Organic Self-organizing Bus-based Communication Systems

Tobias Ziermann, Stefan Wildermann, Jürgen Teich

Hardware-Software-Co-Design

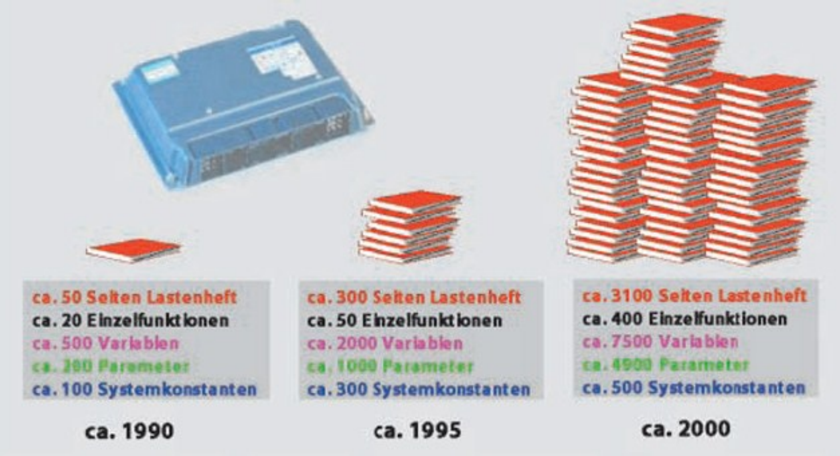
Universität Erlangen-Nürnberg

`tobias.ziermann@informatik.uni-erlangen.de`

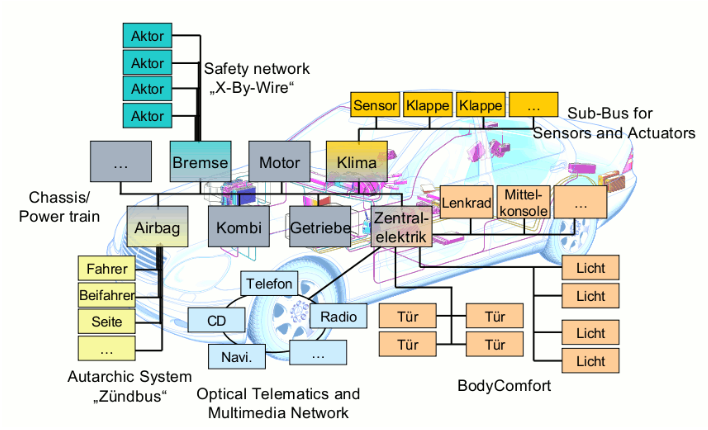
15.09.2011

Motivation

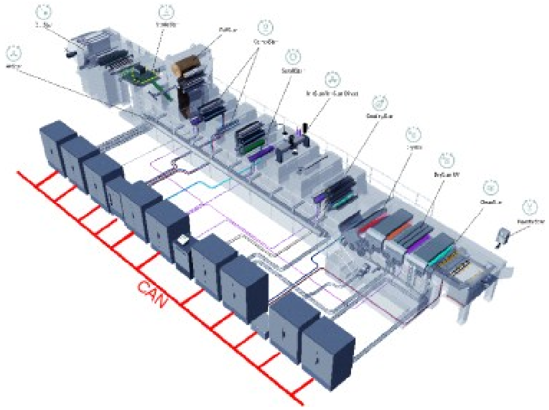
- Increasing complexity in distributed embedded systems
- Increasing demand on the communication
- Wired buses are used today



Source: VW



Source: Daimler AG

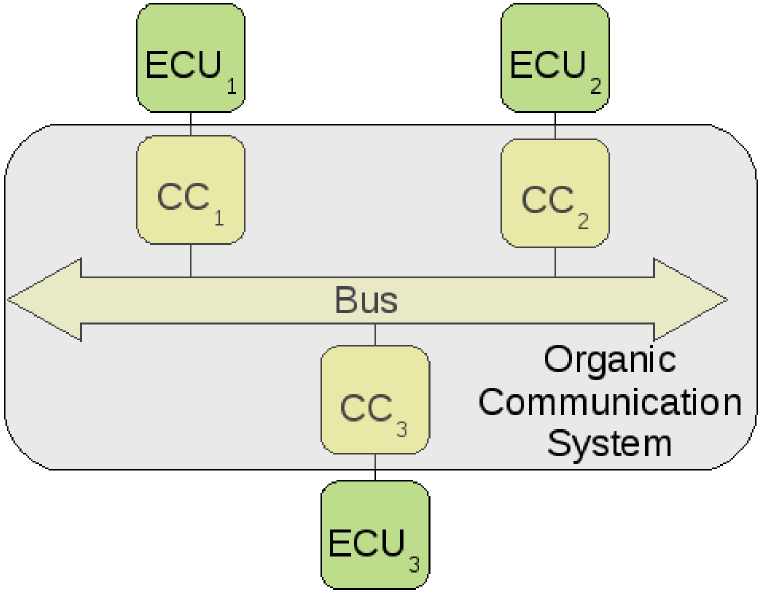


Source: Heidelberger Druckmaschinen AG

Goals of OrganicBus

- Planning of the communication is very difficult
  - Hand-based procedures are not practical
  - Design tools are pessimistic

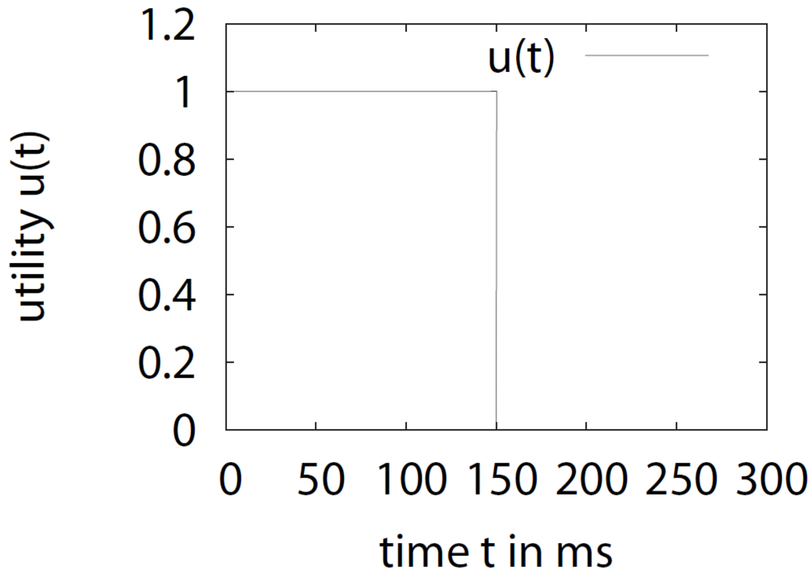
- Solution: Organic Computing approach for priority-based bus communication:
  - Decentralized
  - Self-organizing
  - Self-optimizing
  - …



- Idea: Decentralized run-time communication scheduling using **simple local rules**

Properties of Distributed Systems

- Constraints of messages:
  - Hard deadline
  - Soft deadline
  - Bandwidth



- Occurance of messages:
  - Periodic
  - Sporadic
  - Bandwidth

- Increase overall quality:
  - Satisfaction of safety-critical requirements
  - Increase of number of fulfilled constraints
  - Improvement of bus utilization
  - Guarantee of fairness

Outline

- Motivation and Goals

- Bandwidth sharing
  - Penalty Learning Algorithm (PLA)
  - Results

- Response time reduction
  - Dynamic Offset Adaptation Algorithm (DynOAA)
  - Results

