A-GAP:
Adaptive Monitoring with Accuracy Objectives

Alberto Gonzalez Prieto, Rolf Stadler
Laboratory for Communication Networks
KTH Royal Institute of Technology
Electrical Engineering School

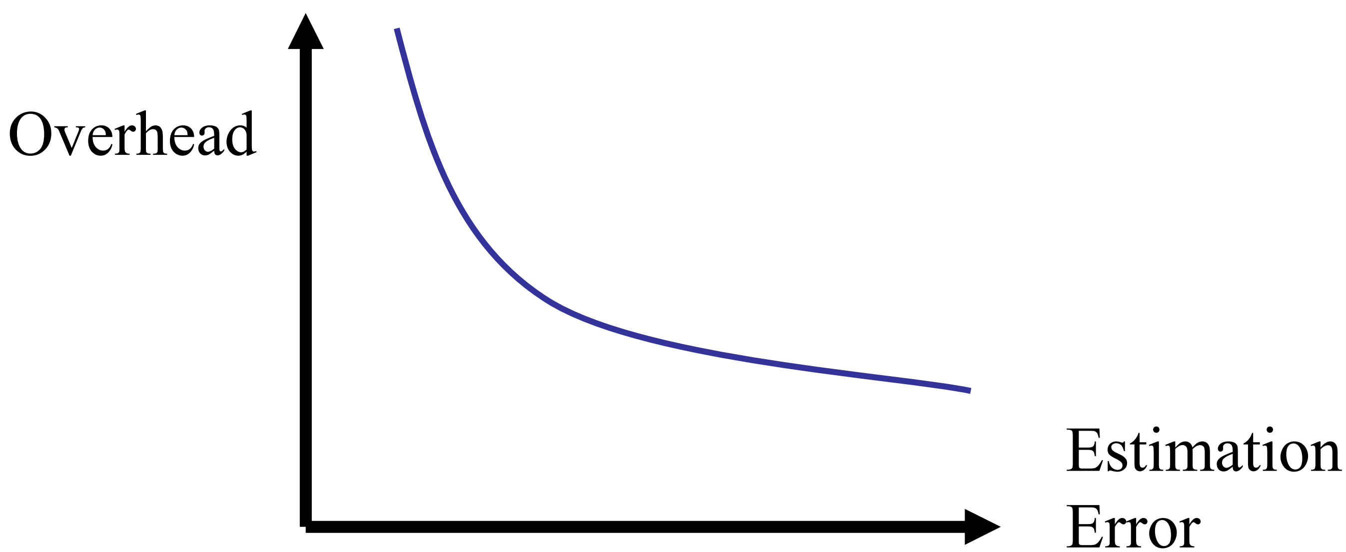
The Problem

- Find an efficient solution for **continuous monitoring** of **aggregates** with **accuracy objectives** in large-scale network environments
  - Aggregation functions, such as SUM, AVERAGE and MAX
  - Sample aggregates: total number of VoIP flows, the maximum link utilization, or a histogram of the current load across routers in a network domain

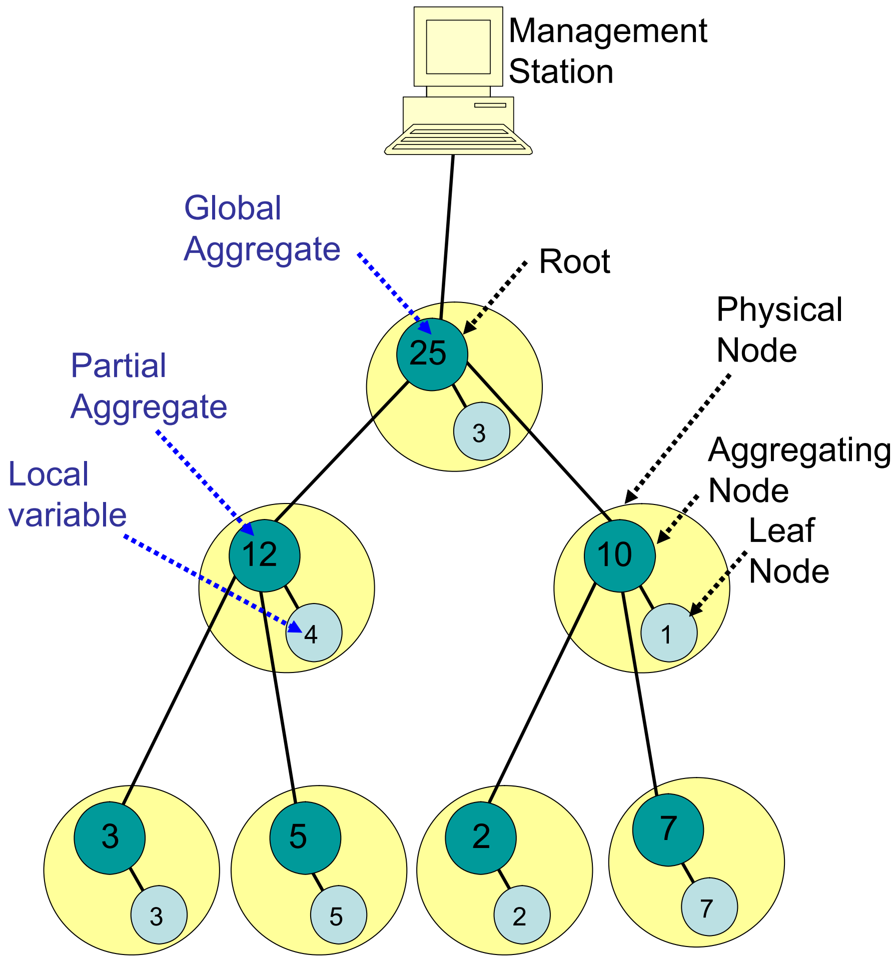
- Key Application Areas: Network Supervision, Quality Assurance, Proactive Fault Management

The Problem (2)

- Network management solutions deployed today usually provide only qualitative control of the accuracy
- **Fundamental trade-off** between accurate estimation of a variable and the management overhead in terms of traffic and processing load

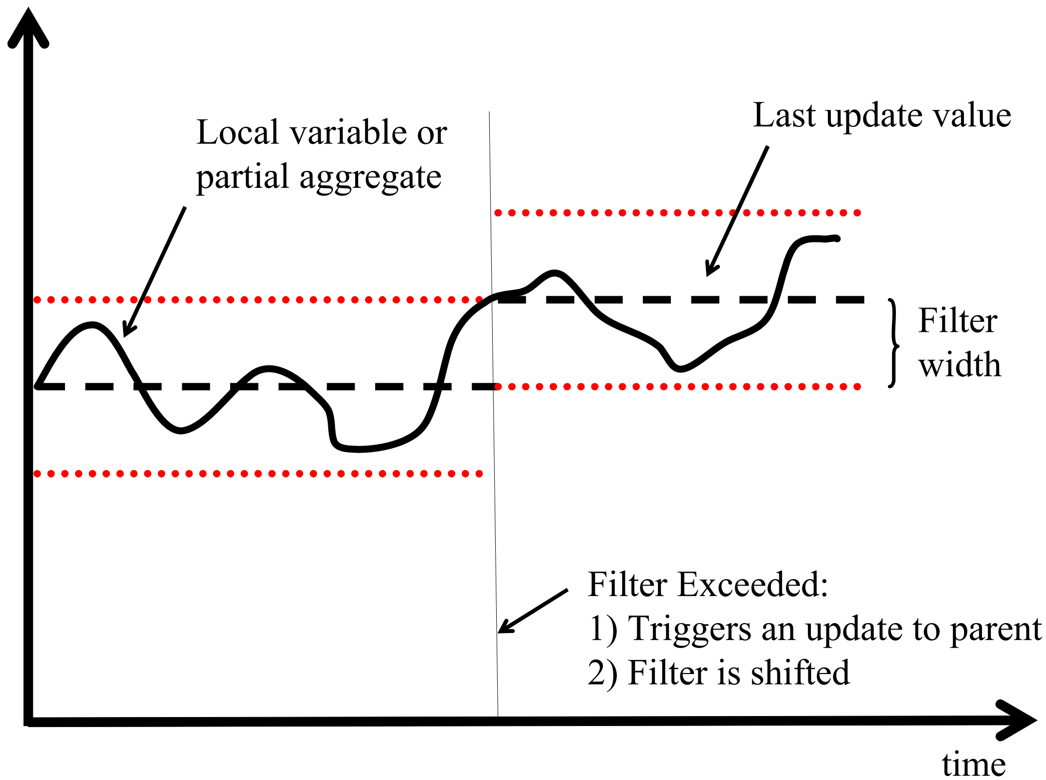


Decentralized in-network Aggregation



- **Self-organizing** Management overlay
  - Spanning tree
- Incremental aggregation
- In-network aggregation
- Push-based
- Local filter that **dynamically** adapts

Local Adaptive Filters



- Each node has a local filter
- **Controls the management overhead** by filtering updates
- **Drops updates** with small variations of its partial aggregate
- Filters **periodically adapt** to the dynamics of network environment

