The Lookahead in a User-Transparent Conservative Parallel Simulator

Presentation of the Results

of the PhD Dissertation

Viliam Solčány

Faculty of Informatics and Information Technologies

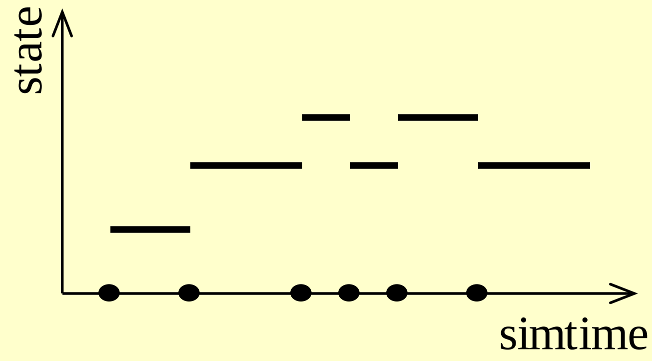
Slovak University of Technology, Bratislava

Outline

- Motivation
- Present State
- Thesis Objectives
- Simulator Design
- Cumulative Timestamp Prediction
- Cumulative Message Pre-Sending
- Experimental Verification and Results
- Conclusions

Motivation

Discrete Event Simulation (DES)



- Instantaneous state changes (events)
- State trajectory over time
- Event calendar and the sequential simulation algorithm

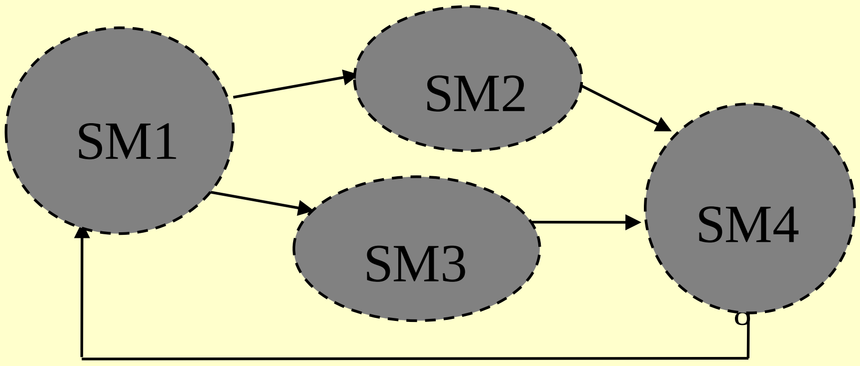
Motivation for Parallel Discrete Event Simulation (PDES)

- Real systems of ever increasing scale and complexity
  $\Rightarrow$ ever increasing demand for high-performance simulation (computational time/memory)
- Interoperability of naturally distributed simulators
- Scientific curiosity: in spite of high degree of parallelism in models they are difficult to simulate in parallel

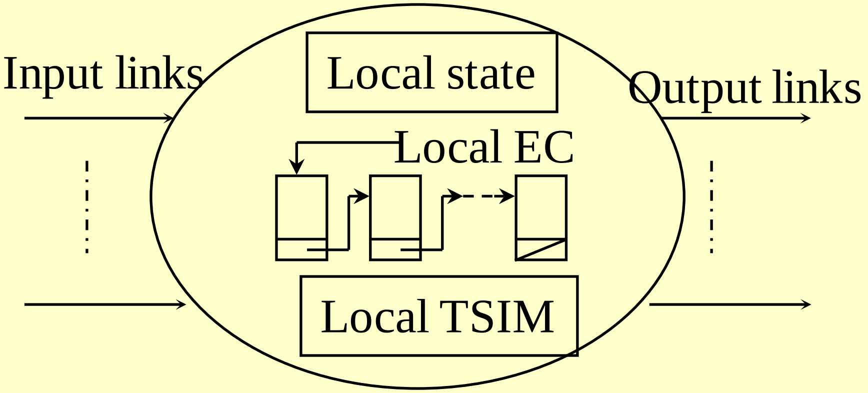
Present State

Parallel Discrete Event Simulation (PDES)