- Summary and Outlook

Problem Description

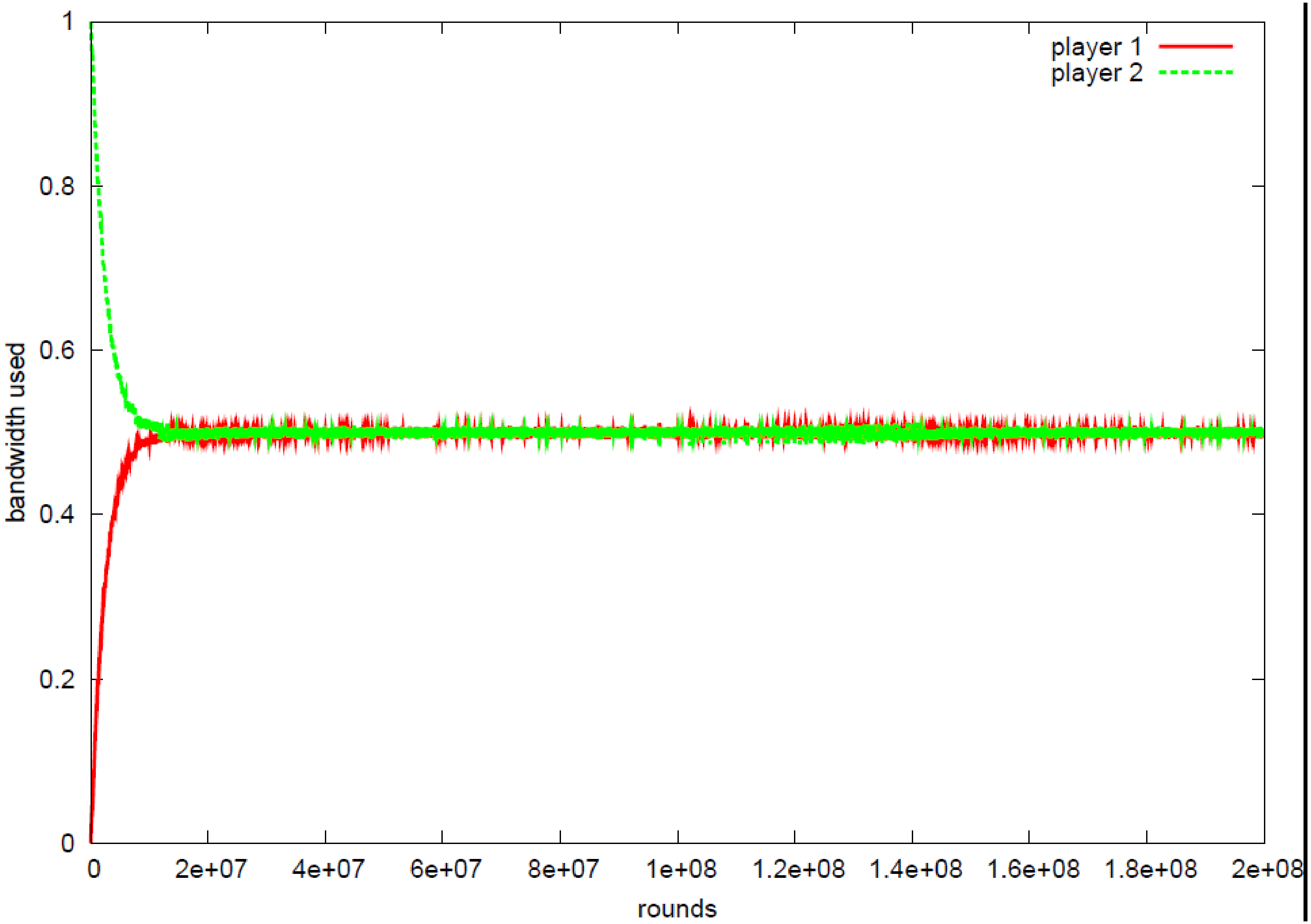
- Several nodes try to stream with maximum bandwidth
- Goal: Every node should get equal bandwidth
- Priority-based access unsuitable
- Description as a Game:
  - Set of Players
  - Set of Strategies
  - Payoff for each combination of played strategies

		Player 2	
		wait	send
Player 1	wait	0,0	0,1
	send	1,0	1,0

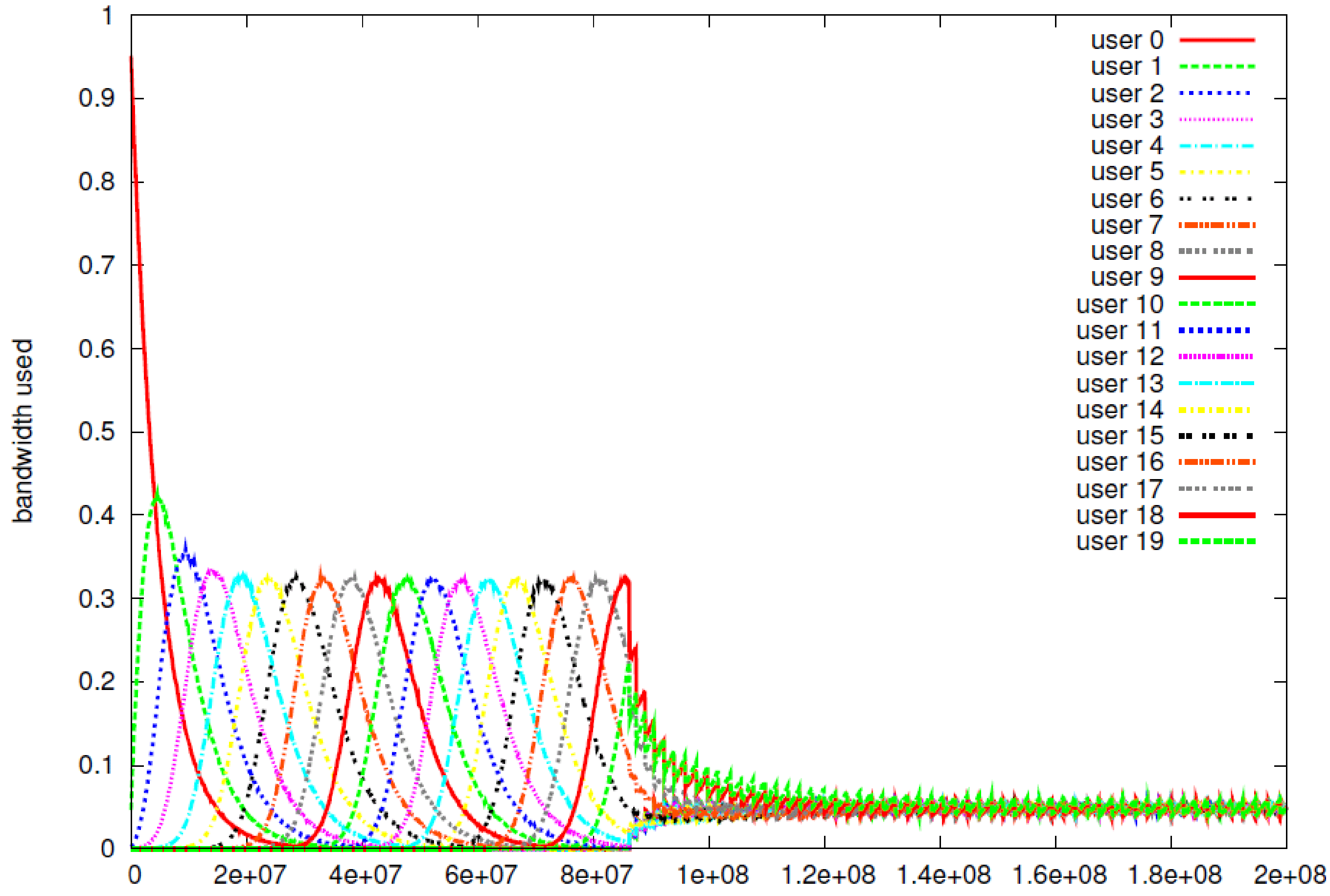
Solution

- Extension of the Game:
  - Sending probability is strategy
  - Demand that a small amount ε of the available bandwidth always stays free.
- Payoff:
  - If sum of sending probabilities is less then 1- ε, then return percentage of successfully sent messages
  - Else return 0
- Fair bandwidth distribution is Nash equilibrium (Proof)
- But not the only one
- Development of multi-agent reinforcement learning algorithm: Penalty Learning Algorithm (PLA)

