Proof of mutual-exclusion and non-starvation of a program: PostgreSQL

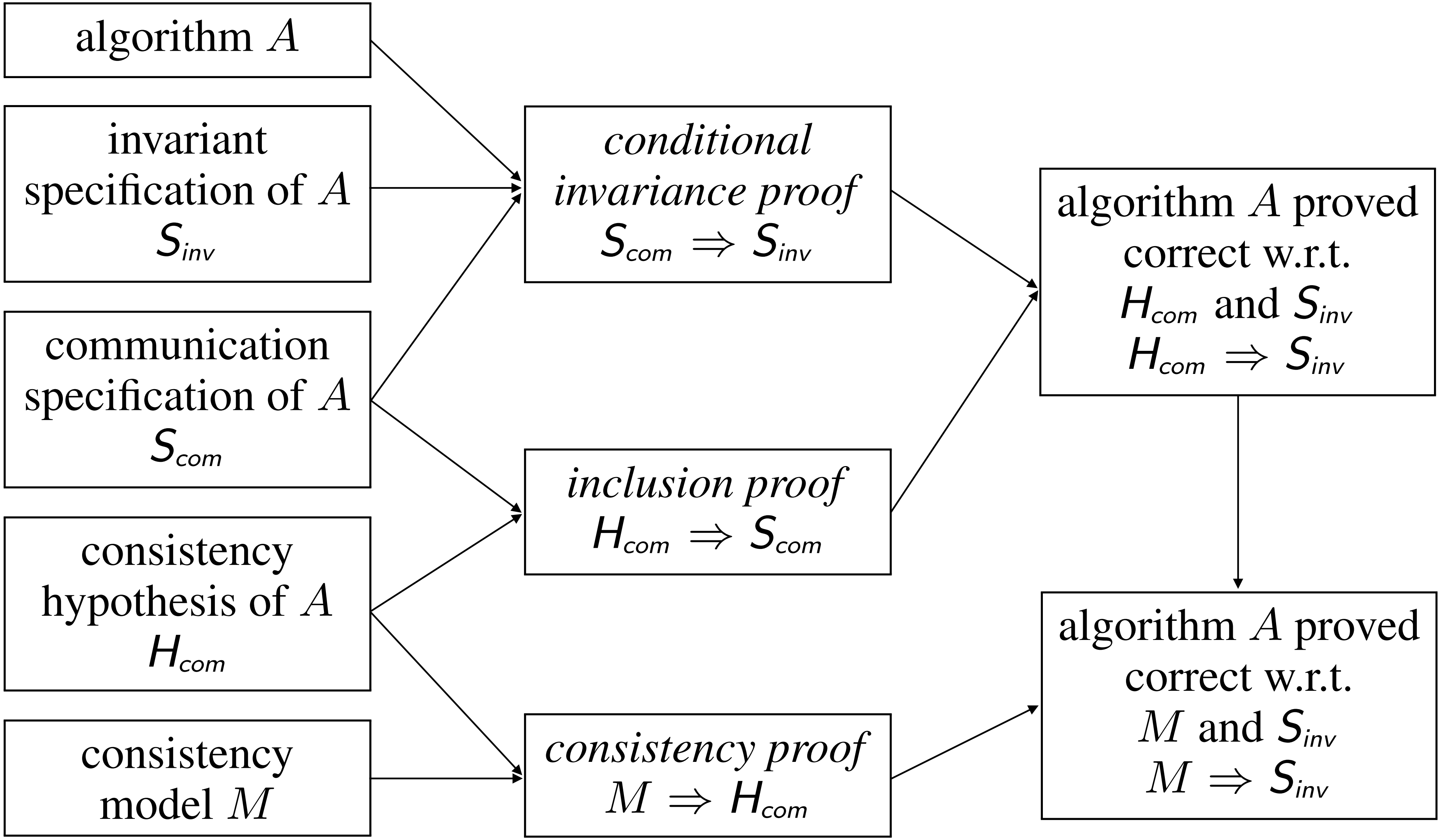
Jade Alglave (MSR-Cambridge, UCL, UK)
Patrick Cousot (NYU, Emer. ENS, PSL)

Dagstuhl Seminar 16471
`http://www.dagstuhl.de/16471`
Concurrency with Weak Memory Models: Semantics, Languages, Compilation, Verification, Static Analysis, and Synthesis
November 22 , 2016

PostgreSQL

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do                                      || 21:do
2:   do                                    || 22:   do
3:     r[] R10 latch0                      || 23:     r[] R11 latch1
4:   while (R10=0)                         || 24:   while (R11=0)
5:   w[] latch0 0                          || 25:   w[] latch1 0
6:   r[] Rf0 flag0                         || 26:   r[] Rf1 flag1
7:   if (Rf0≠0) then                       || 27:   if (Rf1≠0) then
8:     (* critical section *)              || 28:     (* critical section *)
       w[] flag0 0                         ||         w[] flag1 0
9:     w[] flag1 1                         || 29:     w[] flag0 1