- Decomposition into submodels (logical processes)

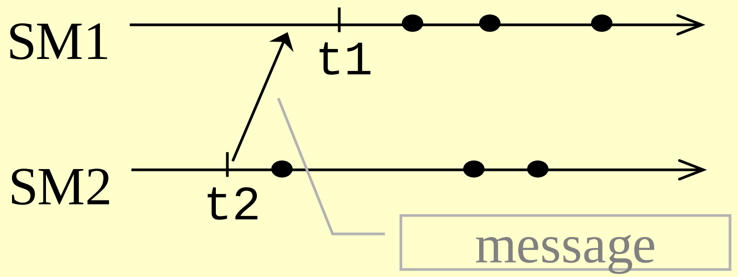


- Submodel structure
  - sequential local executive
  - communication via timestamped messages



Present State (cont.)

<u>Synchronization</u>

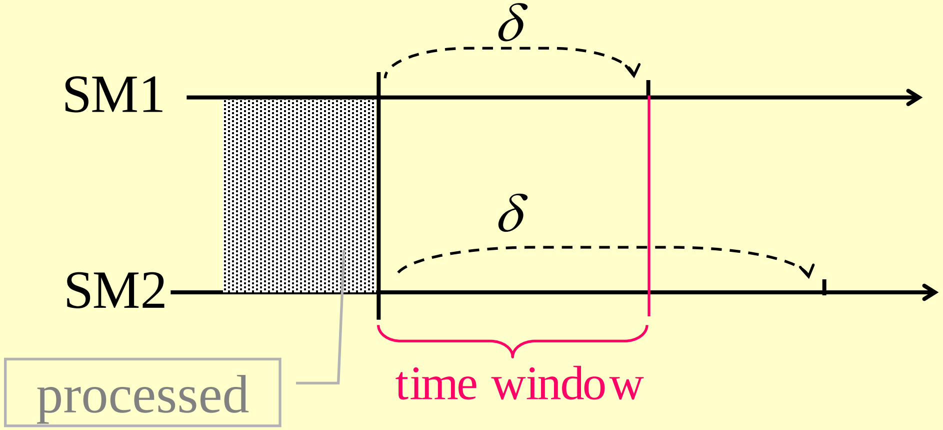


- Unsynchronized local times
  ⇒ causality error

- Synchronization methods
  – numerous approaches, conservative and optimistic
  – no clear favorite

- The YAWNS method (Yet Another Windowing Network Simulator)
  – scalable

- Lookahead in YAWNS
  – message pre-sending
  – conditional timestamp prediction δ



Present State (cont.)

Discrepancies between DES and PDES

- Low impact of PDES in general DES, search for reasons
  - necessity to worry with low level PDES intricacies
  - ignorance of standard modeling methodologies
  - different “languages”
  - ...

- Proposed directions
  - application specific libraries
  - parallel simulation languages
  - support for shared state
  - automatic parallelization

Thesis Objectives

<u>Observations</u>

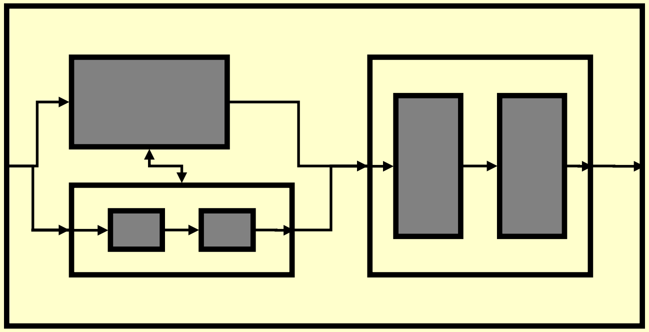
- Integrating established modeling methodologies into PDES
- Focus on large scale systems $\Rightarrow$ scalability
- Importance of the lookahead

<u>Objectives</u>

- User-transparent parallel simulator design with support for large scale models
- Methods for compound submodels, with YAWNS synchronization
  - cumulative timestamp prediction
  - cumulative message pre-sending
- Experimental verification and performance evaluation