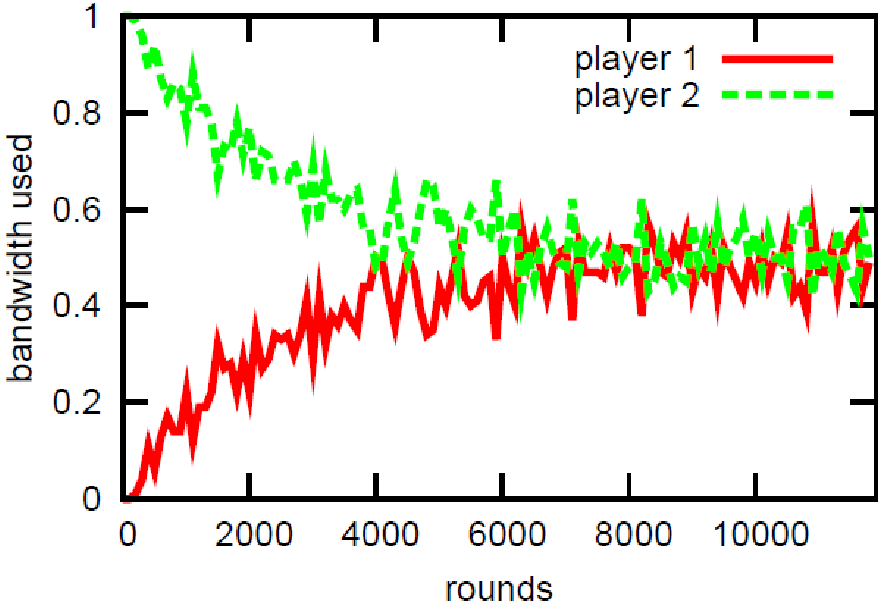
Results


player 1
player 2
bandwidth used
1
0.8
0.6
0.4
0.2
0
0
2e+07
4e+07
6e+07
8e+07
1e+08
1.2e+08
1.4e+08
1.6e+08
1.8e+08
2e+08
rounds

Results (20 Player)



Probabilistic/Periodic Access Method



(a) PLA with $\eta = 0.02$

- Probabilistic: Time independent representation

Outline

- Motivation and Goals

- Bandwidth sharing
    - Penalty Learning Algorithm (PLA)
    - Results

- Response time reduction
    - Dynamic Offset Adaptation Algorithm (DynOAA)
    - Results

- Summary and Outlook

Problem Description

- Properties of control oriented communication:
  - Periodic messages with soft deadline
  - But short response times
  - Limited data rate

- Controller Area Network (CAN) widely used
  - Priority-based event-triggered access method

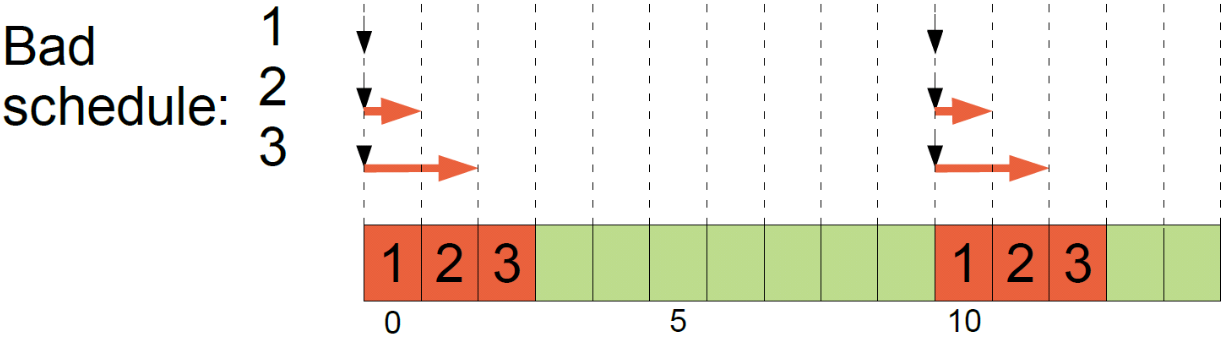
- Problem: Response times increase with workload

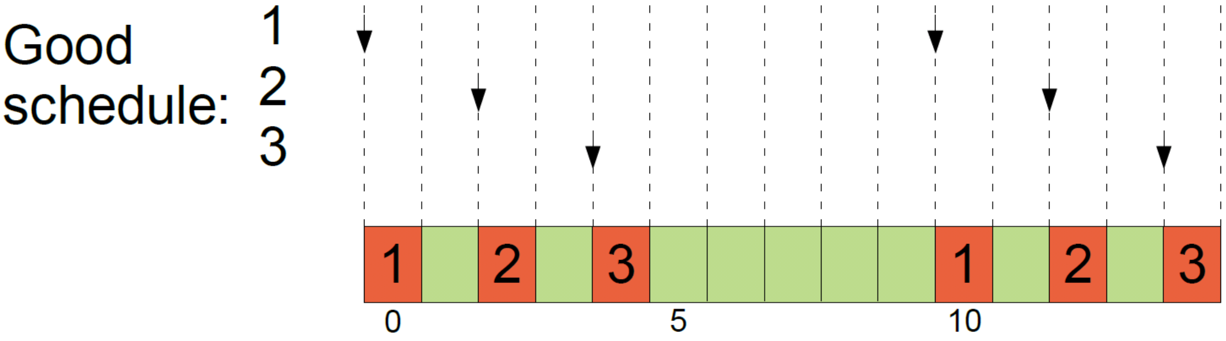
- Reason: On concurrent access messages with low priority get delayed

