Modelling, Analysis and Scheduling with Dataflow Models

Marc Geilen, Bart Theelen,
Twan Basten, Sander Stuijk, AmirHossein Ghamarian,
Jeroen Voeten

Eindhoven University of Technology

Department of Electrical Engineering
Electronic Systems

Overview

- Dataflow model of computation
- Models of multiprocessor SoC and NoCs
  - Integrated model of application, architecture and mapping
- Synchronous Dataflow Graphs
  - Exact algorithms based on state-space exploration
- Kahn Process Networks
  - Implementation of KPNs, deadlock detection and resolution
- Reactive Process Networks
  - Integrating reactive behaviour with dataflow
- Scenarios and Trade-offs
  - Improving WCET estimation
  - Working with trade-offs

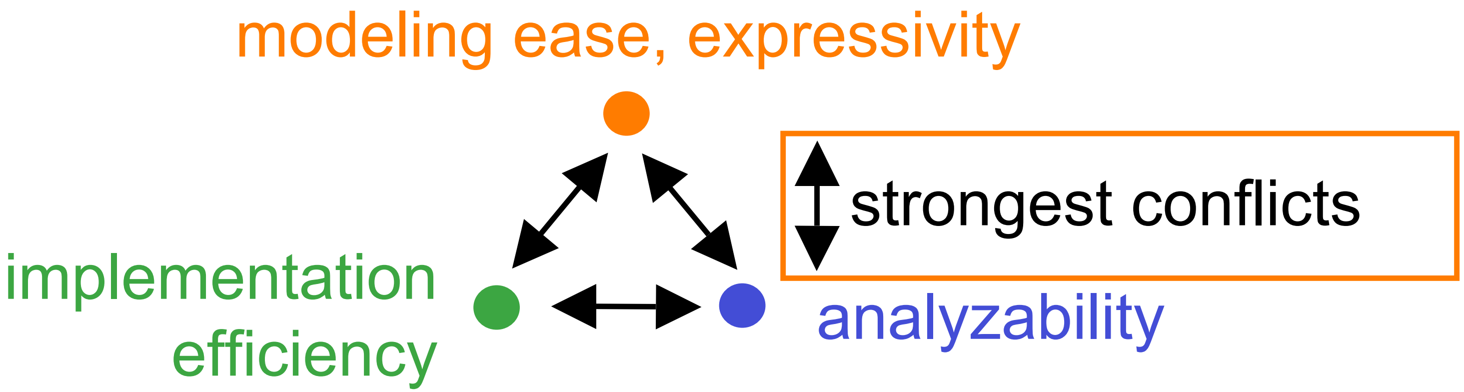
Models of Computation

Basic dimensions	Aspects	Derivatives
processing	functionality concurrency timing energy quality ...	(for complete system models) execution time energy dissipation perceived quality reconfiguration time ...
communication	functionality concurrency timing energy quality ...	
storage		

Our Goals

- Models of computation (and communication)
- Aimed at automated analysis and synthesis
- Solid mathematical foundation
- Basic algorithms and properties
- Domain of embedded systems, often NoC based MPSoc
- Towards variability and sensor networks

Conflicting Requirements



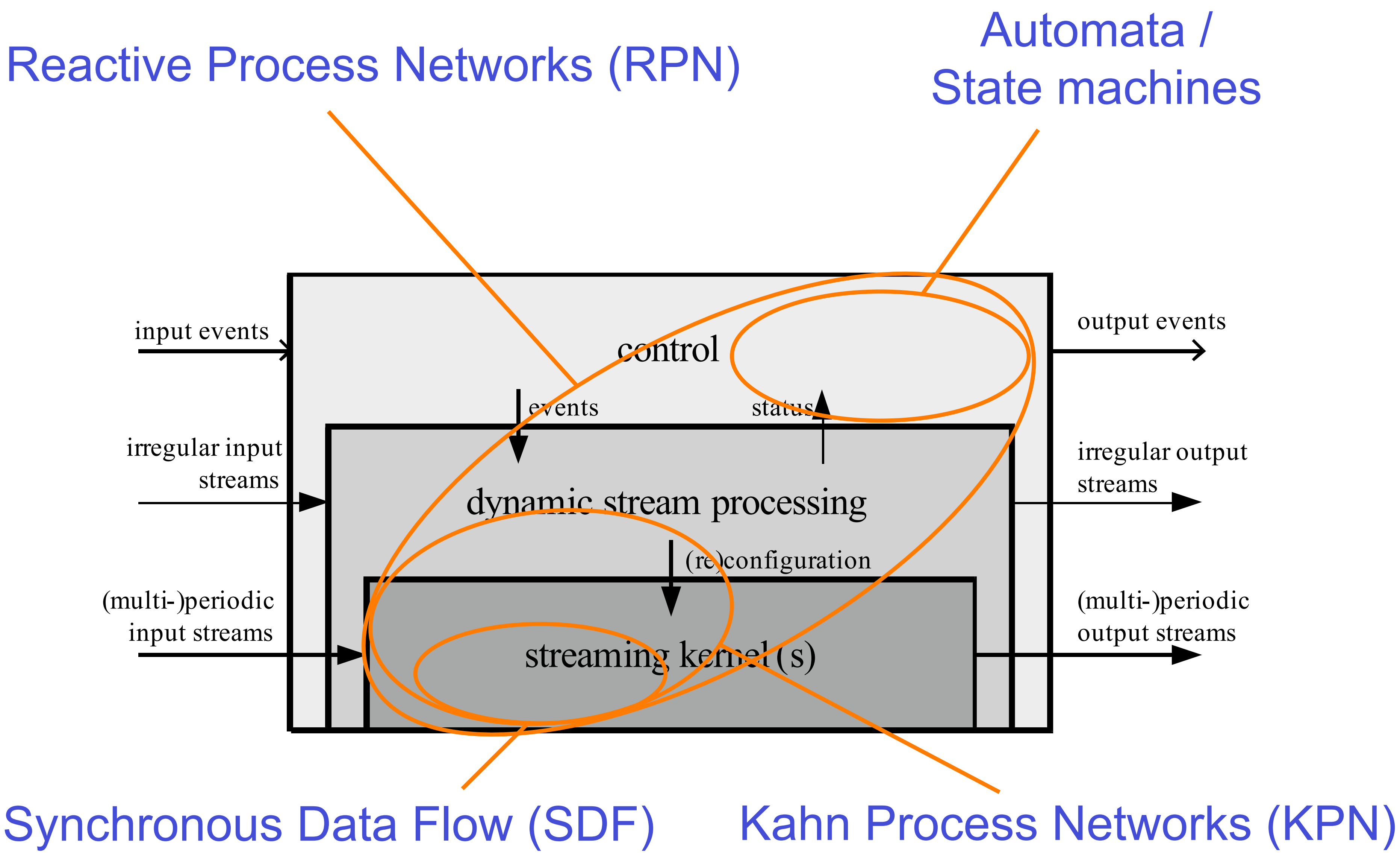
Kahn Process Networks:

+ (streaming), - (only functional behavior), o (no static implementations)

