Measuring and improving quality of parallel application monitoring based on global states

Janusz Borkowski

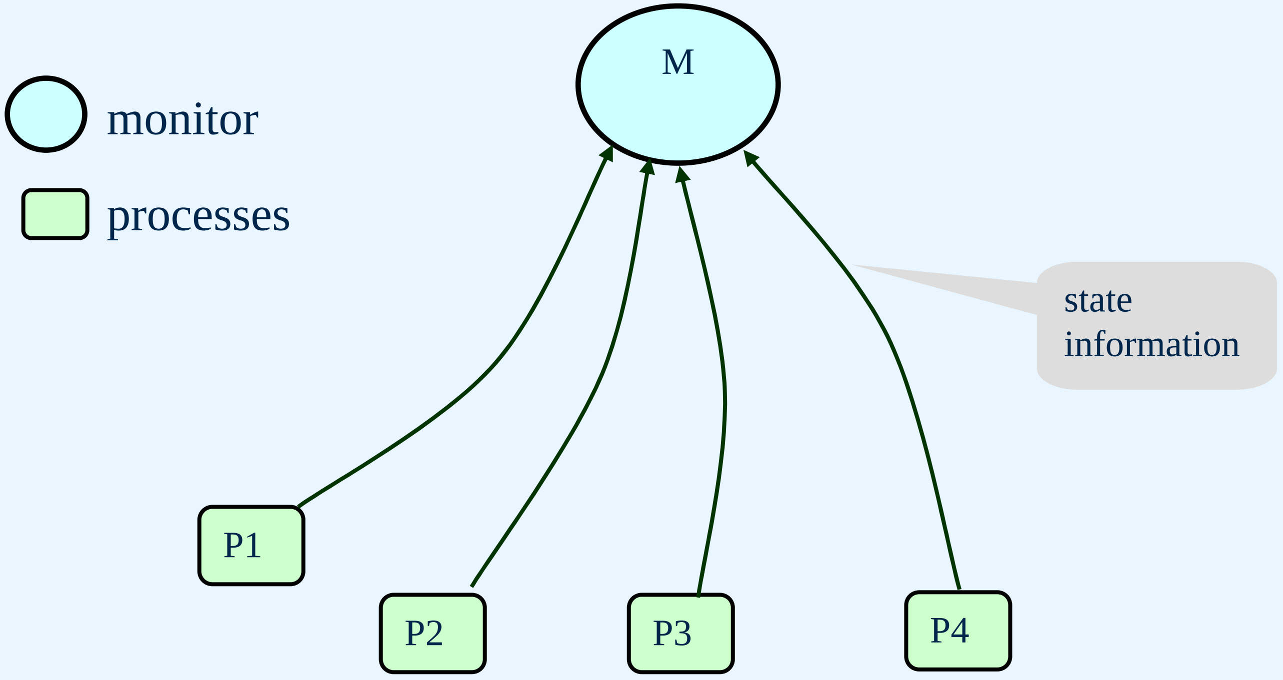
Polish-Japanese Institute of Information Technology,
86 Koszykowa Str., 02-008 Warsaw, Poland
janb@pjwstk.edu.pl

2005.07.04-7, ISPDC Lille, France

Contents:

