Berry-Constructive Programs are Sequentially Constructive

Or:

Synchronous Programming from a Scheduling Perspective

J. Aguado, M. Mendler, R. von Hanxleden, I. Fuhrmann

Aims of Work

- Proof of Claim *[von Hanxleden, Mendler, Aguado, et. al. DATE 2013]* that Sequential Constructiveness is a conservative extension of Berry-Constructiveness.
- New domain-theoretic presentation of Berry‘s constructive „must-cannot“ analysis for pure Esterel:
  – Reconstruction of imperative synchronous programming for multi-threaded shared-variable programs (from a scheduling perspective)
  – New sequential-concurrent reaction model
  – Captures explicit signal initialisation

Structure of Talk

1. Introduction
2. Sequential Constructiveness (Scheduling)
3. Berry Constructiveness (Domain)
4. Examples
5. Conclusion

1 INTRODUCTION

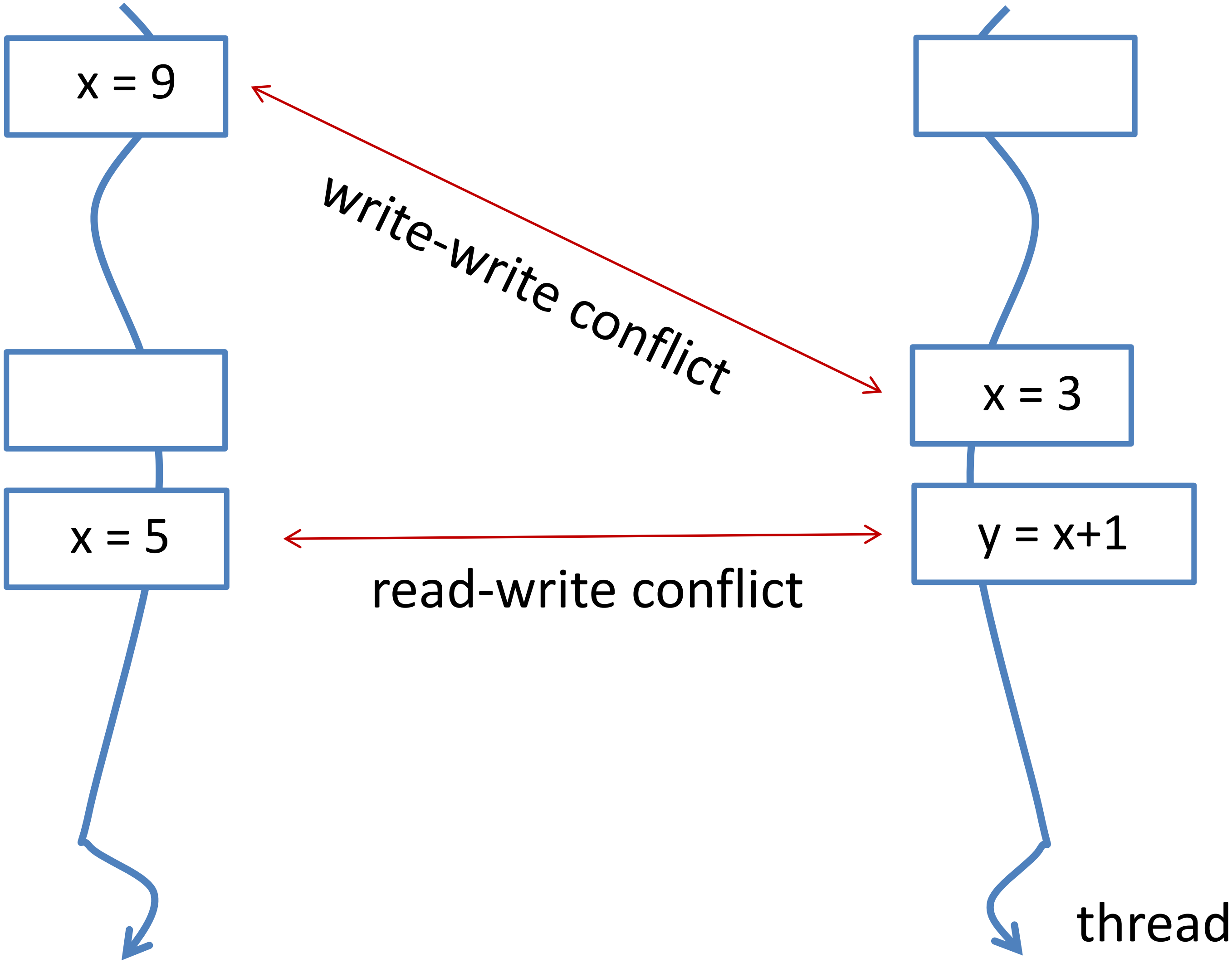
Widening the Scope of a Logical Clock

- Lustre/Signal oversampling clock abstraction
- Quartz clock refinement [Gemünde,Brandt,Schneider 2009]
- Reactive ML clock domains [Mandel,Pasteur,Pouzet 2013]
- Bursty integer clocks [Guatto, et.al. 2012]
- …

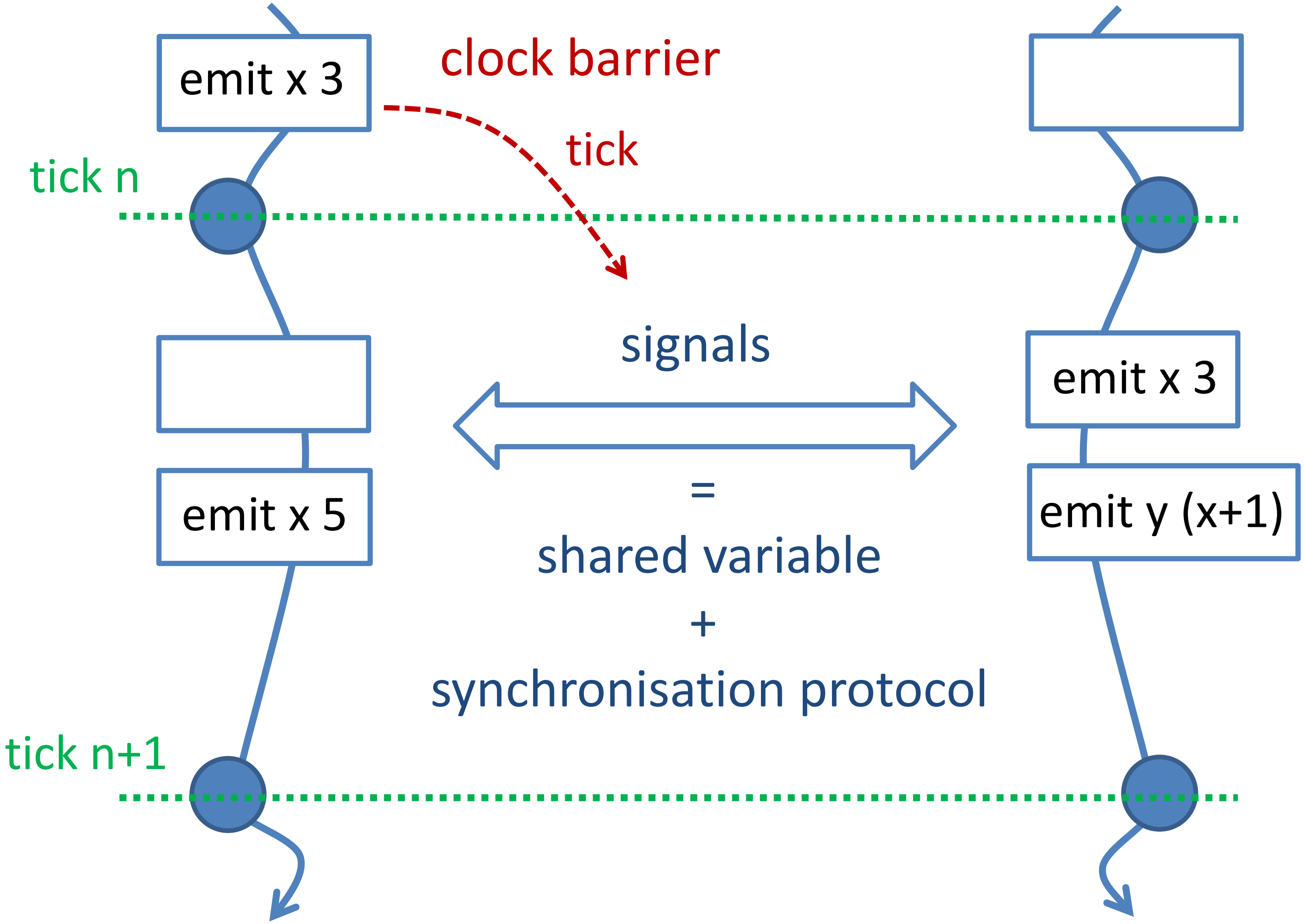
⇒ logical clock
on-a logical clock
on-a logical clock
on-a .... sequential composition

nesting can lead to over-synchronisation...

