Scenario-Aware Dataflow

Bart Theelen (B.D.Theelen@tue.nl)

Department of Electrical Engineering
Electronic Systems

Overview

- Introduction
- Scenario-Aware Dataflow
  - MPEG-4 AVC Shape Decoder
  - MPEG-4 SP Decoder
  - Channel Equalizer
  - MP3 Decoder
- Tool Support
- Conclusions

Introduction

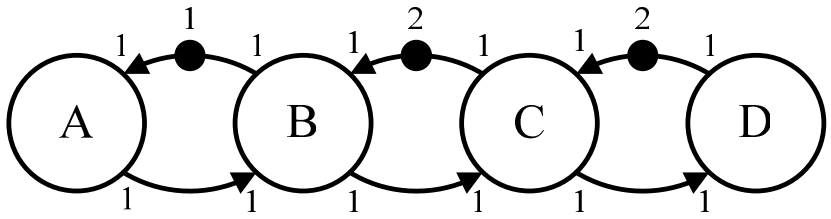
- Goal
  - Design-time performance prediction of hard & soft real-time streaming applications running on platforms with guaranteed and/or (lossless) best-effort services
- Starting point
  - Data flow models allow specifying streaming applications
- But, traditional data flow models
  - Can't express dynamism (SDF)
  - Don't support relevant analysis techniques (KPN)
- Current approach uses SDF to capture worst-case resource usage
  - Large overestimation of required resources
  - Inefficient use of a too expensive platform
  - Suitable for hard real-time/guaranteed services only

4 Scenario-Aware Dataflow (SADF)

- SADF extends SDF with scenarios
  - Scenarios classify behaviour based on resource requirements
- SADF can express
  - Parameterised numbers of communicated tokens
  - Inactive processes
  - Discrete execution time distributions
  - (Hierarchical) correlations between the above for different processes
  - Pipelined reconfiguration
  - Scenario changes do not need to follow reoccurring fixed patterns
- SADF combines
  - Streaming data and control
  - Hard and soft real-time
  - Design-time verification of functional correctness
  - Design-time performance analysis of worst/best case and average case

5 SADF = SDF + Execution Time Distributions

- MPEG-4 AVC Shape Decoder
  - 'HSDF' where execution time of A depends on block type



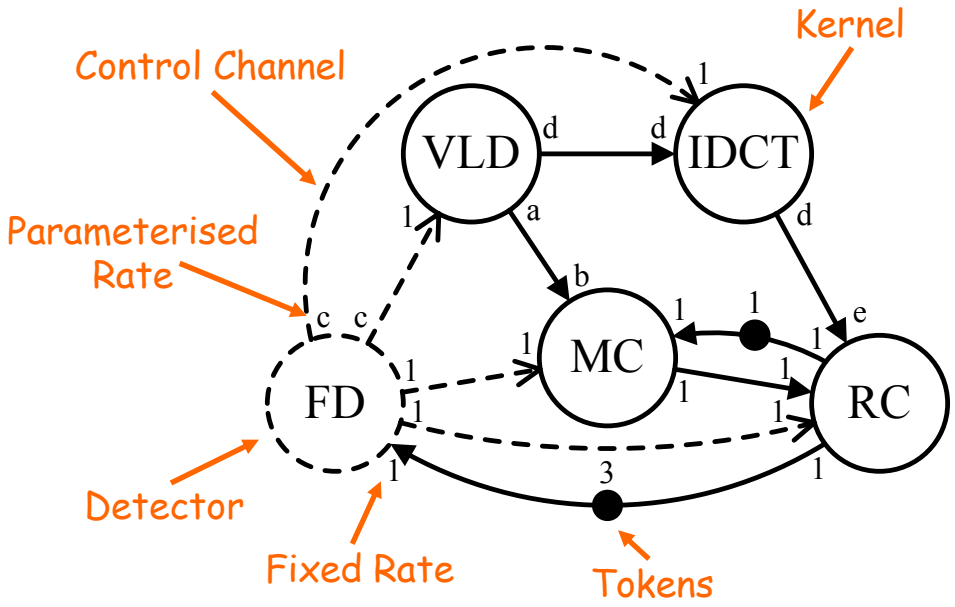
A	S_1	1.858ms
	S_2	4.788ms
	S_3	6.055ms
B	All	0.980ms
C	All	3.816ms
D	All	3.816ms

- Approach 1 – SDF for each block type
  - Throughput for S_1 = 0.262055, S_2 = 0.17337 and S_3 = 0.142146
- Approach 2 - SDF with symbolic execution time [DATE'08]
  - Replace options with symbol t
  - Throughput = 1/max(3.816, 0.98 + t)
- Approach 3 – SADF with only kernels
  - Thee possible samples of distribution
  - Throughput = 0.191256