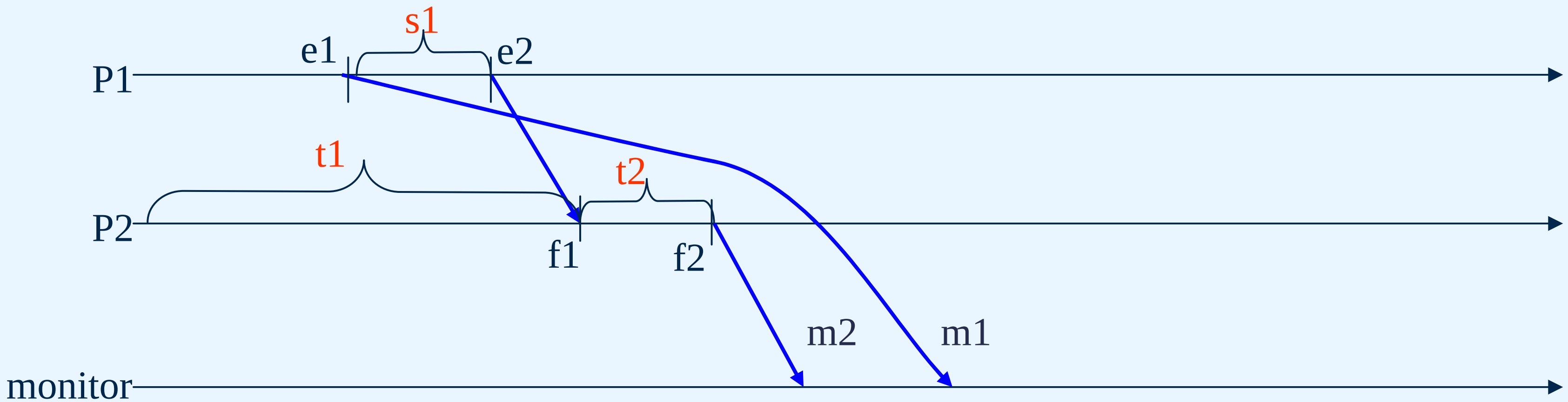
- Consistent Global States (CGS)
- Global predicates for monitoring and control
- Measures of monitoring quality
- 4 SCGS detection algorithms
- Test results
- Conclusions