Simulator Design

The Modeling Approach



- Hierarchical decomposition
- Combined structural and behavioral model description
- Modularity of components, closure under composition

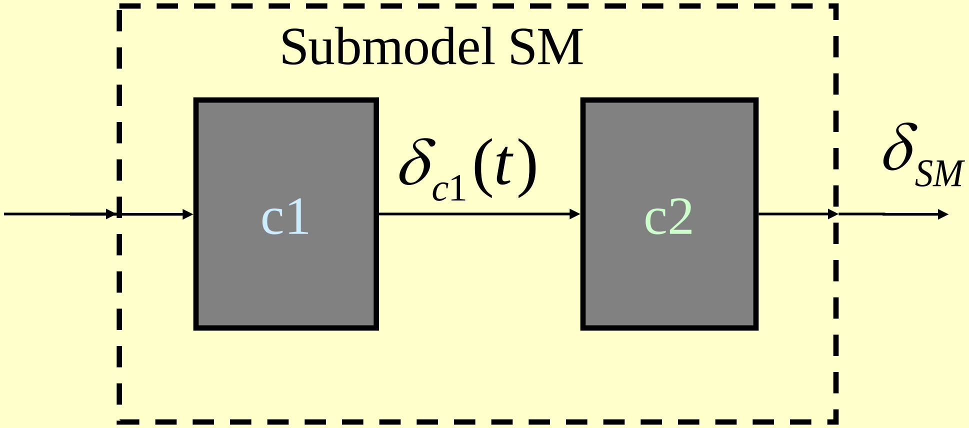
Parallel Execution

- Library of model components
- Components with / without lookahead
- Automatic translation, the user sees just one model
- Cooperation between users and PDES experts

Cumulative Timestamp Prediction

Idea

- Components provide
  - conditional timestamp δ
  - delay (timestamp increment) σ
- Required
  - conditional timestamp δ_{SM} of the submodel



$$\delta_{SM}(t) = \min\left[\delta_{c2}(t), \delta_{c1}(t) + \sigma_{c2}\right]$$

Cumulative Timestamp Prediction (cont.)