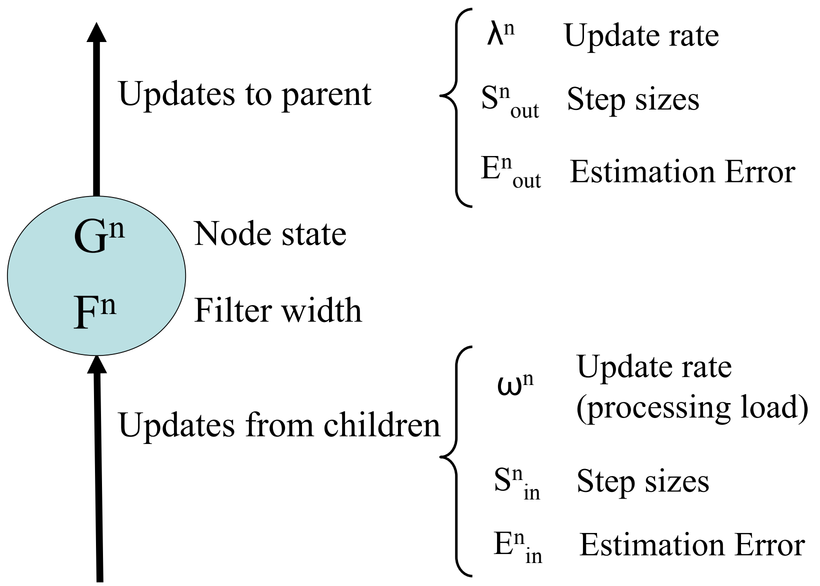
Problem Formalization

- Provide an estimation of the monitored aggregate for a given accuracy ε, with minimal overhead ω^n
  - Overhead: maximum processing load over all management processes
  - Accuracy: average error

$$\text{Minimize } \underset{n}{Max}\left\{\omega^n\right\} \quad \text{s.t.} \quad E\left(\left|E^{root}\right|\right) \le \varepsilon$$

An Stochastic Model for the Monitoring Process

- For each node *n*, the model relates
  - the error of the partial aggregate of *n*,
  - the evolution of the partial aggregate
  - the rate of updates *n* sends
  - the width of the local filter.



- Based on discrete-time Markov chains
- Permits to compute the distribution of estimation error and the overhead on each node

Stochastic Model (leaf)

Estimation of local variable evolution:

$$j^n = \begin{cases} i^n + X^n & -F^n \le i^n + X^n \le F^n \\ 0 & otherwise. \end{cases}$$

Transition Matrix:

$$t_{ij}^n = \begin{cases} P(X^n = j^n - i^n) & \left|j^n\right| \le F^n, j^n \ne 0 \\ P(X^n = -i^n) + P(F^n - i^n < X^n < -F^n - i^n) & j^n = 0 \end{cases}$$

Step Size:

$$P(S_{out}^n = s) = \begin{cases} \sum_{z=s-F^n}^{s+F^n} P(X^n = z)P(G^n = s - z) & \left|s\right| > F^n \\ \sum_{d=-F^n}^{F^n} \sum_{z=d-F^n}^{d+F^n} P(X^n = z)P(G^n = d - z) & s = 0 \\ 0 & otherwise. \end{cases}$$

Estimation Error: $E_{out}^n = G^n$

Management Overhead: $\lambda^n = (1 - P(S_{out}^n = 0))$

Stochastic Model (aggregating node)

Input → **Output**

Step Size:

$$P\left(S_{in}^{n}=s\right)=\frac{\sum_{c\in\gamma^{n}}\left(P\left(S_{out}^{c}=s\right)\cdot\Delta^{c}\right)}{\sum_{c\in\gamma^{n}}\Delta^{c}}$$

→

$$P(S_{out}^{n}=s)=\begin{cases}\sum_{k=s-F^{n}}^{s+F^{n}}P(S_{in}^{n}=k)P(G^{n}=s-k) & |s|>F^{n}\\ \sum_{d=-F^{n}}^{F^{n}}\sum_{k=d-F^{n}}^{d+F^{n}}P(S_{in}^{n}=k)P(G^{n}=d-k) & s=0\\ 0 & otherwise\end{cases}$$

Estimation Error:

$$E_{in}^{n}=\sum_{c\in\gamma^{n}}E_{out}^{c}$$

→

$$E_{out}^{n}=E_{in}^{n}+G^{n}$$

Management Overhead:

$$\omega^{n}=\sum_{c\in\gamma^{n}}\lambda^{c}.$$

→

$$\lambda^{n}=\Delta^{n}(1-P(S_{out}^{n}=0))$$

Transition Matrix:

$$t_{ij}^{n}=\begin{cases}P(S_{in}^{n}=j^{n}-i^{n}) & \left|j^{n}\right|\le F^{n},\ j^{n}\ne 0\\ P(S_{in}^{n}=-i^{n})+P(F^{n}-i^{n}<S_{in}^{n}<-F^{n}-i^{n}) & j^{n}=0\end{cases}$$

A-GAP: A Distributed Heuristic

- The global problem is **mapped onto a local problem** for each node

$$\text{Minimize } \underset{\pi}{Max}\{\omega^{\pi}\} \quad \text{s.t.} \quad E\left(\left|E_{out}^{n}\right|\right) \leq \varepsilon^{n}$$

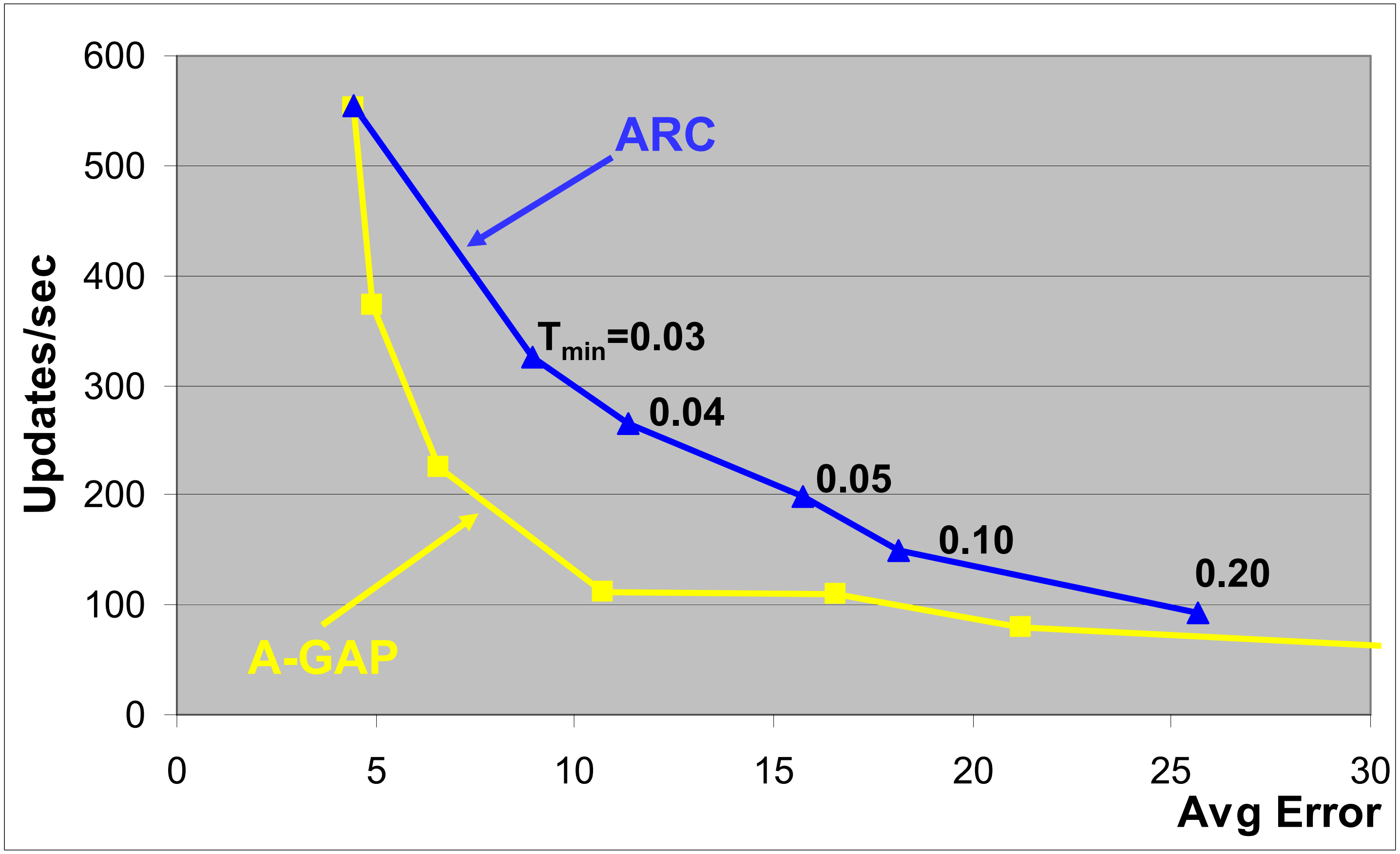
- Attempts to minimize the maximum processing load on all nodes by minimizing the load within each node's neighborhood
- Filter computation: **decentralized** and **asynchronous**
- Each node **independently** runs a control cycle:

every τ seconds
  request model variables from all children γ^n
  select $\Omega \subseteq \gamma^n$ n children
  compute new filters for Ω
  compute new accuracy objectives for γ^n
  send new filters to Ω and objectives to γ^n
  compute local variables