Global State monitoring



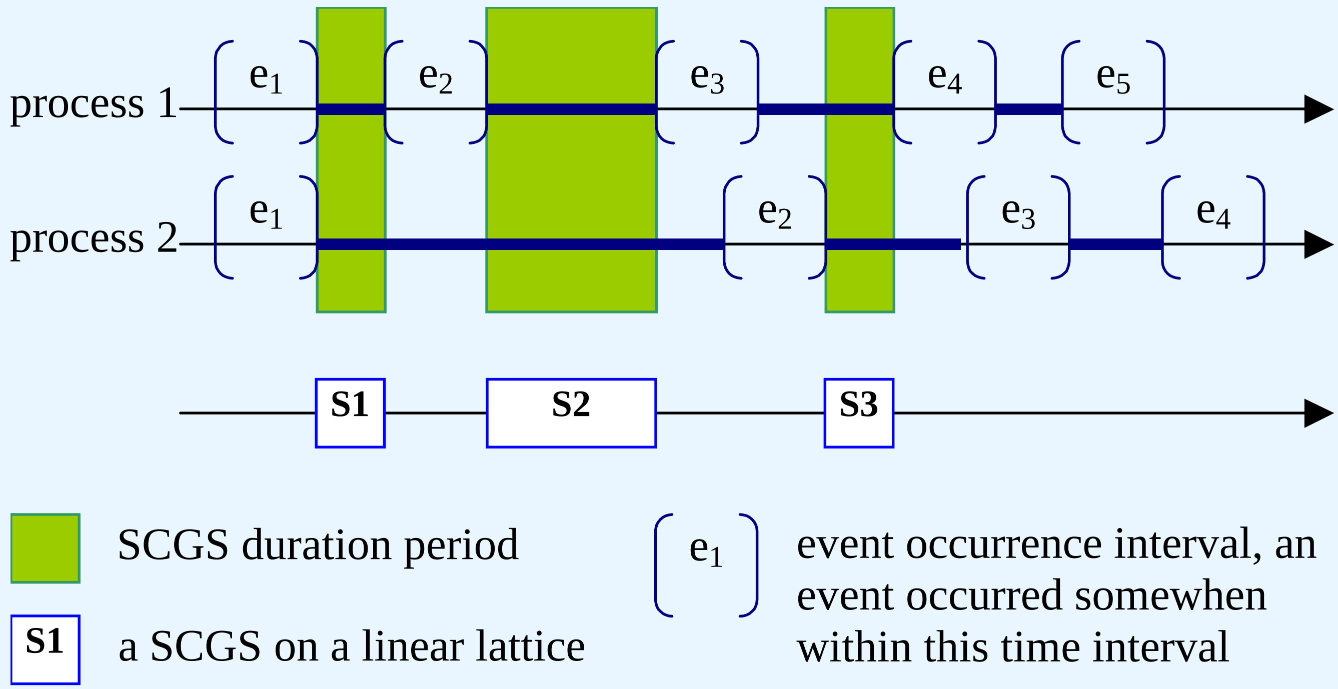
Consistent Global States

- There is no global clock, no shared memory, only message passing
- A monitor receives reports about process events
- The monitor must be able to order properly incoming events to build Consistent Global States (CGS)
- CGS is a combination of process local states, one state from each process, such that the local states are pairwise concurrent. E.g. <s1,t1> is a CGS, <s1,t2> is not.



Consistent Global States with real time

- Process local clocks are roughly synchronized (NTP, radio, RBS, ..)
- Interval timestamps describe time periods within which events occured
- When we can tell for sure the state of each of the process simultaneously, then we have a Strongly Consistent Global State SCGS