10:    w[] latch1 1                        || 30:     w[] latch0 1
11:  fi                                    || 31:   fi
12:while true                              || 32:while true
13:                                        || 33:
```

Methodology



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

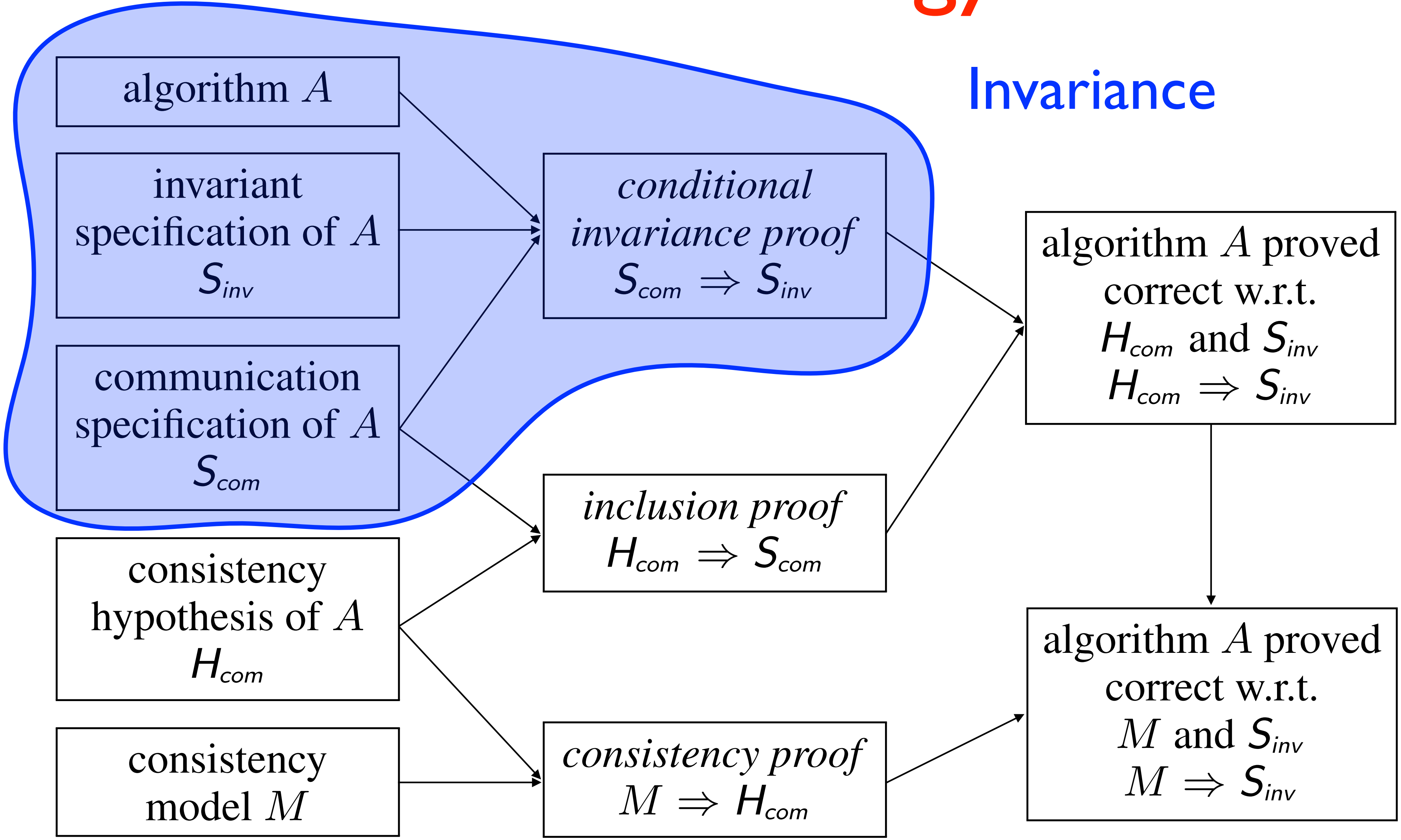
inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Methodology



Invariance

algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

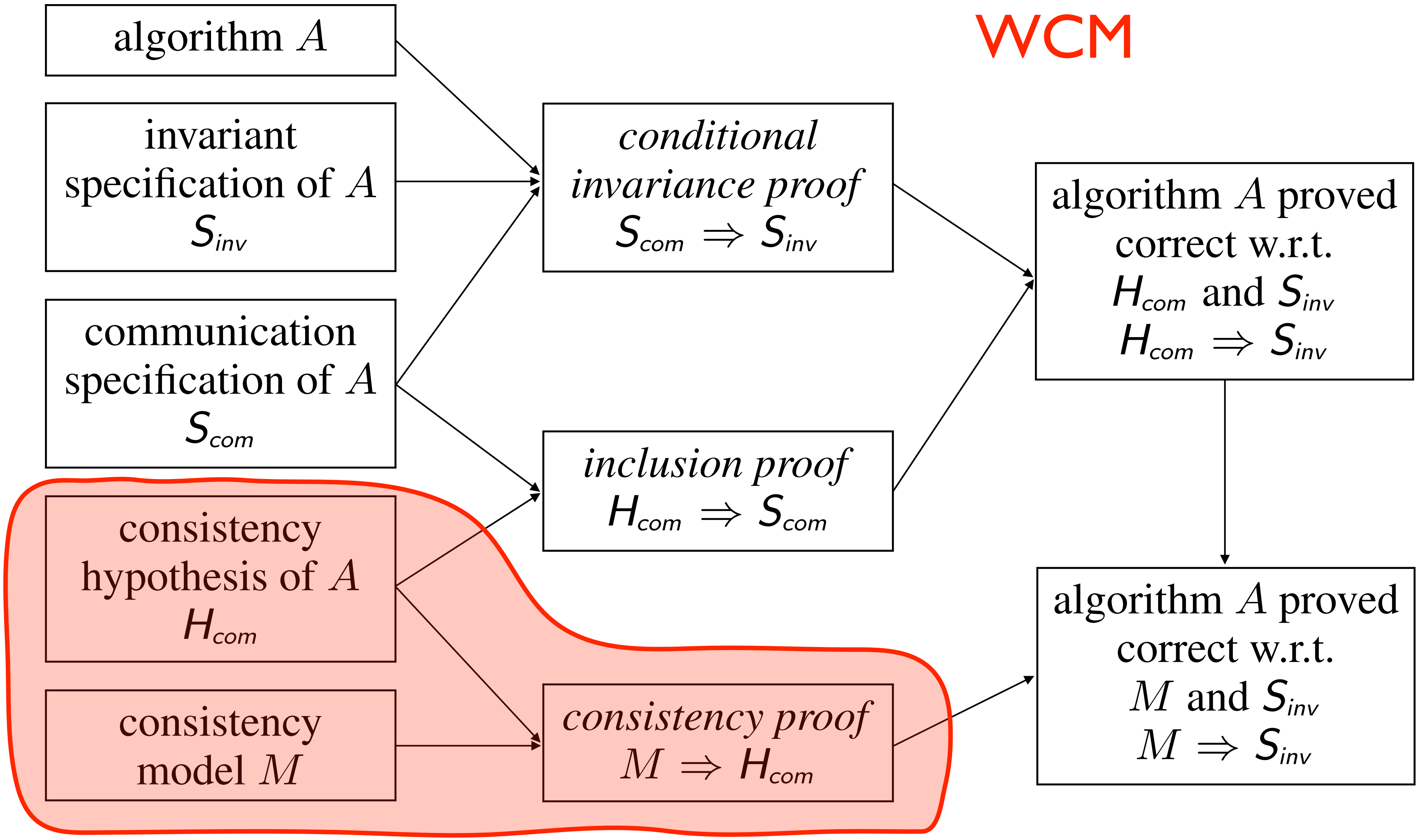
consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Methodology

WCM



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

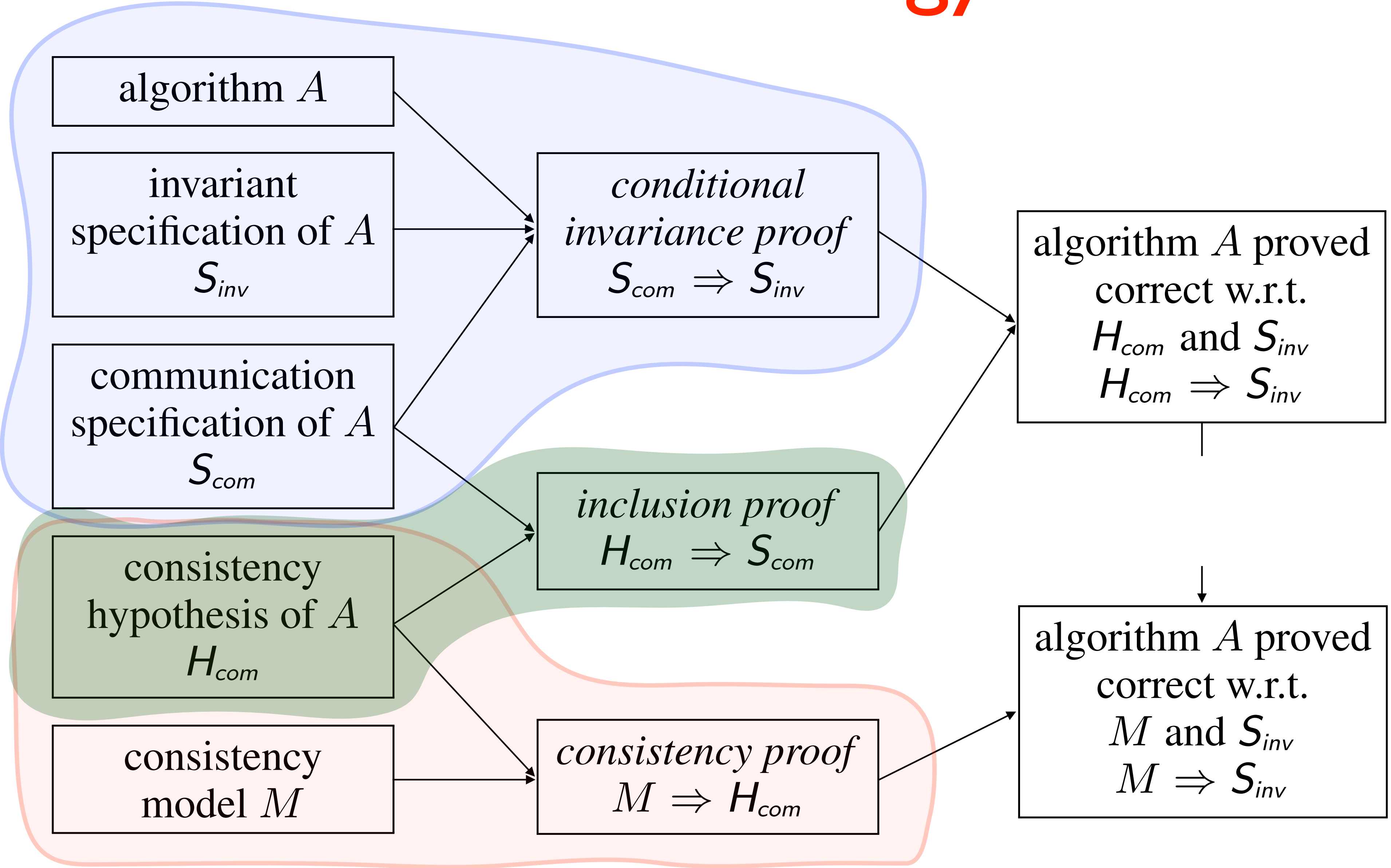
inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Methodology



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

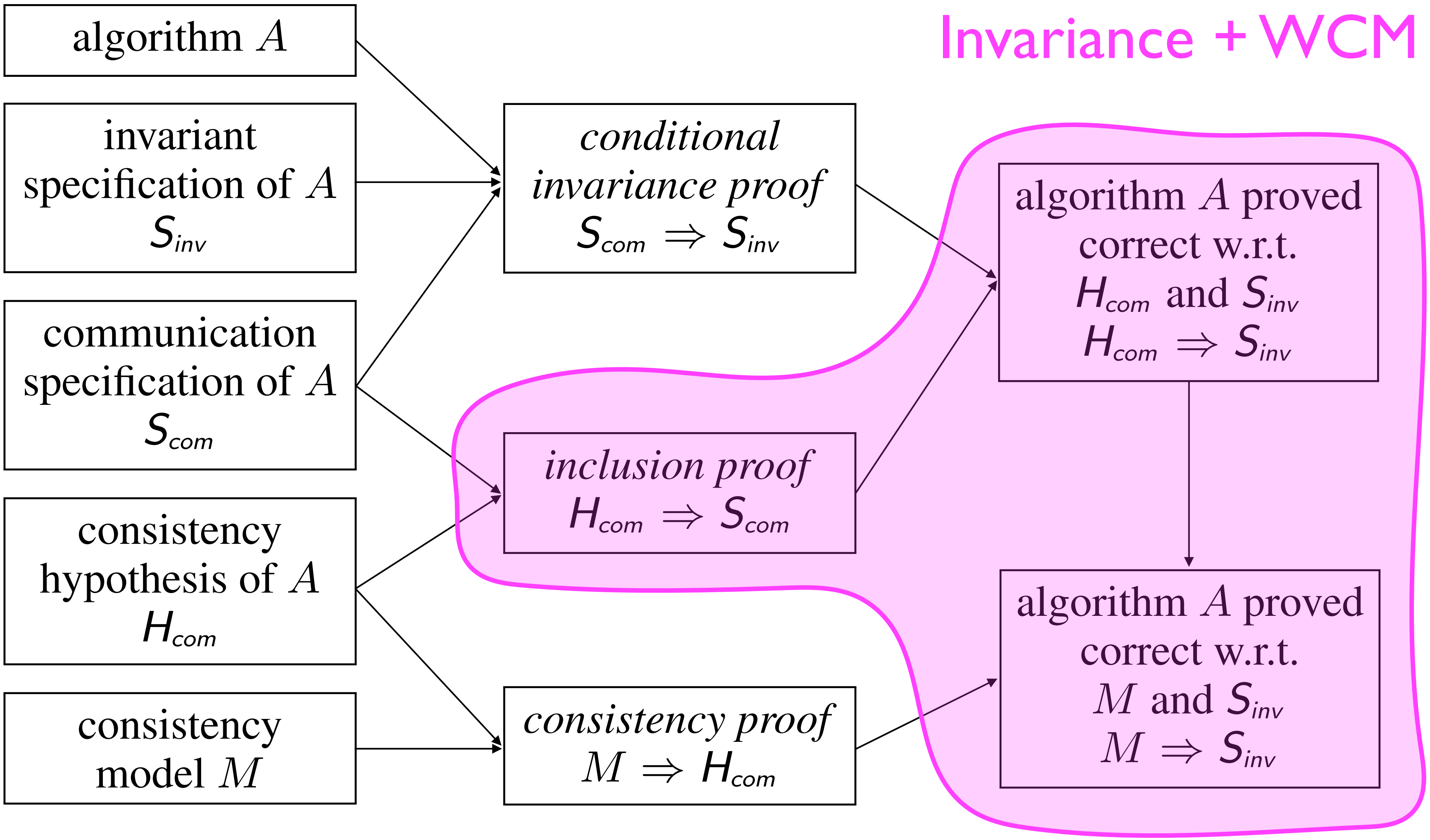
inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Methodology



Invariance + WCM

algorithm A

invariant specification of A
S_{inv}

communication specification of A
S_{com}

