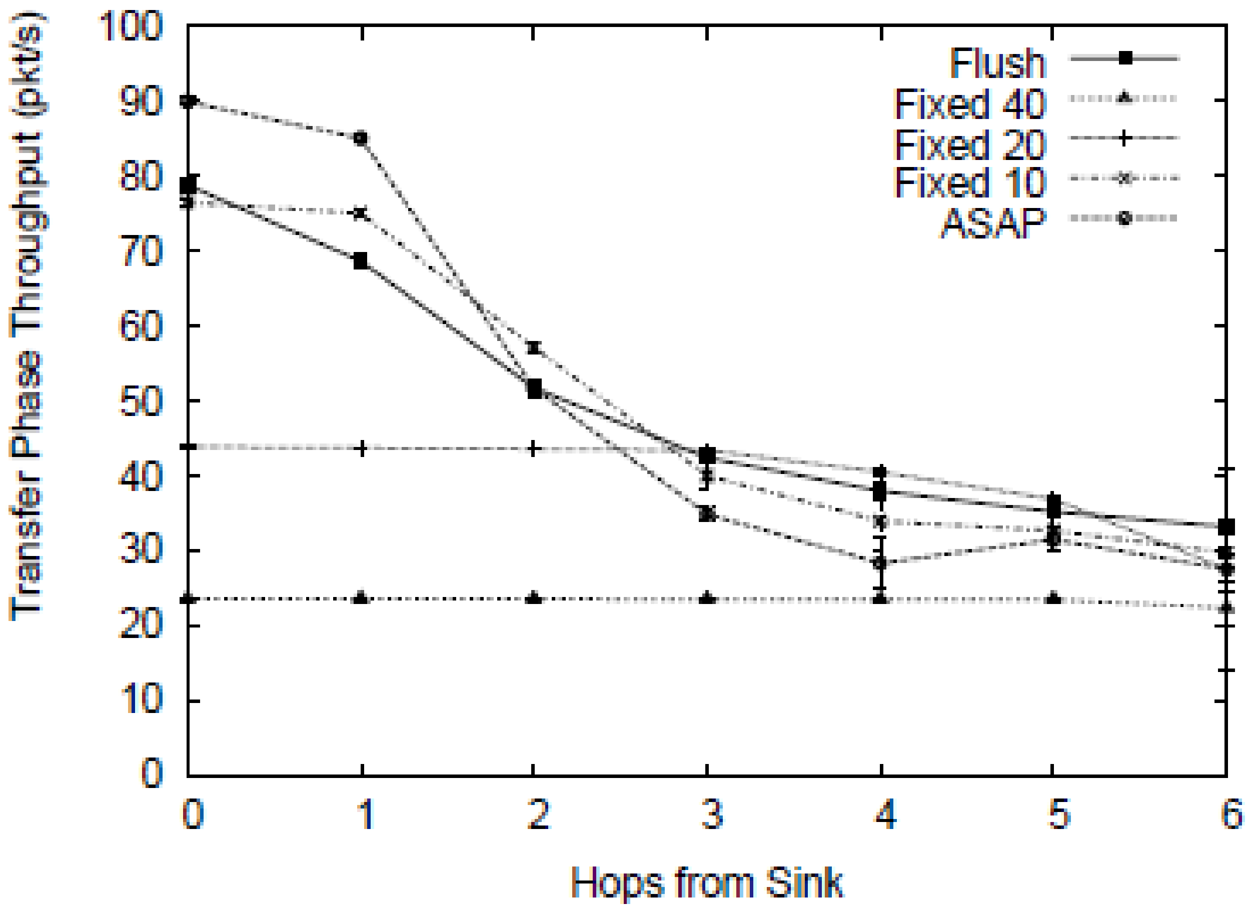


Transfer Phase Byte Throughput



Transfer phase byte throughput. Flush results
take into account the extra 3-byte rate control
Header. Flush achieves a good fraction of the throughput of "ASAP", with a 65% lower loss rate.

Transfer Phase Packet Throughput



Transfer phase packet throughput. Flush provides comparable throughput with a lower loss rate.