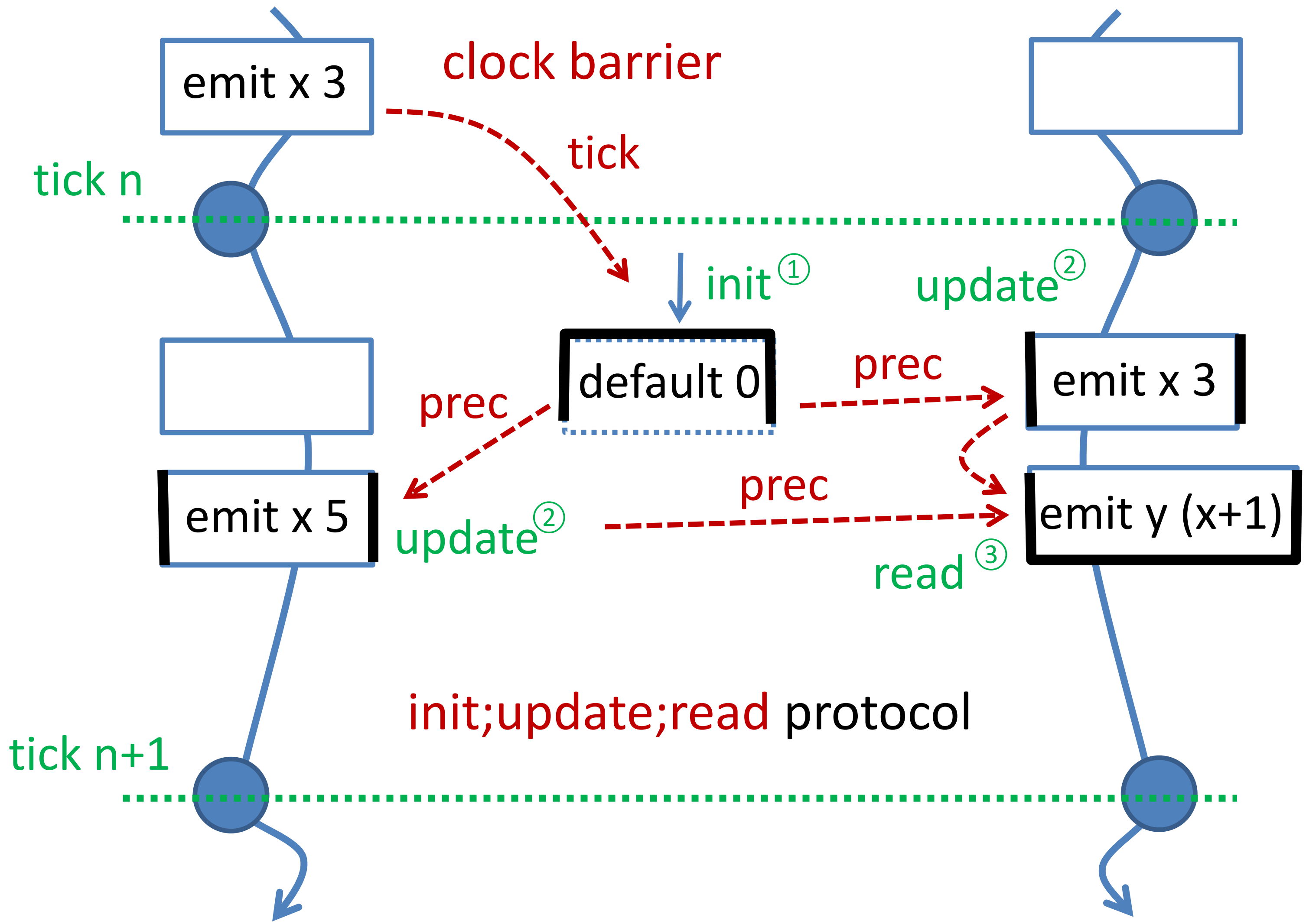
Shared Memory Multithreading



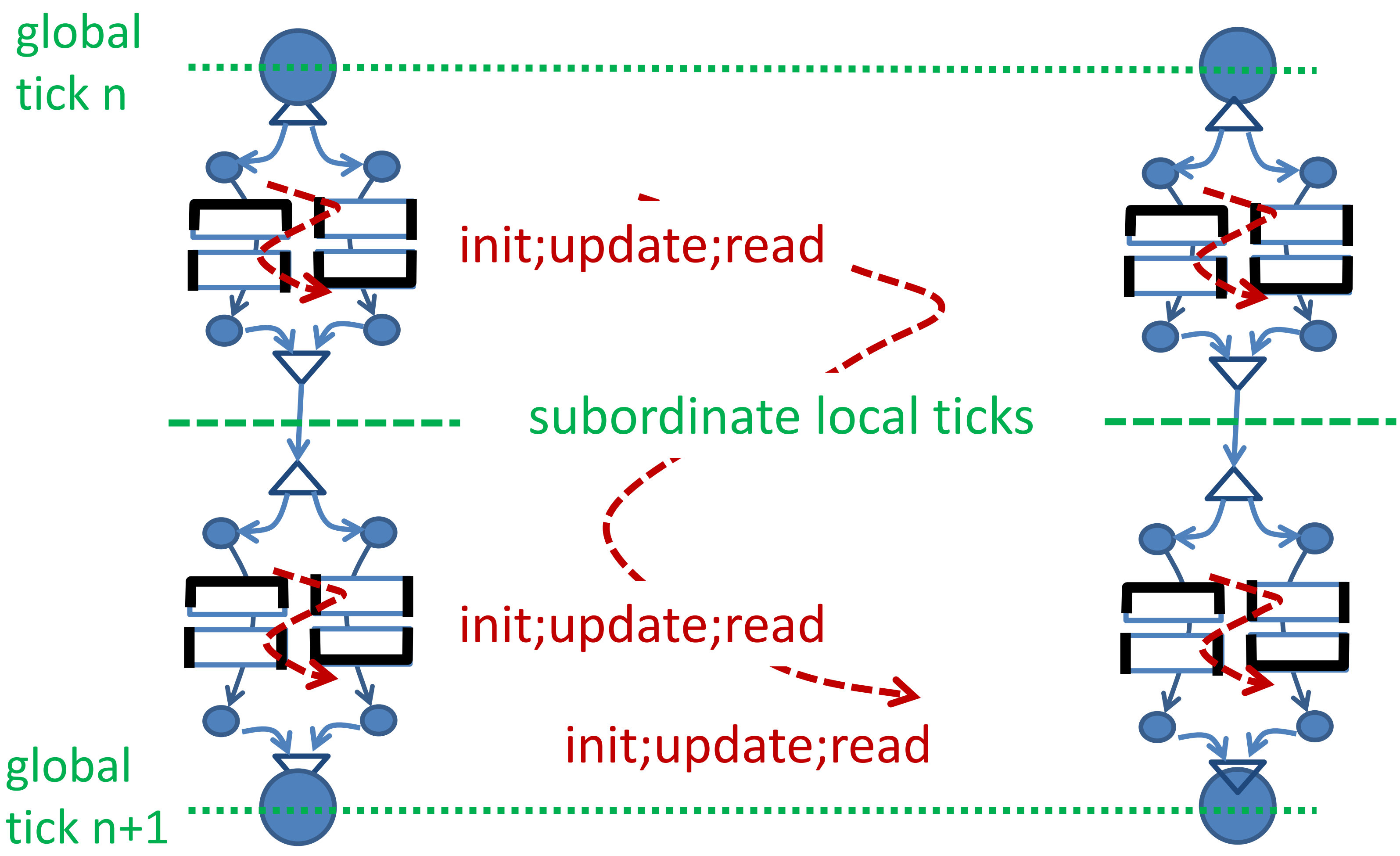
Clocked Shared **Signal** Multithreading