Auto-concurrency excluded implicitly for (H)SDF

6 SADF = SDF + Parameterised Rates

- MPEG-4 SP Decoder



Rate	I	P_0	P_x
a	0	0	1
b	0	0	x
c	99	1	x
d	1	0	1
e	99	0	x

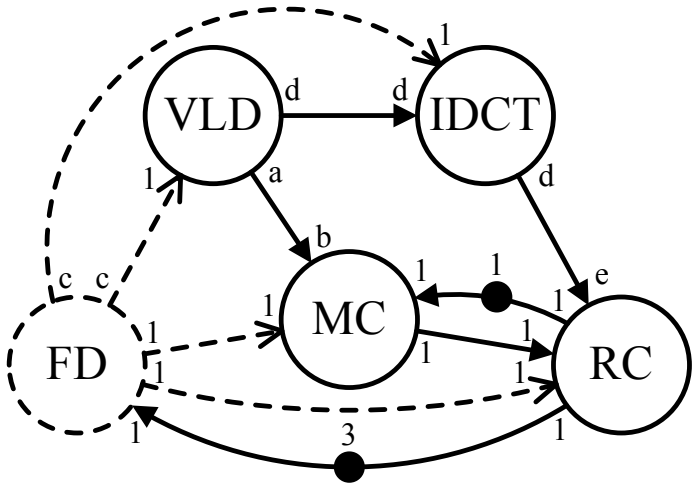
x = {30, 40, 50 ,60, 70, 80, 99}

VLD and IDCT fire per macro block
FD, MC and RC fire per frame

- FD detects frame type (9 scenarios)
  - I-frame (99 macro blocks, 0 motion vectors)
  - P_x-frame (x macro blocks, x motion vectors)
  - P_0-frame (still video)

7 SADF = SDF + Parameterised Rates

- MPEG-4 SP Decoder

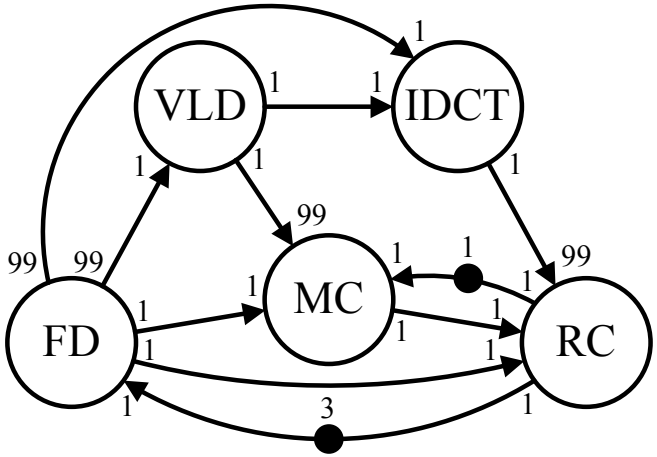


VLD	P_0	0
	All except P_0	40
IDCT	P_0	0
	All except P_0	17
MC	I, P_0	0
	P_{30}	90
	P_{40}	145
	P_{50}	190
	P_{60}	235
	P_{70}	265
	P_{80}	310
	P_{99}	390
RC	I	350
	P_0	0
	P_{30}, P_{40}, P_{50}	250
	P_{60}	300
	P_{70}, P_{80}, P_{99}	320
FD	All	0

- Worst-case resource requirements do not occur in a single scenario
  - Worst-case rates in P_{99}
  - Worst-case execution times in I & P_{99}

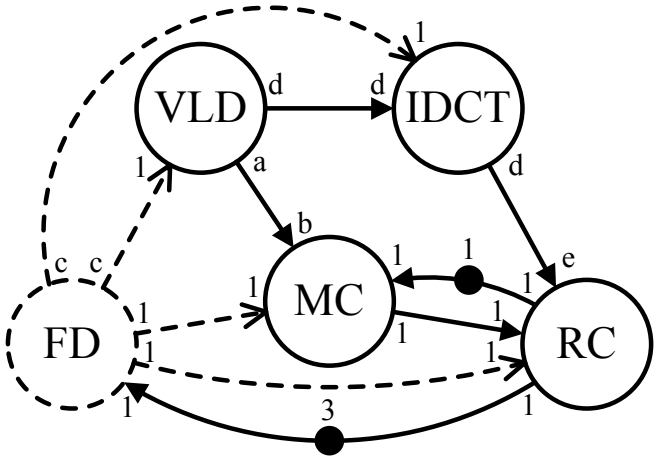
SDF versus SADF

- SDF worst-case approximation is inadequate



SDF

Throughput = 0.252525



SADF

Throughput = 0.425571

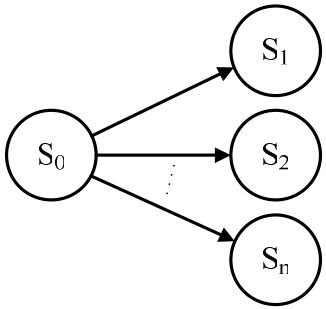
Taking dynamism into account is good idea

Auto-concurrency excluded implicitly for SDF

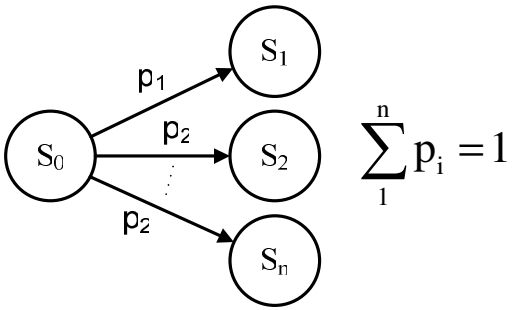
SADF Semantics (1)