Evaluation through Simulation

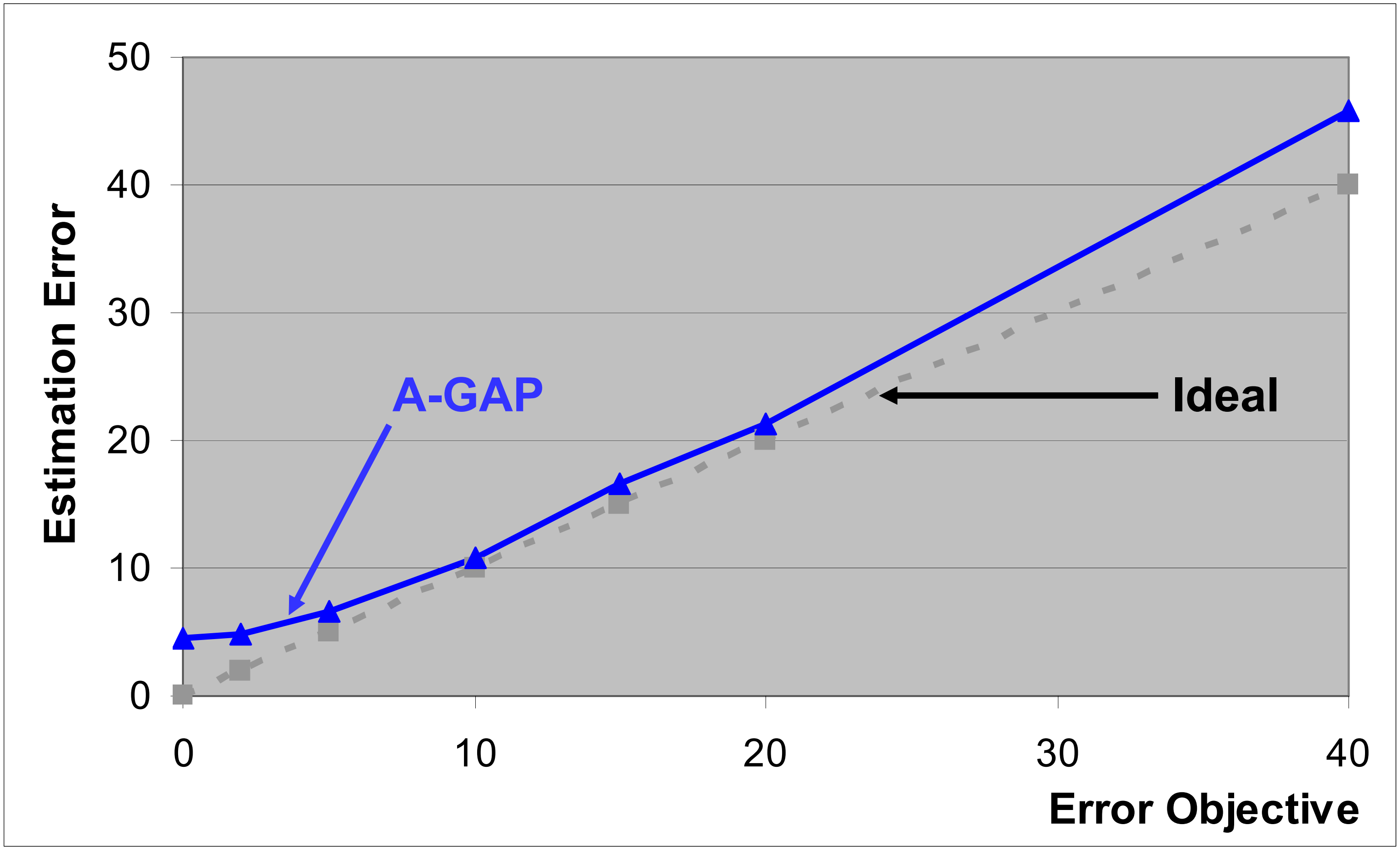
- Different overlay **topologies**:
  - Abovenet (654 nodes, 1332 links)
  - Grids: 25, 85, 221, 613 nodes (4 neighbors)
- Monitored variable: Number of **http flows** in the domain
- **Real traces**
  - From two 1 Gbit/s links that connect University of Twente to a research network
- Control cycle τ=1 sec

Trade-off: accuracy vs overhead



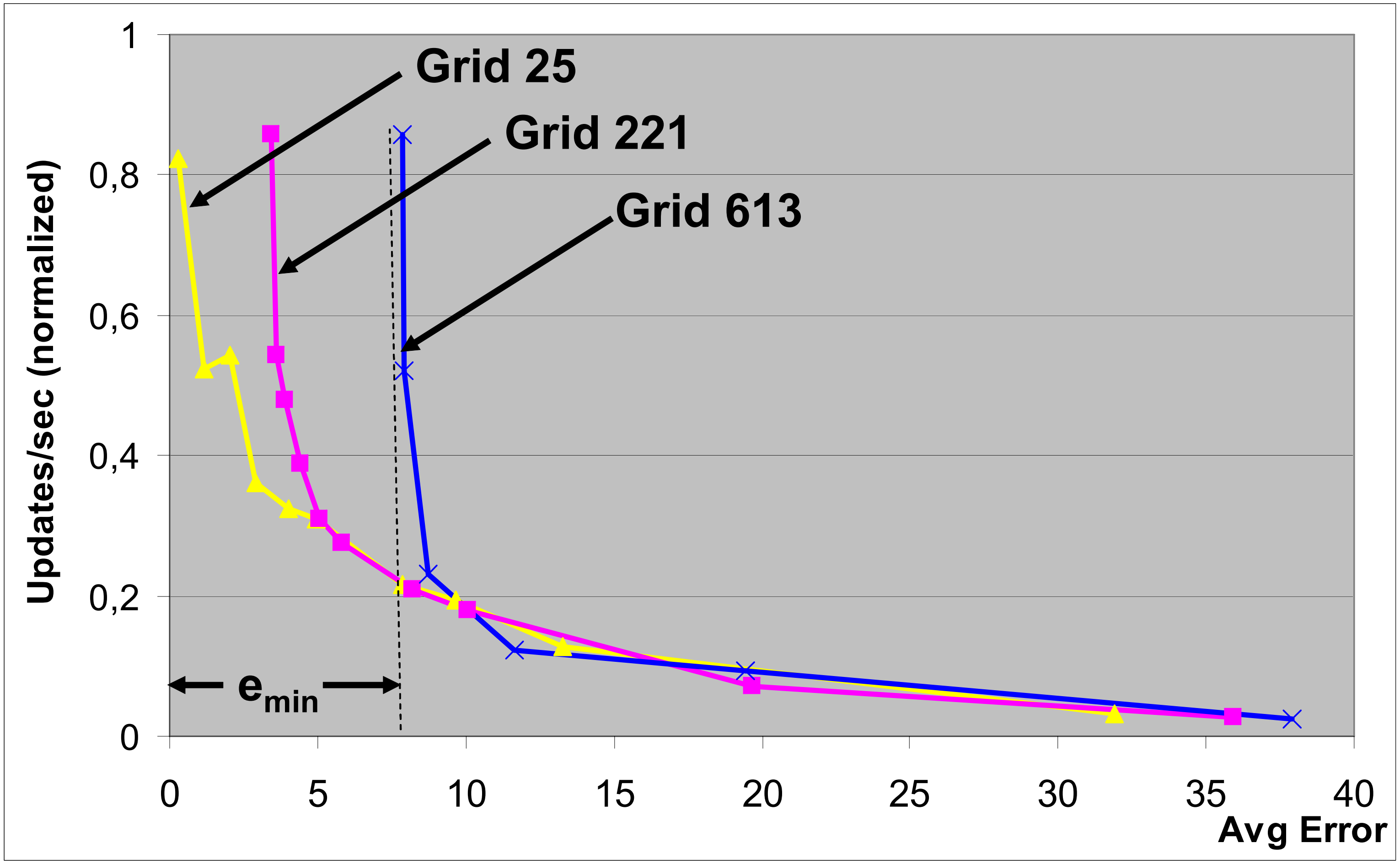
- Overhead **decreases monotonically**
- Overhead depends on the **changes of the aggregate**, not on its value.
- A-GAP **outperforms** a rate-control scheme (ARC)