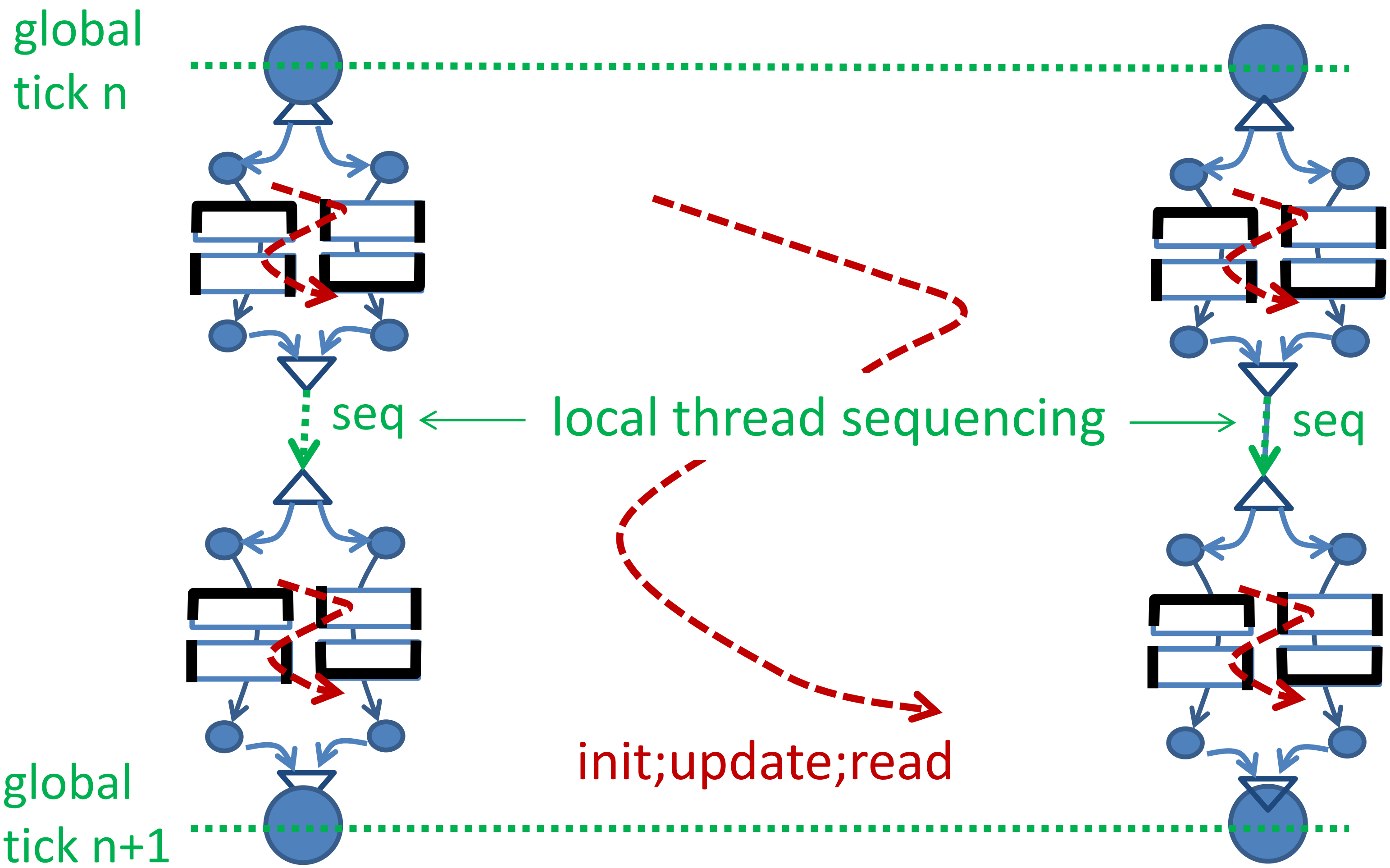
Clocked Shared Signal Multithreading



Clock Refinement



Prescriptive Sequentiality Replaces Local Clock



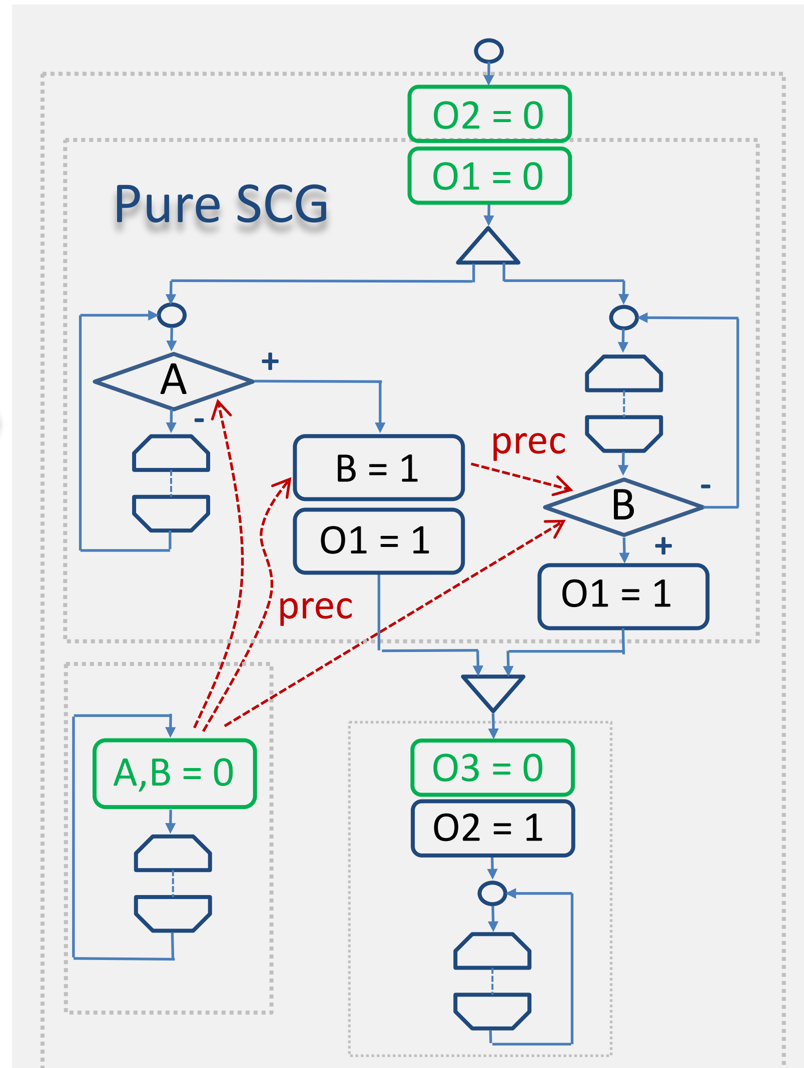
2 SEQUENTIAL CONSTRUCTIVENESS

Pure Signals = Synchronised Boolean Variables

```
module ABOs—Esterel
inputoutput A,B;
signal O2 in
 signal O1 in
  [  % Thread A
     await immediate A;
     emit B;
     emit O1;
  ||
     % Thread B
     await B;
     emit O1;
  ];
 end;
 signal O3 in
  emit O2;
 end;
 halt;
end
```

presence ↦ 1
absence ↦ 0

init ↦ x = 0
update ↦ x = 1



Synchronous Instants

A synchronous instant is a maximal micro-sequence

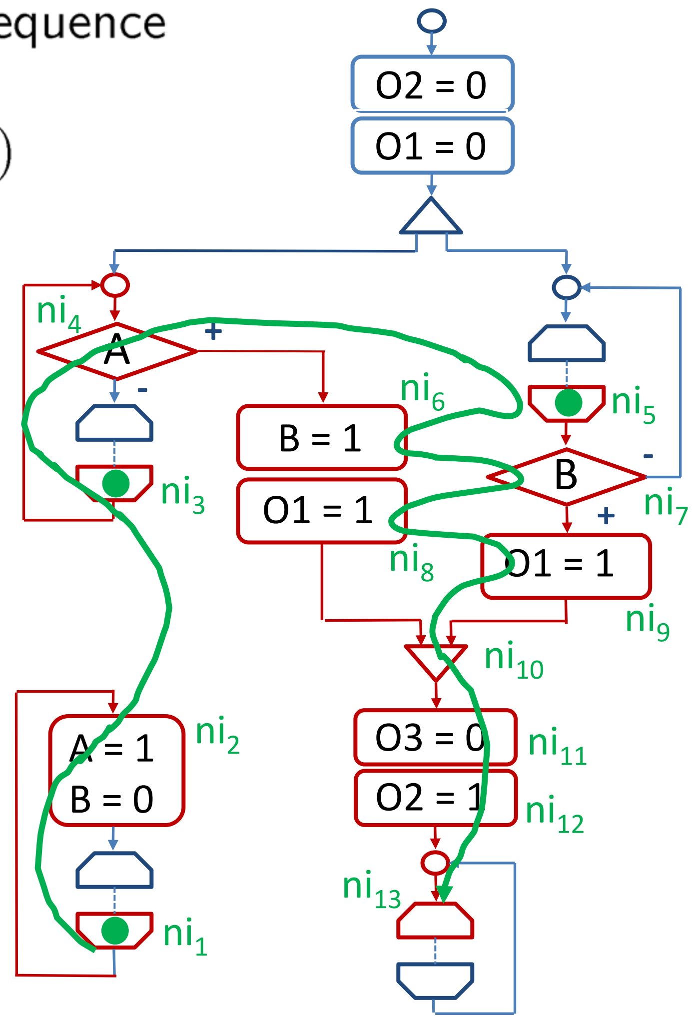
$$R : (\Sigma_0, \rho_0) \overset{ni_1}{\rightarrow}_{\mu s} (\Sigma_1, \rho_1) \cdots \overset{ni_k}{\rightarrow}_{\mu s} (\Sigma_k, \rho_k)$$

of node instances $ni_j = R(j)$.