- Kernel semantics
  - Wait for scenario-valued control token on control inputs
  - Fix scenario and hence rates and execution time distribution
  - Finalise firing as SDF actor

- Detectors contain automata to capture occurrences of scenarios
  - Real-life: data-dependent control behaviour (normal state machine)
  - Worst/best case only model: non-deterministic state machine
  - Worst/best case and average case model: Markov chain



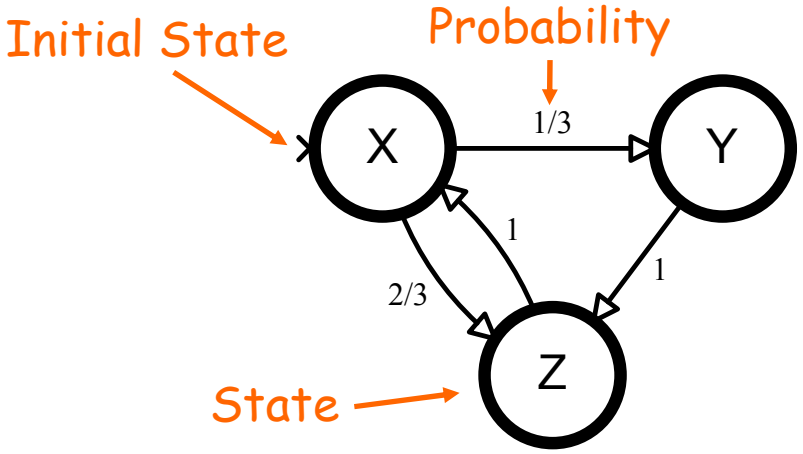
non-deterministic state machine



Markov chain

SADF Semantics (2)

- Detector semantics
  - Wait for scenario-valued control token on control inputs
  - Fix scenario (select automaton corresponding to control tokens values)
  - Fix subscenario based on next state of selected automaton
    - Subscenario gives rates and execution time distribution
  - Finalise firing as SDF actor
    - Tokens produced to control channels are scenario valued



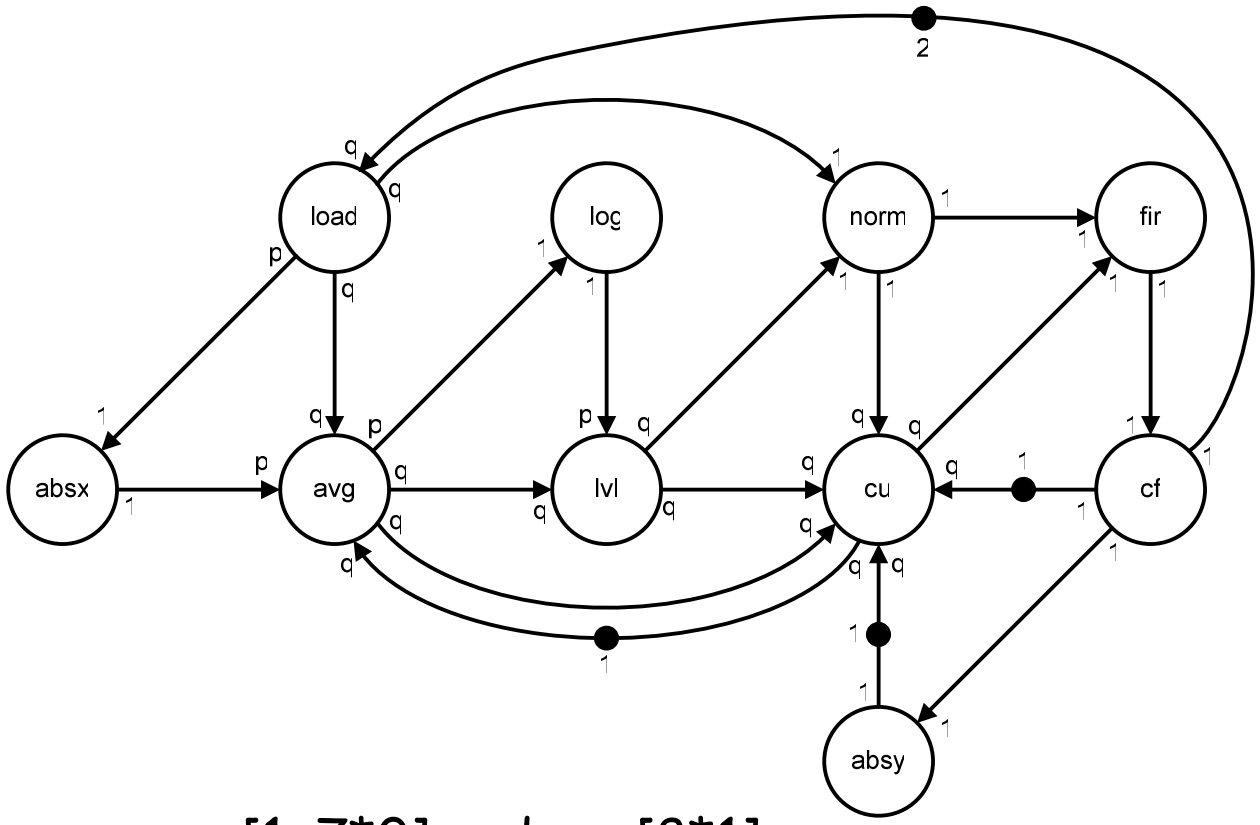
State	SubScenario
X	S_1
Y	S_2
Z	S_2

Captures sequences
- S_2, S_2, S_1 with probability 1/3
- S_2, S_1 with probability 2/3

