Statically Analyzable Programming Model for Dynamic Streaming

Peter Poplavko[1,2]

joint work with: Pascal Fradet[1], Alain Girault[1], Ali Erdem Ozcan[3]

[1] INRIA Rhône-Alpes, [2] CRI Grenoble, France

[3] STMicroelectronics Inc., Ottawa, Canada

29 June 2011, Map2MPSoC

plan

- motivation
- basic model of computation
- status
  - proposed model of computation
  - tools
- summary

motivation

- the advent many-core SoCs:

  tens / hundreds of processors

  high-performance applications e.g. HDTV

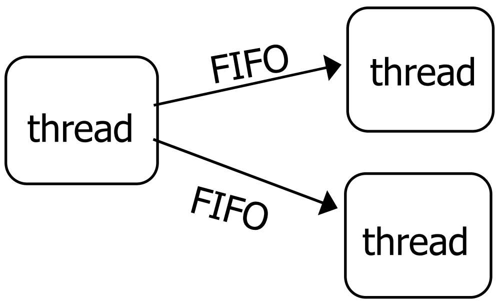
  => A and B

A: thread-level parallelism

- data parallelism (clone threads for different data)
- functional parallelism (different threads in a pipeline)

B: point-to-point FIFO connections

A + **B** = **dataflow**



requirements

dataflow model of computation (MoC) :

expressive

- data-dependent communication rates

  wide dynamic range

pure dataflow execution

- fully distributed
- where no central controllers intervene

compile-time verifiable for:

- absence of deadlock / liveness
- bounded memory requirements

state of the art

- static-rate and static-structure streaming

  video sample rate conversion, noise reduction, FFT, …

  - **Synchronous Dataflow – SDF** [E.A.Lee and D.G.Messerschmitt 1987]

- dynamic-rate and dynamic-structure streaming

  MP3/MPEG4 codecs, video surveillance, interactive TV …

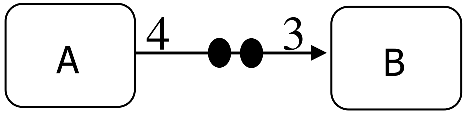
  - which model of computation (MoC) ???

KPN	- Kahn Process Networks	[G.Kahn 1974]
	------ extensions of SDF: -------------	
HDF	- heterochronous	[A.Girault *et al* 1999]
PSDF	- parametric	[B.Bhattacharyya and S.S.Bhattacharyya 2001]
SADF	- scenario-aware	[B.D.Theelen *et al* 2006]
VRDF	- variable-rate	[M.H.Wiggers *et al* 2008]

… no complete answer to our goals

starting point: SDF

SDF graph (initial state):



2 initial tokens in FIFO

one balance equation per edge:

$\#_A \cdot 4 = \#_B \cdot 3$

minimal solution = SDF iteration:

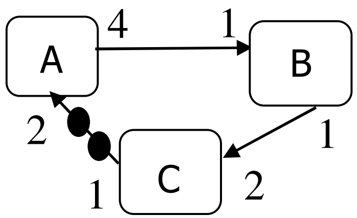
$\#_A = 3; \quad \#_B = 4;$

after iteration → initial state

schedule 1: A A A B B B B = A^3 B^4

schedule 2: A B B A B A B = A B^2 $(AB)^2$

a larger SDF:



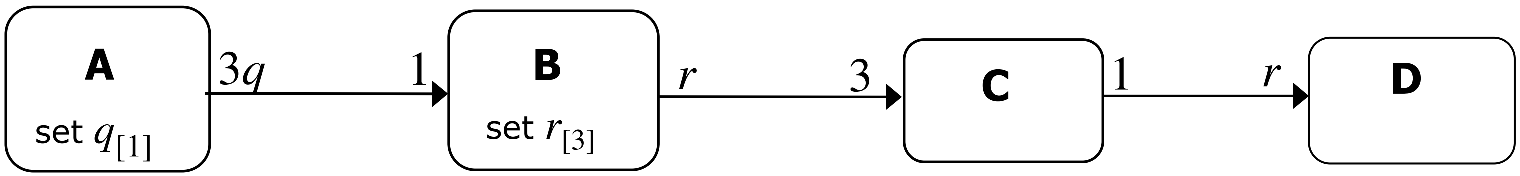
system of balance equations

a schedule: A B^4 C^2

proposed MoC: SPDF

Schedulable Parametric Dataflow (SPDF)