The synchronisation protocol depends on ...

- concurrency $ni_6 \mid_R ni_9$
- precedence $ni_6 \rightarrow_{prec} ni_7$
- confluence $ni_8 \sim_R ni_9$

of node instances.



Concurrency, Precedence, Confluence (Informal)

- $ni_1 \mid_R ni_2$: ni_1 and ni_2 are concurrent in R if
  - ni_1 and ni_2 have been instantiated in R as siblings of the same common (least ancestor) fork node.
- $ni_1 \rightarrow_{prec} ni_2$: ni_1 precedes ni_2 in the init;update;read protocol if
  - $ni_1 \mid_R ni_2$,
  - ni_1 performs an init/update on a variable that is read by ni_2, or
  - ni_1 performs an init and ni_2 performs an update.
- $ni_1 \sim_R ni_2$: ni_1 and ni_2 are confluent in R if
  - ni_1 and ni_2 can be executed in any order, with the same resulting configuration.

Sequential Constructiveness SC

- A micro-sequence R is sequentially admissible iff for all node instances the following scheduling condition is satisfied:

  If $ni_{j_1} \rightarrow_{prec} ni_{j_2}$, then $j_1 \leq j_2$ or $ni_{j_1} \sim_R ni_{j_2}$.

- A (closed) process P is sequentially constructive (SC) iff for all admissibly reachable configurations (Σ_0, ρ_0) of P:
  - there exists a sequentially admissible synchronous instant $R : (\Sigma_0, \rho_0) \stackrel{ni_1}{\rightarrow}_{\mu s} (\Sigma_1, \rho_1) \cdots \stackrel{ni_k}{\rightarrow}_{\mu s} (\Sigma_k, \rho_k)$ and
  - every sequentially admissible synchronous instant R leads to the same quiescent response ρ_k.

Conservative Approximations of SC

Static Approximation

- A program P is ASC-schedulable iff no control-flow cycle in the SCG of P contains a $\rightarrow_{prec}$ dependency edge.

Theorem 1 (DATE 2013). *Every ASC-schedulable program is SC.*

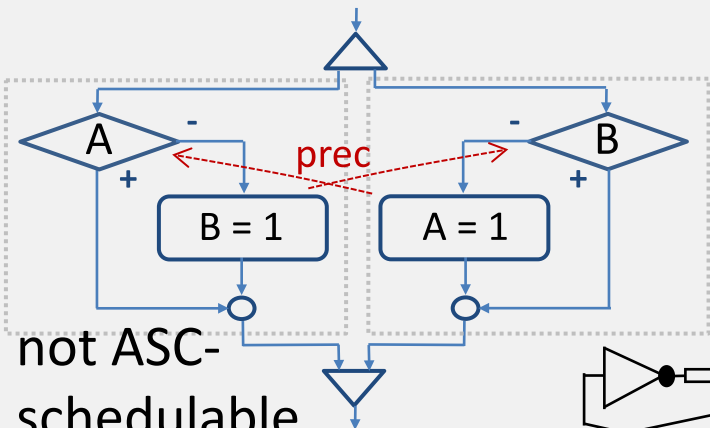
Data-dependent Approximation

Theorem 2 (New). *Every Berry-constructive program is SC.*

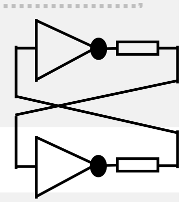
What means Berry-constructive in the shared-variable setting with explicit initialisations ? ...

Examples

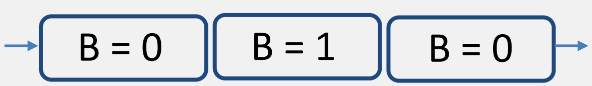
not sequentially constructive



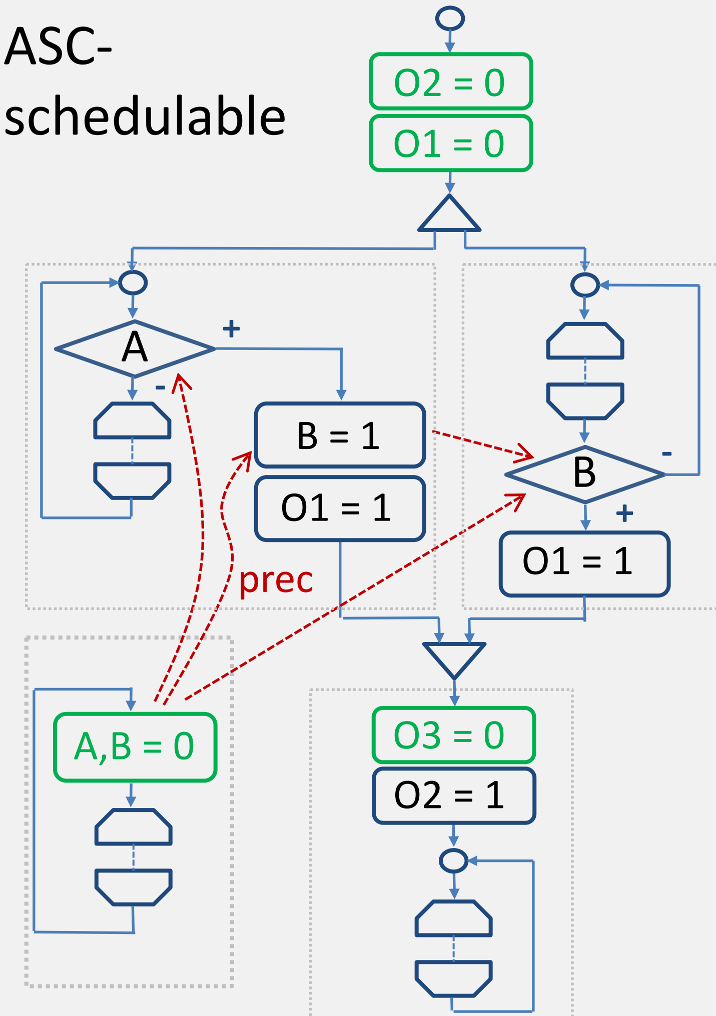
not ASC-
schedulable



Every non-concurrent program
is sequentially constructive



ASC-
schedulable



3 BERRY CONSTRUCTIVENESS

BC Semantics – Variable Status

- Variables $\mathcal{V}$ participate in the synchronisation protocol by changing status in a linear 4–valued domain

$$
\begin{aligned}
(\mathbb{D}, \leq) &= \{\bot \leq 0 \leq 1 \leq \top\} \\
&= \{\text{pristine} \leq \text{initialised} \leq \text{updated} \leq \text{crashed}\}
\end{aligned}
$$