Meeting the accuracy objective



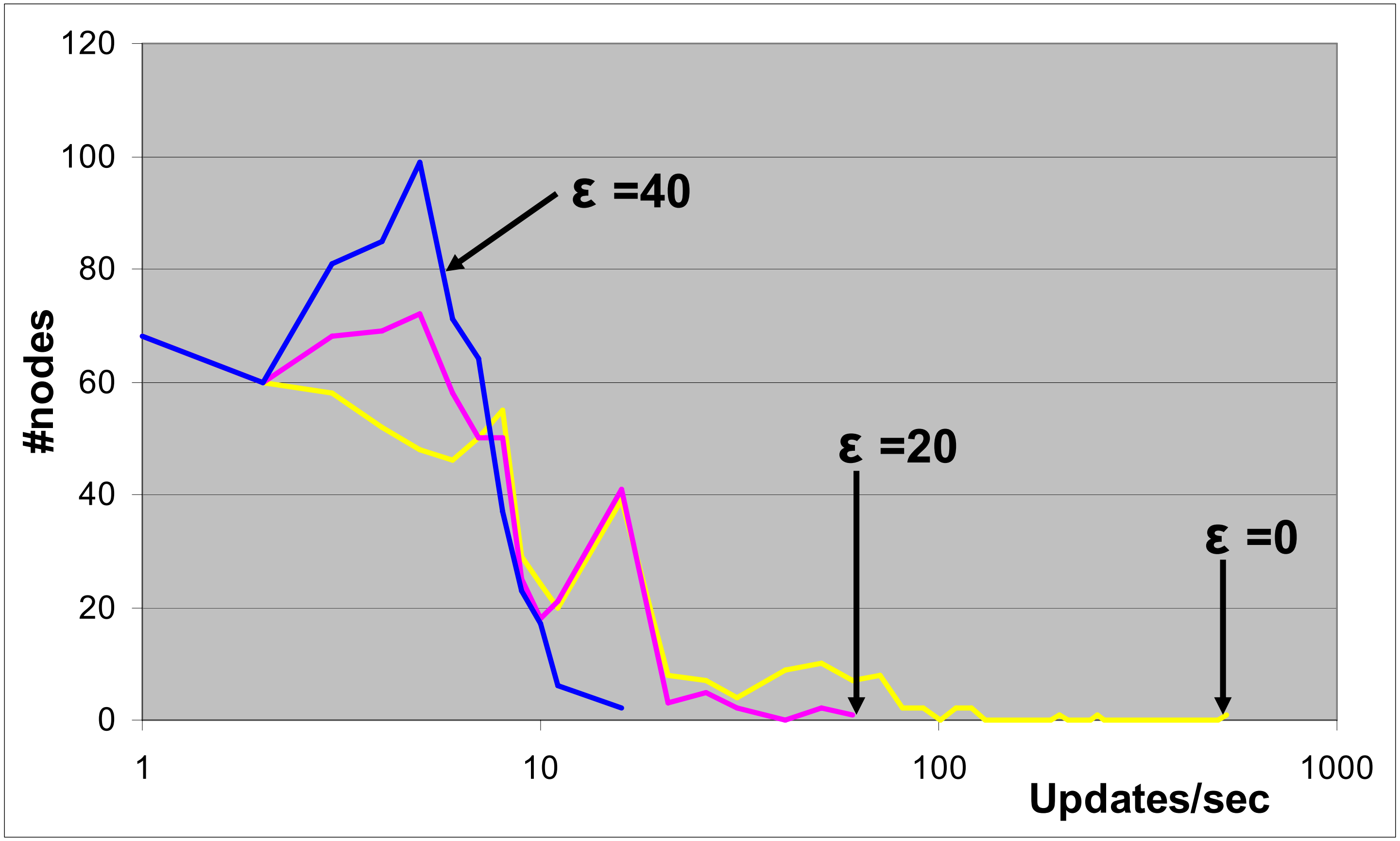
Network and processing **delays** affect estimation error

Scalability



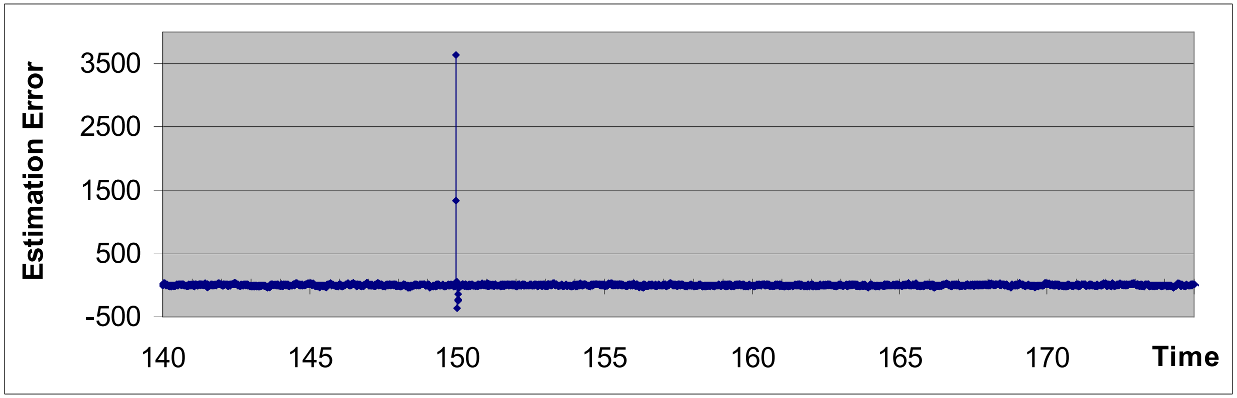
- The minimum error e_{min} increases with the network size
- Maximum load increases linearly with network size for same error objective

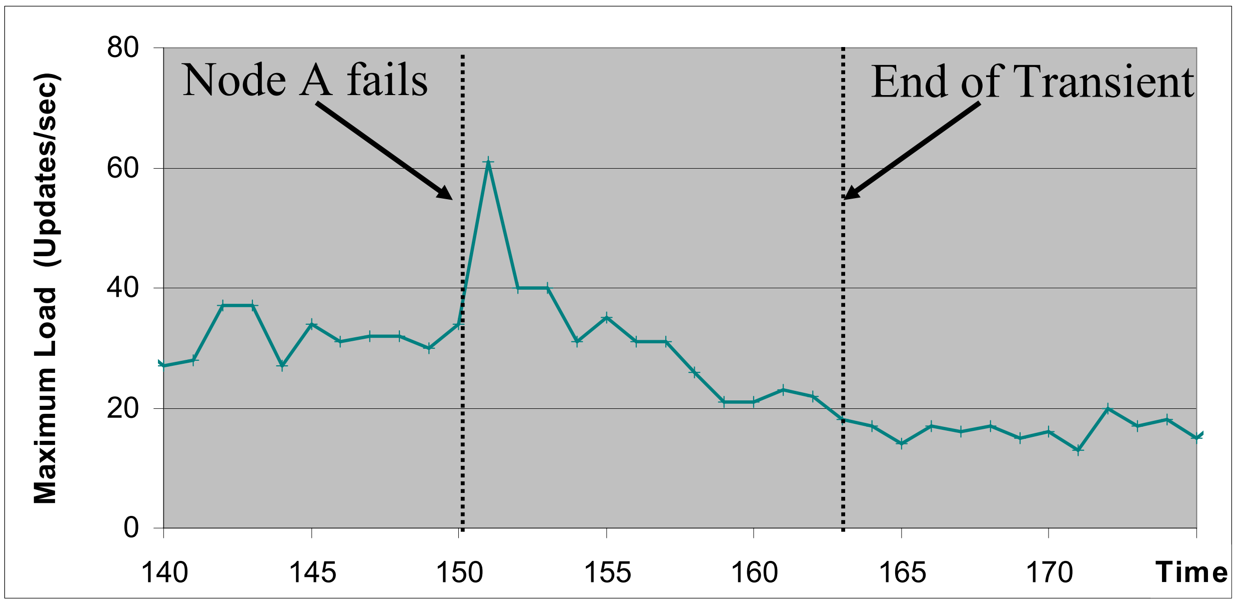
Load Distribution



- Allowing for a larger error reduces both the maximum load and the average load
- The larger the error we allow, the better the system balances the load