- rates: parametric or constant
- a parameter is set by an actor
  - parameter changes every "period" actor executions
  - notation: set $parameter_{[period]}$



a schedule: A $(B^3\ C^r\ D)^q$

quasi-static schedule

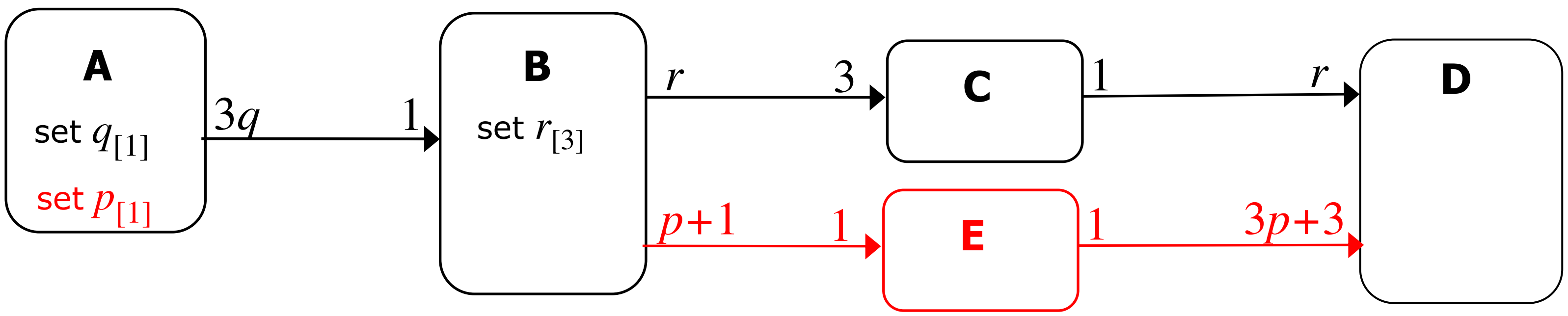
SPDF details

parameter expressions

- polynomials with positive integer coefficients
- Boolean expressions

no fixed rule on which actors may set a given parameter

the tools check inconsistencies in parameter communication



tools [1]

implement parameter communication

static analysis:

- rate consistency
  - solutions of balance equations must exist
- parameter change safety
  - rates may change only at certain points
- liveness of the cycles

compute a quasi-static schedule

compile time!

---

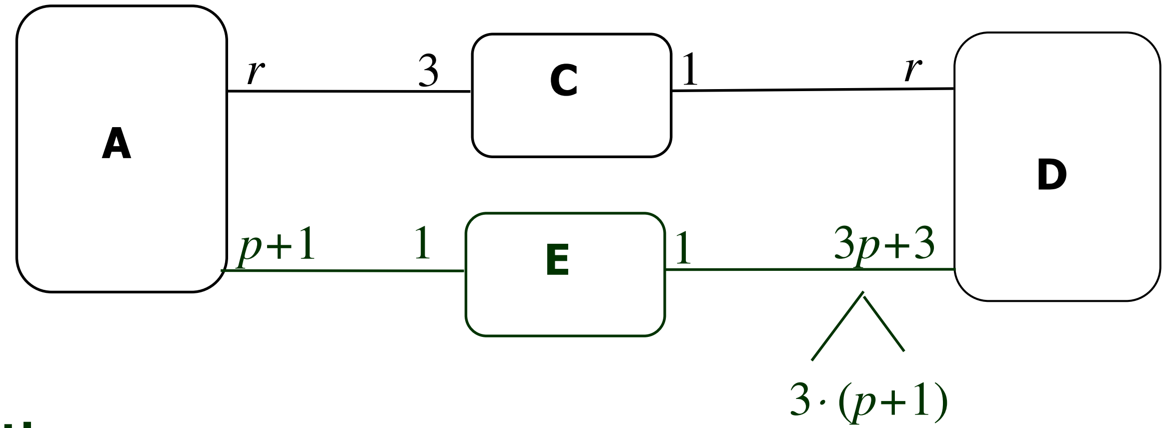
[1] **"Static Analysis of Parametric Dataflow Graphs", to appear soon as *INRIA Technical Report*.**