consistency hypothesis of A
H_{com}

consistency model M

conditional invariance proof
$S_{com} \Rightarrow S_{inv}$

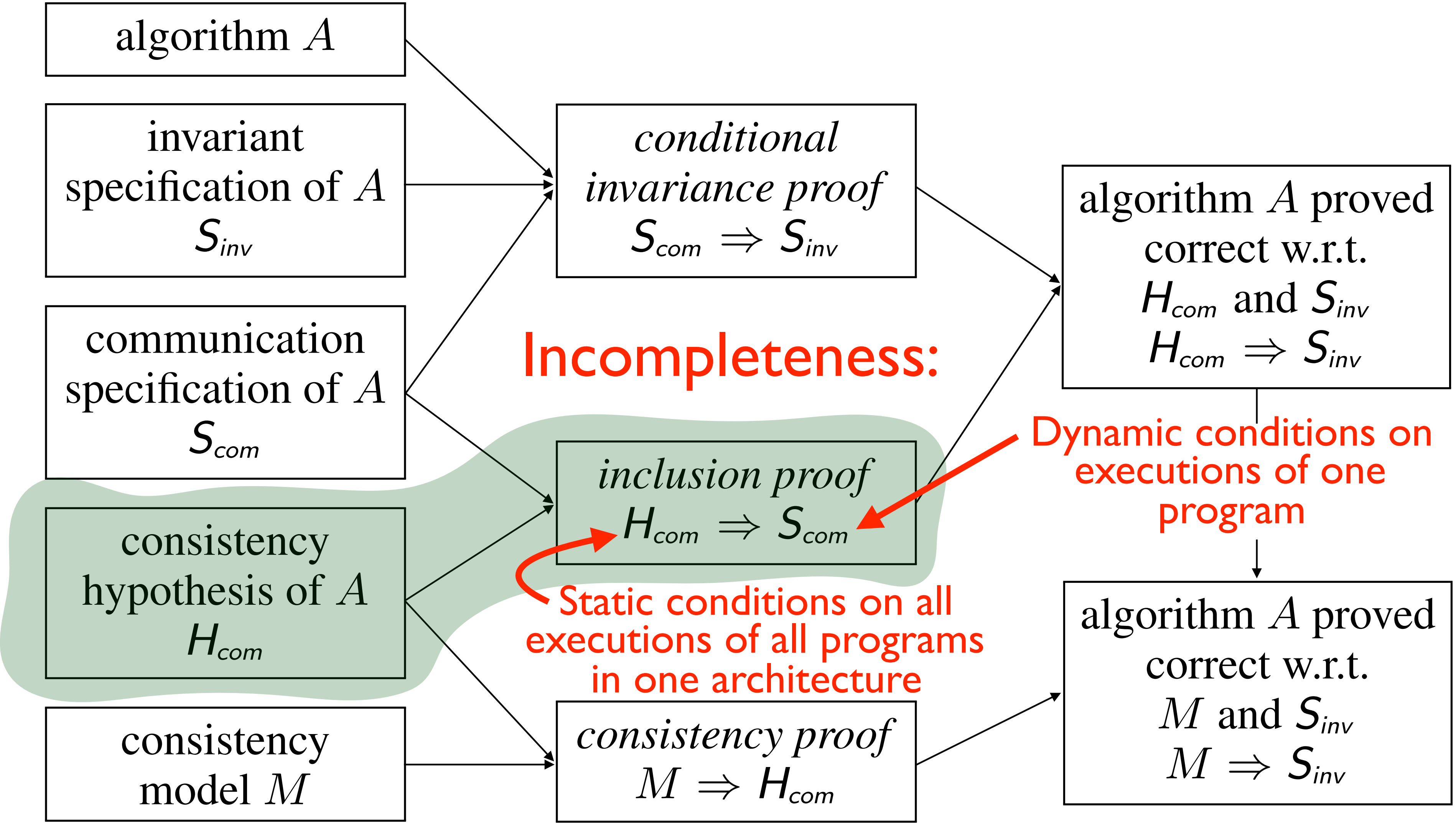
inclusion proof
$H_{com} \Rightarrow S_{com}$

consistency proof
$M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv}
$H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv}
$M \Rightarrow S_{inv}$

Methodology



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

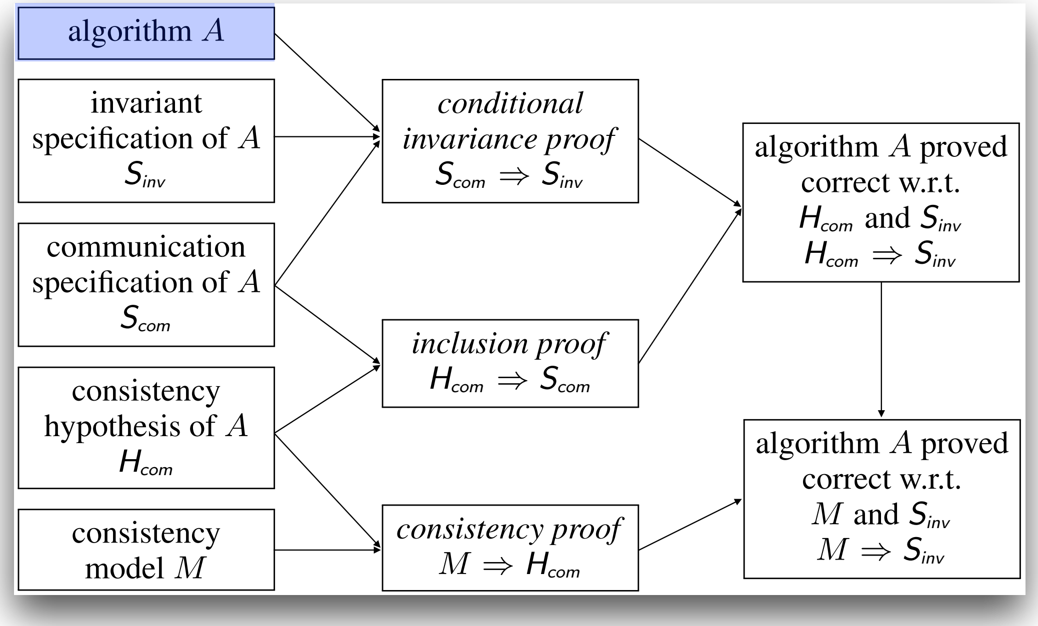
Incompleteness:

Dynamic conditions on executions of one program

Static conditions on all executions of all programs in one architecture

Conditional invariance proof: Mutual exclusion

Algorithm


algorithm A
invariant specification of A S_{inv}
communication specification of A S_{com}
consistency hypothesis of A H_{com}
consistency model M
conditional invariance proof $S_{com} \Rightarrow S_{inv}$
inclusion proof $H_{com} \Rightarrow S_{com}$
consistency proof $M \Rightarrow H_{com}$
algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$
algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