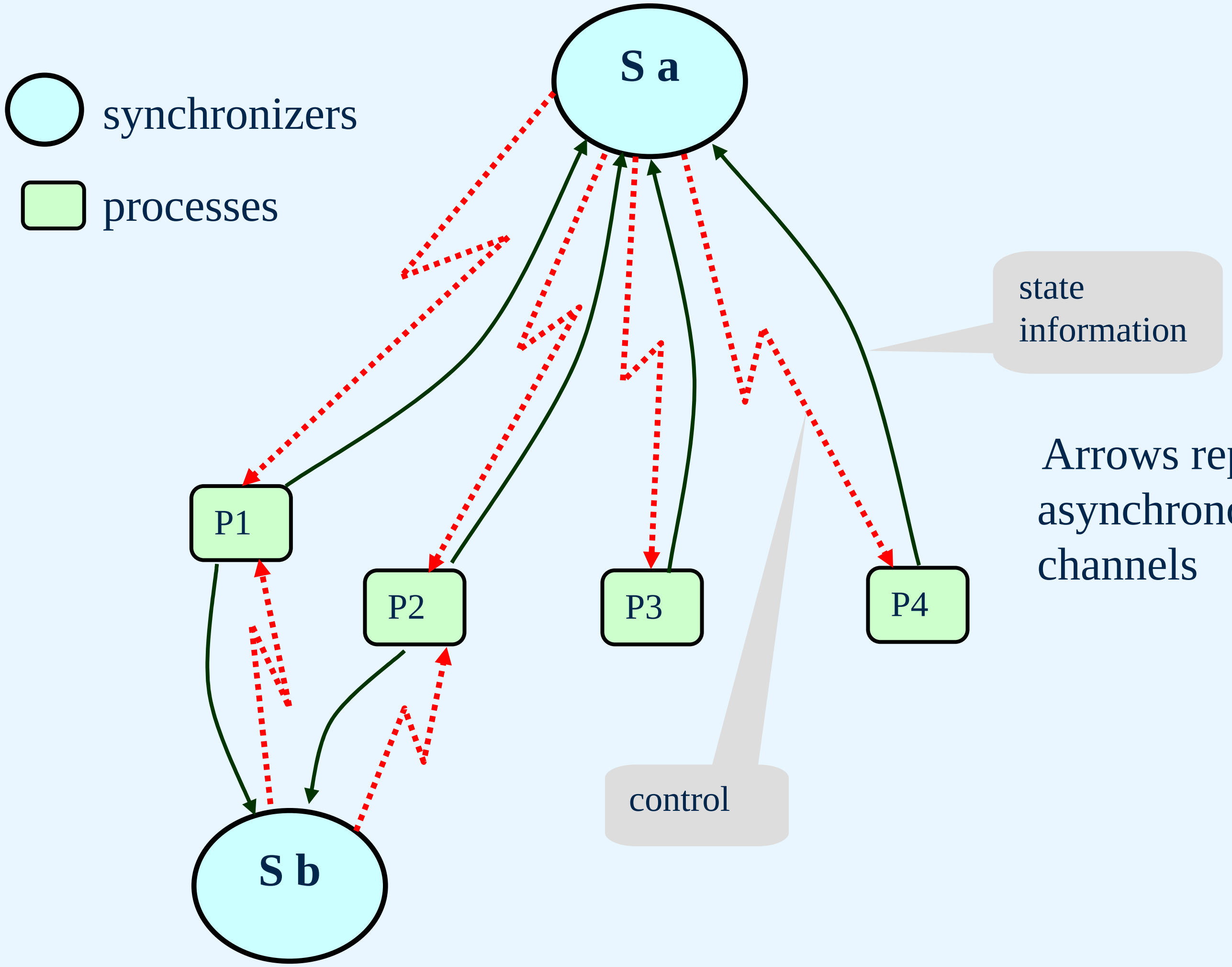


CGS applications

- Parallel / distributed systems debugging
- Monitoring – detection of erroneous system states
- Control – making control decisions based on current CGS

Control based on global states

Processes can communicate with a number of active monitors. Monitors learn state information from processes and send back control information.



Arrows represent reliable, asynchronous communication channels

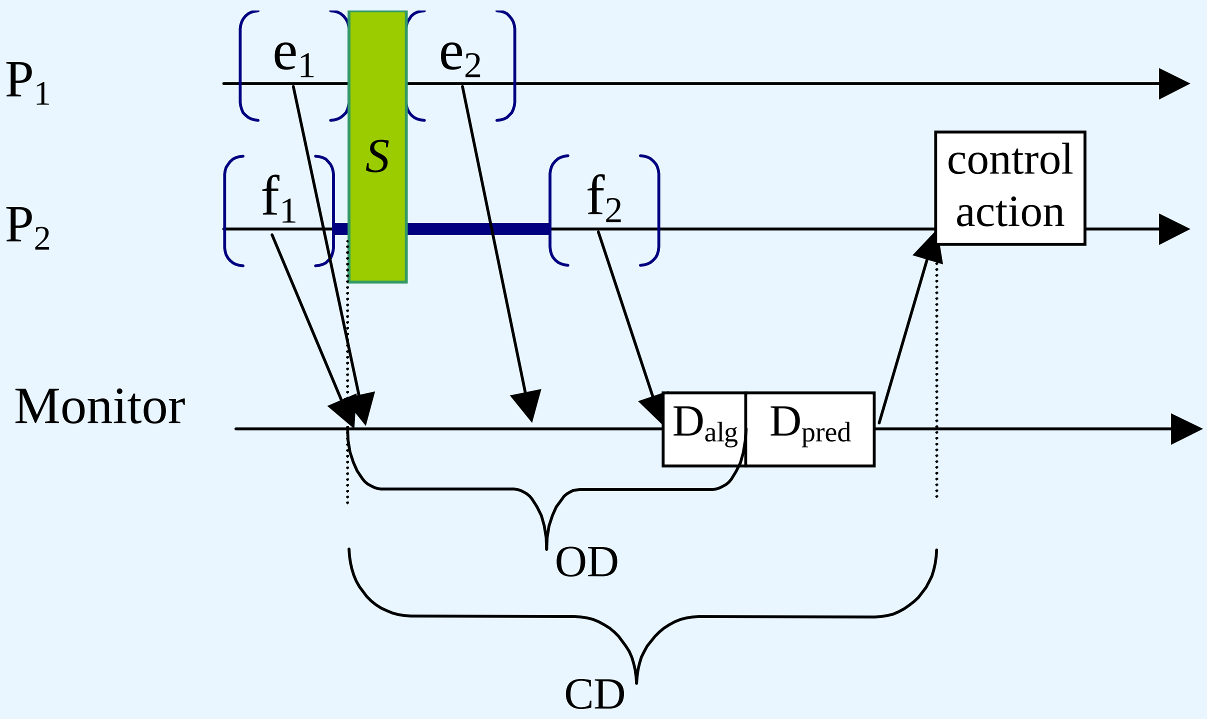
Measures of monitoring quality

Three desired features of a monitoring system:

1. The incoming information must be processed quickly
2. The incoming information must be up to date
3. The information must arrive constantly