Real world Experiment

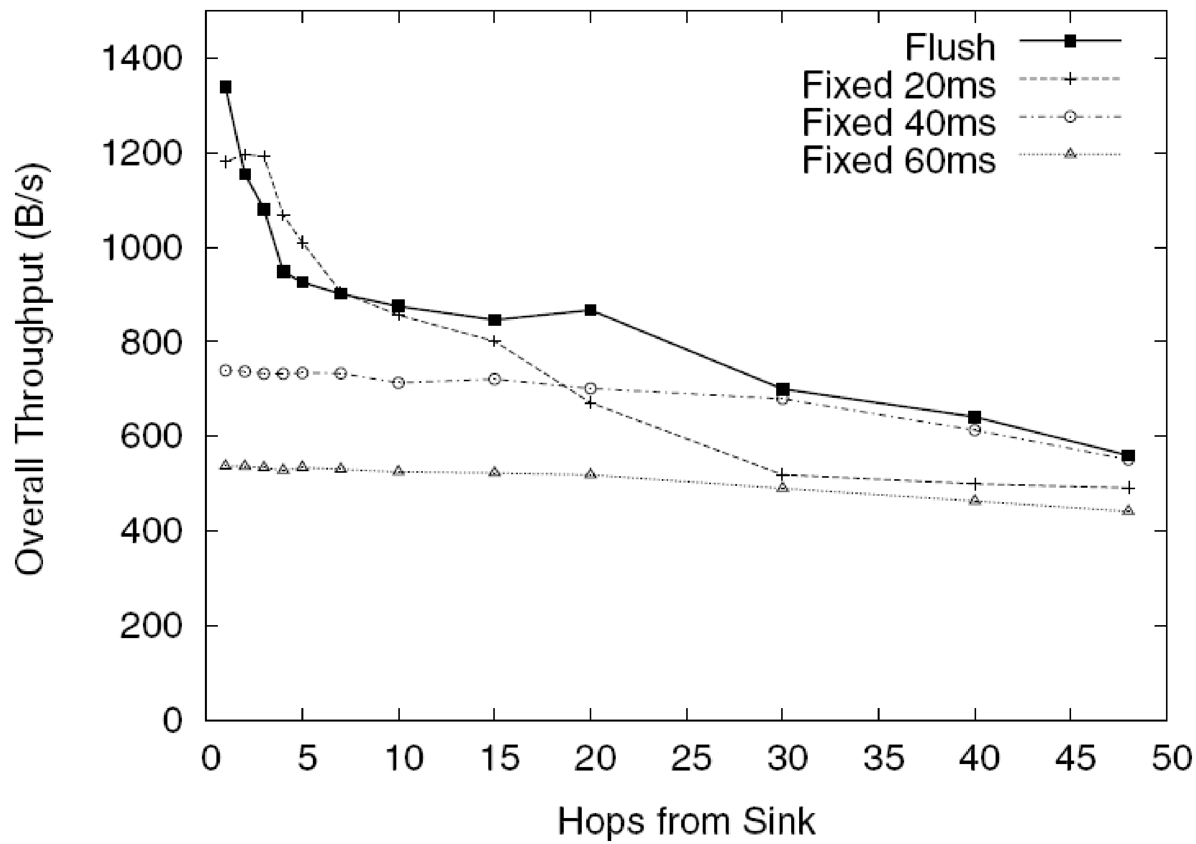
Real world experiment

79 nodes

48 Hops

3 Bytes Flush Header

35 Bytes payload



Evaluation – Memory and code footprint

	Memory Footprint		Code Size	
Component	Regular	Flush	Regular	Flush
Queue	230	265	380	1,320
Routing	754	938	76	2,022
Proto Eng	-	301	-	2,056
Delay Est	-	109	-	1,116
Total	984	1,613	456	6,514
Increase	629		6,058	

Conclusion

- VoxNet is easy to deploy and flexible to be used in different applications.
- The usage of rate based algorithms are better than the window based algorithms.
- Flush is one of the good algorithms when the nodes are somewhat in chain topology

Refrences

- VoxNet: An Interactive Rapidly Deployable Acoustic Monitoring Protocol By
  - Michel Allen Cogent Computing ARC Coventry University
  - Lewis Girod, Ryan Newton, Samuel Madden, MIT/CSAIL
  - Daniel Blumstein, Deborah Estrin, CENS, UCLA
- Flush: A Reliable Bulk Transport Protocol for Multihop Wireless Networks By
  - Sakun Kim, Rodrigo Fonsecca, Prabal Dutta, Arsalan Tavakoli, David Culler, Philip Levis, Computer Science Division, UC Berkeley
  - Scott Shenker, ICSI 1947 Center Street Berkeley, CA 94704
  - Ion Stoica, Computer Systems Lab, Satnford University

Questions?