Robustness



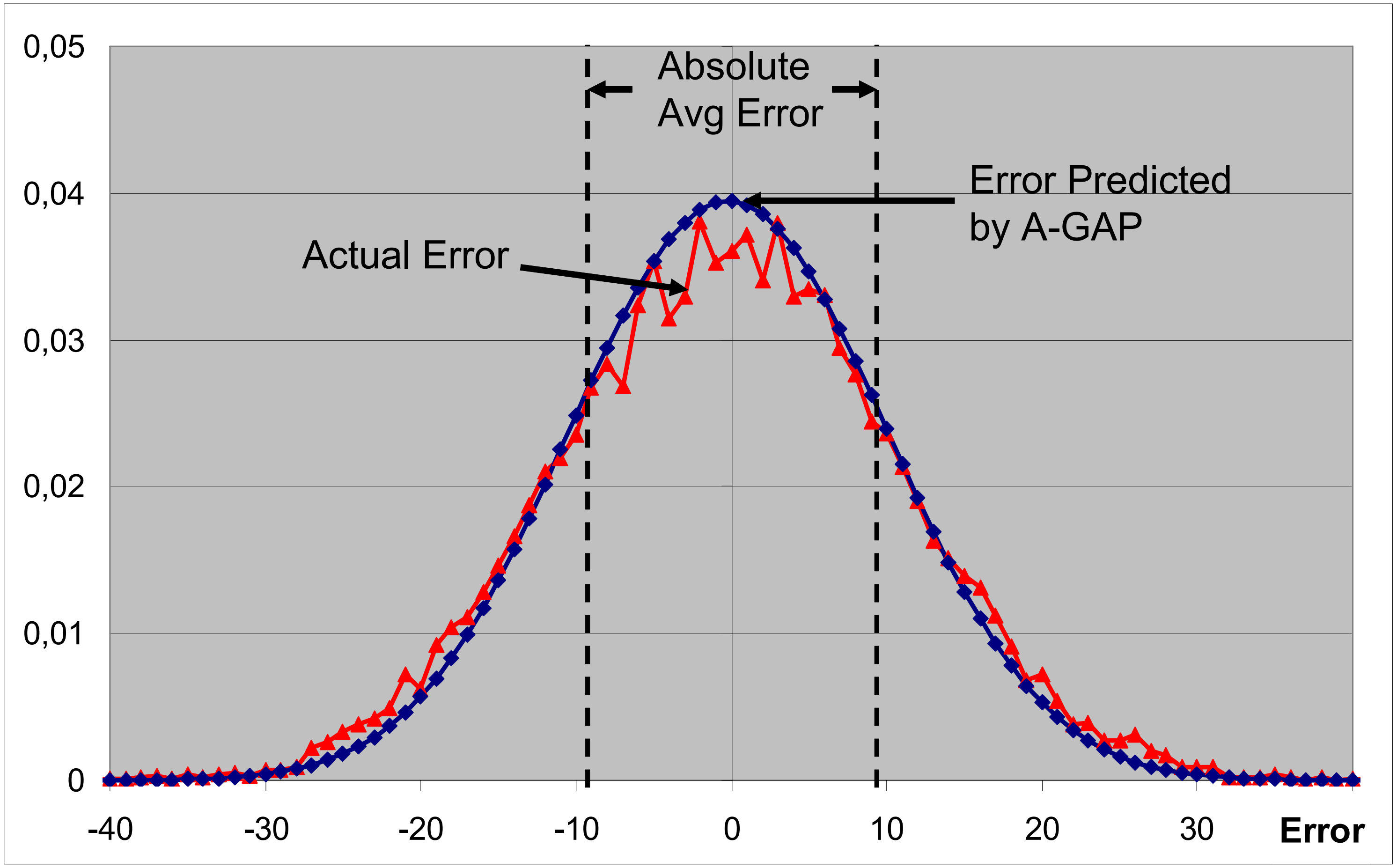


- Estimation error: several spikes (positive and negative) during a sub-second transient period
- Overhead: a single peak with a long transient

Robustness (2)

			Maximum Estimation Error at the Root Node		Transient Period Duration (error)		Maximum Load (relative to average load)		Transient Period Duration (load)	
Node	Dist. From root	#Nodes subtree	ε=3	ε=20	ε=3	ε=20	ε=3	ε=20	ε=3	ε=20
A	*1*	*100*	*4912*	*4910*	*0,12*	*0,1*	*139 %*	*268 %*	*3*	*19*
B	1	*30*	*1421*	*1429*	*0,12*	*0,1*	*Ø*	*227 %*	*Ø*	*19*
C	4	*28*	*1432*	*1439*	*0,12*	*0,11*	*Ø*	*239 %*	*Ø*	*11*
D	4	*7*	*312*	*303*	*0,02*	*0,02*	*Ø*	*129 %*	*Ø*	*11*
E	8	*6*	*310*	*278*	*0,02*	*0,02*	*Ø*	*Ø*	*Ø*	*Ø*
F	8	*3*	*171*	*130*	*0,02*	*0,02*	*Ø*	*Ø*	*Ø*	*Ø*

Error Prediction by A-GAP vs Actual Error



- **Accurate prediction** of the error distribution
- Maximum error >> average error (one order of magnitude)

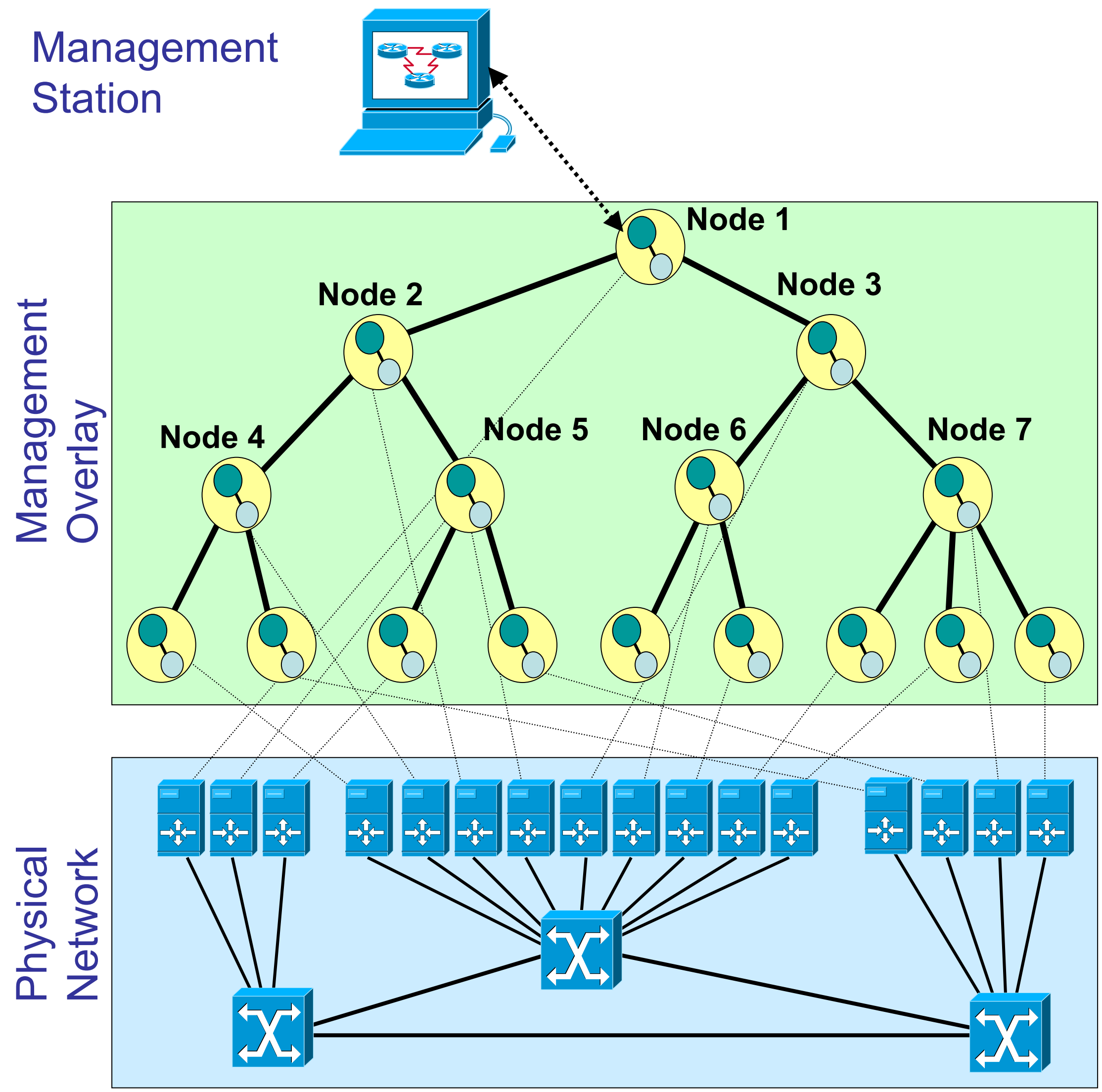
Contribution

- A-GAP: a protocol for continuous monitoring with accuracy objectives
  - Decentralized and asynchronous
  - Accuracy objective is expressed as the **average error**
  - Evaluation through **simulation**
  - **Control the trade-off** between estimation accuracy and protocol overhead
  - **Effective:** significant saving in overhead (up to 97%)
  - **Estimation error prediction** in real-time
  - **Dynamically adapts** to changes in evolution of local management variables, network topology, and node failures