Measures of monitoring quality

- Control Delay and Observation Delay



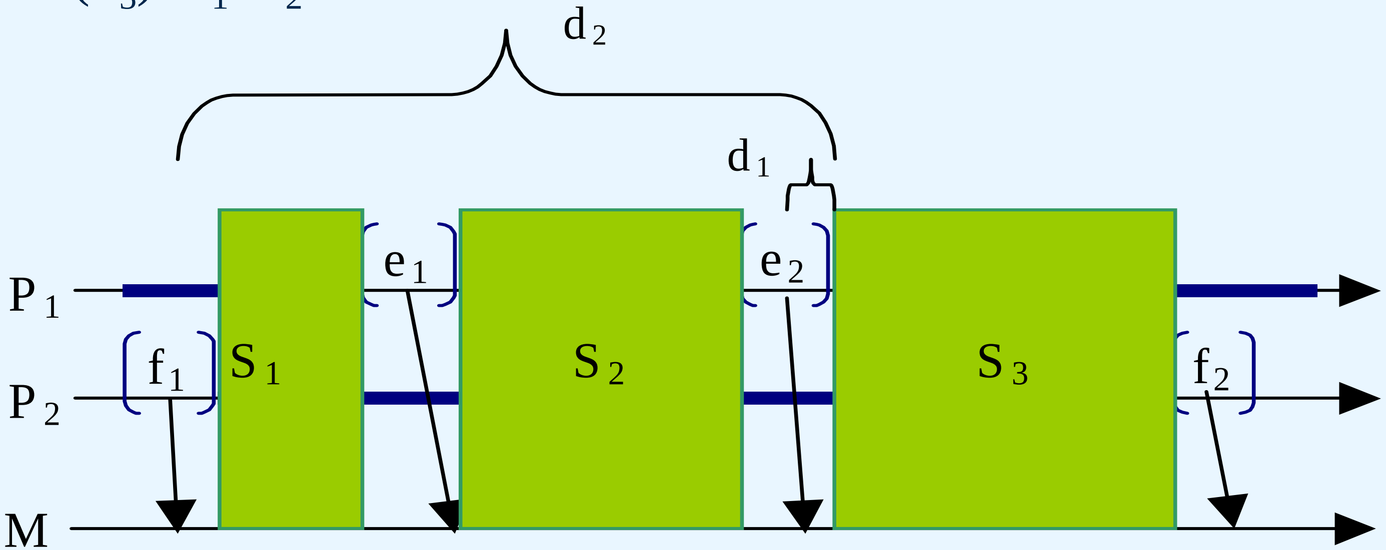
Measures of monitoring quality

- State Currency

Consistent state S_1 uses fresh information about P_2

The information about P_2 used by S_3 can be outdated

$SC(S_3)=d_1+d_2$



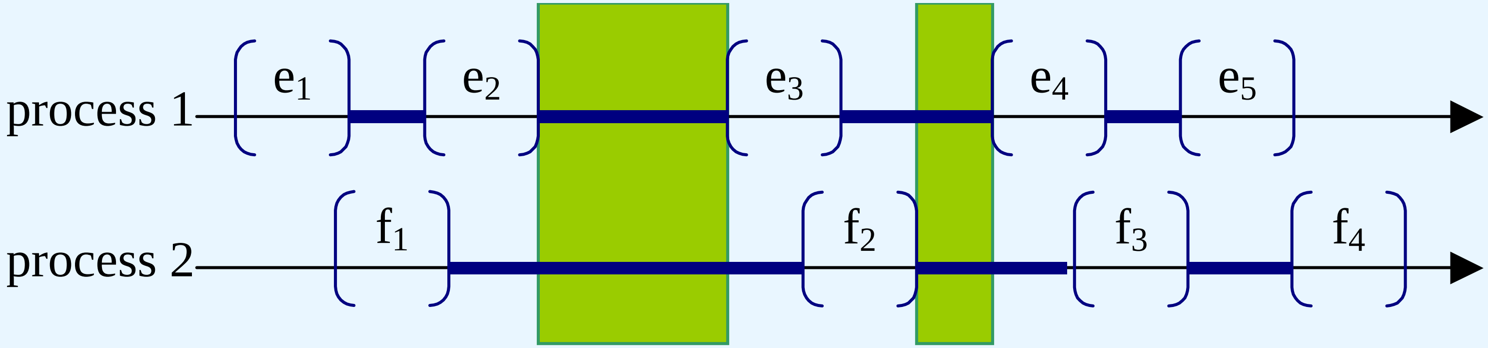
Measures of monitoring quality

•Observation Frequency

How often a global state is observed and examined?