Solution

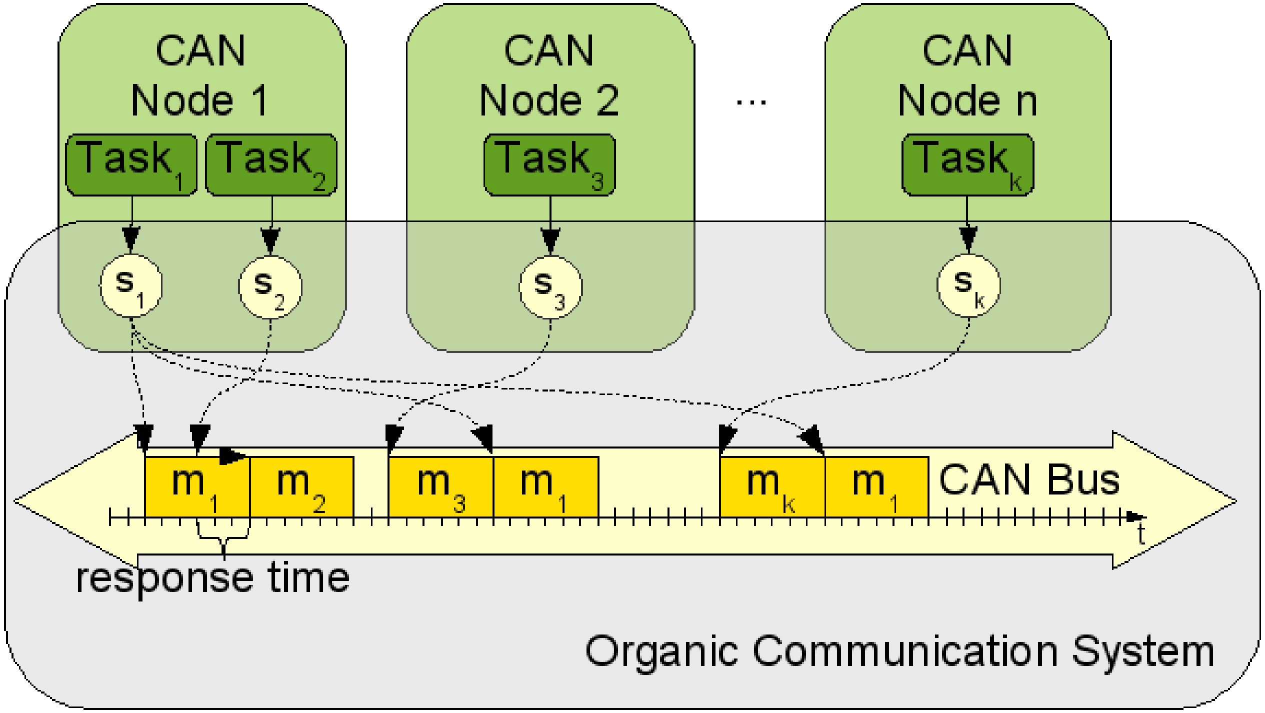
- Scheduling of messages to avoid concurrent access

- Example:



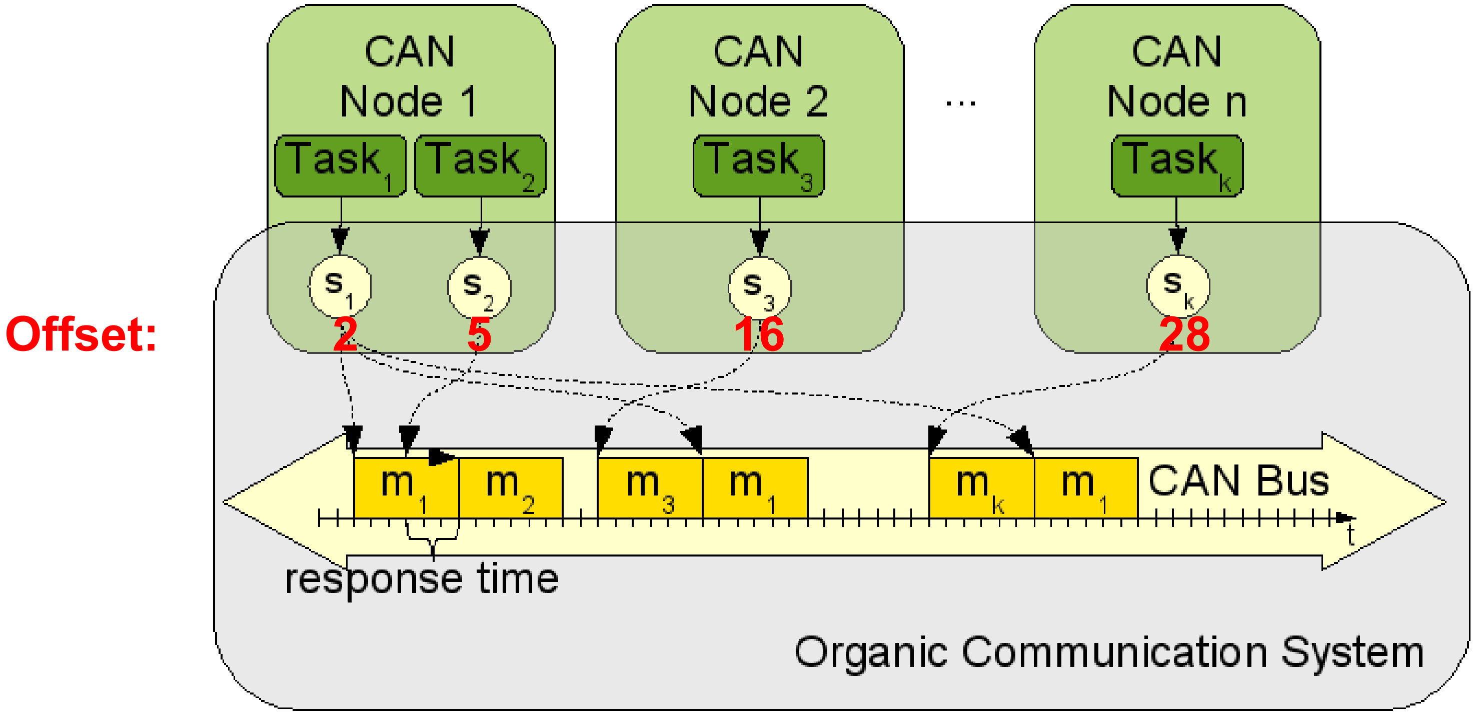


System Model



- Given a set of streams that periodically send messages
- Worst case response time (WCRT) is largest observed message delay during a given interval of time