A-GAP Prototype

Runs on KTH Testbed

- 16 nodes
- 3 switches
- java



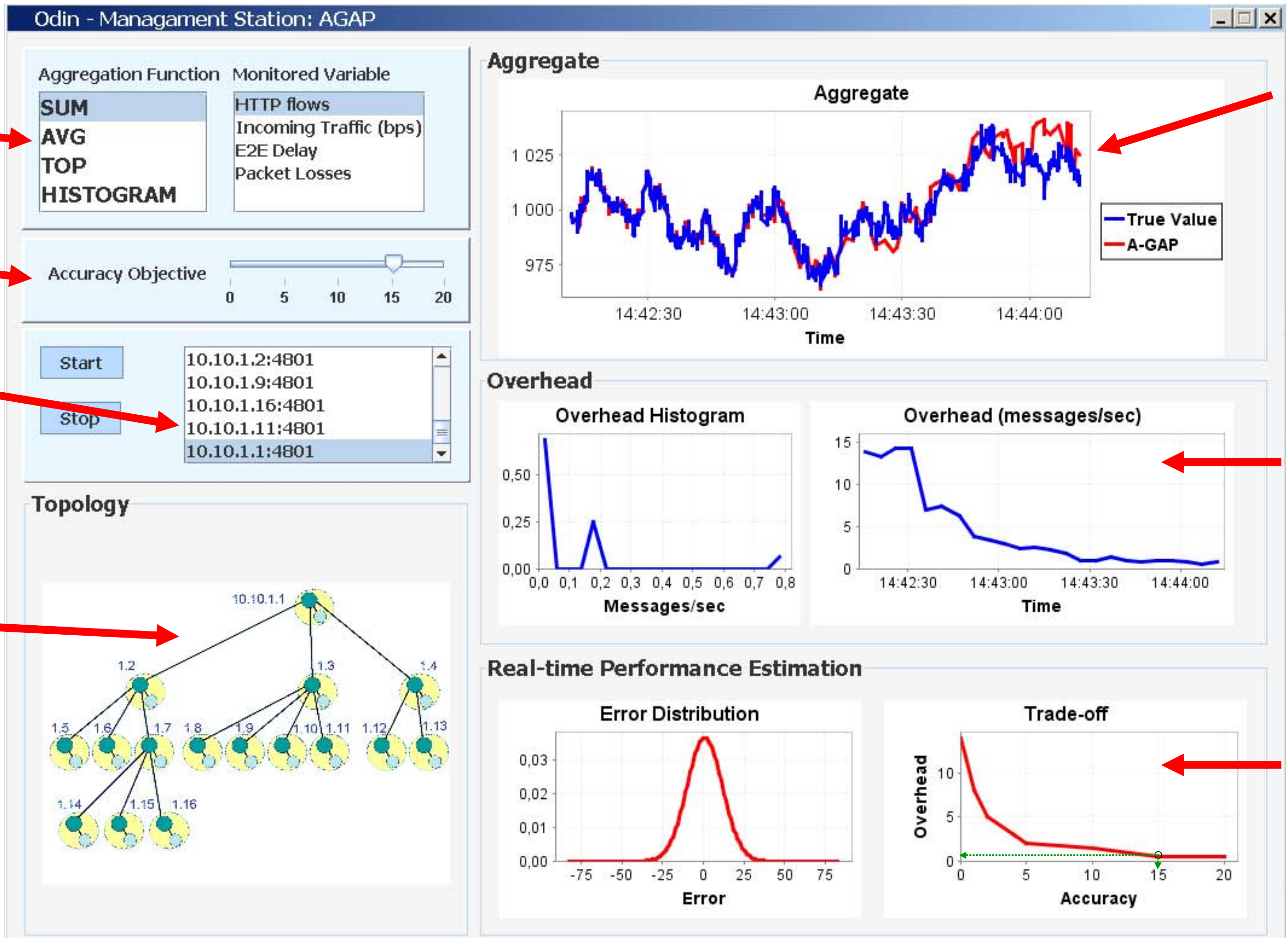
Implementation Evaluation

Select Aggregation Function

Select Accuracy Objective

Select Root Node

Show Aggregation Tree



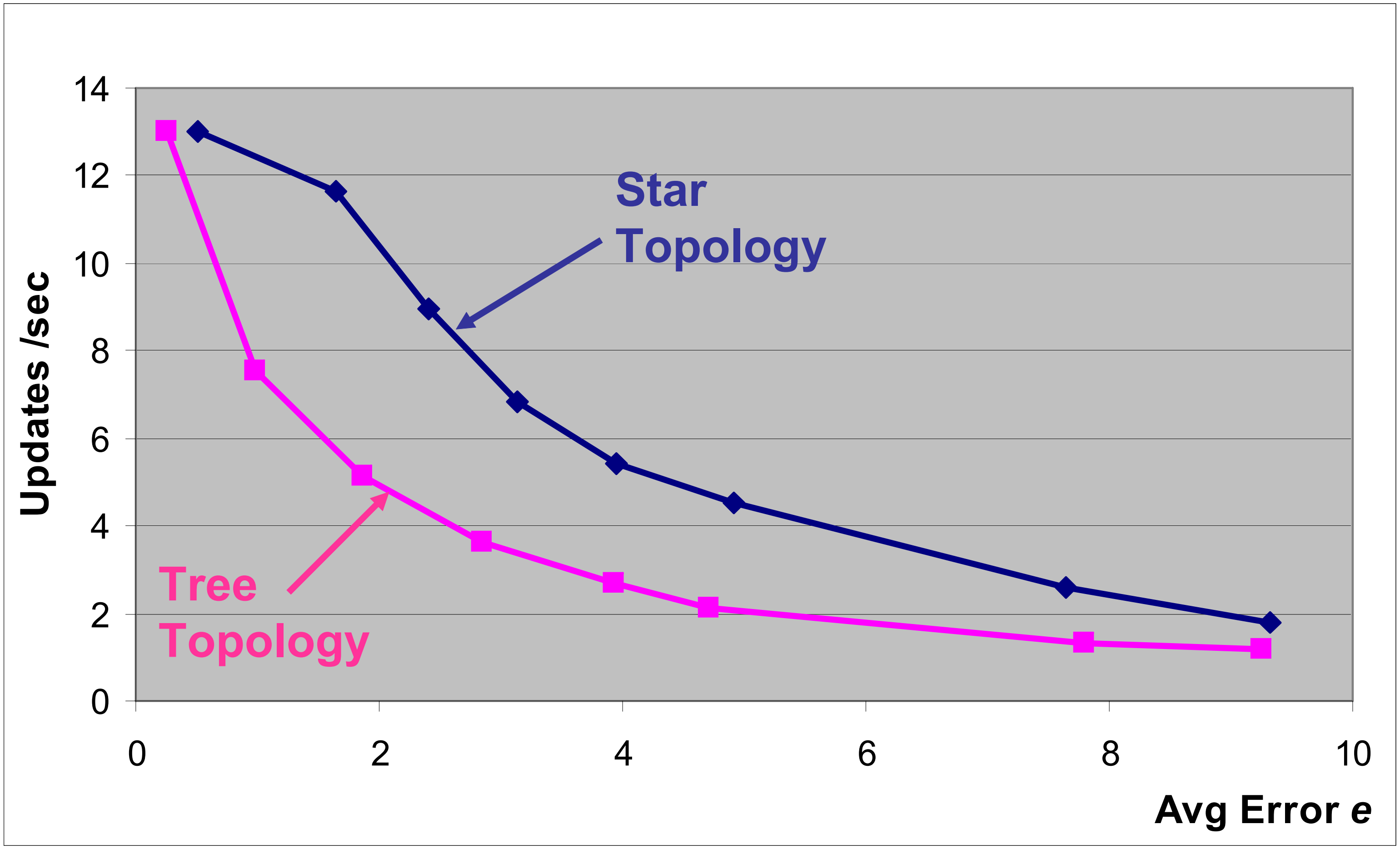
Evolution of the Aggregate

(True Value and A-GAP Estimation)

Overhead Distribution and Evolution

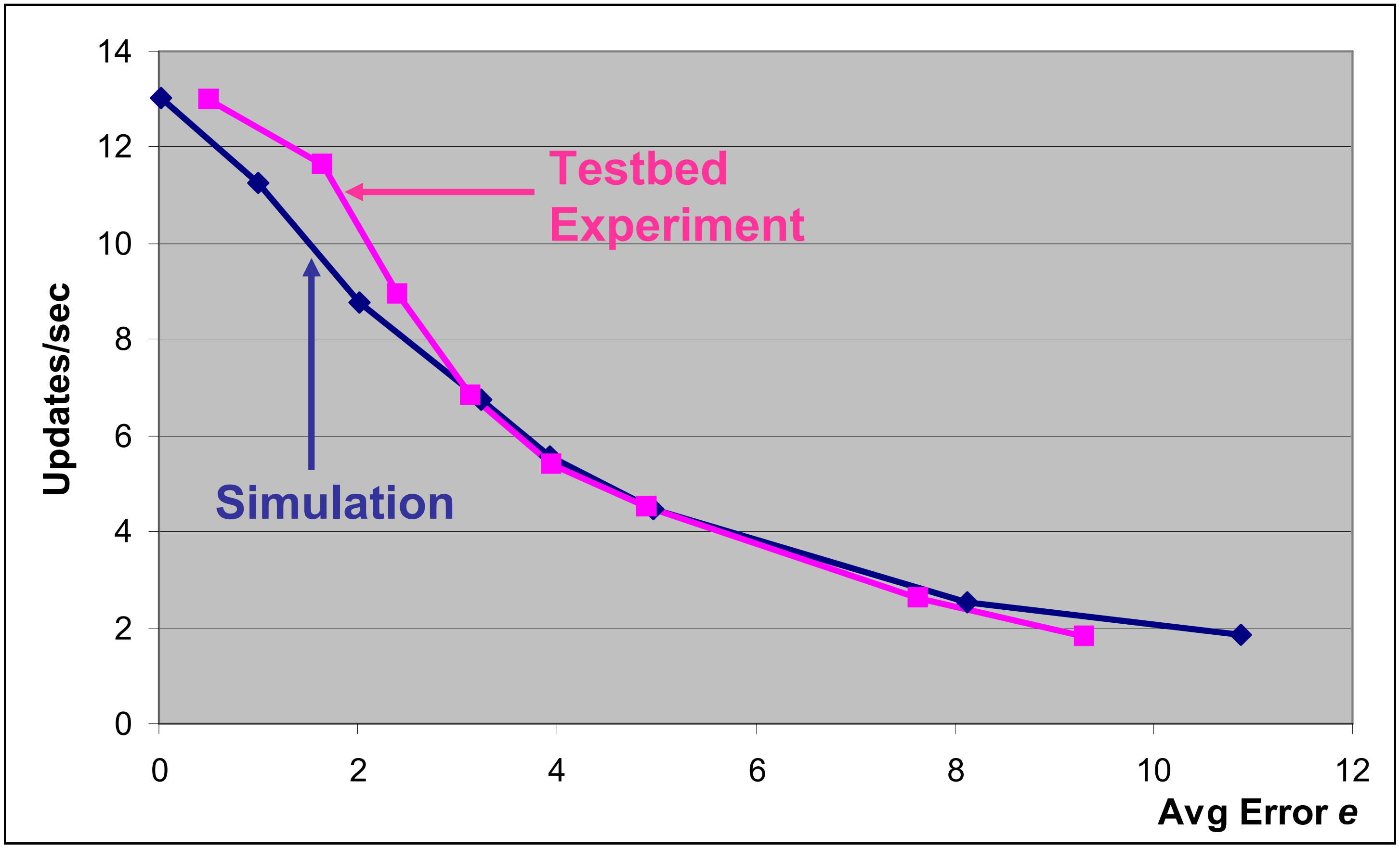
Real-time Estimation of Error Distribution and Trade-off