Synchronous Dataflow Graphs:

- (periodic kernels), + (timing, schedulability), + (static implementations)

Application Domain: Multimedia Applications



Streaming MoCs and the Design Flow

Reactive Process Networks (RPN)

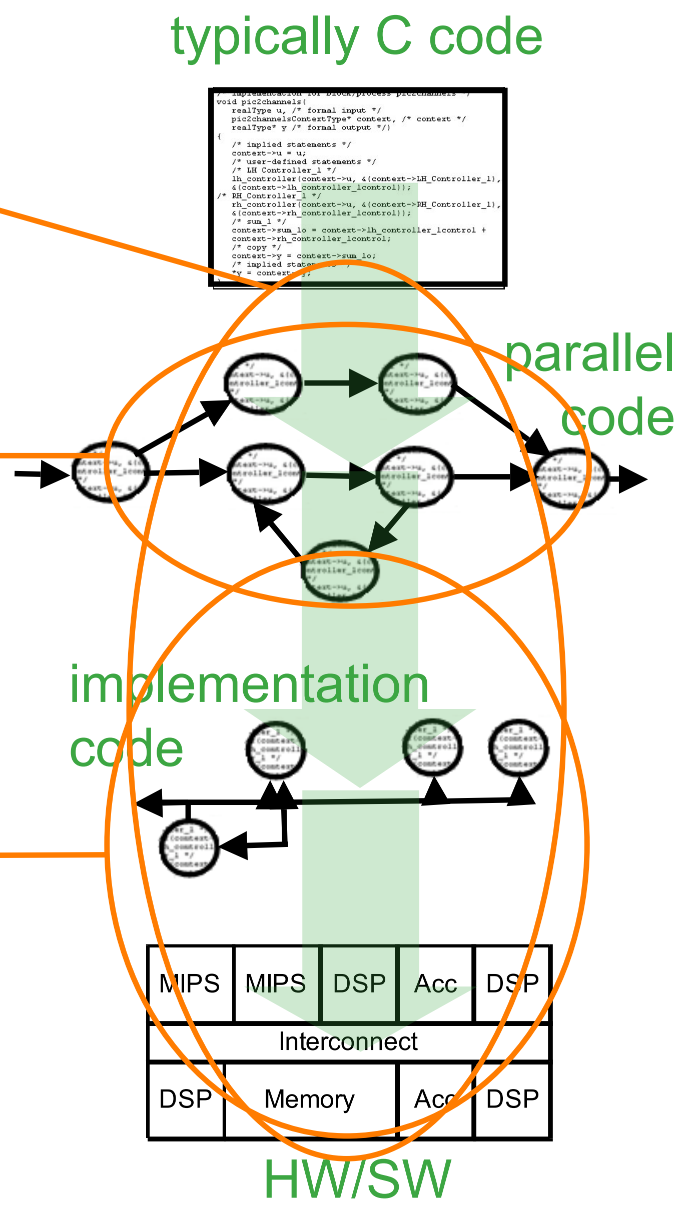
(Semantic framework)

Kahn Process Networks (KPN)

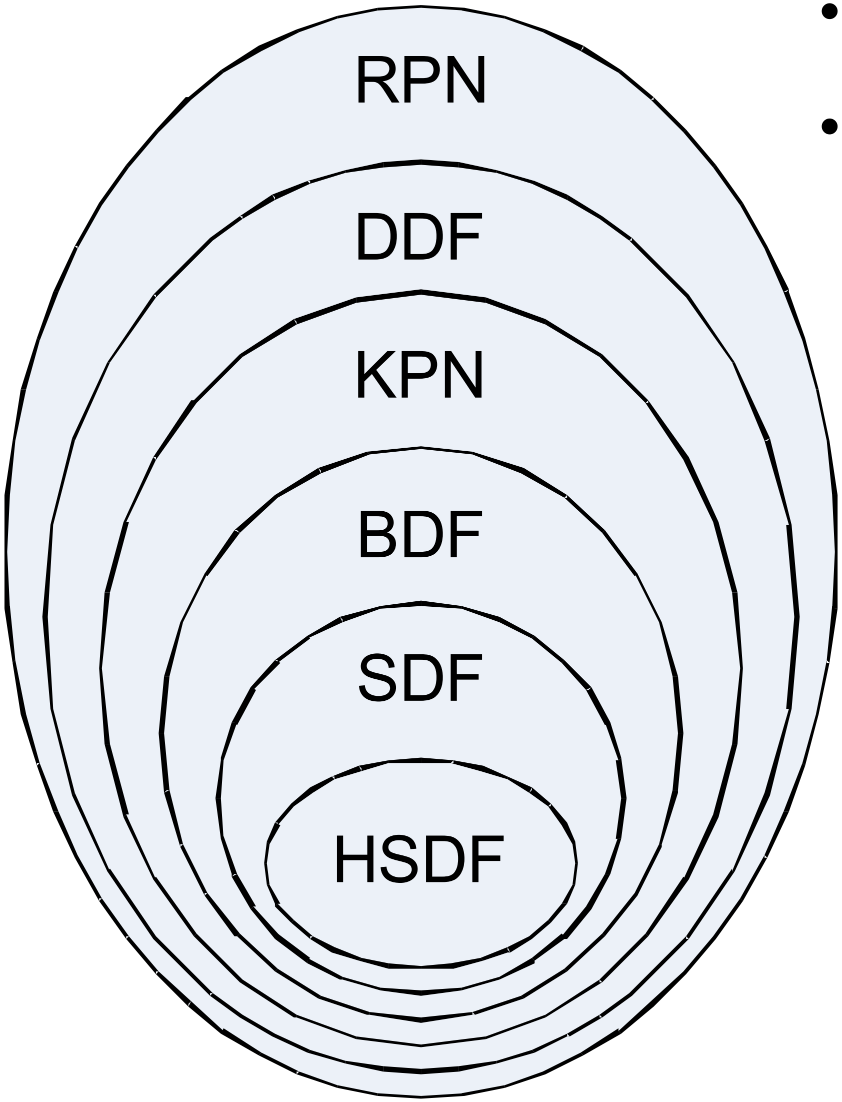
(Executable specifications)

(Synchronous) Data Flow

(Implementation and analysis)