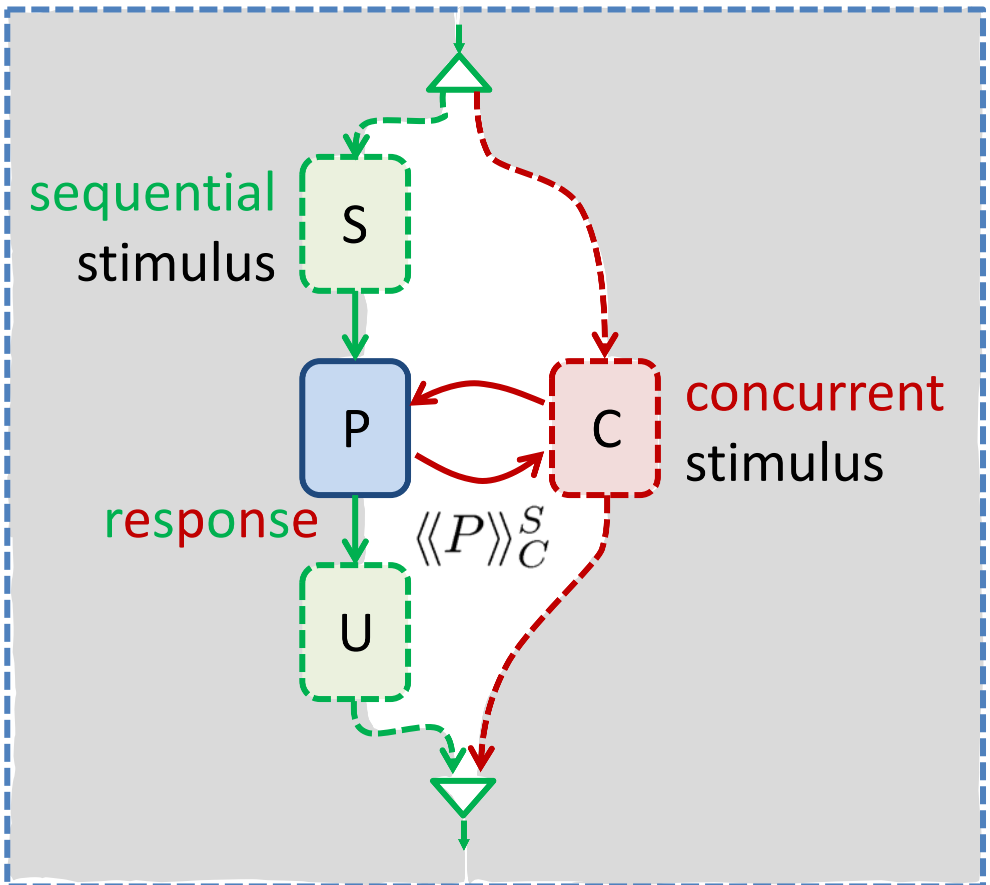
A variable $s \in \mathcal{V}$ with status $\gamma \in \mathbb{D}$ is denoted by s^{γ}.

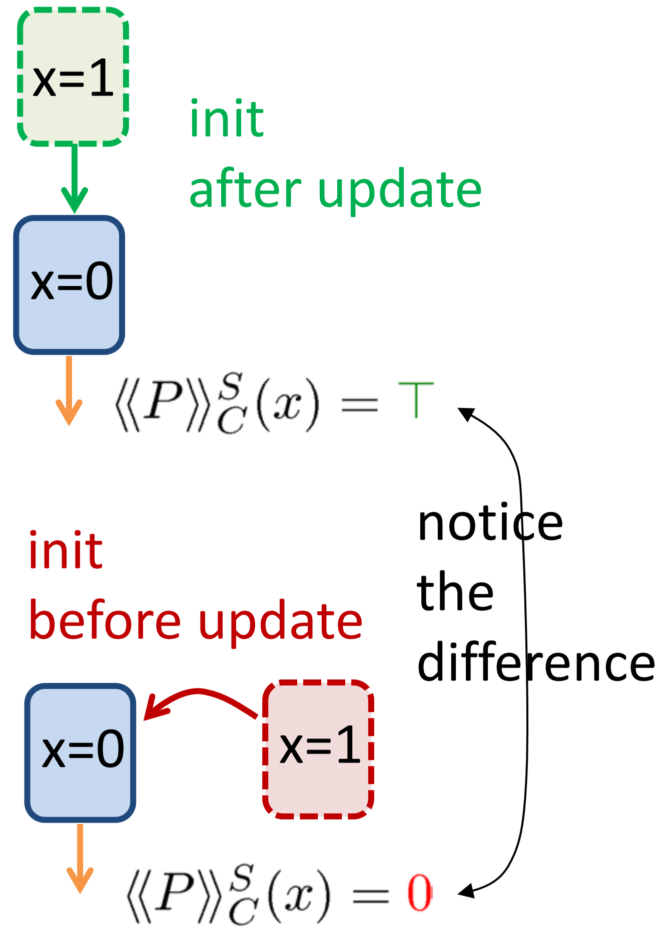
In the fixed point analysis these values are approximated using

- intervals ...
- in a 2-dimensional sequential-concurrent reaction model ...

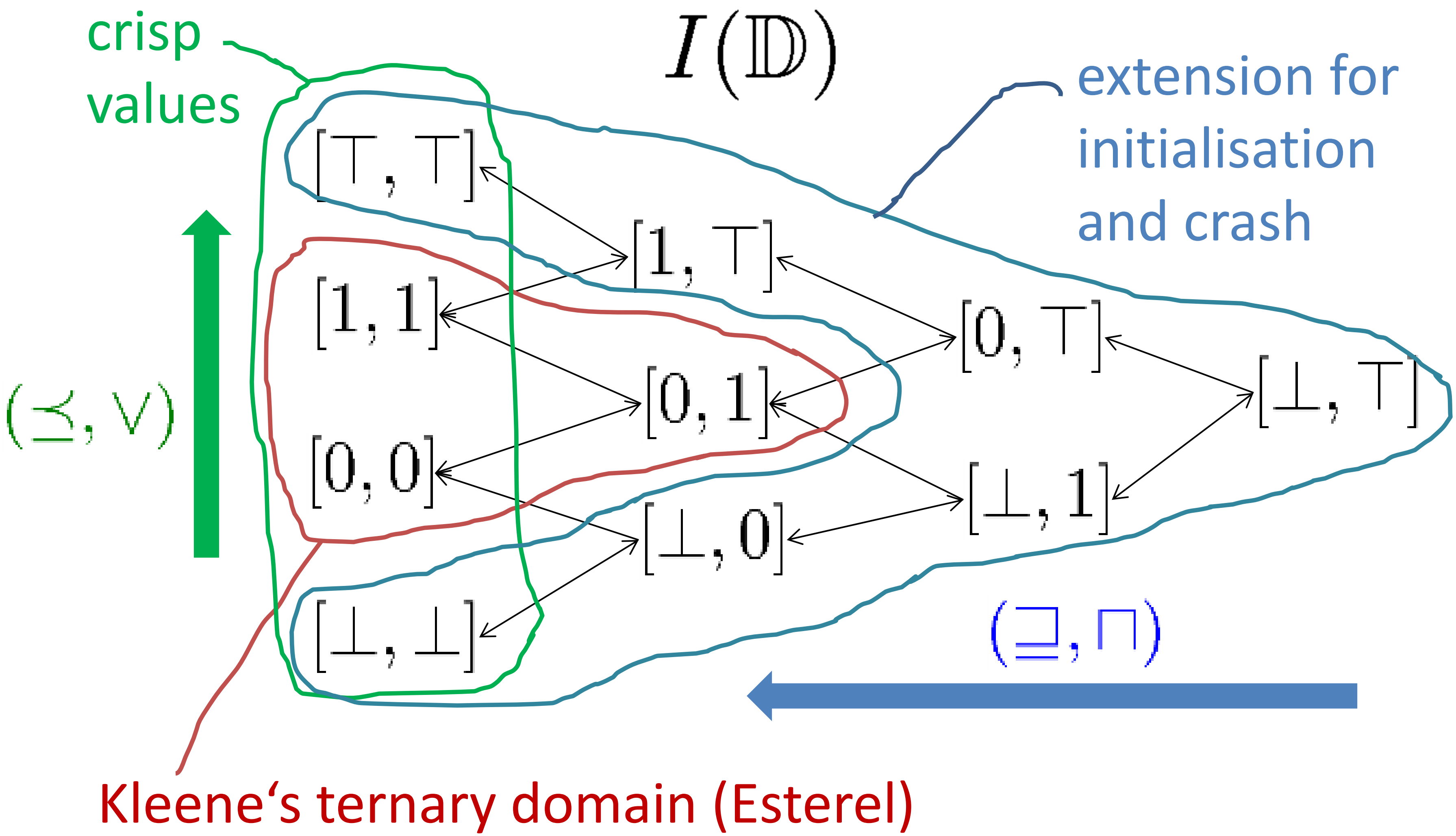
Sequential-Concurrent (SC) Reaction Model