Not every event must start SCGS !

OF = #SCGS/sec

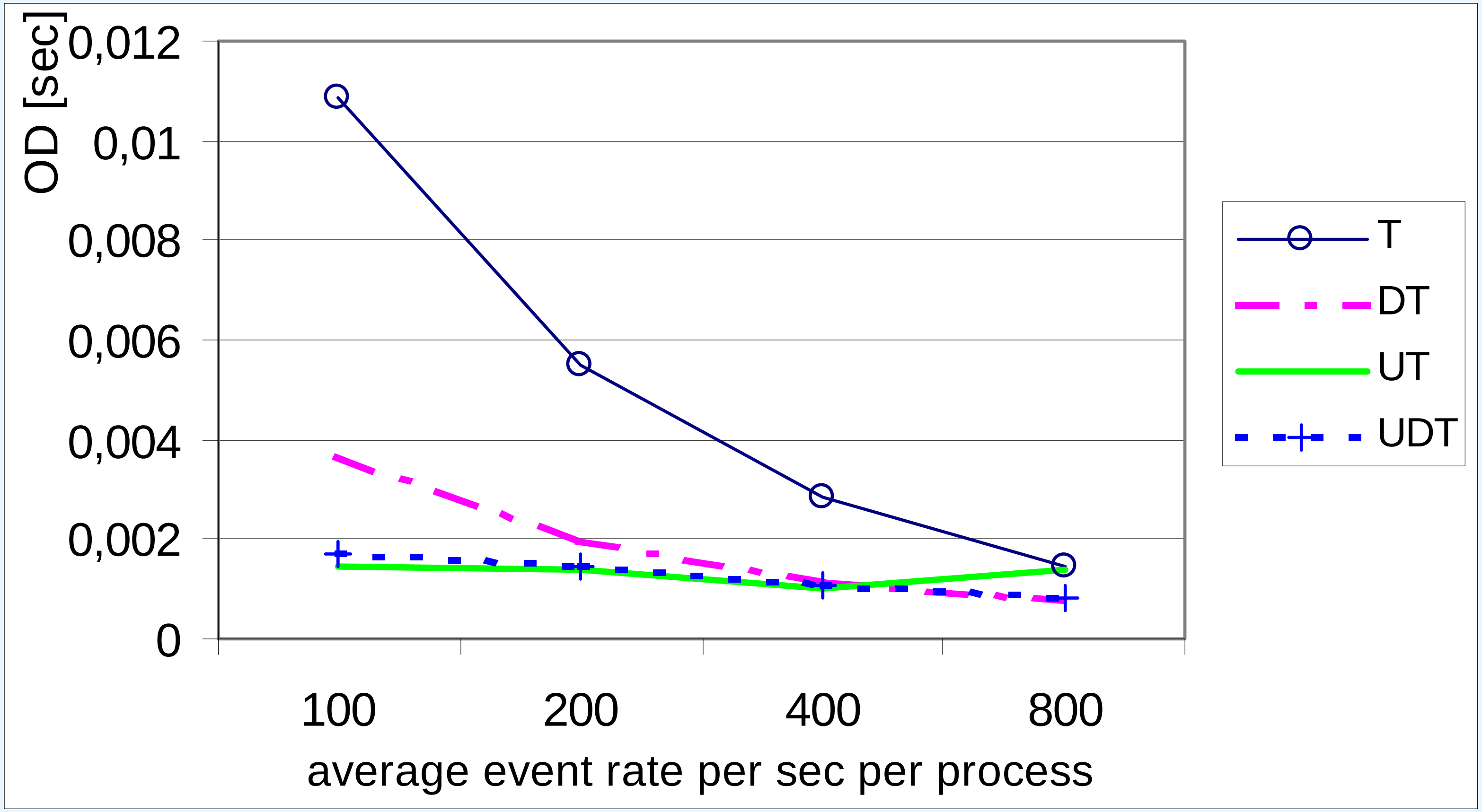


4 SCGS algorithms

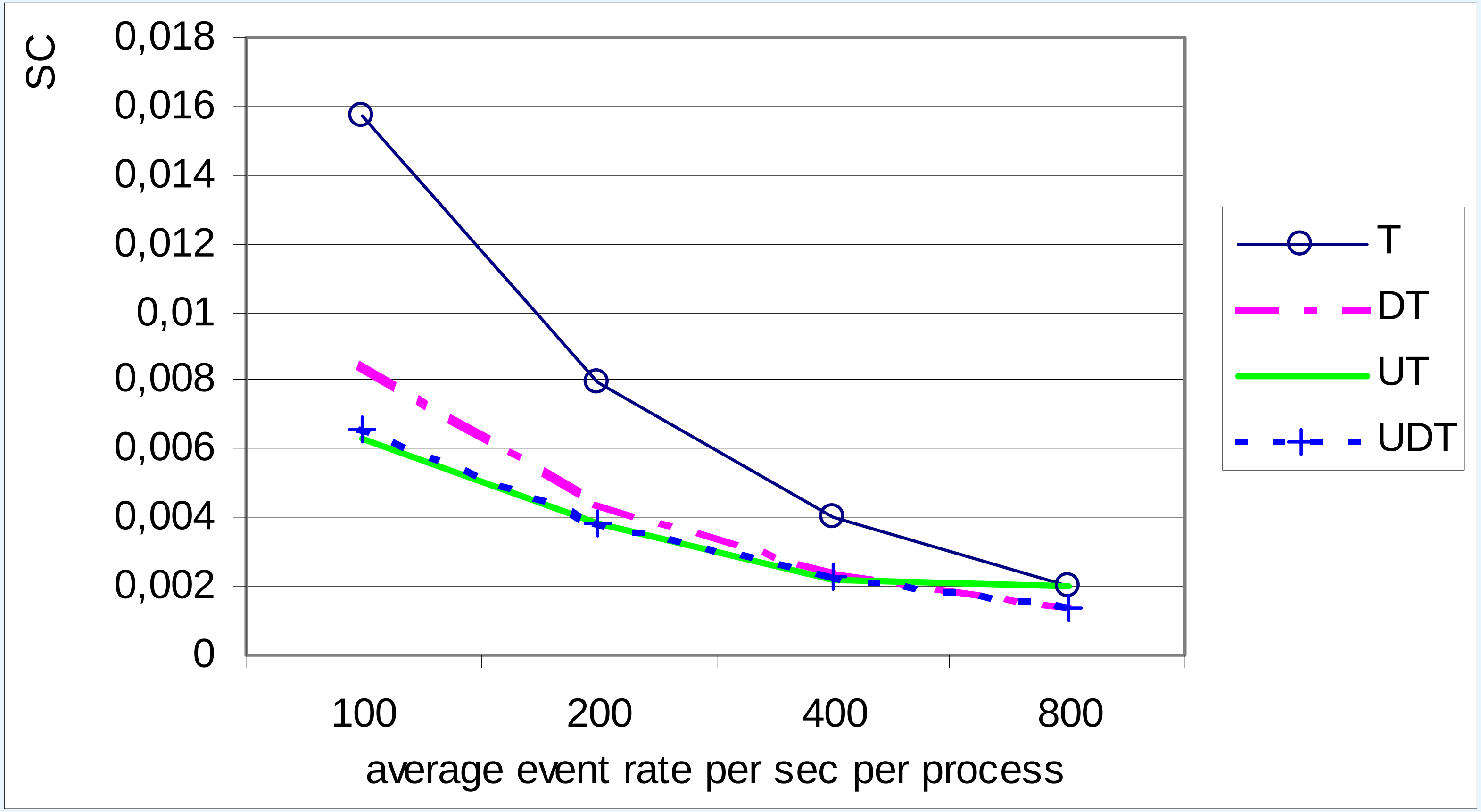
- **T** the standard algorithm working with terminated local states
- **UT** works with unterminated local states, requires that the message transfer time is bounded and uses the bound
- **DT** works with unterminated local states under assumption, that local state durations cannot be shorter than a known *delta* value (newly introduced here)
- **UDT** combines UT and DT (newly introduced here)