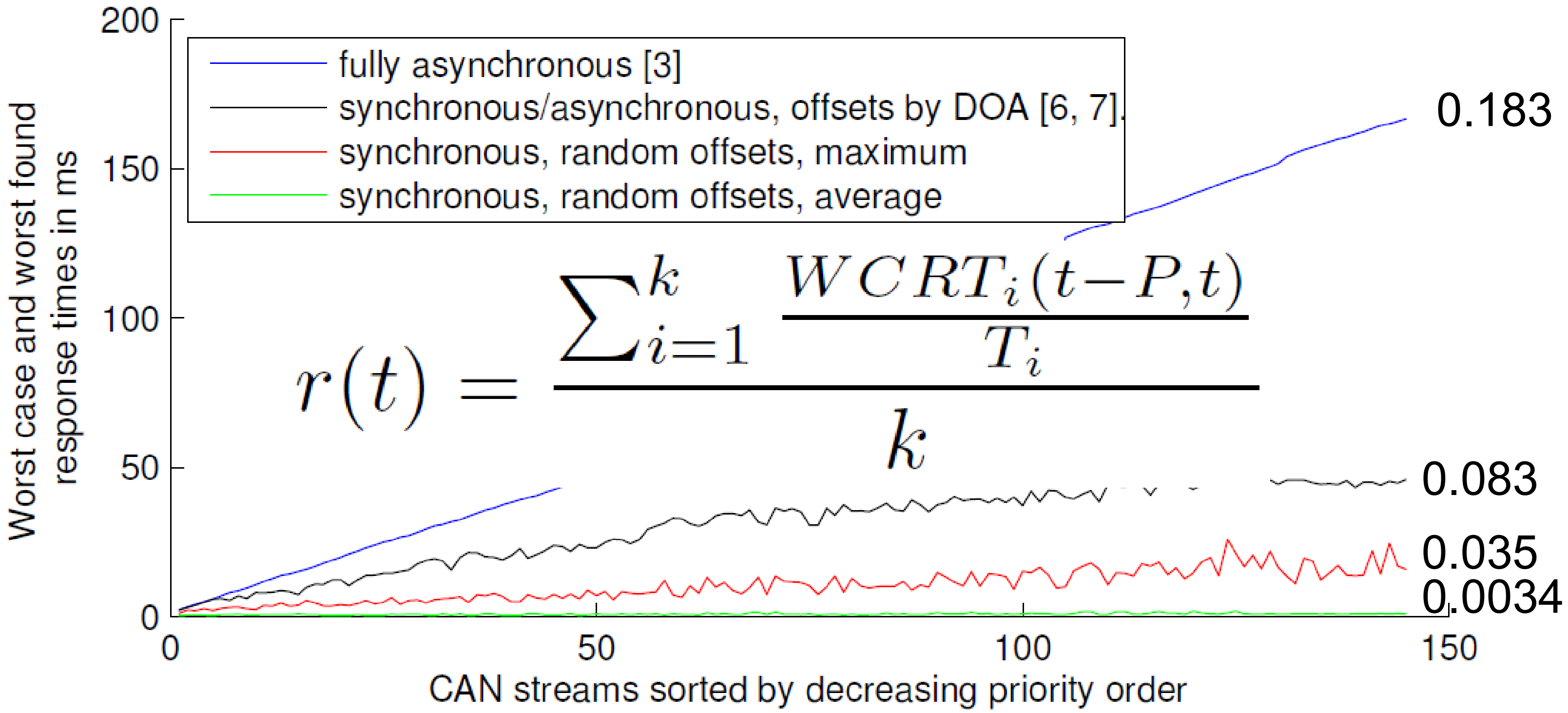
Goal



- Find offsets to reduce WCRT
- Online algorithm because streams are asynchronous

Rating Approach

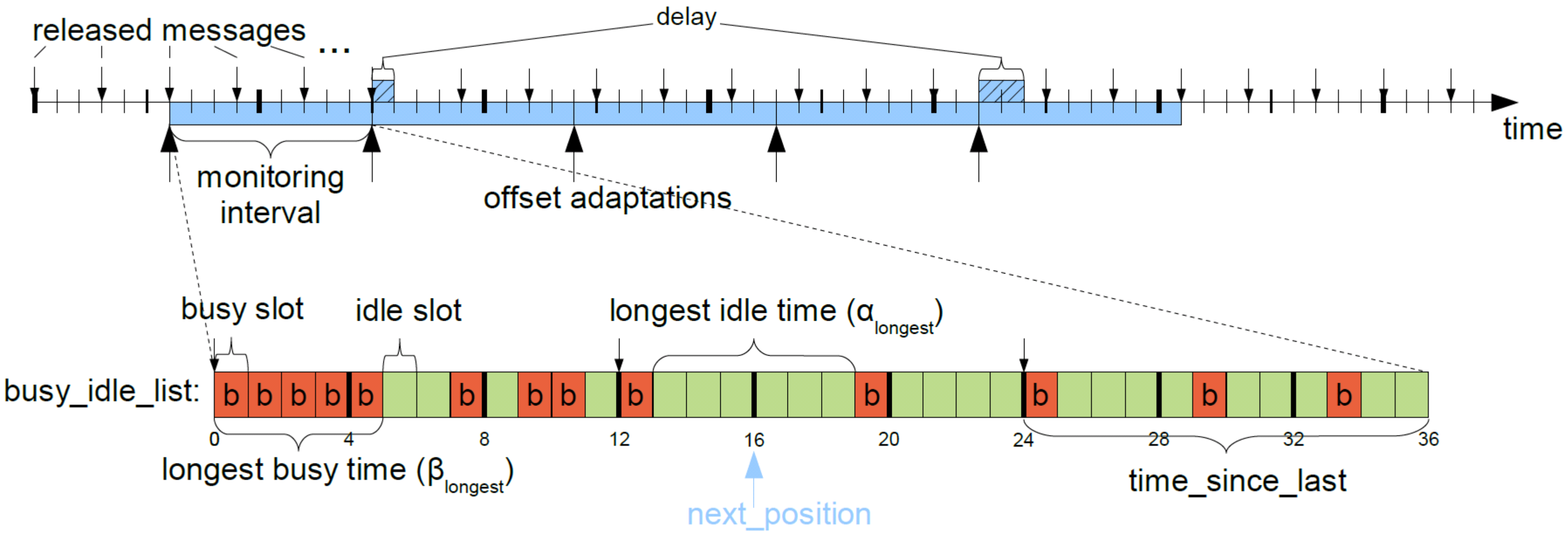
- Single-processor task scheduling:
  - Binary schedulability criterion for hard real-time tasks not applicable
- Diagram of the WCRTs of all streams
- Our approach: Rating function



$$r(t) = \frac{\sum_{i=1}^{k} \frac{WCRT_i(t-P,t)}{T_i}}{k}$$

Dynamic Offset Adaptation Algorithm (DynOAA)

- Run on each node independently and forever:
  1. Monitor current bus communication
  2. Decide whether to adapt
  3. Adapt according to monitoring information

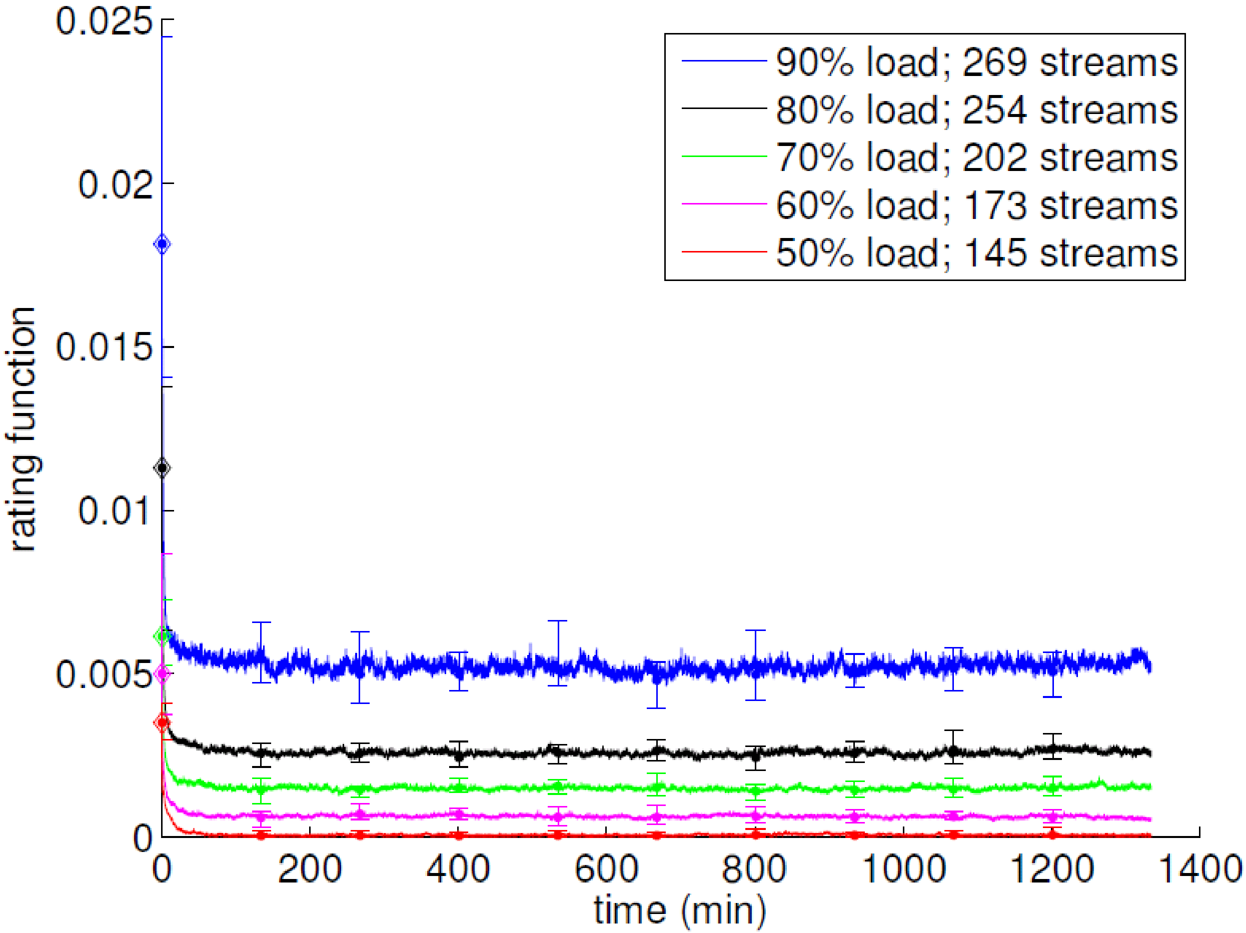


Simulation

- Evaluation by simulation
- Bit-accurate CAN simulator
  - Error free case
  - Worst-case bit stuffing
  - Synchronous simulation
  - Integrated online adaptation
- Test scenarios from Netcarbench (http://www.netcarbench.org/)
  - Typical automotive scenarios
  - 125 kbit/s data rate
  - Workload ranging from 50% to 90%