Streaming MoCs Expressiveness Hierarchy



- BDF and larger: Turing complete
- Better notions of expressiveness needed

RPN: Reactive Process Networks

KPN: Kahn Process Networks

DF: Data Flow

DDF: Dynamic DF

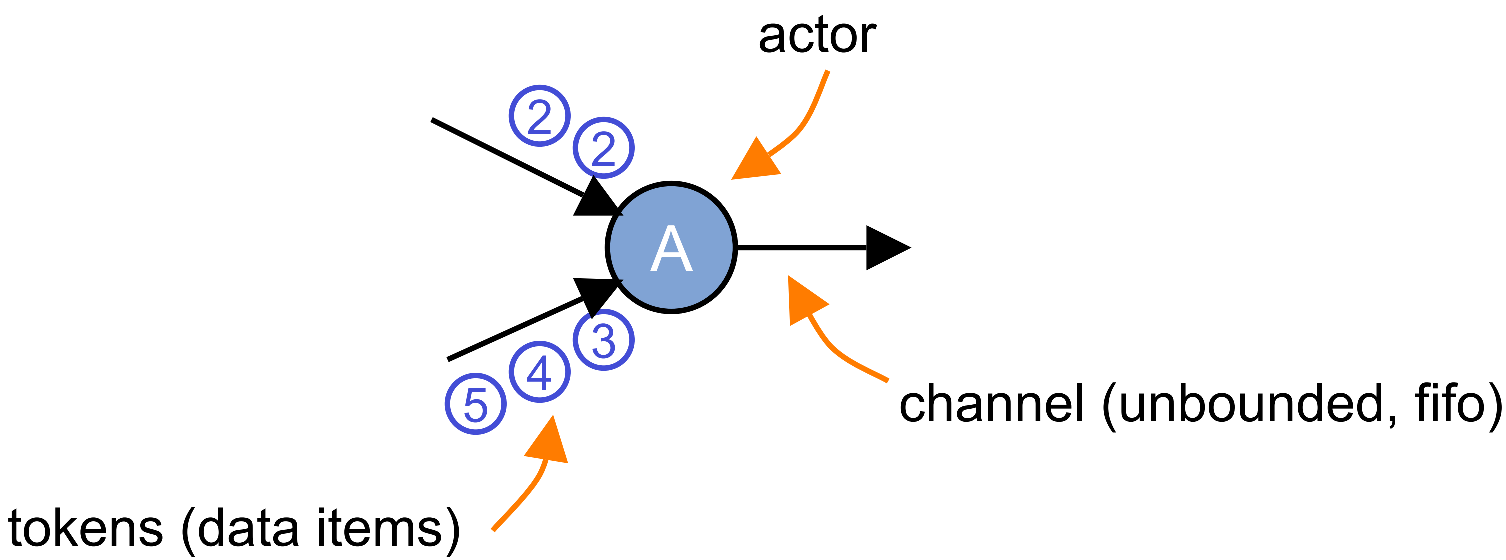
BDF: Boolean DF

SDF: Synchronous DF

HSDF: Homogeneous SDF

Dataflow Actors

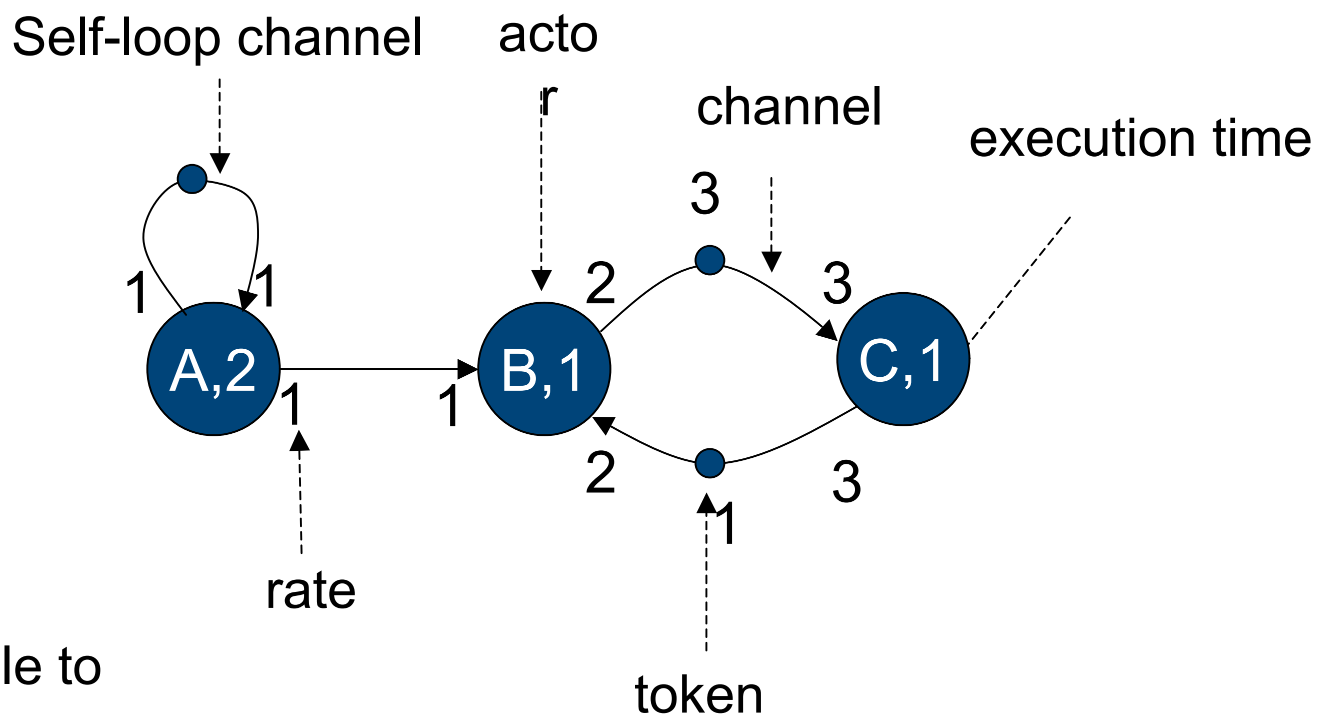
- Basic building blocks of any type of dataflow graph



- Actor firing: consumption and production of tokens

Synchronous Dataflow (SDF) Graphs

- Actor fires atomically, reading and writing fixed number of tokens on the inputs and outputs
- Formal semantics models conservatively the behaviour of real kernels



Comparable to

Computation Graphs (Karp, Miller)

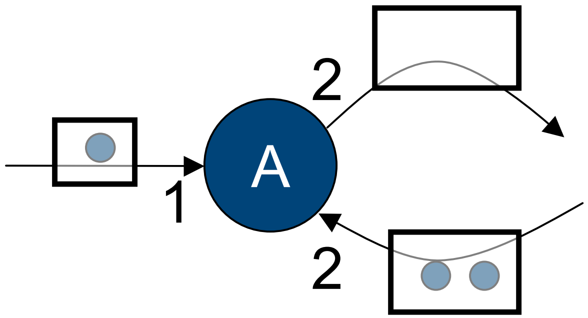
Weighted Marked (Timed) Graphs (Petri-Nets)

11 Synchronous Dataflow (SDF) Graphs

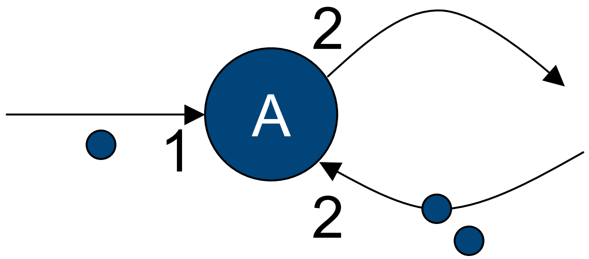
- Worst-case, conservative approximation of actual read and write instances and variable (bounded) execution time.
- Can model buffer storage limitations
- Can model scheduling decisions
- Models application mapping. E.g., communication latency or memory access latency
- Models multimedia applications
- Composibility, resource virtualization, TDMA