rate consistency

do balance equations have a solution for any parameter value?

turn all directed edges into undirected edges

consider every cycle:



theorem:

we have solutions

iff

the factors are balanced in every cycle

parameter change safety

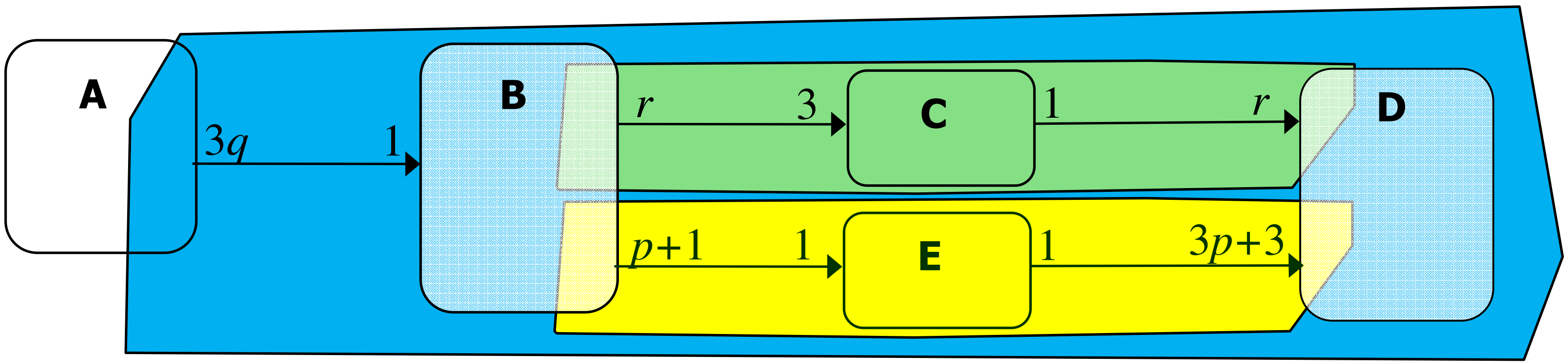
check the correctness of the **periods**

algorithm:

1. cover the graph by a hierarchy of subgraphs.
2. for every subgraph bottom-up:

   solve local balance equations

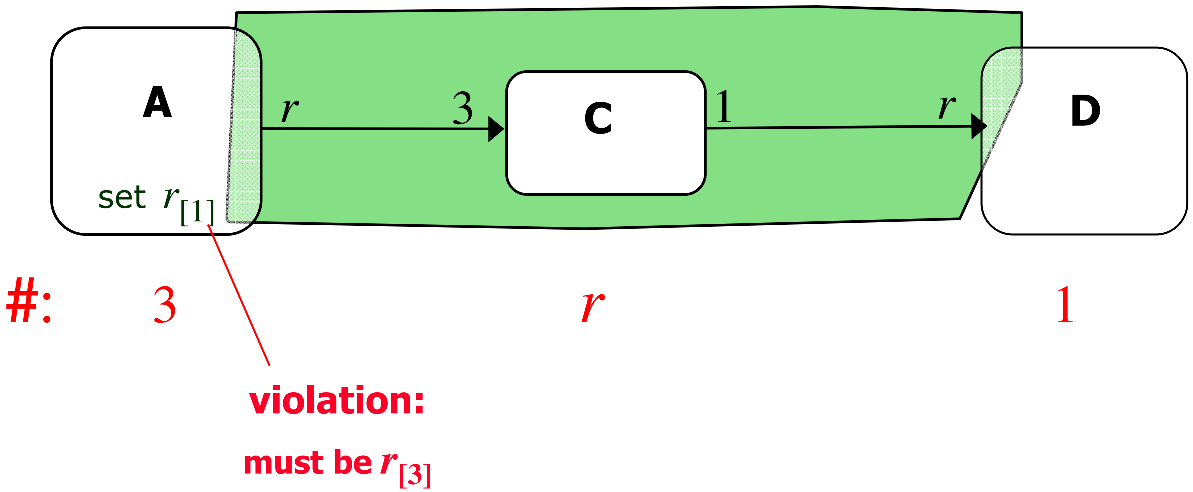
   #A, #B… - safe periods for setting parameters at A, B, …