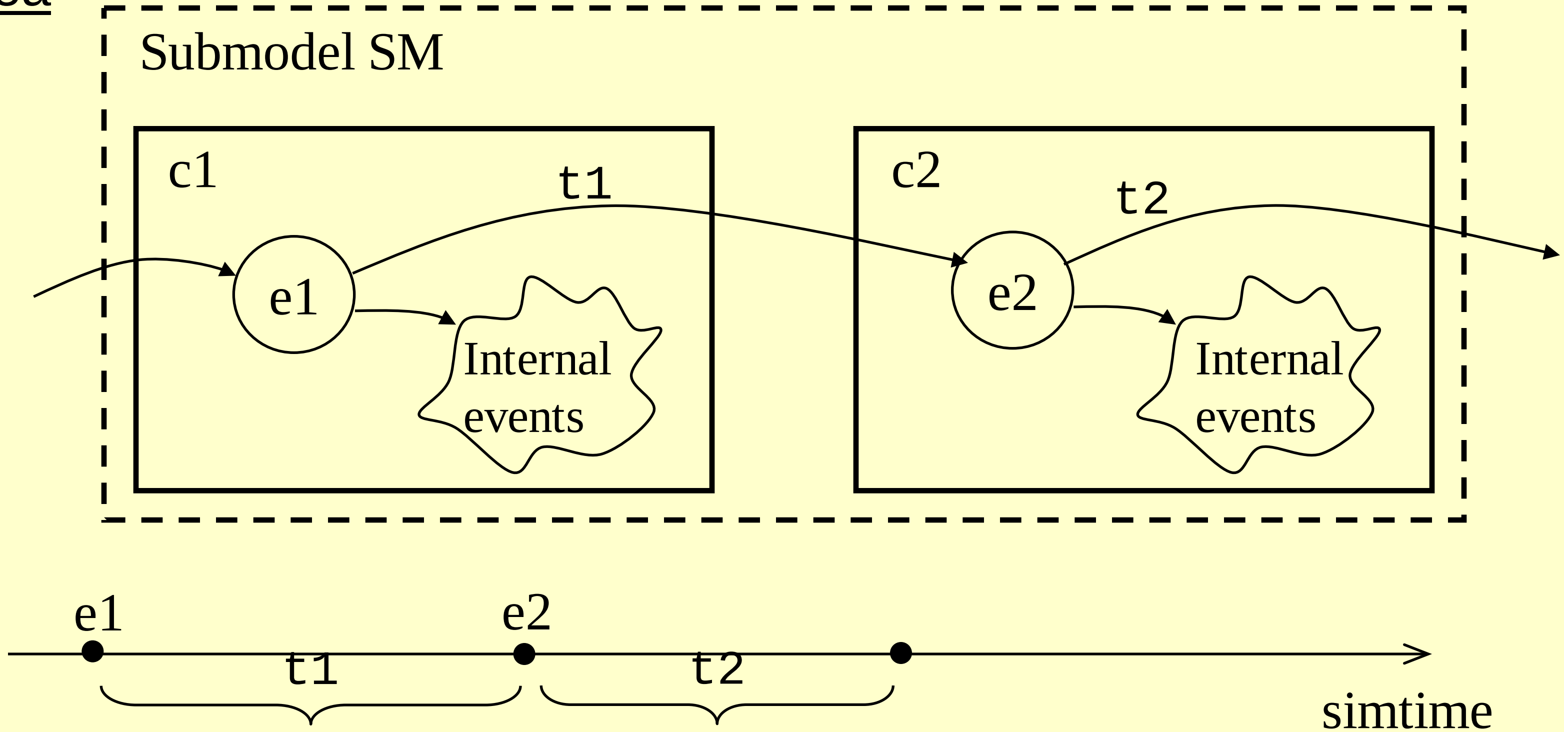
General Solution: **DeltaSM Algorithm**

begin
$SOL = \emptyset$;
foreach ($c \in C_{SM}$)
  $\gamma_c(t) = \delta_c(t)$; $Q.insert(c)$;
while ($\exists\, x \in C_{out}$ such that $x \notin SOL$) {
  $c = Q.get\_min()$;
  foreach ($x \in SUCC_c$)
    if ($\gamma_x(t) > \gamma_c(t) + \sigma_x(\gamma_c(t))$) {
      $\gamma_x(t) = \gamma_c(t) + \sigma_x(\gamma_c(t))$;
      $Q.sort(x)$;
    }
  $SOL = SOL \cup \{c\}$;
}
$\delta_{SM}(t) = \min_{\forall c \in Cout} (\gamma_c(t))$;
end

- Derived from Dijkstra's shortest path algorithm
- Can compute exact δ_{SM} or lower bound on it
- Hierarchical models first transformed using the FHG transform

Cumulative Message Pre-Sending

Idea



- Non-cumulative pre-sending: `t2` time units ahead of time
- e2 processed at the same time as e1
  ⇒Cumulative pre-sending: `t1+t2` time units ahead of time

Cumulative Message Pre-Sending (cont.)

The Method: **Immediate Message Forwarding (IMF)**

- Conditions
  - special component construction with F-events
    - $\Rightarrow$ IMF-capable components
  - timestamp order of input messages
    - $\Rightarrow$ IMF-enabled components

- Simulation executive support
  - identification of IMF-enabled components
    - $\Rightarrow$ The IEE algorithm
  - higher priority of F-events

- Other properties
  - applicable to hierarchical models
  - applicable to other synchronization schemes

Experimental Verification and Results

Objectives of the Experiments

- Verification of the simulator design and of the feasibility of developed methods
- Measurement of performance contribution of cumulative lookahead