Conservative model

model execution

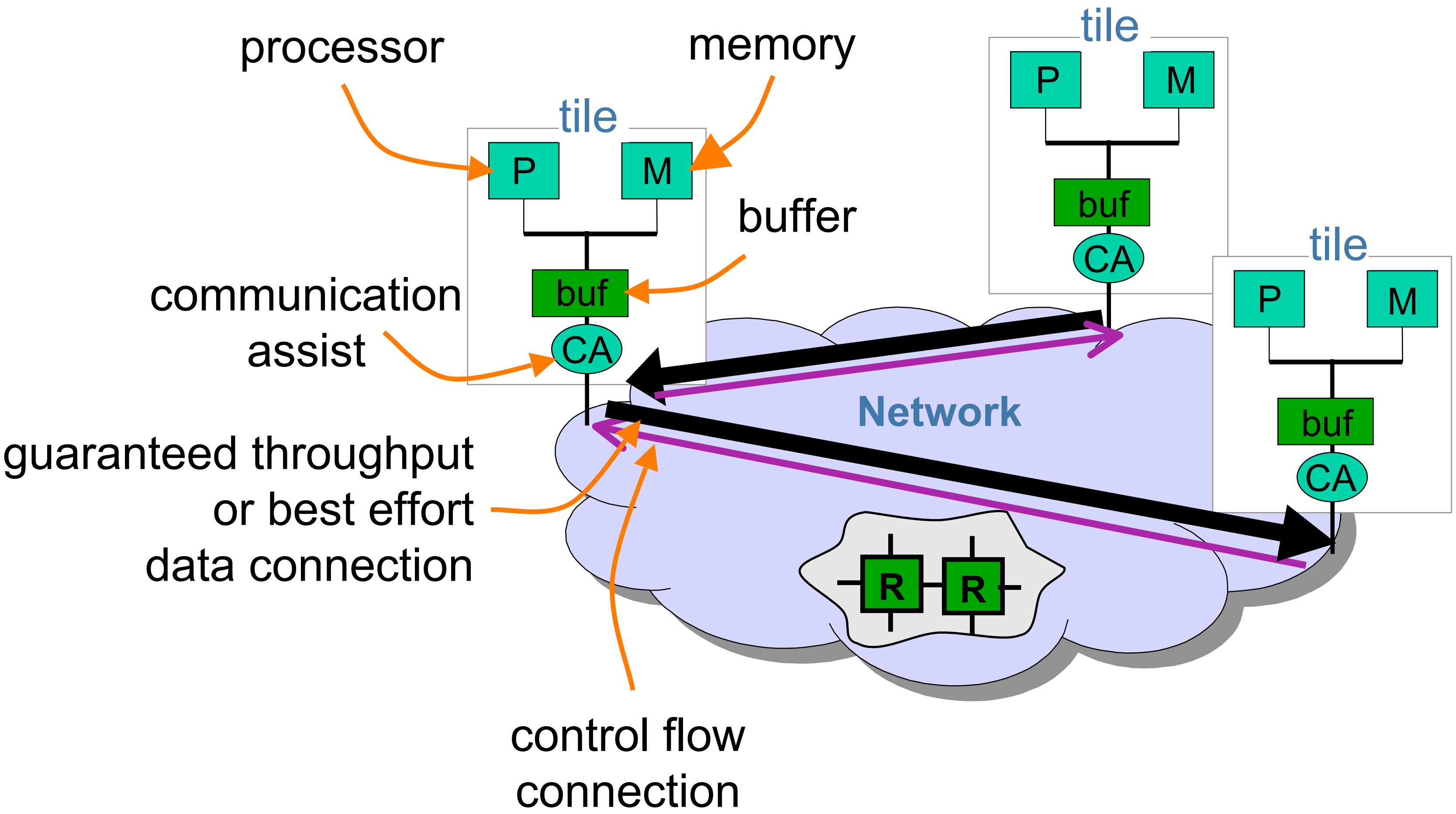


actual execution

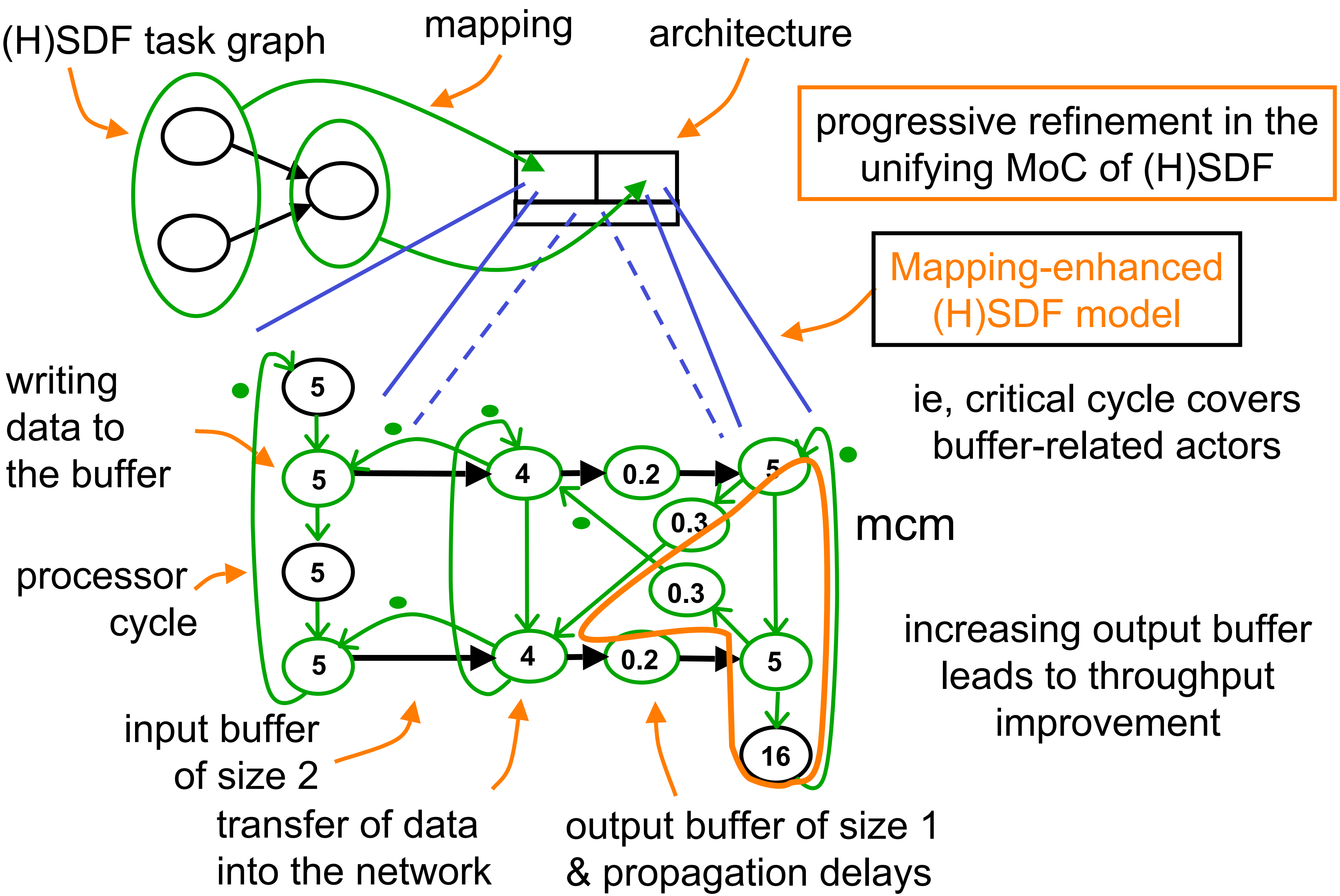


conservativity + monotonicity

A NoC-based SoC Platform



Philosophy



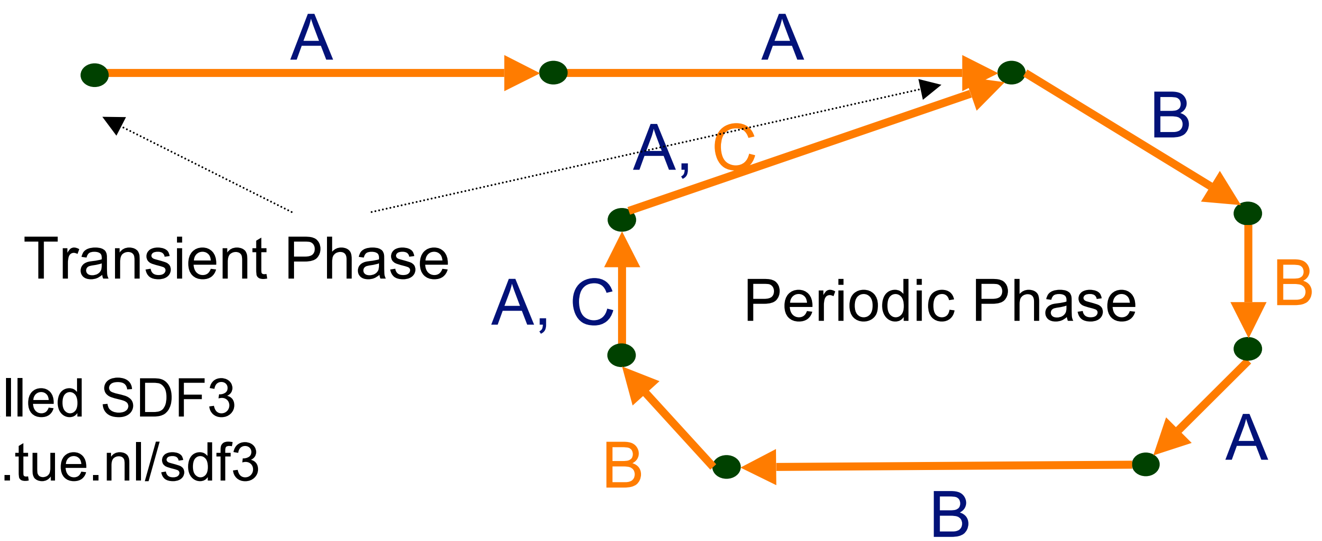
(H)SDF task graph
mapping
architecture
progressive refinement in the unifying MoC of (H)SDF
Mapping-enhanced (H)SDF model
ie, critical cycle covers buffer-related actors
mcm
increasing output buffer leads to throughput improvement
writing data to the buffer