Program P in Context

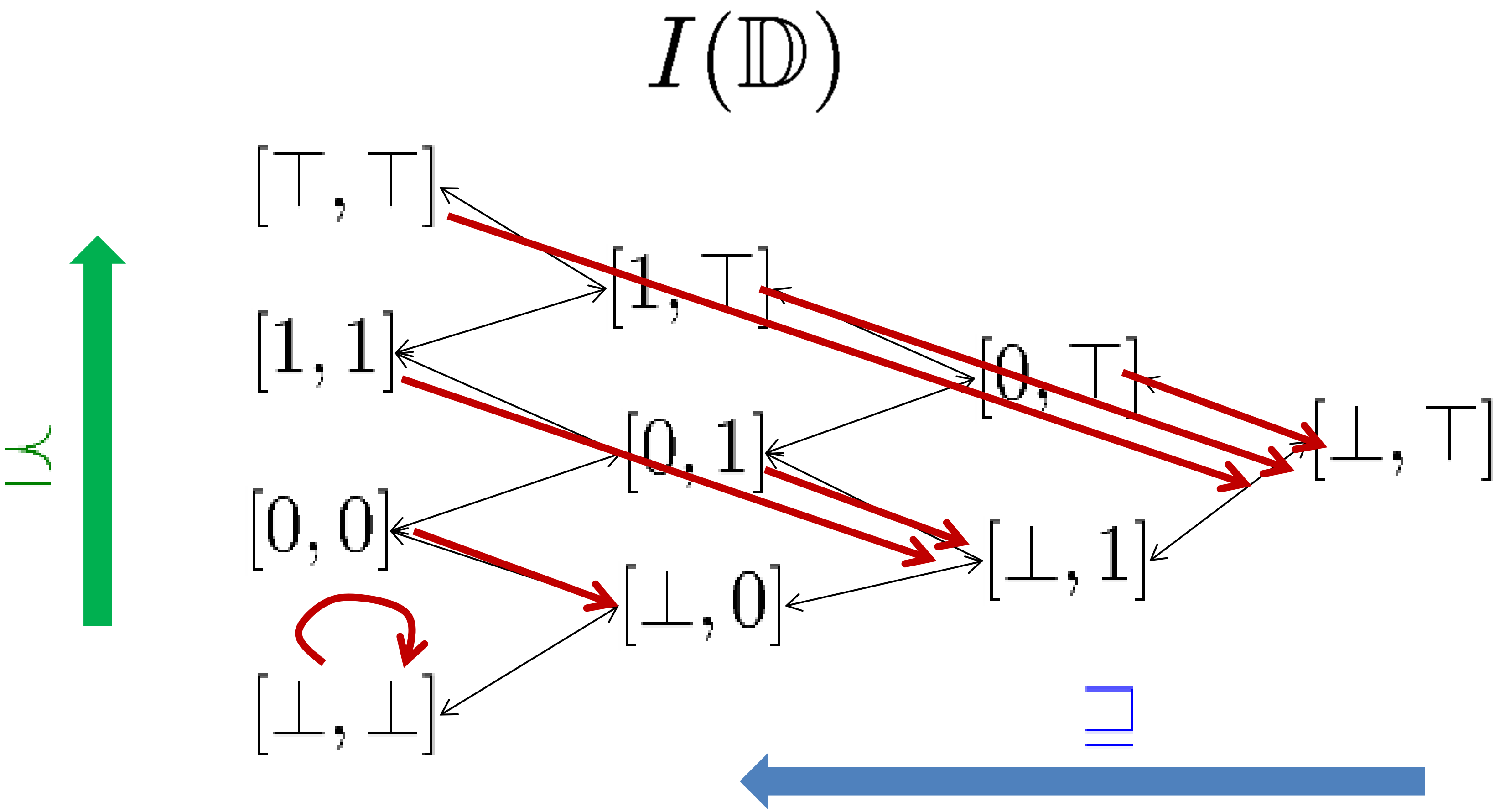




Interval Domain of Variable Statuses

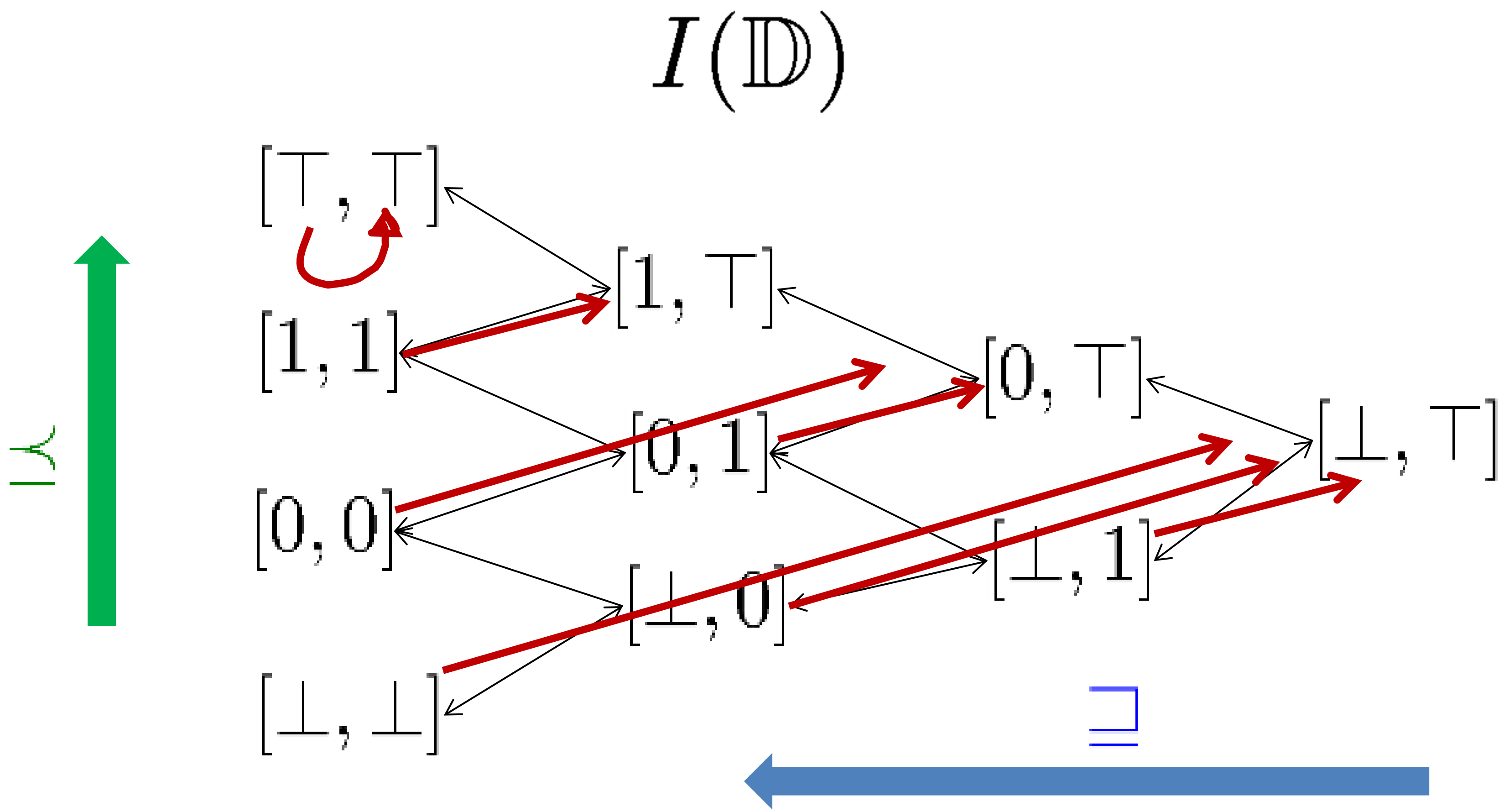


Interval Domain of Variable Statuses



Upper Projection *low* extracts "cannot" information

Interval Domain of Variable Statuses



Lower Projection *low* extracts "must" information

Berry Constructiveness

Theorem 3 (Berry Semantics). *For reset-free program P and ternary environment C,*

- $s \in must(P, C)$ *iff* $s^1 \in \langle\!\langle P \rangle\!\rangle^0_C$ *and*
- $s \in cannot(P, C)$ *iff* $s^0 \in \langle\!\langle P \rangle\!\rangle^0_C$.

Thus, Berry's must-cannot fixed point response of reset-free P coincides with $\mu C. \langle\!\langle P \rangle\!\rangle^0_C$.