- MPEG-4 SP Decoder
  - Scenario occurrences captured with fully connected 9-state Markov chain

SADF includes CSDF

- Channel Equalizer



load	[8*224]
absx	114
avg	[704, 7*416]
log	114
lvl	[440, 7*24]
norm	328
cu	[4944, 7648, 6*4966]
fir	114
cf	328
absy	114

- p = [1, 7*0] and q = [8*1]
- Weakly consistent
- Throughput = 0.000162443

Auto-concurrency excluded implicitly for CSDF

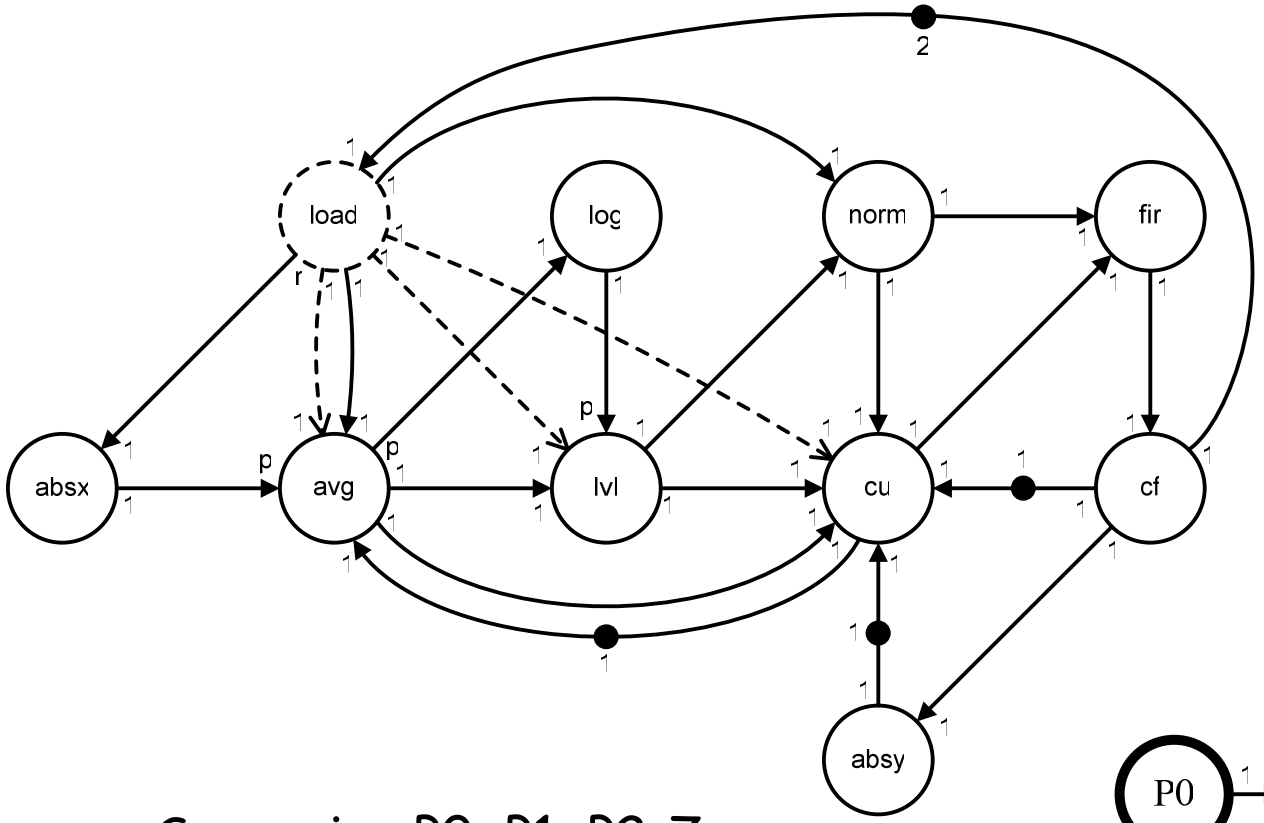
SADF includes CSDF

	P0	P1-7
p	1	0

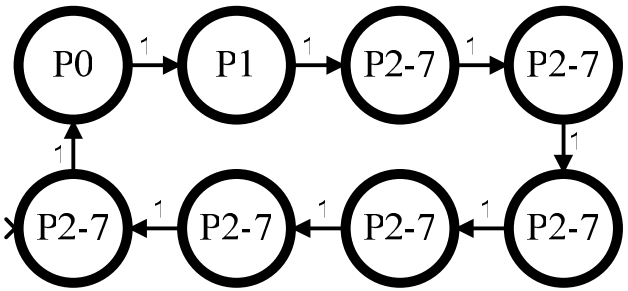
	P0	P1	P2-7
r	1	0	0

load	All	224
absx	Default	114
avg	P0	704
	P1-7	416
log	Default	114
lvl	P0	440
	P1-7	24
norm	Default	328
cu	P0	4944
	P1	7648
	P2-7	4966
fir	Default	114
cf	Default	328
absy	Default	114