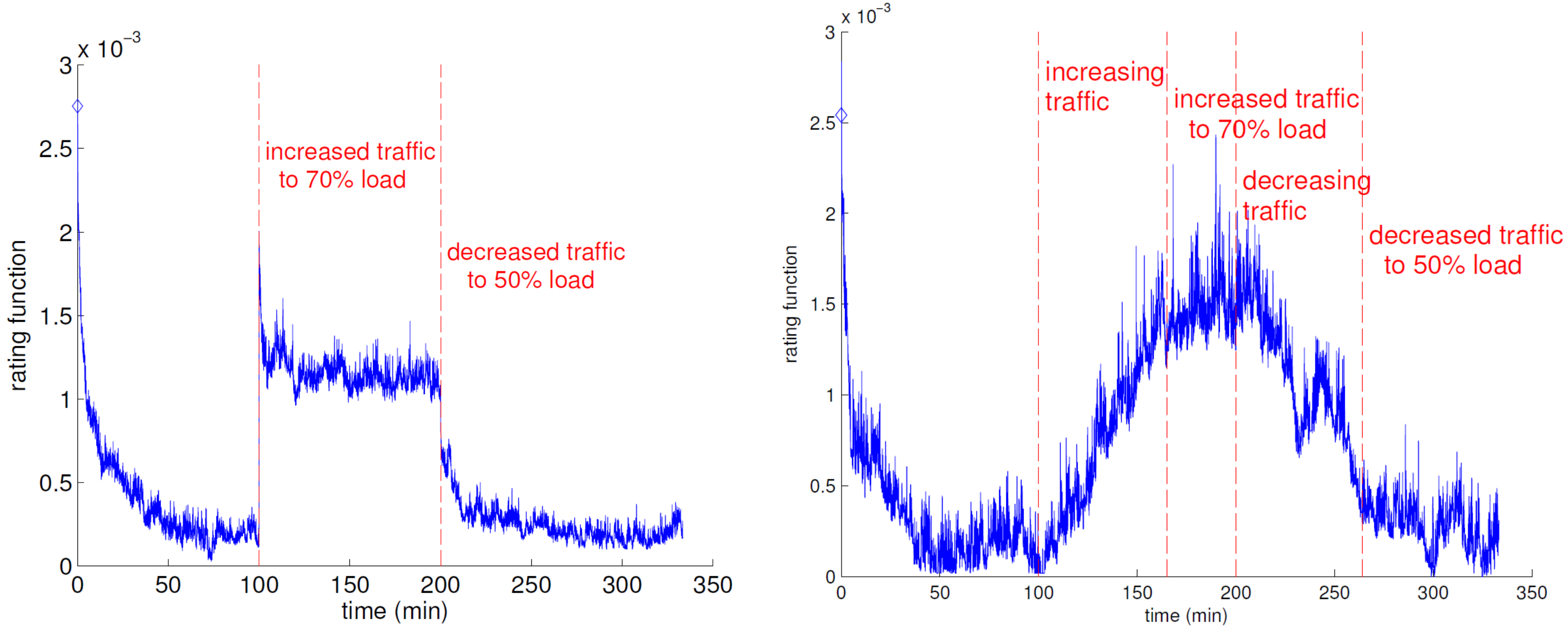
Results

- Rating over time with 10 random initial offsets for different scenarios



Adaptation to Changing System

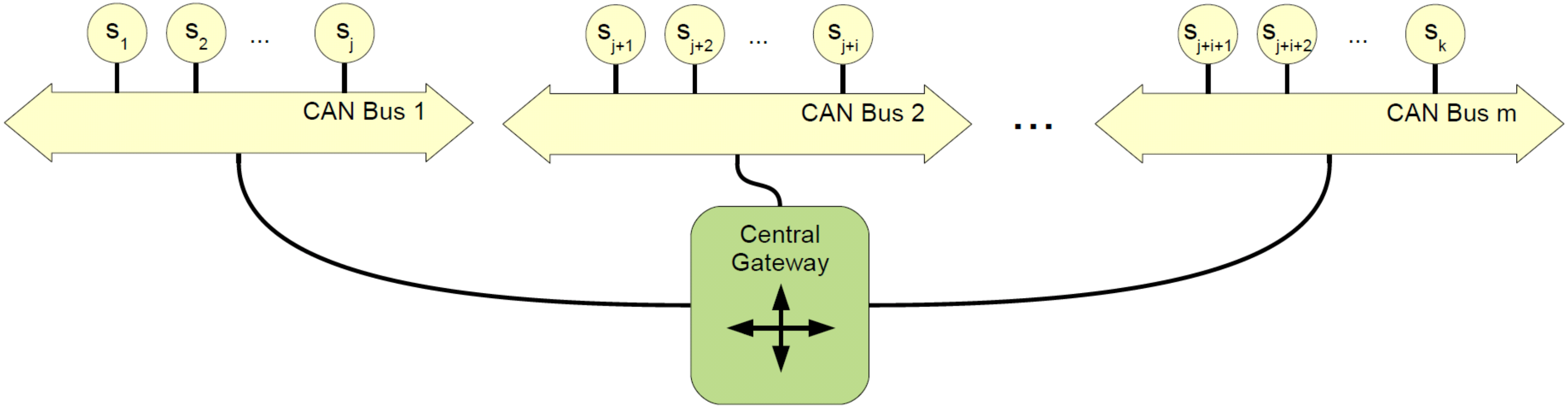
- Simulation shows robustness to changing system during run-time



Multi-segment System Model

- Stream model extended by a source bus and a set of destination buses

- Central gateway:
  - Delays neglected
  - Priority-based access
  - Immediate start of retransmission after full reception