Definition 2. *A program P is*

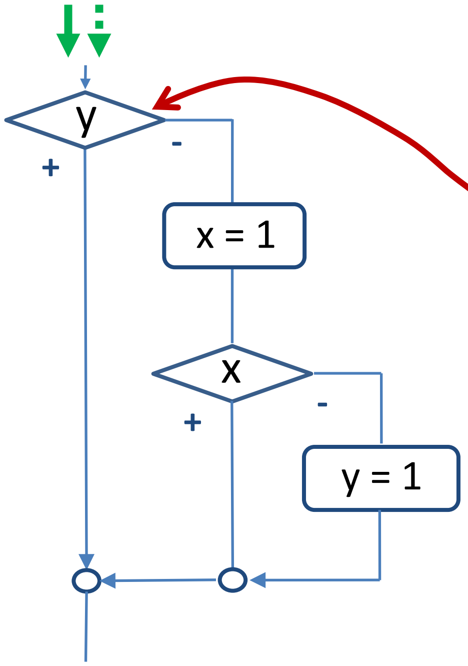
- Berry-constructive *iff* $\forall x \in \mathcal{V}.\ \mu C. \langle\!\langle P \rangle\!\rangle^0_C(x) \in \{0, 1\}$;
- strongly Berry-constructive *iff* $\forall x \in \mathcal{V}.\ \mu C. \langle\!\langle P \rangle\!\rangle^\perp_C(x) \in \{\perp, 0, 1\}$.

Theorem 4. *P is strongly Berry-constructive $\Rightarrow$ P is Berry-constructive.*

4 EXAMPLES

if y else $(x = 1$; if x else $y = 1)$

$S_0 = \{x^0, y^0\}$



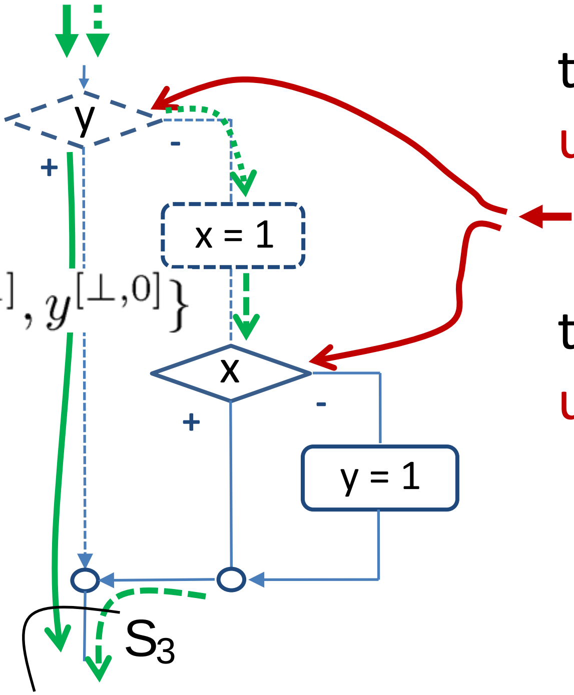
test $S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

$C_0 = \{x^{[\bot,\top]}, y^{[\bot,\top]}\}$

if y else $(x = 1$; if x else $y = 1)$

$S_0 = \{x^0, y^0\}$



$S_1 = \{x^{[\perp,1]}, y^{[\perp,0]}\}$

$S_4 = S_0$ _ upp S_3

test $S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

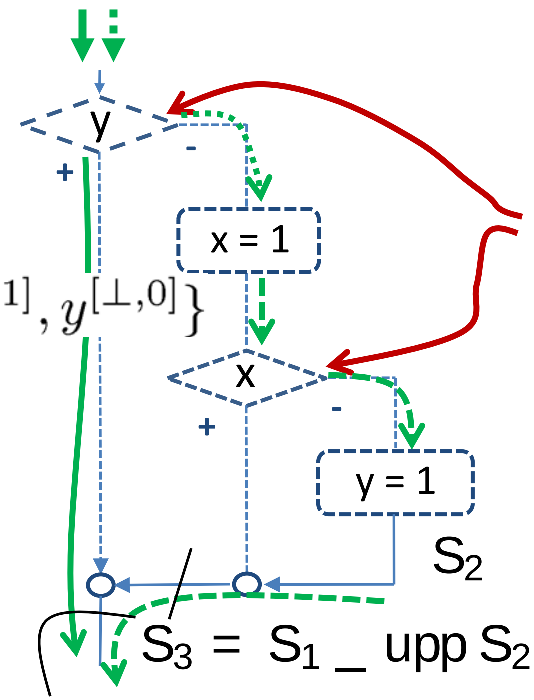
$C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

test $S_1 \vee C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

undecided

if y else $(x = 1 \; ; \text{ if } x \text{ else } y = 1)$

$S_0 = \{x^0, y^0\}$



$S_1 = \{x^{[\perp,1]}, y^{[\perp,0]}\}$

S3 = S1 _ upp S2

S4 = S0 _ upp S3

test $\quad S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

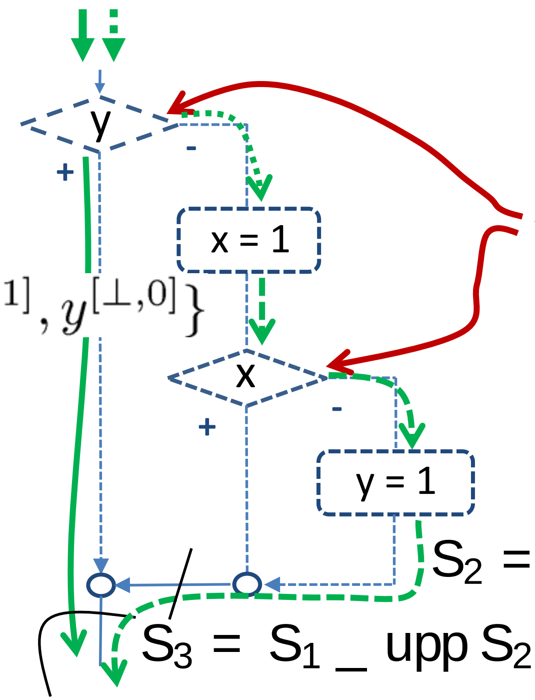
$C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

test $\quad S_1 \vee C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

undecided

if y else $(x = 1\ ;$ if x else $y = 1)$

$S_0 = \{x^0, y^0\}$



$S_1 = \{x^{[\perp,1]}, y^{[\perp,0]}\}$

test $\quad S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

$C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

test $\quad S_1 \vee C_0 = \{x^{[\perp,\top]}, y^{[\perp,\top]}\}$

undecided

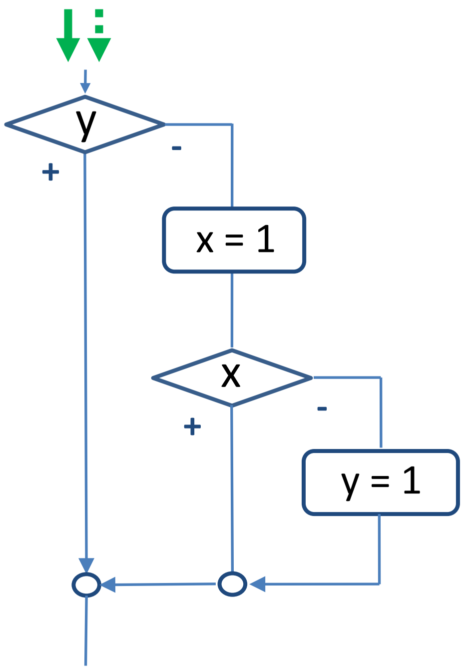
$S_2 = \text{f}\ x^{[?\ ;1]}; y^{1}\text{g}$