PostgreSQL

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}                                    || 21:do {$\ell$}
2:   do {$j_i$}                                || 22:   do {$m_\ell$}
3:     r[] Rl0 latch0 {$\leadsto L0^i_{j_i}$}  || 23:     r[] Rl1 latch1 {$\leadsto L1^\ell_{m_\ell}$}
4:   while (Rl0=0) {$k_i$}                     || 24:   while (Rl1=0) {$n_\ell$}
5:   w[] latch0 0                              || 25:   w[] latch1 0
6:   r[] Rf0 flag0 {$\leadsto F0^i$}           || 26:   r[] Rf1 flag1 {$\leadsto F1^\ell$}
7:   if (Rf0$\neq$0) then                      || 27:   if (Rf1$\neq$0) then
8:     (* critical section *)                  || 28:     (* critical section *)
       w[] flag0 0                             ||         w[] flag1 0
9:     w[] flag1 1                             || 29:     w[] flag0 1
10:    w[] latch1 1                            || 30:     w[] latch0 1
11:  fi                                        || 31:  fi
12:while true                                  || 32:while true
13:                                            || 33:
```

Stamps

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do $\{i\}$                                  || 21:do $\{\ell\}$
2:   do $\{j_i\}$                              || 22:  do $\{m_\ell\}$
3:     r[] Rl0 latch0 $\{\leadsto L0^{i}_{j_i}\}$    || 23:    r[] Rl1 latch1 $\{\leadsto L1^{\ell}_{m_\ell}\}$
4:   while (Rl0=0) $\{k_i\}$                    || 24:  while (Rl1=0) $\{n_\ell\}$
5:   w[] latch0 0                              || 25:  w[] latch1 0
6:   r[] Rf0 flag0 $\{\leadsto F0^{i}\}$            || 26:  r[] Rf1 flag1 $\{\leadsto F1^{\ell}\}$
7:   if (Rf0$\neq$0) then                       || 27:  if (Rf1$\neq$0) then
8:     (* critical section *)                  || 28:    (* critical section *)
       w[] flag0 0                             ||         w[] flag1 0
9:     w[] flag1 1                             || 29:    w[] flag0 1
10:    w[] latch1 1                            || 30:    w[] latch0 1
11:  fi                                        || 31:  fi
12:while true                                  || 32:while true
13:                                            || 33:
```

Ensure that events are unique (your choice)

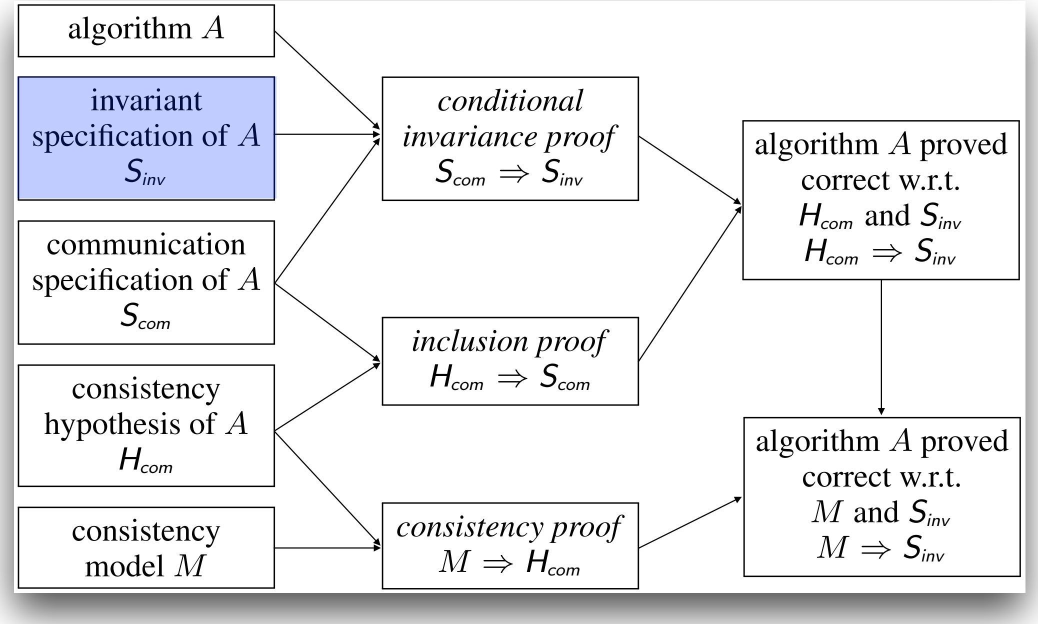
Variables in Hoare logic & L/O-G

- program variables: `int x;`
- in predicates you need to name the value of variable `x` to express properties of this value of x:
  - valueof(x)
  - x
- WCM: no notion of "the" value of a shared variable `x`
- The only way to know something about "the" value of a shared variable `x` is to read it
- Pythia variable: name given to the read value
- Not necessary in the semantics, only in assertions (but we put them in the semantics)

Pythia variables

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}                              || 21:do {$\ell$}
2:   do {$j_i$}                          || 22:   do {$m_\ell$}
3:     r[] Rl0 latch0 {$\rightsquigarrow L0^i_{j_i}$}   || 23:     r[] Rl1 latch1 {$\rightsquigarrow L1^\ell_{m_\ell}$}
4:   while (Rl0=0) {$k_i$}               || 24:   while (Rl1=0) {$n_\ell$}
5:   w[] latch0 0                        || 25:   w[] latch1 0