processor cycle
input buffer of size 2
transfer of data into the network
output buffer of size 1 & propagation delays
5
5
5
5
4
4
0.2
0.2
0.3
0.3
5
5
16

Analysis

- Many problems NP Complete, defined on HSDF only
  buffer sizes, throughput, (deadlock?)

- Conversion SDF to HSDF leads to potentially exponential blowup

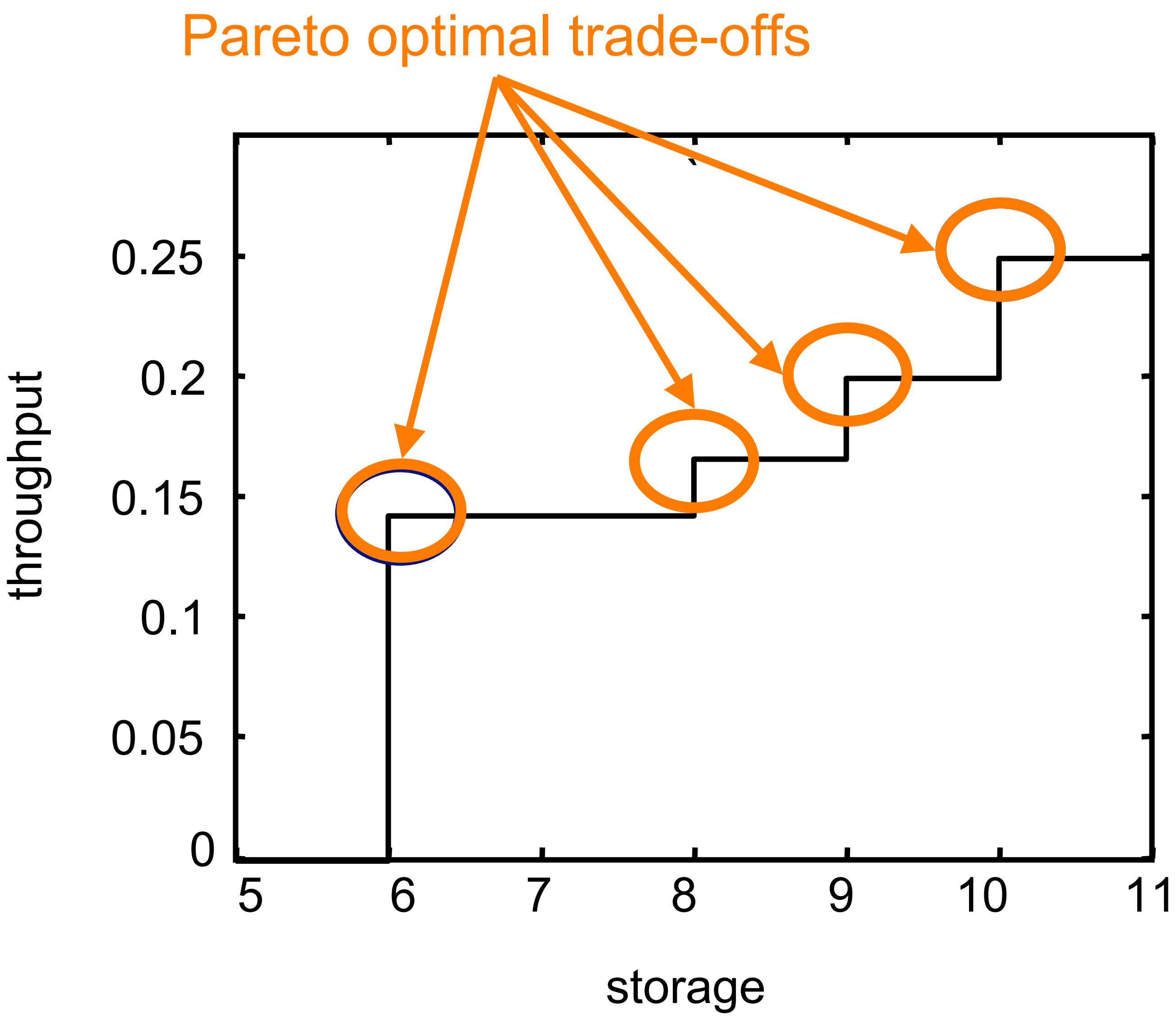
- We have implemented exact methods based on sate-space exploration
  Surprisingly efficient compared to Maximum Cycle Mean methods