$S_3 = S_1\ \_\ \text{upp}\ S_2 = \text{f}\ x^{[?\ ;1]}; y^{[?\ ;1]}\text{g}$

$S_4 = S_0\ \_\ \text{upp}\ S_3 = \text{f}\ x^{[0;1]}; y^{[0;1]}\text{g}$

if y else $(x = 1\ ;$ if x else $y = 1)$

$S_0 = \{x^0, y^0\}$

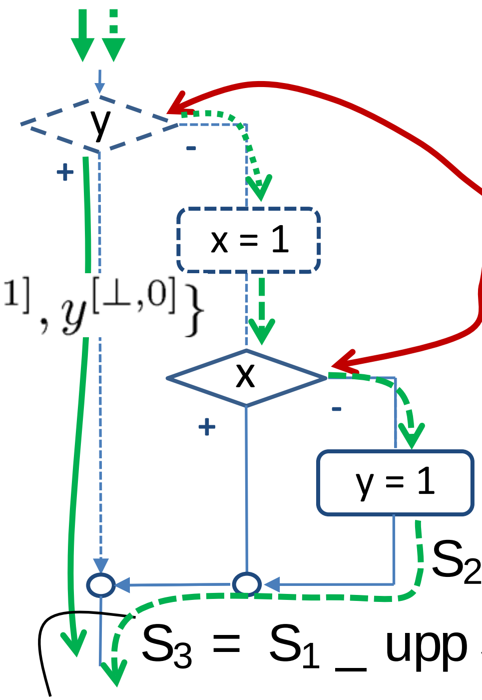


$C_1 = \{x^{[0,1]}, y^{[0,1]}\}$

S_4 = S_0 _ upp S_3 = f $x^{[0;1]}$; $y^{[0;1]}$g

if y else $(x = 1\ ;$ if x else $y = 1)$

$S_0 = \{x^0, y^0\}$



$S_1 = \{x^{[\bot,1]}, y^{[\bot,0]}\}$

test $\quad S_0 \vee C_1 = \{x^{[0,1]}, y^{[0,1]}\}$

undecided

$C_1 = \{x^{[0,1]}, y^{[0,1]}\}$

test $\quad S_1 \vee C_1 = \{x^{[0,1]}, y^{[0,1]}\}$

undecided

$S_2 =$ f $x^{[?\,;1]}; y^{1}$g

$S_3 = S_1$ _ upp $S_2 =$ f $x^{[?\,;1]}; y^{[?\,;1]}$g

$S_4 = S_0$ _ upp $S_3 =$ f $x^{[0;1]}; y^{[0;1]}$g

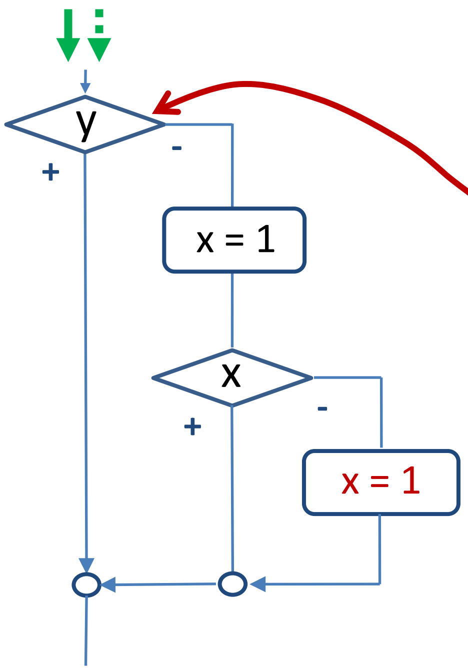
Fixed Point

response undecided

not Berry-constructive

if y else $(x = 1 \; ; \text{if } x \text{ else } x = 1)$

$S_0 = \{x^0, y^0\}$



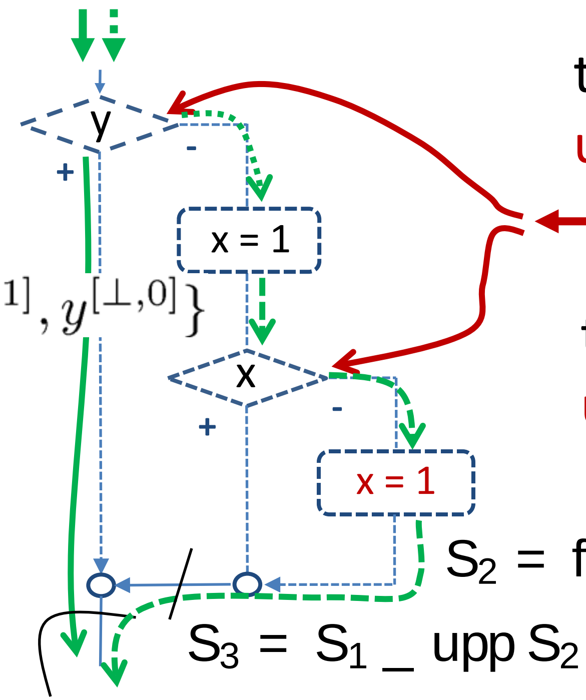
test $S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

$C_0 = \{x^{[\bot,\top]}, y^{[\bot,\top]}\}$

if y else $(x = 1 \,;$ if x else $x = 1)$

$S_0 = \{x^0, y^0\}$



test $\quad S_0 \vee C_0 = \{x^{[0,\top]}, y^{[0,\top]}\}$

undecided

$C_0 = \{x^{[\bot,\top]}, y^{[\bot,\top]}\}$

$S_1 = \{x^{[\bot,1]}, y^{[\bot,0]}\}$

test $\quad S_1 \vee C_0 = \{x^{[\bot,\top]}, y^{[\bot,\top]}\}$

undecided

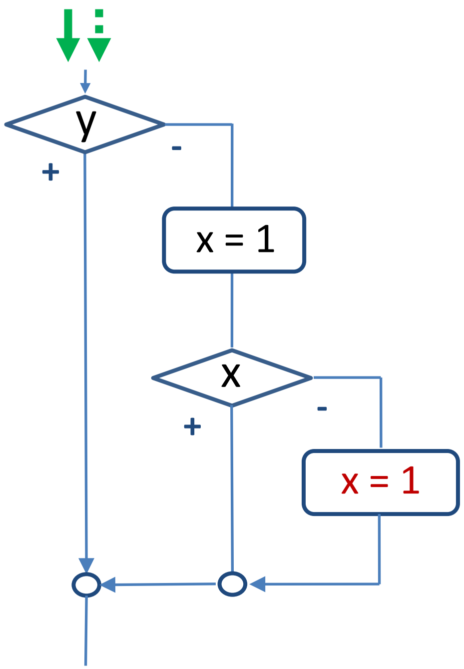
$S_2 = \mathrm{f}\; x^1; y^{[?\;;0]}\mathrm{g}$

$S_3 = S_1 \_ \text{ upp } S_2 = \mathrm{f}\; x^{[?\;;1]}; y^{[?\;;0]}\mathrm{g}$

$S_4 = S_0 \_ \text{ upp } S_3 = \mathrm{f}\; x^{[0;1]}; y^0\mathrm{g}$

if y else $(x = 1\ ;$ if x else $x = 1)$

$$S_0 = \{x^0, y^0\}$$

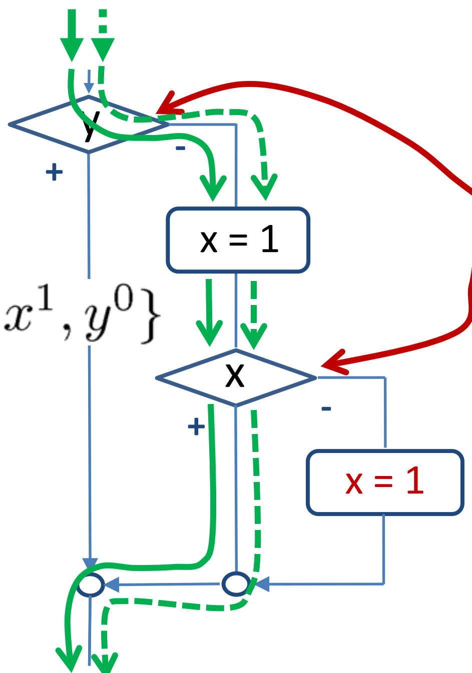


$$\leftarrow C_1 = \{x^{[0,1]}, y^0\}$$

S_4 = S_0 _ upp S_3 = f $x^{[0;1]}$; y^0g

if y else $(x = 1\ ;\ \text{if } x \text{ else } x = 1)$

$$S_0 = \{x^0, y^0\}$$



$$S_1 = \{x^1, y^0\}$$

test $\quad S_0 \vee C_1 = \{x^{[0,1]}, y^0\}$

decided

$$C_1 = \{x^{[0,1]}, y^0\}$$

test $\quad S_1 \vee C_1 = \{x^1, y^0\}$

decided

$$S_1 = \{x^1, y^0\}$$

Fixed Point

crisp response

Berry-constructive

5 CONCLUSION

Conclusions

- Signals can be emulated and generalised using shared variables + synchronisation constraints.
) „SC-Thesis“: Signals in Esterel are syntactic sugar.
- Sequential Constructiveness permits
  – arbitrary (init;update;read;)*-tick cycles.
- Berry-constructive reactions correspond to
  – a single $(\text{init;update;read;})^1$-tick cycle;
  – fixed point analysis on sequential-parallel lattice $I(\mathbb{D})$
- Yes, SC is a conservative extension of BC

Open Problems

- extend results to full Esterel (V7) syntax
- develop fixed-point semantics for SC