- Channel Equalizer

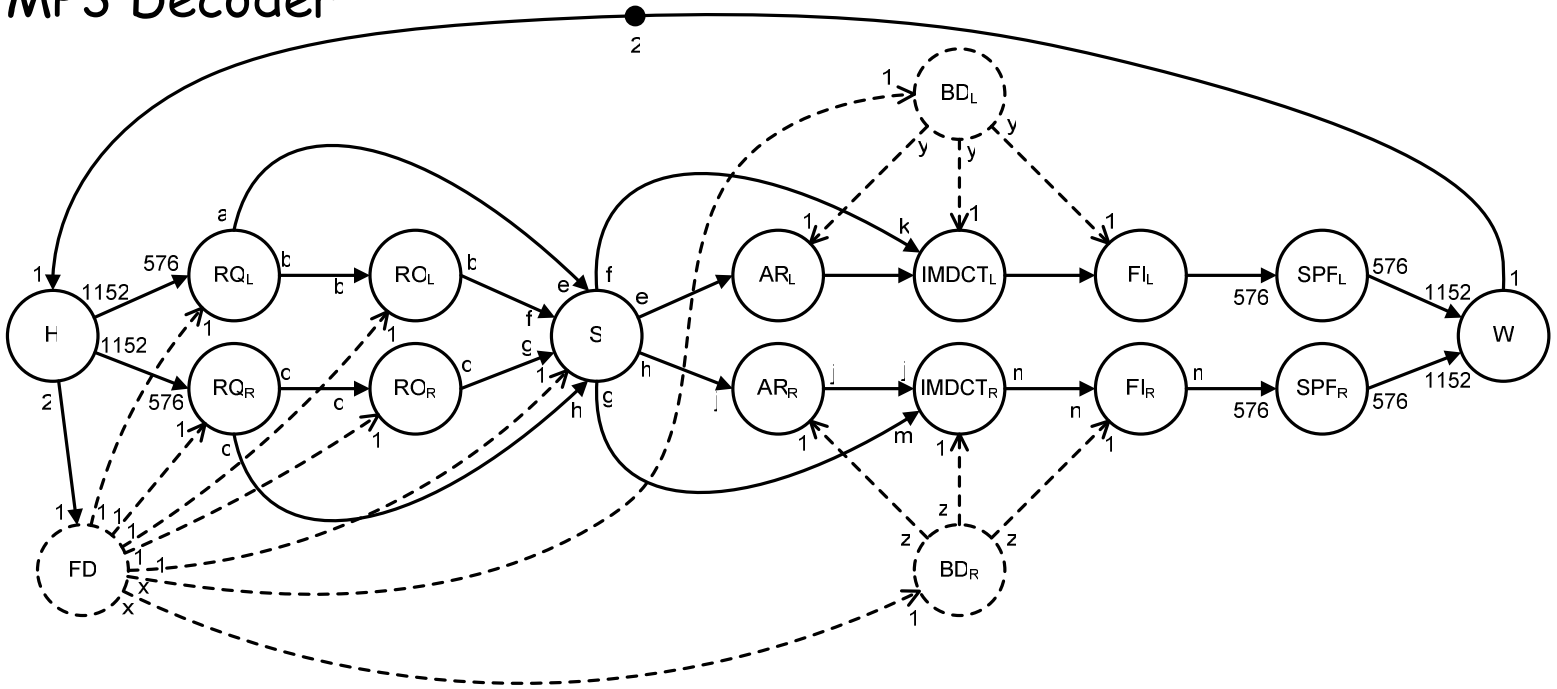


- Scenarios P0, P1, P2-7
  - Control tokens to avg and lvl are valued P1-7 in scenario P1 and P2-7
- Throughput = 0.000162443



SADF > Similar SDF Graphs

- MP3 Decoder



- 3 Frame types for each granule of both channels
  - Short frame = 96 short blocks (6 frequency components)
  - Long frame = 32 long blocks (18 frequency components)
  - Mixed frame = 2 long blocks succeeded by 90 short blocks
- 5 Scenarios LL, SS, LS, SL, M

SADF > Similar SDF Graphs

- MP3 Decoder

	LL	SS	LS	SL	M
e	0	576	0	576	540
f	576	0	576	0	36
g	0	576	576	0	36
h	576	0	0	576	540
x	1	1	1	1	2