“regions of influence” of parameters: e.g. region r, region p

parameter change safety (2)

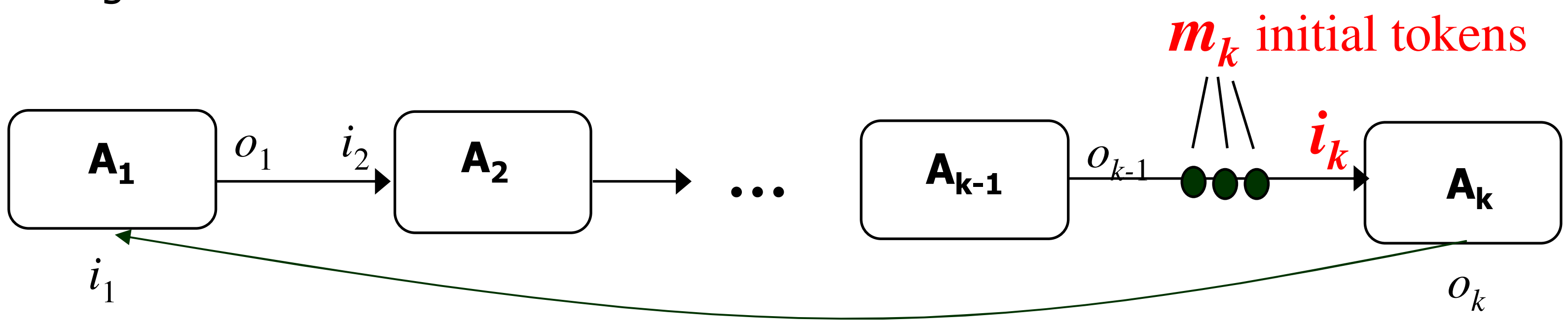
for example, region r



during an iteration of subgraph, rates should not change

liveness of cyclic paths

enough initial tokens



sufficient condition:

$\#^c$ – solution of local balance equations for the **cyclic path**

$$\exists\, k : \quad m_k \geq i_k \cdot \#^c(A_k)$$

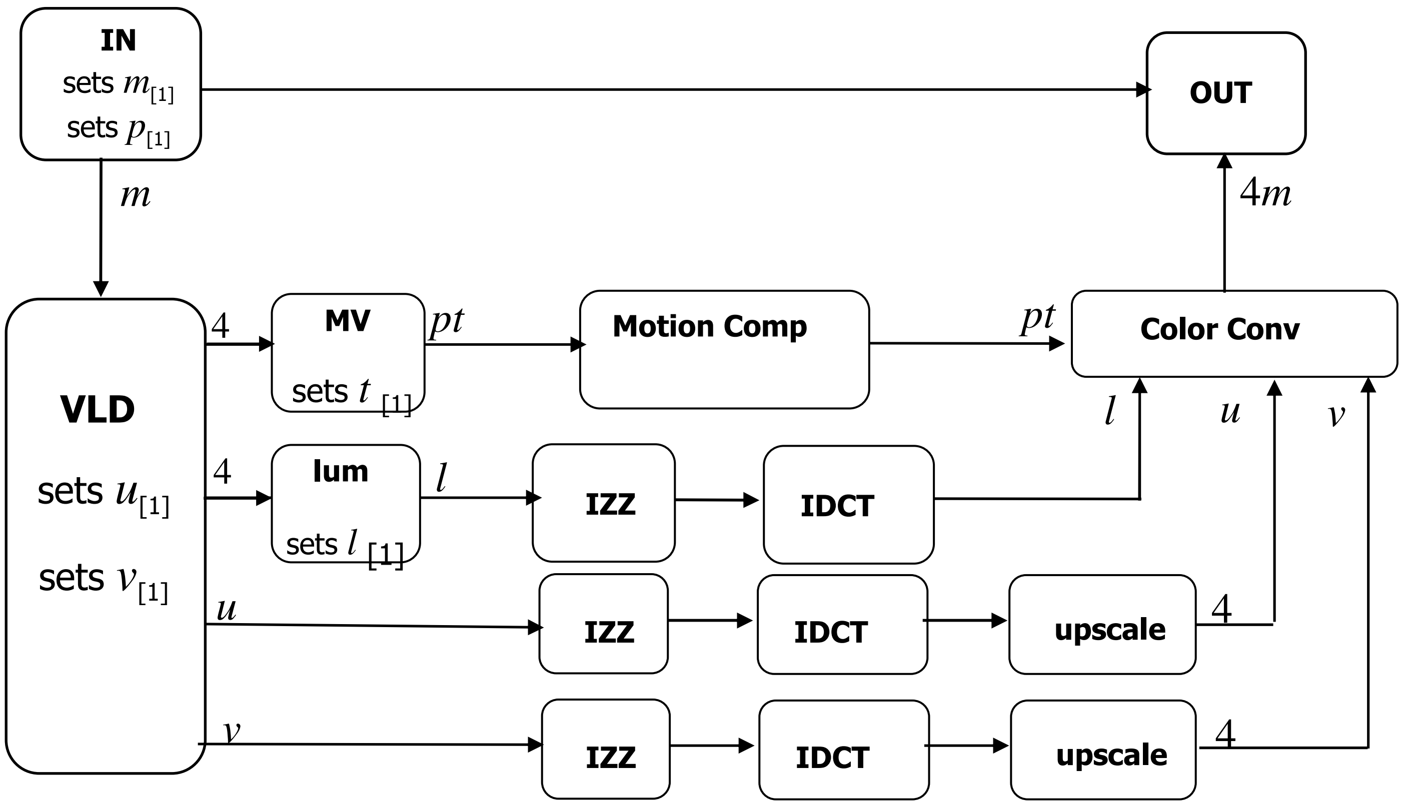
the above is easy to verify

m_k - a compile-time constant

i_k , $\#^c$ - expressed by Boolean functions and **positive-coefficient** polynomials