Trade-off: accuracy vs overhead



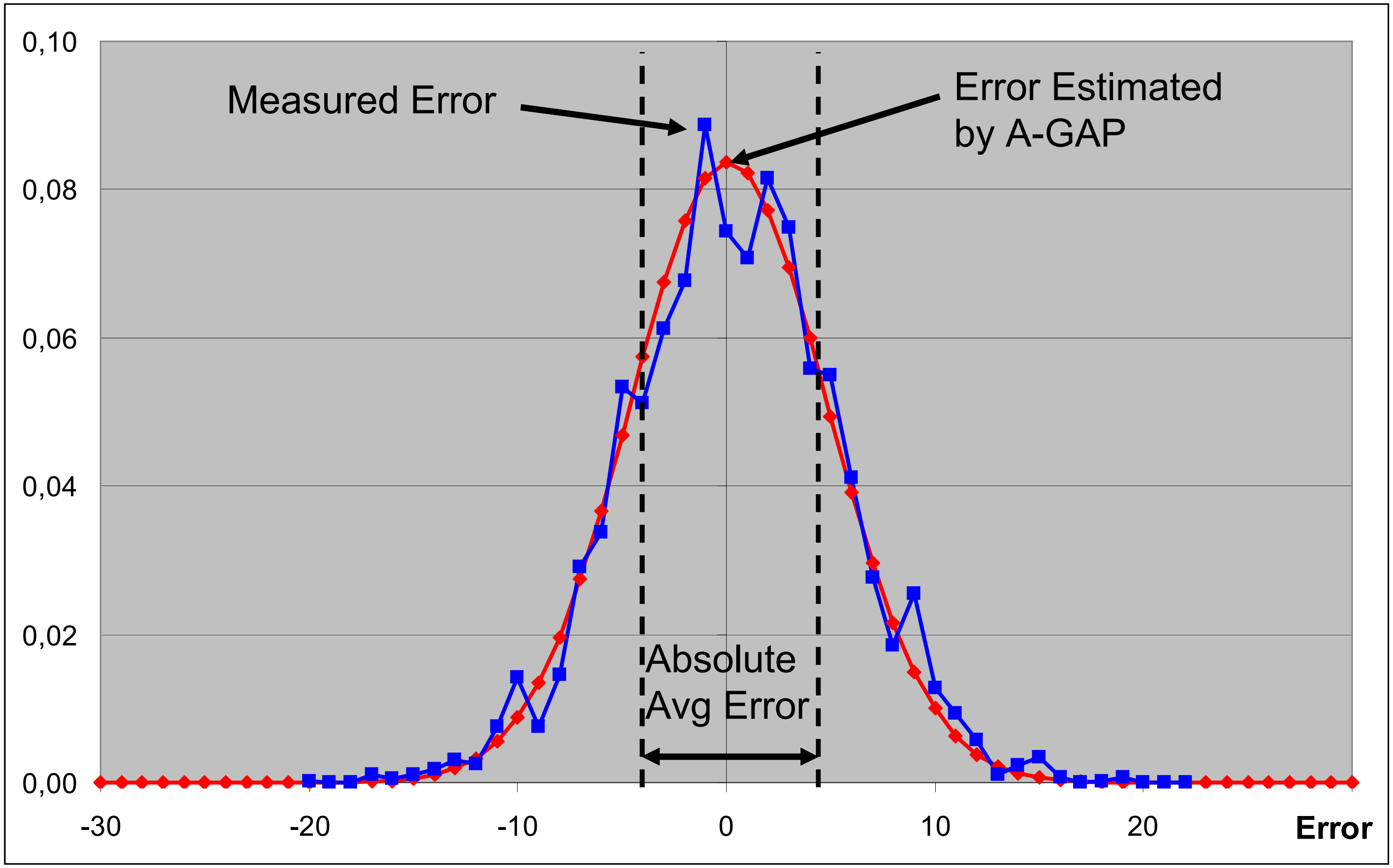
- Overhead **decreases monotonically**
- Impact of the **overlay topology** (factor of 2)
  – positive and negative errors compensate

Trade-off: accuracy vs overhead (2)



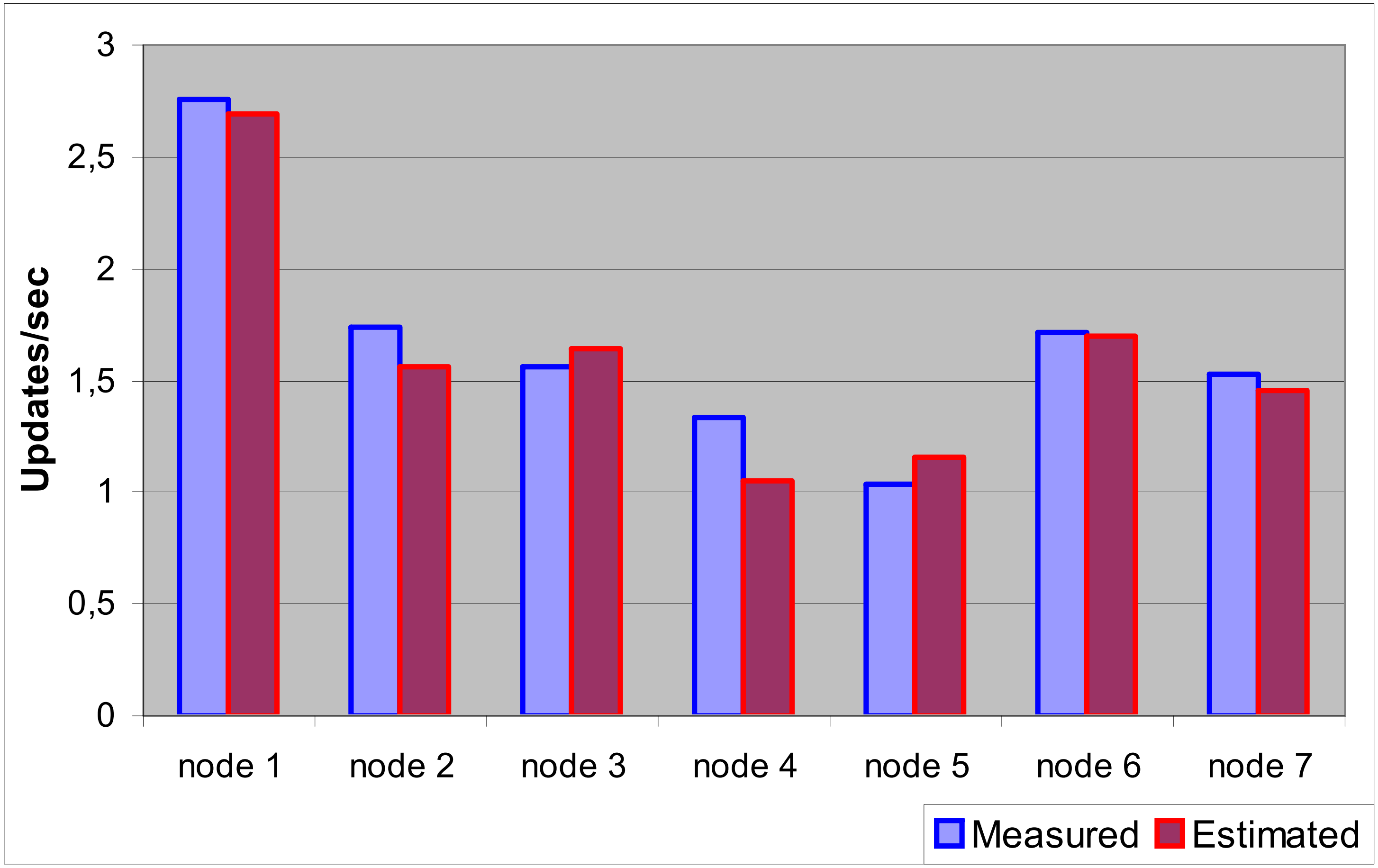
- Curves are very close (difference in overhead is below 3,5%)
- Simulation model validation