6:   r[] Rf0 flag0 {$\rightsquigarrow F0^i$}   || 26:   r[] Rf1 flag1 {$\rightsquigarrow F1^\ell$}
7:   if (Rf0≠0) then                     || 27:   if (Rf1≠0) then
8:     (* critical section *)            || 28:     (* critical section *)
       w[] flag0 0                       ||          w[] flag1 0
9:     w[] flag1 1                       || 29:     w[] flag0 1
10:    w[] latch1 1                      || 30:     w[] latch0 1
11:  fi                                  || 31:   fi
12:while true                            || 32:while true
13:                                      || 33:
```

Invariant specification S_{inv}



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Mutual exclusion

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}
2:   do {$j_i$}
3:     r[] R10 latch0 {$\rightsquigarrow L0^i_{j_i}$}
4:   while (R10=0) {$k_i$}
5:   w[] latch0 0
6:   r[] Rf0 flag0 {$\rightsquigarrow F0^i$}
7:   if (Rf0≠0) then
8:     ¬at{28}
       (* critical section *)
       w[] flag0 0
9:     w[] flag1 1
10:    w[] latch1 1
11:  fi
12:while true
13:
```

||

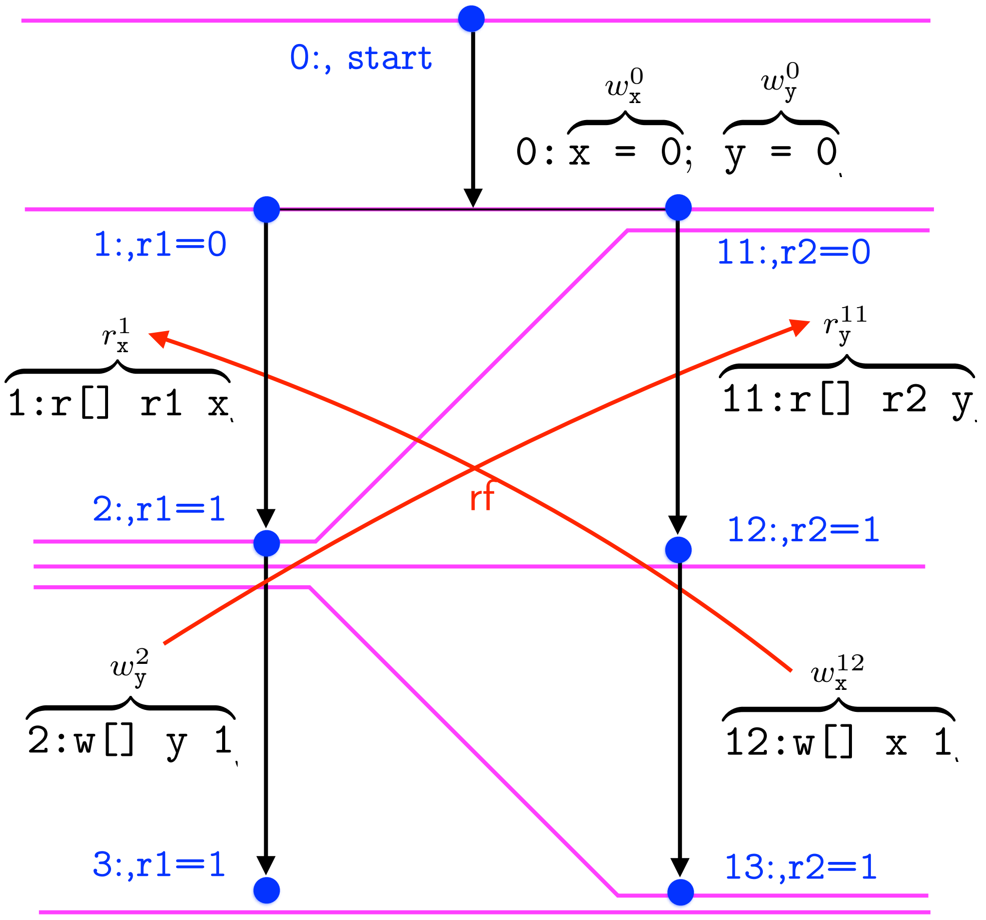
```