Experimental Models

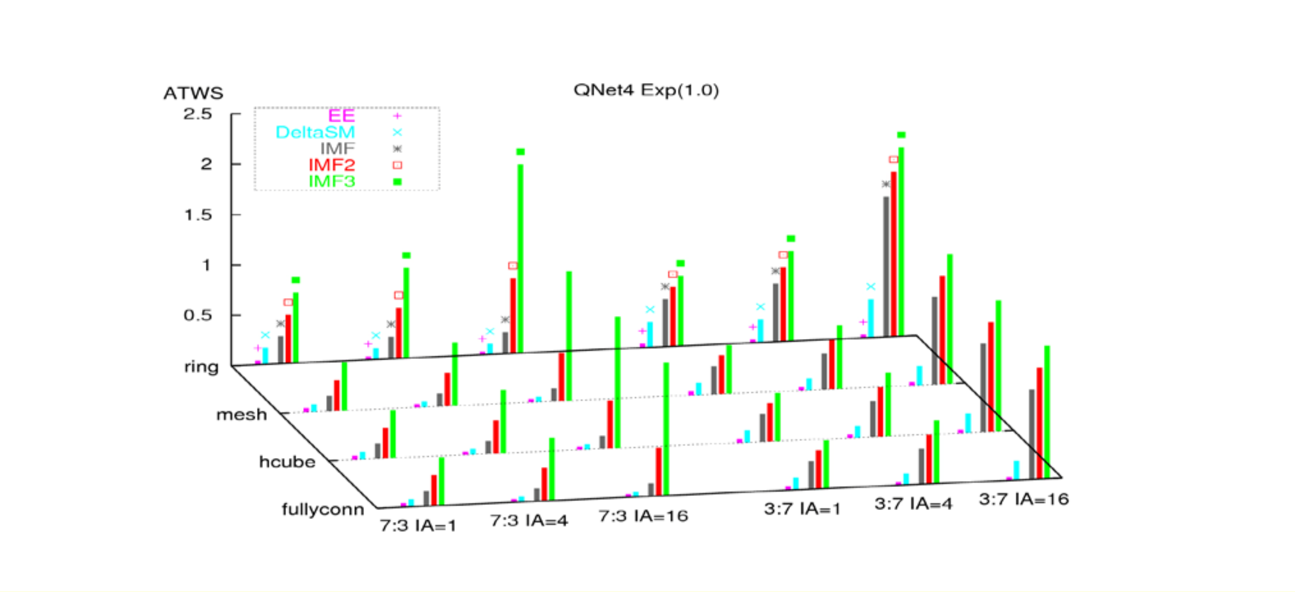
- Closed networks of 64 nodes decomposed into 8 submodels
- 2 or 4 components within a node
- Parameters
  - topology (ring, mesh, hypercube, fully connected)
  - node type (QNet2, QNet4, PHold)
  - time distribution (const, shifted exp, exp)
  - time distribution ratio (3 : 7, 7 : 3)
  - initial arrivals per node (1, 4, 16)
  - lookahead level (2 non-cumulative, several cumulative)

Experimental Verification and Results (cont.)

Performance Measures

- Average time window size (ATWS)
- Average number of events per window (ANEW)

Results



- Performance depends on parameters
- Up to 57-fold increase of ATWS

Conclusions

<u>Fulfilment of the Objectives</u>

- Parallel simulator design
  - hierarchical model decomposition/composition
  - user transparency through component re-use
  - no “logical process” model view
  - LLG transformation

- Cumulative timestamp prediction:
  - the DeltaSM algorithm
  - proof of its properties
  - its complexity analysis
  - FHG transformation for hierarchical models

Conclusions (cont.)

Fulfilment of the Objectives (cont.)

- Cumulative message pre-sending
  - Immediate message forwarding
  - conditions for IMF and its applicability
  - IEE algorithm
  - cumulative timestamp prediction based on min. delays
  - applicability to other synchronization methods
  - IMF in hierarchical models

- Experimental verification and evaluation
  - verification of the simulator design
  - verification of the feasibility of cumulative lookahead
  - measurement of performance contribution
    - ⇒ cumulative lookahead can significantly contribute

Conclusions (cont.)

Thesis Contributions Summary

- Theoretical: in-depth study of the issue of cumulative lookahead
- Practical: simulator providing a solution to PDES modeling problems

Possible Future Directions

- Application of cumulative lookahead in more models
- Implementation and evaluation of cumulative lookahead in other synchronization methods
- Analytical evaluation of cumulative lookahead