- Tool set, called SDF3
  www.es.ele.tue.nl/sdf3

Analysis

- Throughput computation
- Latency definition for SDF
- Liveness, boundedness, consistency
  Based on throughput analysis
- Minimum buffer sizes
  State-space exploration; critical cycle analysis
- Buffer size vs. Throughput (latency) trade-offs
  Generalization of minimum buffer size problem


Pareto optimal trade-offs
0.25
0.2
0.15
0.1
0.05
0
throughput
5
6
7
8
9
10
11
storage

Limitations of SDF Graphs

Limitations

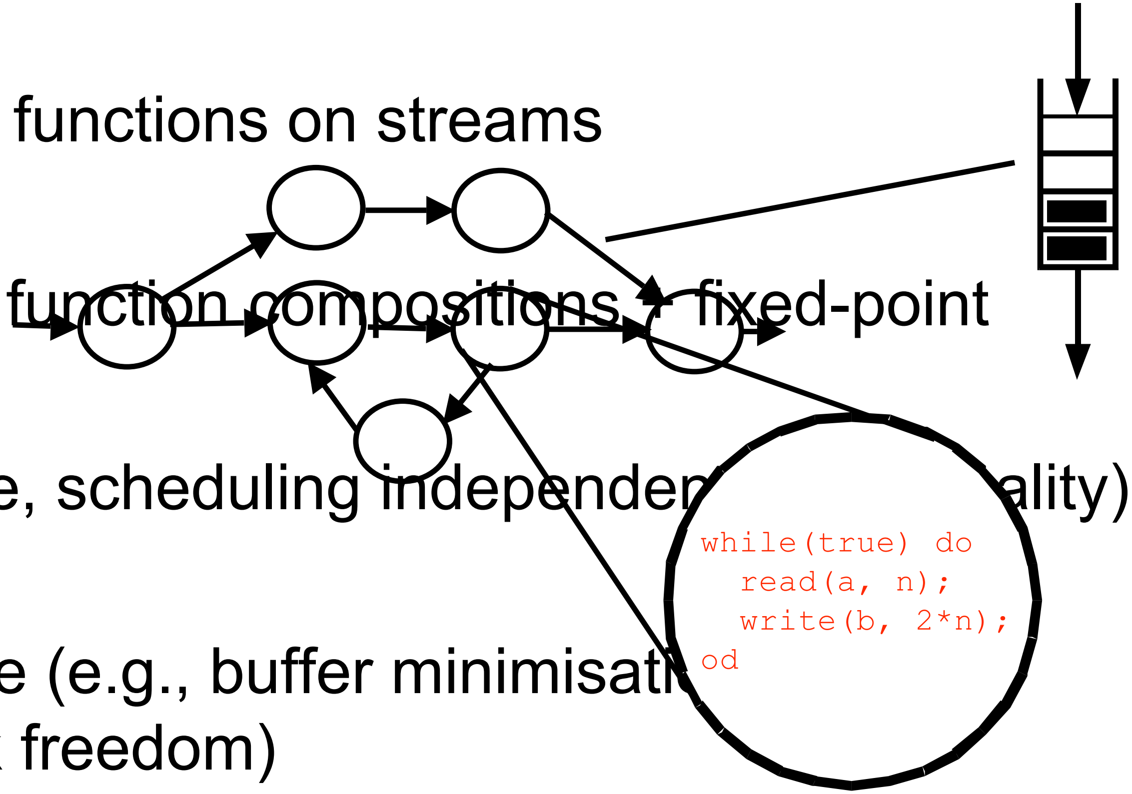
- Static, periodic behaviour
- Fixed execution times
  (potentially high over-estimation of worst-case)
- No Dynamic instantiation, reconfiguration

Extensions

- Kahn Process Networks (KPN)
- Scenarios (Scenario Aware Dataflow)
- Reactive Process Networks (RPN)

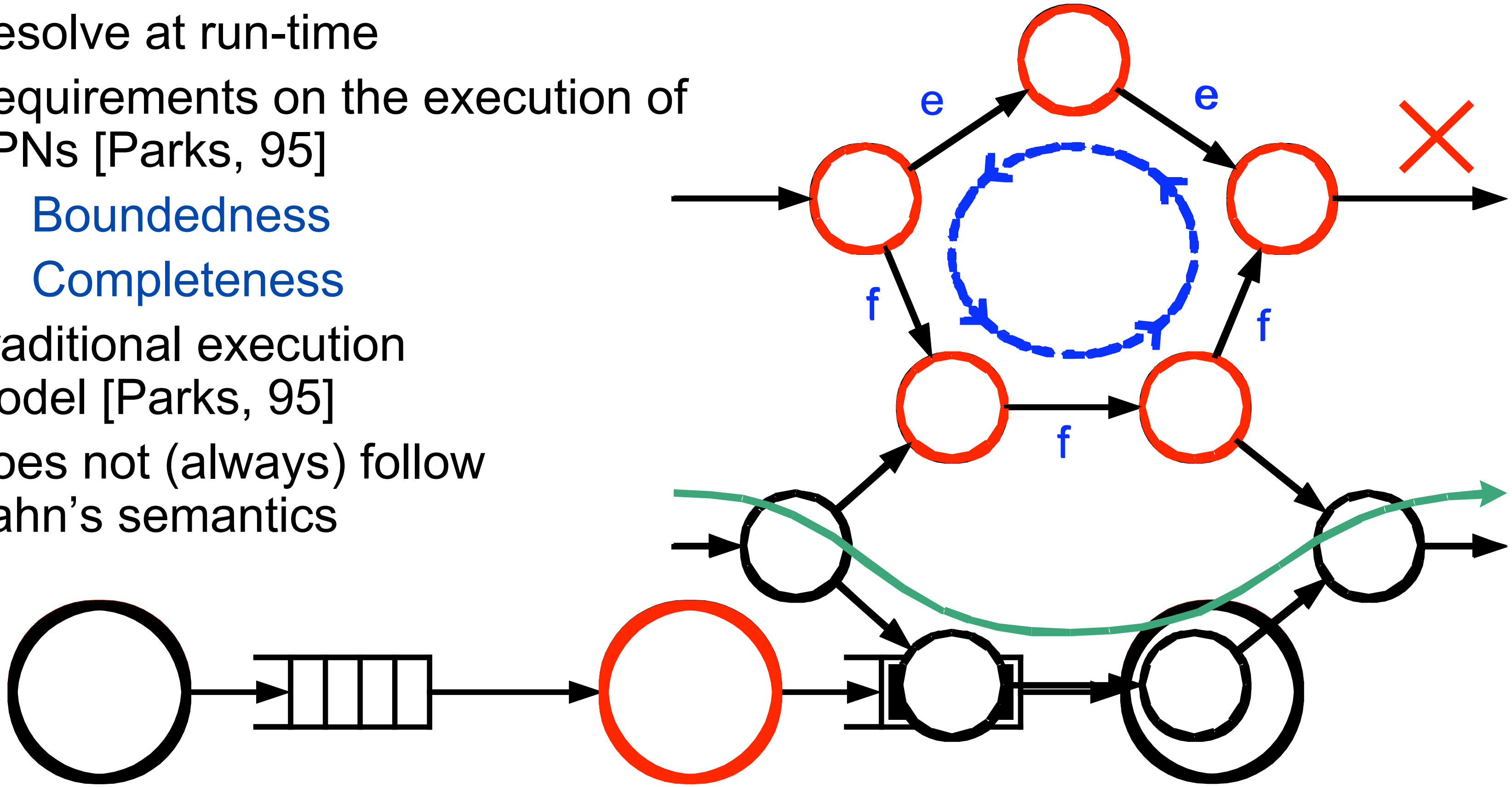
Kahn Process Networks (KPN)