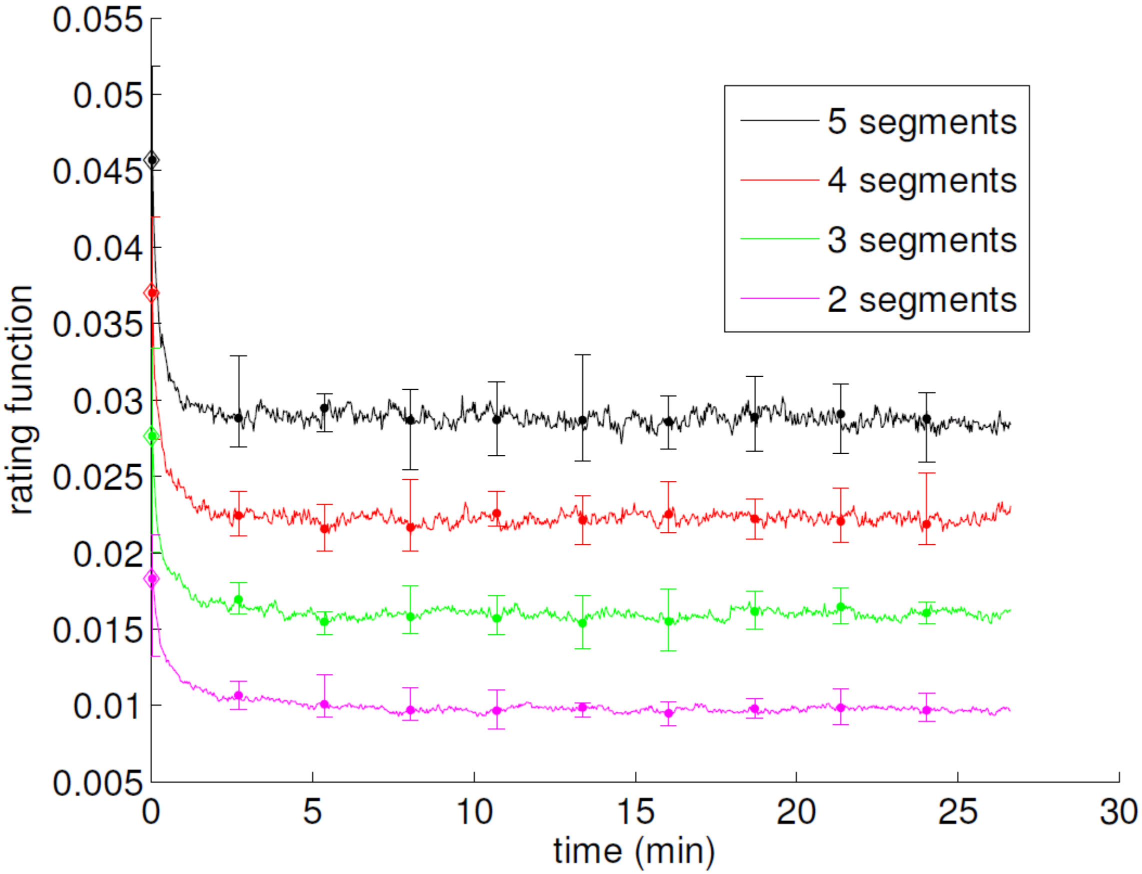


Multi-segment

- Difference: Handling of routed streams as non-adapting streams
- Modified algorithm to allow partial adaptation
- Scenarios are generated from single-segment scenarios:
    - Assigning source streams uniformly
    - routing
- Preliminary results show the performance of DynOAA in multi-segment systems

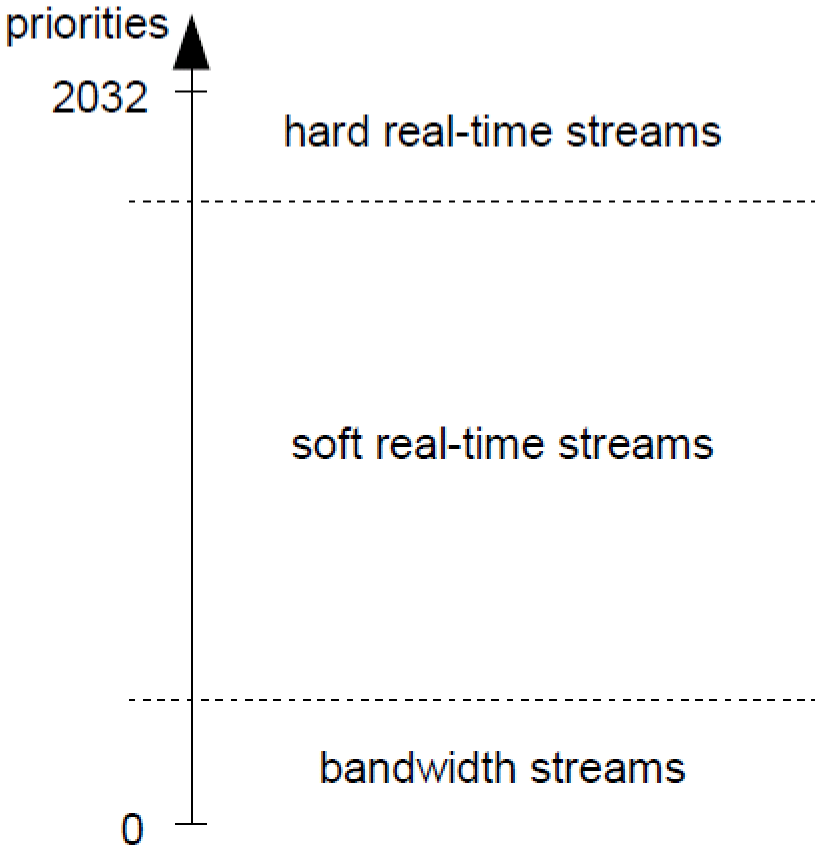
Results

- Rating over time for different number of segments where all streams are routed to all other segments



Integration of All Approaches

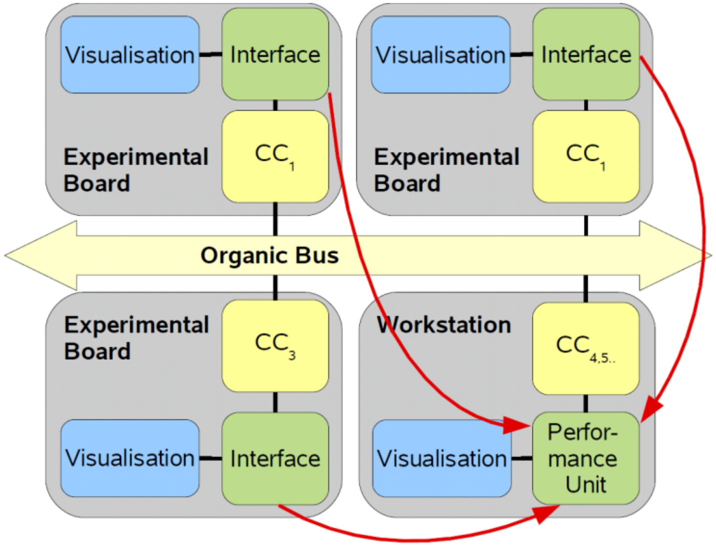
- Hard deadline
  - Highest priorities
  - Analytical approach, e.g. EPOC
- Soft deadline
  - Periodic: DynOAA
  - Sporadic: Priority access
- Bandwidth
  - Lowest priority
  - PLA