Performance of the algorithms has been compared using a simulation environment build with OMNeT++ package

4 SCGS algorithms – Observation Delay



4 SCGS algorithms – State Currency

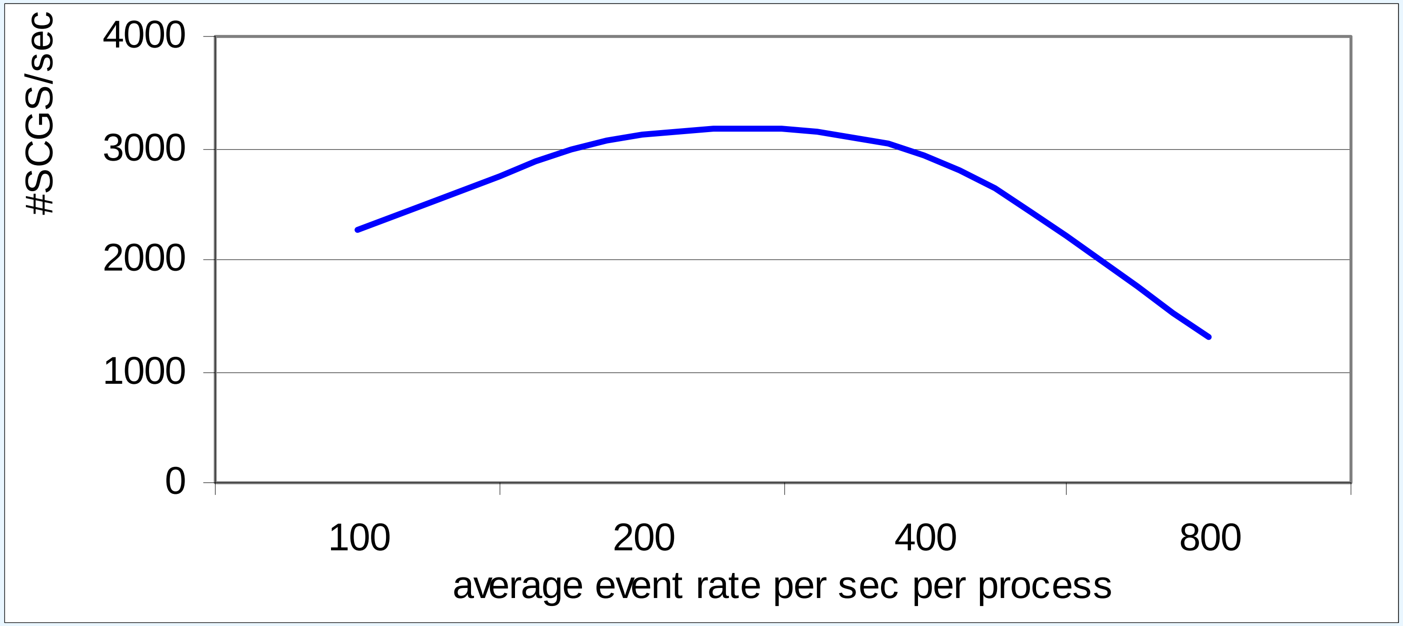


Observation Frequency

More events (higher event rate) create more SCGS

But only up to a treshhold

For high event rates many events get discarded and less SCGS emerge



Conclusions

- Universal measures for CGS monitoring quality have been defined
  - Control Delay / Observation Delay
  - State Currency
  - Observation Frequency
- 4 SCGS algorithms tested for OD and SC
  - UT performs best for low event rates
  - DT performs best for high event rates
  - SC is always better for high event rates
- To maximize Observation Frequency it may be necessary to diminish State Currency and vice versa