- Data-dependent data rates
  (for instance Variable Length Decoding)
- Asynchronous reads and writes
- Continuous functions on streams
- Semantics: function compositions + fixed-point
- Determinate, scheduling independen ality)
- Undecidable (e.g., buffer minimisati
  or deadlock freedom)



KPN

- Buffer management needs to be done at run-time
- Too small buffers lead to artificial deadlock
  Resolve at run-time
- Requirements on the execution of KPNs [Parks, 95]
  - Boundedness
  - Completeness
- Traditional execution model [Parks, 95]
- Does not (always) follow Kahn's semantics



21 Improved KPN scheduler

- A scheduler that is correct for every KPN cannot exist

- A scheduling algorithm has been defined which is correct for every bounded and effective KPN. It is executed in bounded memory by our scheduler and produces the complete output.

1. Schedule enabled processes (in a fair way)
2. Until (local) deadlock occurs (cyclic causal chain)
3. Resolve deadlock by increasing the smallest full FIFO

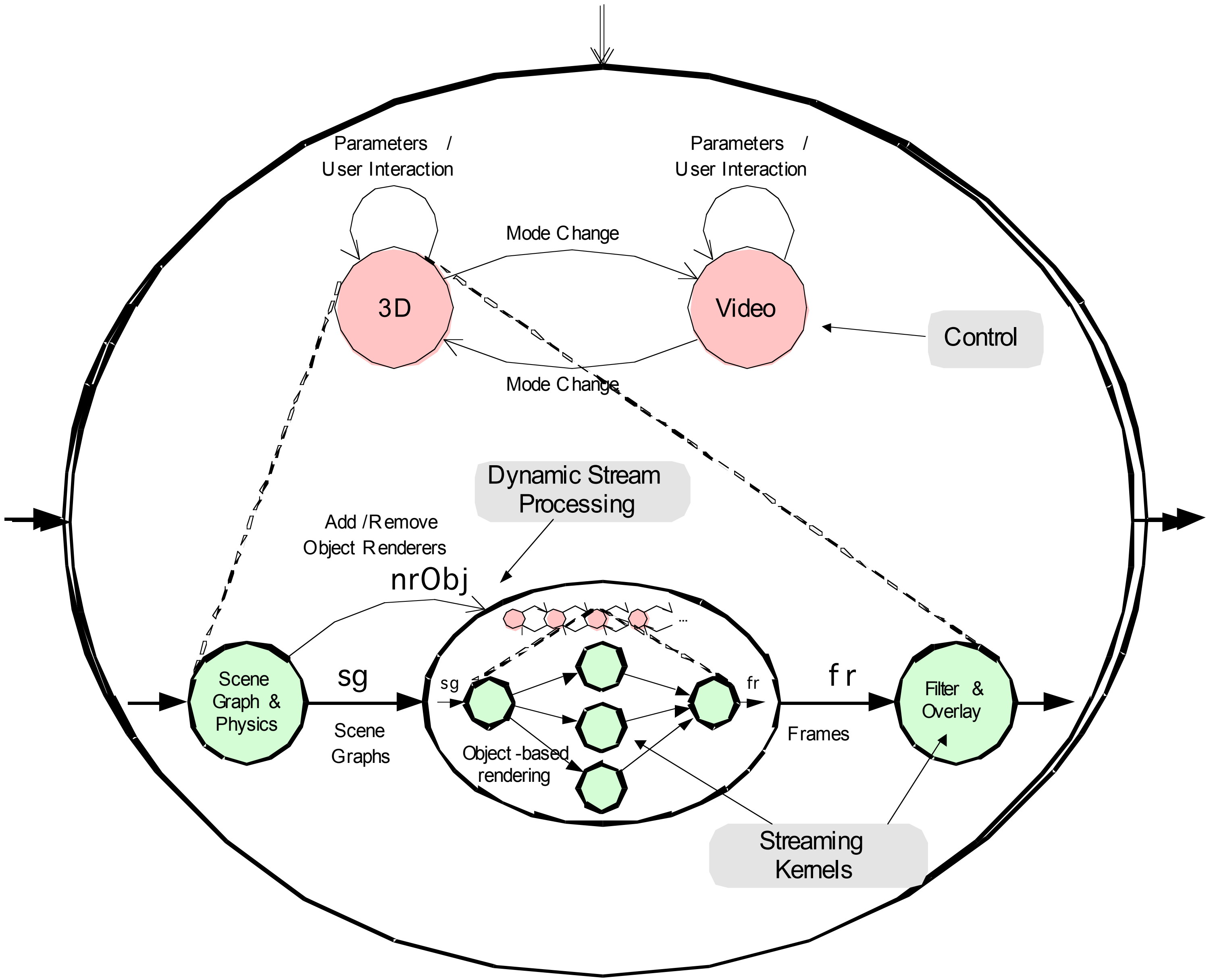
Reactive Process Networks

Limitations of KPN

- No non-determinism / reactive behaviour
- Only continuous functions

Extension Reactive Process Networks

- Integrate reactive behaviour, (sporadic) event communication with determinate dataflow models
- Coordination between streaming and control required to get some level of predictability
- Determinacy is lost



Parameters /
User Interaction
Parameters /
User Interaction
Mode Change
3D
Video
Control
Mode Change
Dynamic Stream
Processing
Add /Remove
Object Renderers
nrObj
...
Scene
Graph &
Physics
sg
Scene
Graphs
sg
fr
f r
Frames
Filter &
Overlay
Object -based
rendering
Streaming
Kernels