Error Estimation by A-GAP vs Actual Error



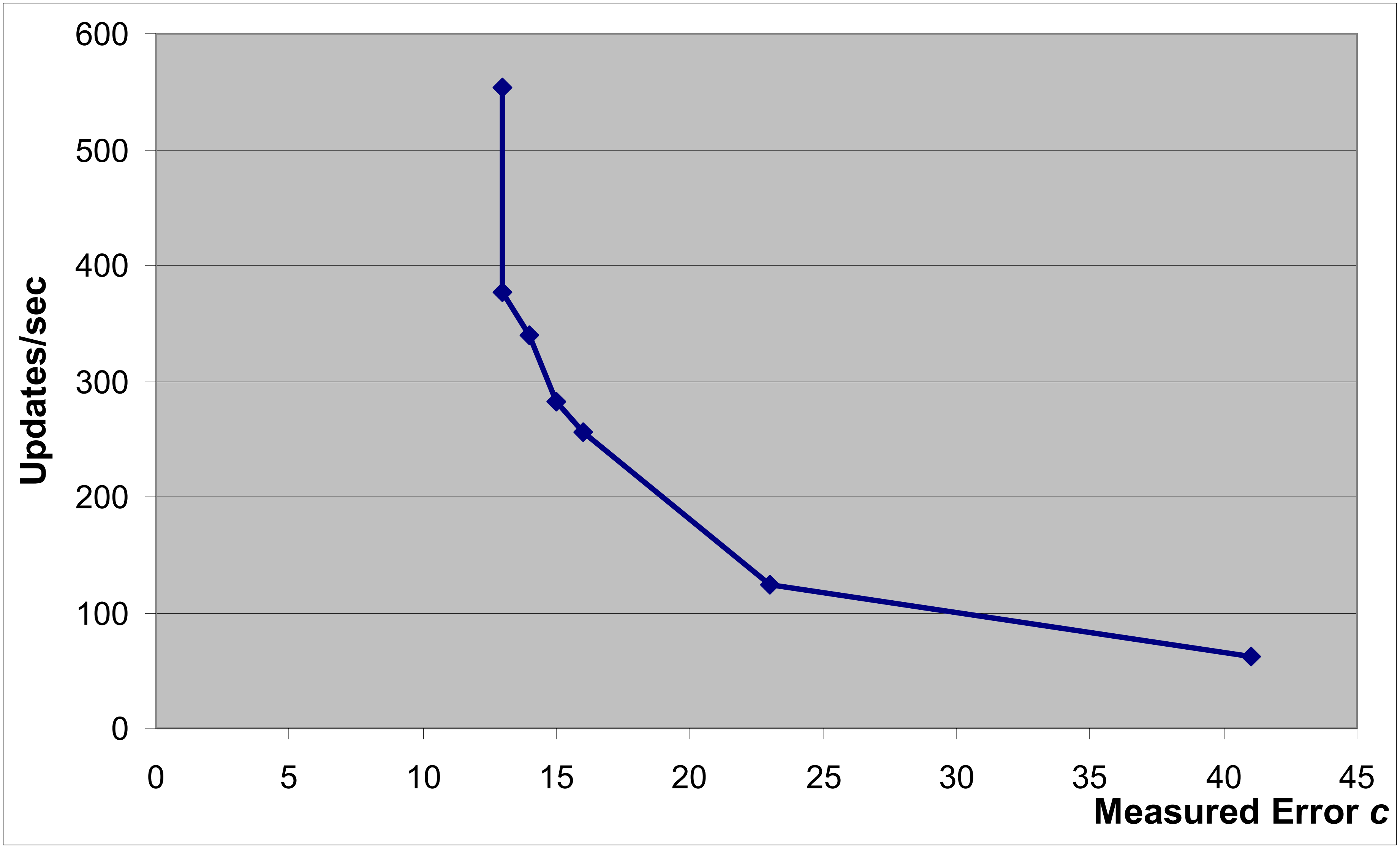
- **Accurate estimation** of the error distribution
- Maximum error >> average error (one order of magnitude)

Overhead Estimation by A-GAP vs Actual Overhead

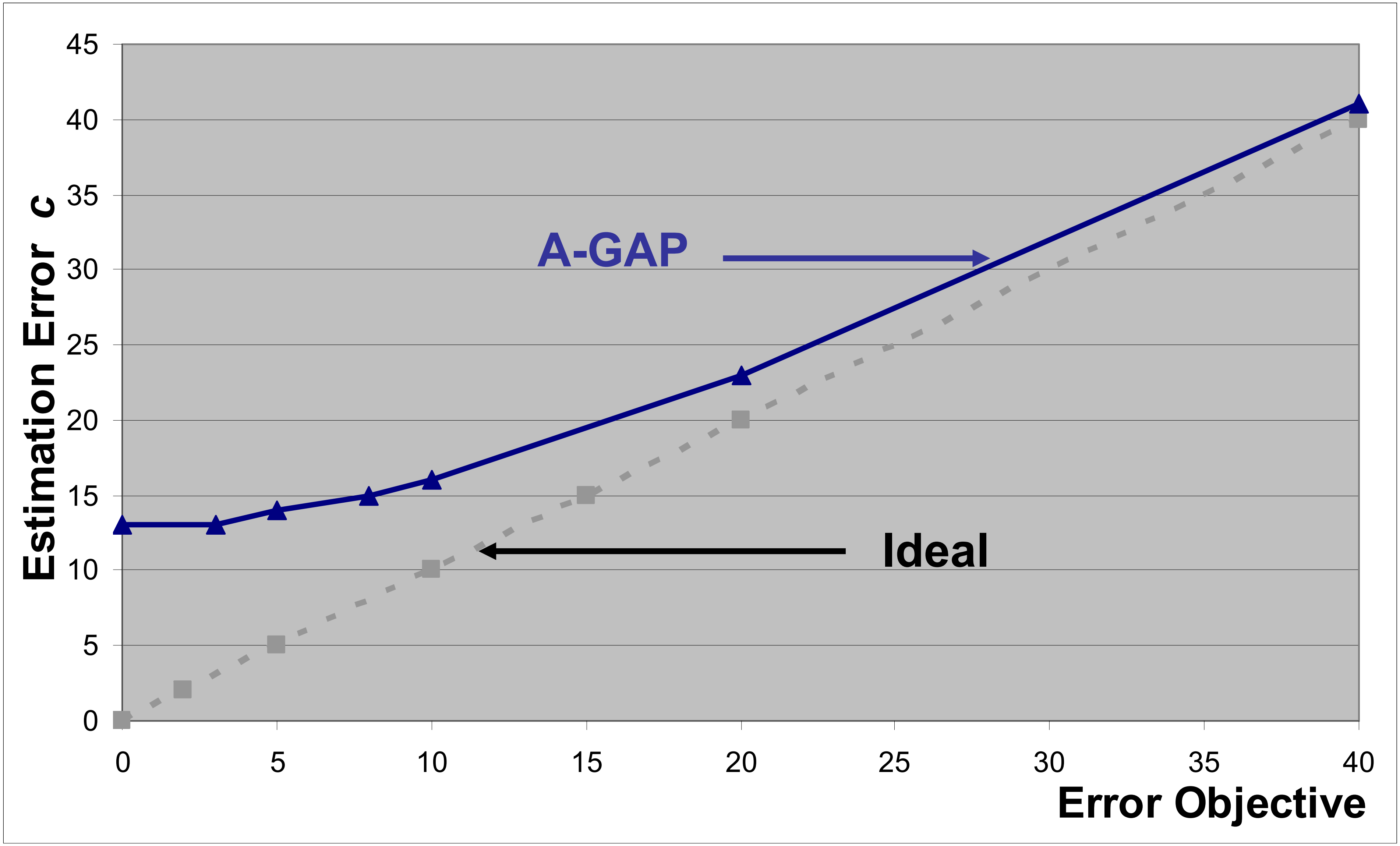


- **Accurate estimation** of the overhead
- tends to be more accurate for **nodes closer to the root**

Supporting Percentile Error Objectives



Supporting Percentile Error Objectives (2)



Contribution

- A-GAP: a protocol for continuous monitoring with accuracy objectives
  - Decentralized and asynchronous
  - Accuracy objective is expressed as the **average error**
  - Evaluation through **simulation and prototype**
  - **Control the trade-off** between estimation accuracy and protocol overhead
  - **Effective:** significant saving in overhead (up to 97%)
  - **Estimation error prediction** in real-time
  - **Dynamically adapts** to changes in evolution of local management variables, network topology, and node failures