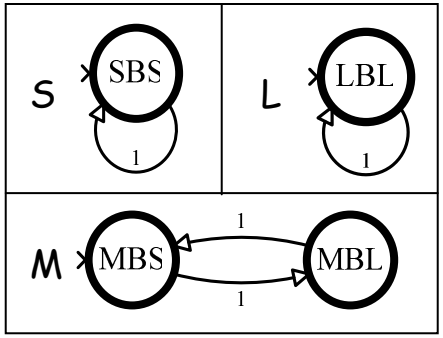
	L	S	M
a, c	576	0	36
b, d	0	576	540

	BL	BS
i, j	18	0
k, m	0	6
l, n	18	6

	LBL	SBS	MBL	MBS
y, z	32	96	2	90

- FD contains fully connected 5-state Markov chain
- Control tokens to RQ, RO are values L, S, M
- Control tokens to AR, IMDCT, FI are valued BL, BS
- Throughput = 2.68096e-07

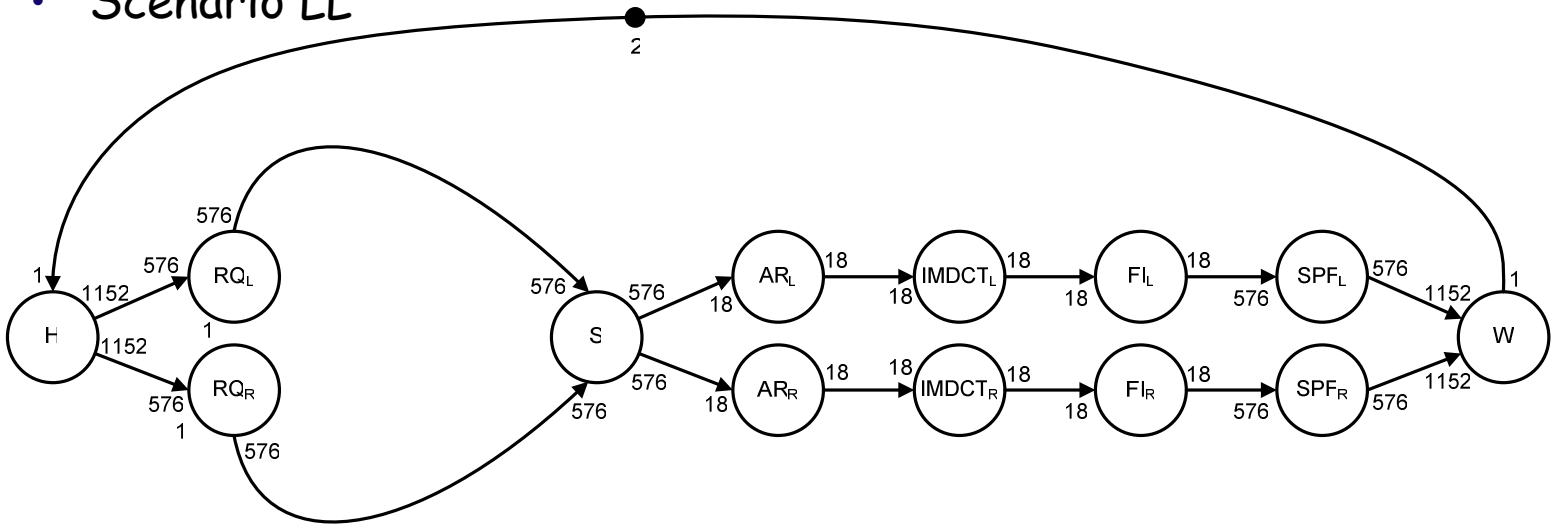
Detectors BD



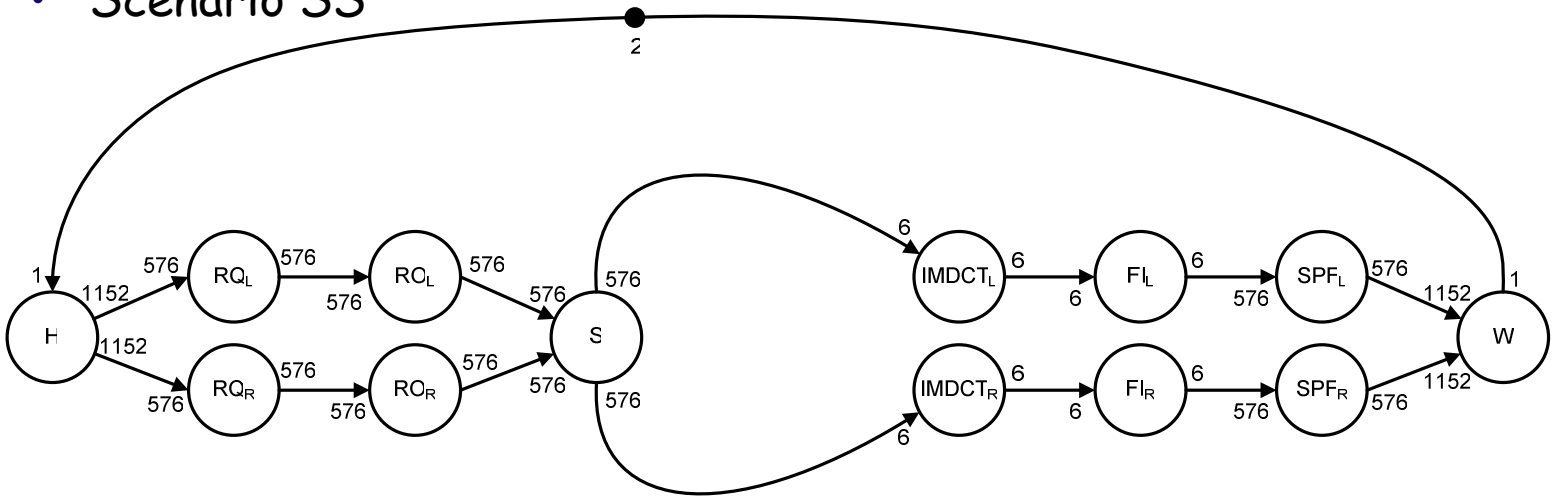
H	default	151977
RQ	L	56431
	S	72695
	M	42911
RO	L	0
	S	34684
	M	17568
S	LL	39763
	SS	38832
	LS, LS	19
	M	53602
AR	BL	409
	BS	0
IMDCT	BL	7414
	BS	5561
FI	BL, BS	4912
SPF	default	1865001
W	default	42896
FD	All	0
BD	All	0