24 Scenarios

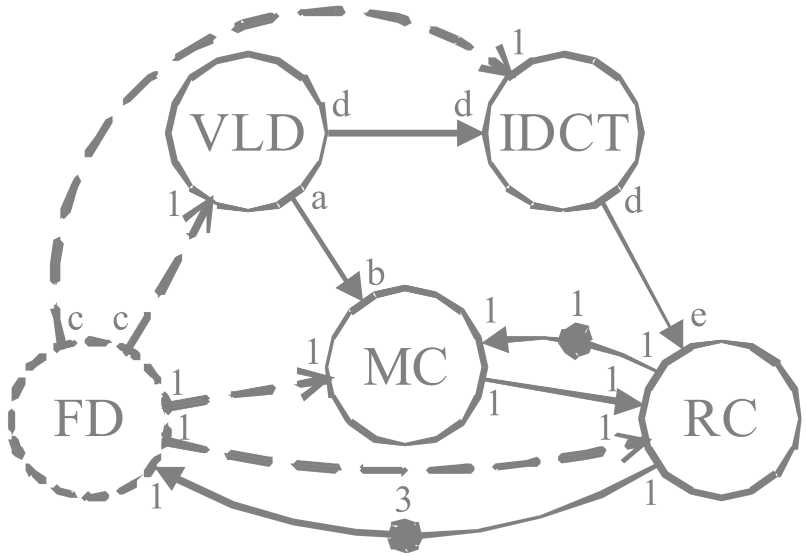
- Worst-case execution time estimation is overly pessimistic in many modern dynamic applications
- Scenarios are detectible situations in which better bounds can be given. This can e.g. be exploited by on-line scheduling and frequency and voltage scaling methods.

25 Scenario-Aware Data Flow (SADF)

- Problem - traditional data flow models are not capable of expressing the dynamism of modern streaming applications

- Dynamism supported by SADF
  - Parameterized numbers of consumed/produced tokens
  - Inactive processes
  - Discrete execution time distributions
  - Correlations between above for different processes

- Example: Pipelined MPEG-4 SP

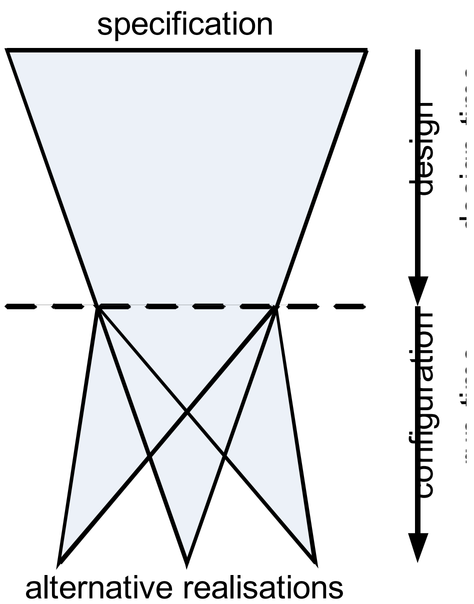


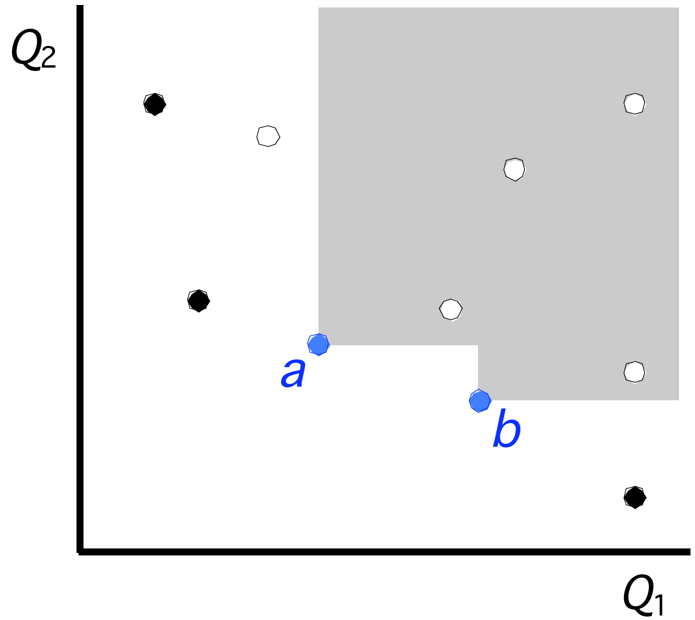
- Analysis of correctness as well as worst-case and long-run average performance is possible

- Tool support for exhaustive and simulation-based performance analysis

Analysing trade-offs

- Inherent trade-offs between computation, communication, storage and quality or other non-functional properties
- Pareto points are abstractions of multi-dimensional optimisation criteria / cost functions
- An Algebra of Pareto Points
  Incremental, compositional computation of system trade-offs.





Challenges

- Expressivity while maintaining analyzability
- Dynamism and reconfiguration
- Abstraction without loosing accuracy
- Composability and compositionality
- MoCs for non-functional aspects
- Unification or integration of MoCs

References

Can be found on www.es.ele.tue.nl/~epicurus

- Throughput Analysis of Synchronous Data Flow Graphs
  A.H. Ghamarian, M.C.W. Geilen, S. Stuijk, T. Basten, A.J.M. Moonen, M.J.G. Bekooij, B.D. Theelen and M.R. Mousavi. ACSD 2006

- Application Scenarios in Streaming-Oriented Embedded System Design
  S.V. Gheorghita, T.Basten, H. Corporaal.
  In International Symposium in System-on-Chip (SoC 2006)Execution of KPN [ESOP 2003]

- A Scenario-Aware Data Flow Model for Combined Long-Run Average and Worst-Case Performance Analysis
  B.D. Theelen, M.C.W. Geilen, T. Basten, J.P.M. Voeten, S.V. Gheorghita and S. Stuijk. MEMOCODE 2006

- Exploring Trade-Offs in Buffer Requirements and Throughput Constraints for Synchronous Dataflow Graphs
  S. Stuijk, M.C.W. Geilen and T. Basten. In Design Automation Conference, DAC 06,

- An Algebra of Pareto Points
  M.C.W. Geilen, T. Basten, B.D. Theelen, R.H.J.M. Otten. ACSD 2005

- A toolkit: SDF^3, http://www.es.ele.tue.nl/sdf3 [ACSD 2006]

- Liveness and Boundedness of Synchronous Data Flow Graphs
  A.H. Ghamarian, M.C.W. Geilen, T. Basten, B.D. Theelen, M.R. Mousavi and Sander Stuijk. FMCAD 2006.

- Predictable embedding of large data structures in multiprocessor networks-on-chip
  S. Stuijk, T. Basten, B. Mesman and M.C.W. Geilen. In Digital System Design, 8th Euromicro Conference, DSD 05

- Reactive Process Networks.
  M.C.W. Geilen and T. Basten. EMSOFT 2004

- Requirements on the Execution of Kahn Process Networks
  M.C.W. Geilen and T. Basten. ESOP 2003