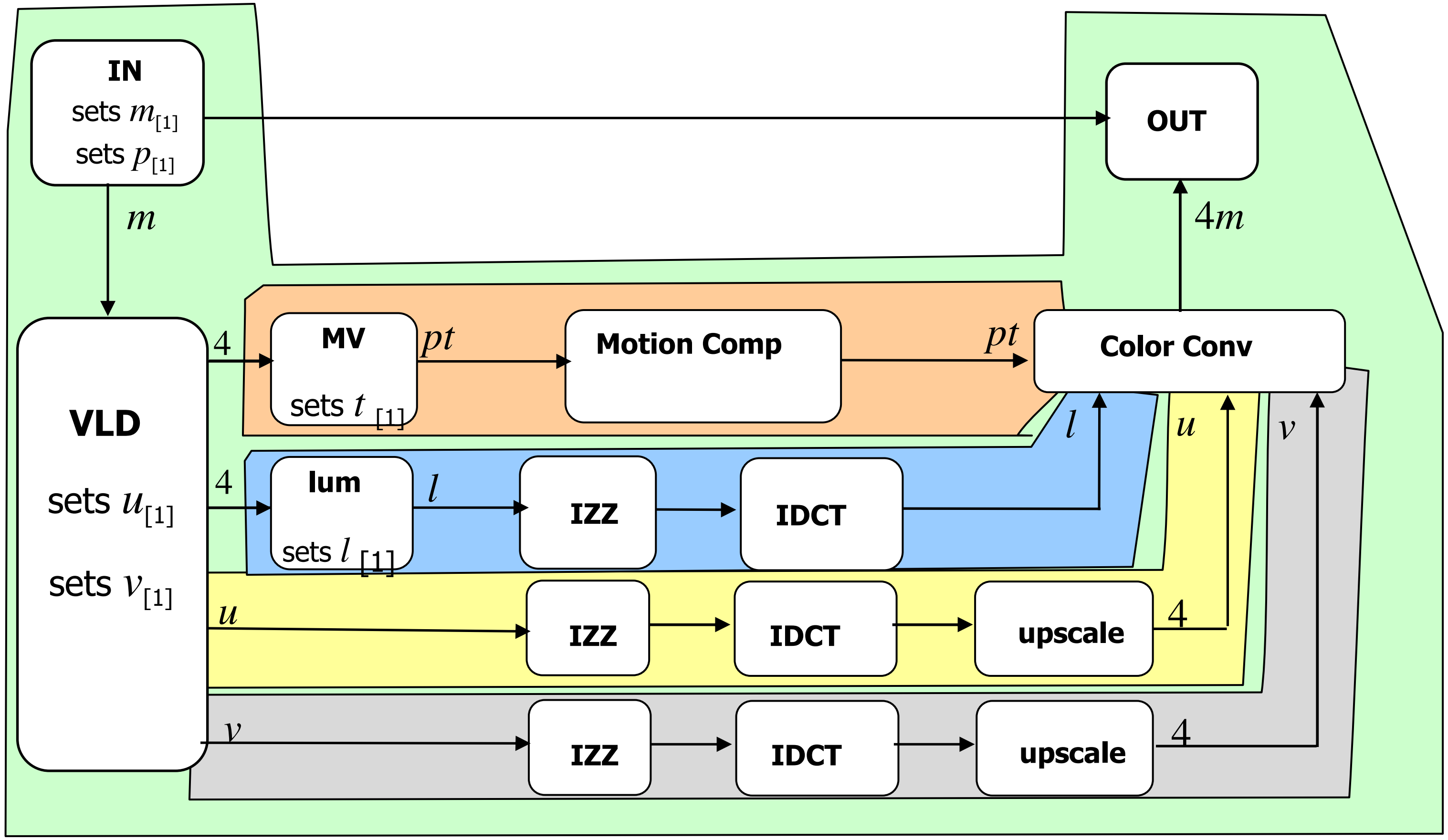
max values of parameters should be provided

video decoder example



(rates equal to 1 are omitted here)

the regions of parameters



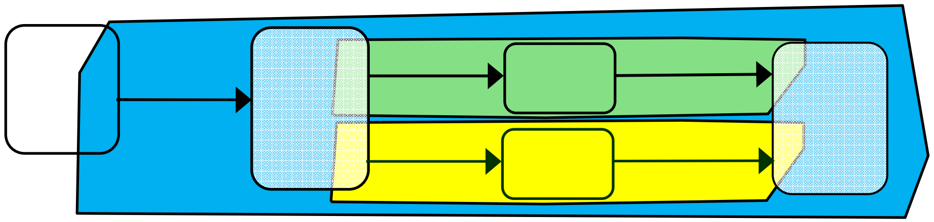
summary

- dynamic streaming model of computation
    - schedulable at compile-time
    - preserves many advantages of SDF

future work

- memory minimization
- reconfiguration
- MPSoC mapping and scheduling
- verify the performance analytically
    - worst-case, average-case analysis

questions?



[1] **“Static Analysis of Parametric Dataflow Graphs”, to appear soon as *INRIA Technical Report*.**