SADF > Similar SDF Graphs

- MP3 Decoder
  - Scenario LL



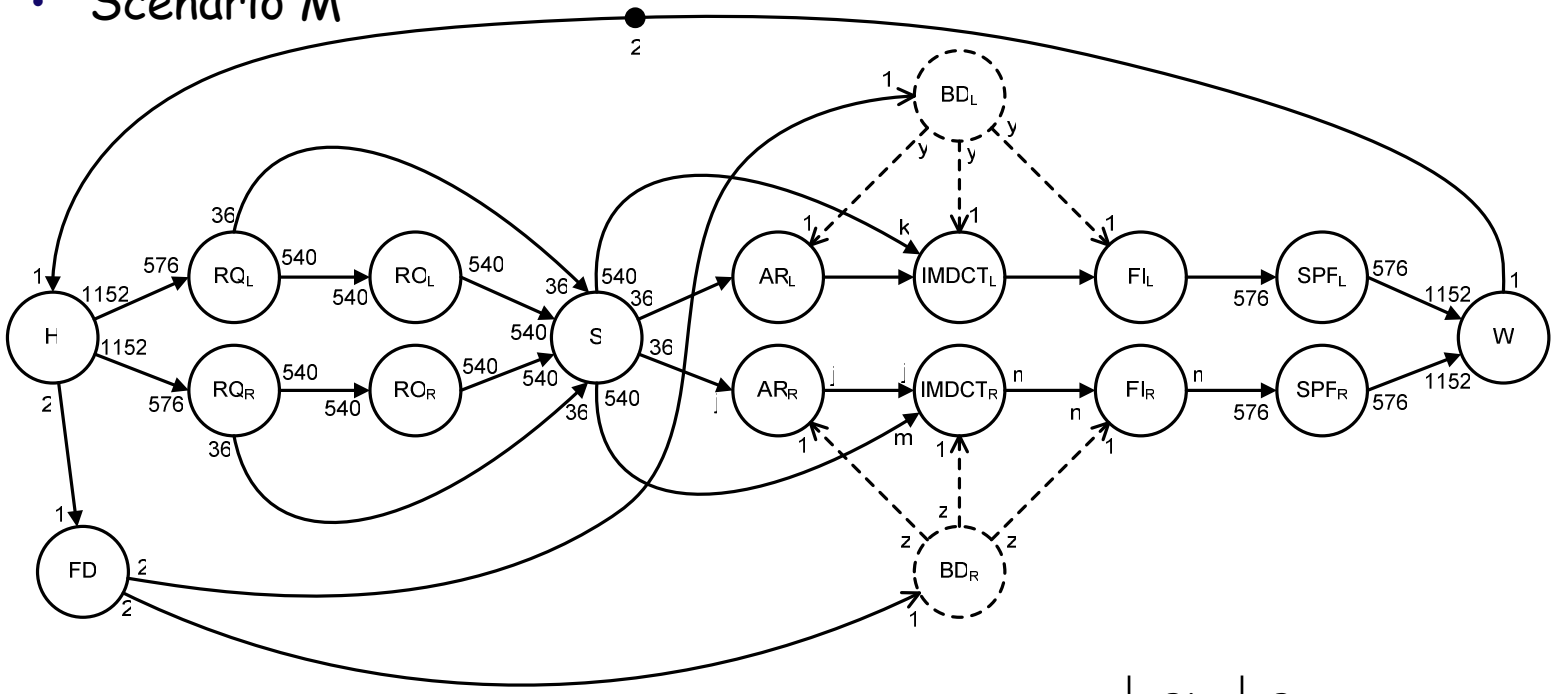
  - Scenario SS

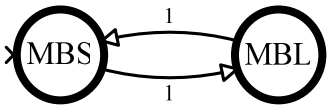


Auto-concurrency excluded implicitly for SDF

SADF > Similar SDF Graphs

- MP3 Decoder
  - Scenario M





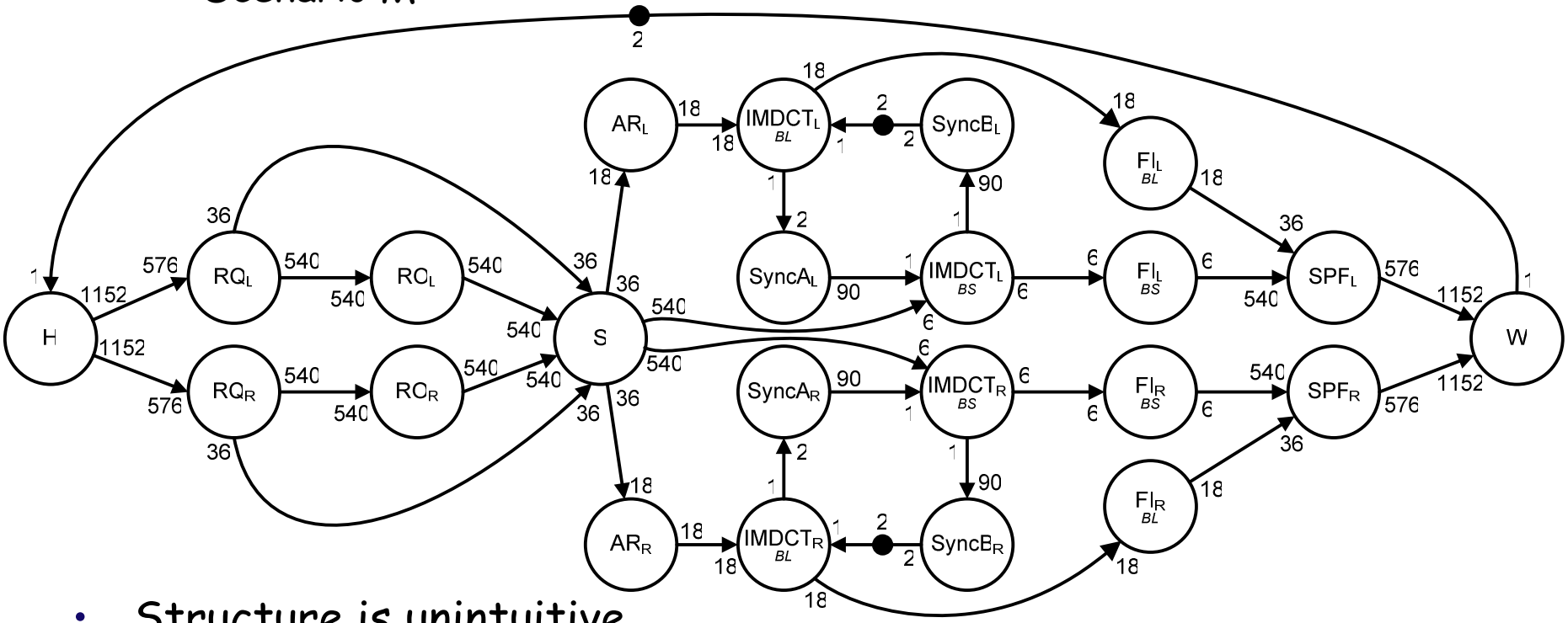
	MBL	MBS
y, z	2	90

	BL	BS
i, j	18	0
k, m	0	6
l, n	18	6

- Markov chain of BD for scenario M captures order of blocks

SADF > Similar SDF Graphs

- MP3 Decoder
  - Scenario M



- Structure is unintuitive
  - Duplicates & artefacts
- Structure SADF graph matches structure of code
  - Weakly consistent

Auto-concurrency excluded implicitly for SDF

Expressiveness

- SDF and CSDF are subsets of SADF
  - Even with execution time distributions
- SADF is suitable to model for example

  PSDF, SSDF, HDF, CDDF, BDF, ICDF, KPN, CTG, DDF, RPN

  by capturing dynamism with detectors
  - Scheduling settings of environment (SSDF)
  - Data-dependent functional implementation details (all except SSDF)
  - Continuous control as abstraction of reactive control (RPN)
- SADF is design-time analysable

Performance Analysis

- Model checking techniques in SDF^3
  - Take probabilities into account for state-space exploration when needed
  - Probabilities capture relative occurrence of scenarios and execution times
- Worst/best case metrics (probabilities irrelevant)
  - min/max response delay
  - max buffer occupancy
  - min/max inter-firing latency
- Expected/probabilistic reachability metrics (probabilities relevant)
  - expected response delay
  - response delay deadline miss probability
- Long-run metrics (probabilities relevant)
  - throughput
  - av/var inter-firing latency
  - periodic deadline miss probability
  - av/var buffer occupancy
- May suffer from state-space explosion
  - Simulation-based estimation offered as alternative

Conclusions & Future

- SADF extends SDF with scenarios
- SADF is fully analysable at design-time
- SDF^3 implements analysis techniques for SADF

- More efficient analysis techniques
  - Compositional per scenario approach instead of whole at once
  - Both for verification of correctness and performance analysis
- Extension with parametric execution times

www.es.ele.tue.nl/sadf