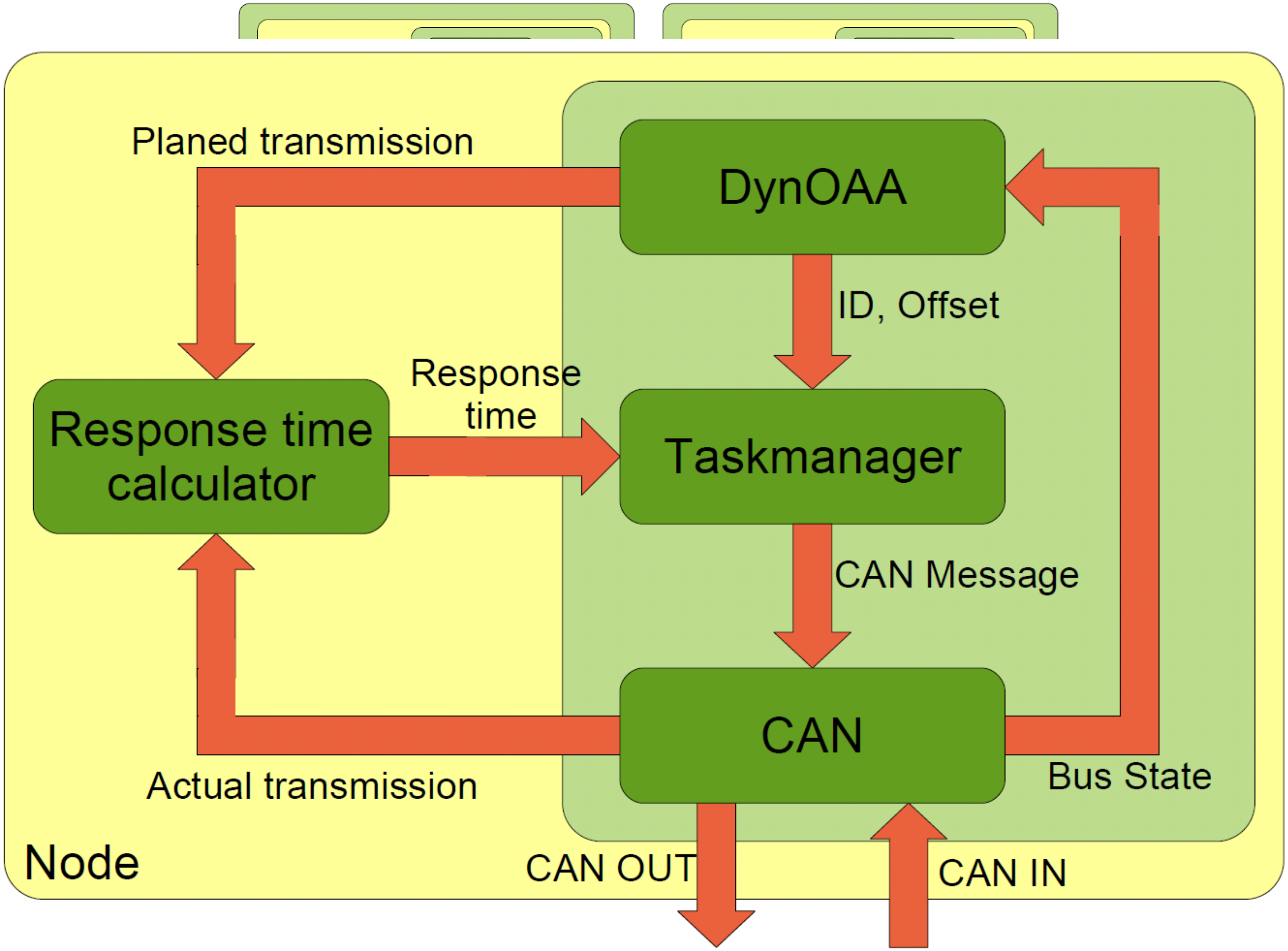
Outlook

- Implement the algorithms on real hardware
  - Analyze overhead of organic bus protocol
  - Consideration of asynchronous communication with Controller Area Network (CAN)
  - Provide prototype and demonstrator

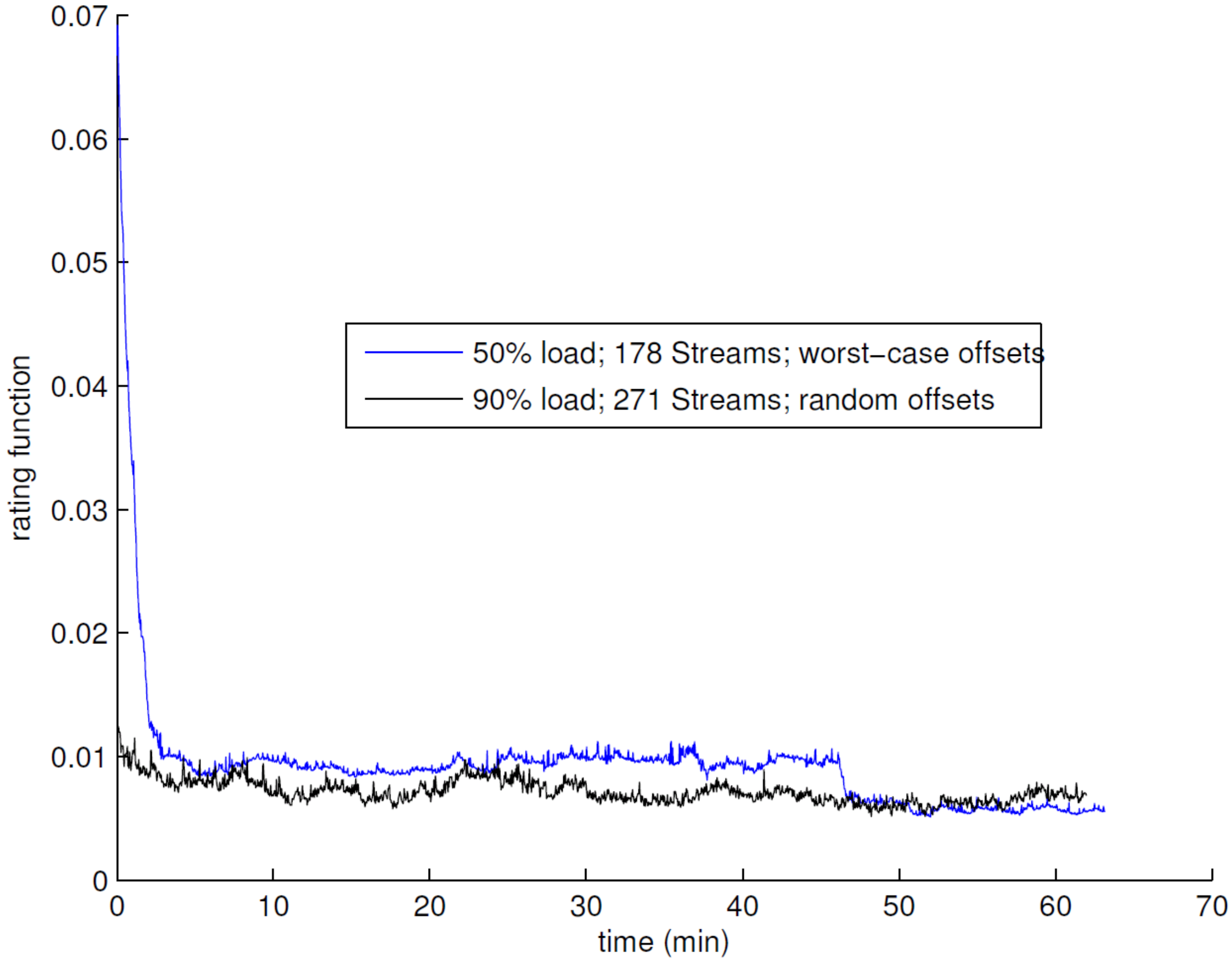
- Considered Platforms:
  - Standard PC
  - Prototype on FPGA
    - Softcore processor
    - Pure hardware



Hardware Architecture



Preliminary Results



Summary

- Modeling and analysis of decentralized bus bandwidth allocation algorithms using game theory

- Development and simulation of two algorithms:
  - Penalty Learning Algorithm for bandwidth constraints
  - Dynamic Offset Adaptation Algorithm for soft real-time constraints

- Decentralized approach avoids single point of failure

- Online adaptation allows adjustment to current traffic
  - Allows higher utilization of bus

- Prototype will provide proof of concept

Thanks for your attention

- Project page:
  - www12.informatik.uni-erlangen.de/research/organicbus/

- Contact:
  - Tobias Ziermann
  - tobias.ziermann@informatik.uni-erlangen.de
  - www12.informatik.uni-erlangen.de/people/ziermann