21:do {$\ell$}
22:  do {$m_\ell$}
23:    r[] R11 latch1 {$\rightsquigarrow L1^\ell_{m_\ell}$}
24:  while (R11=0) {$n_\ell$}
25:  w[] latch1 0
26:  r[] Rf1 flag1 {$\rightsquigarrow F1^\ell$}
27:  if (Rf1≠0) then
28:    ¬at{8}
       (* critical section *)
       w[] flag1 0
29:    w[] flag0 1
30:    w[] latch0 1
31:  fi
32:while true
33:
```

(invariant Si_{nv} is elsewhere `true`)

Analytic semantics =
Anarchic semantics +
communication constraints

Analytics semantics with cuts

```
0:{ x = 0; y = 0; }
P0          || P1          ;
1:r[] r1 x  || 11:r[] r2 y;
2:w[] y 1   || 12:w[] x 1 ;
3:          || 13:         ;
```



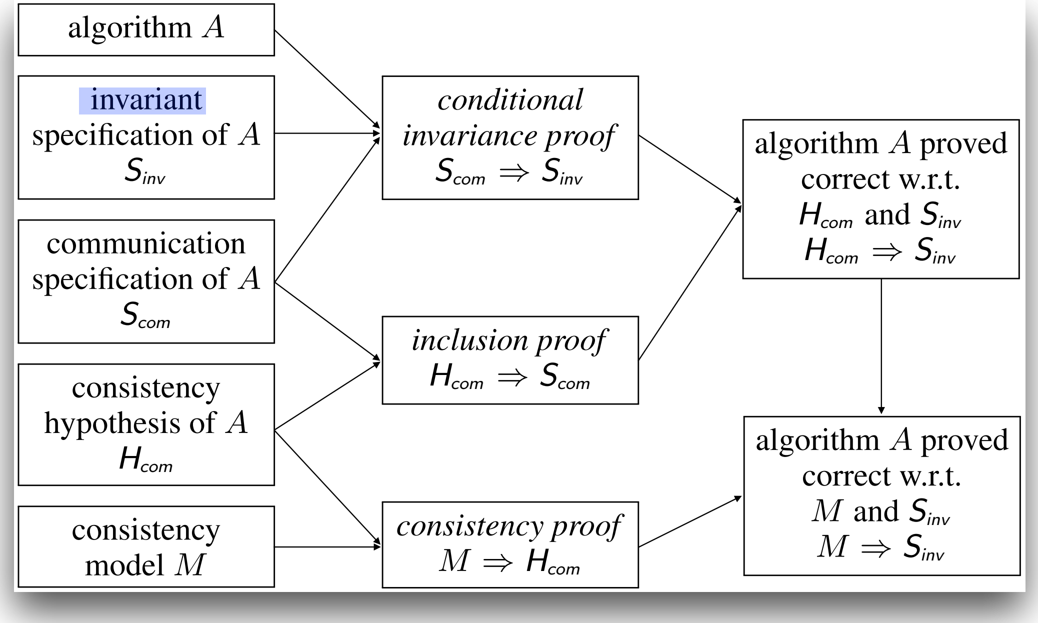
- Anarchic semantics: set of executions:

  $\pi = \varsigma \times \pi \times \mathsf{rf}$

  - ς is the *computation*
  - π is the *cut sequence*
  - rf is the *communication*

- Communication semantics: restrictions on rf in `cat`

Local invariants



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

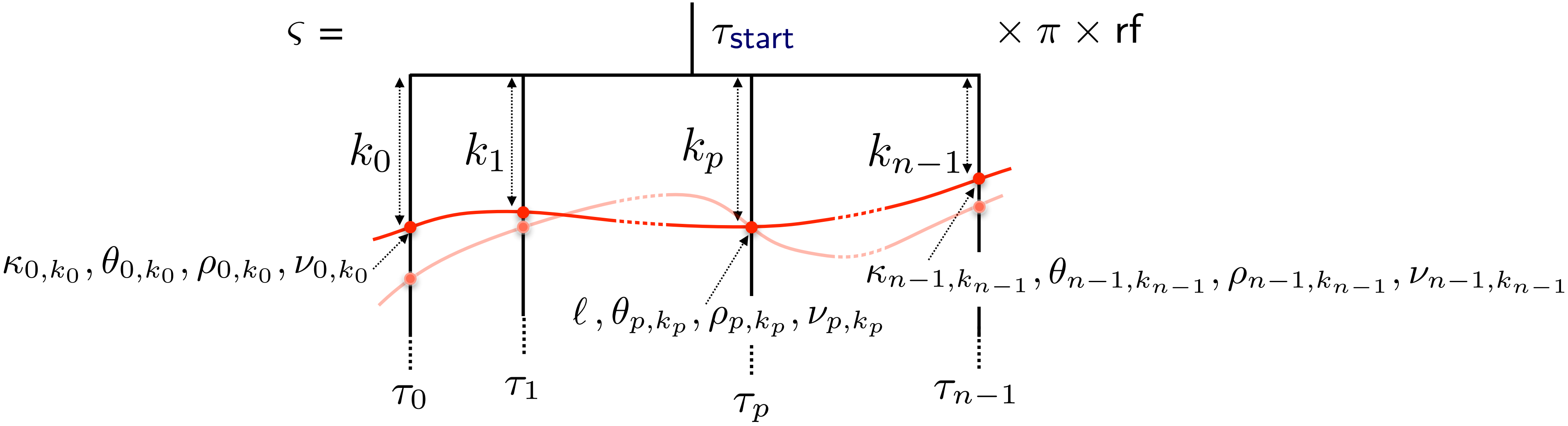
inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

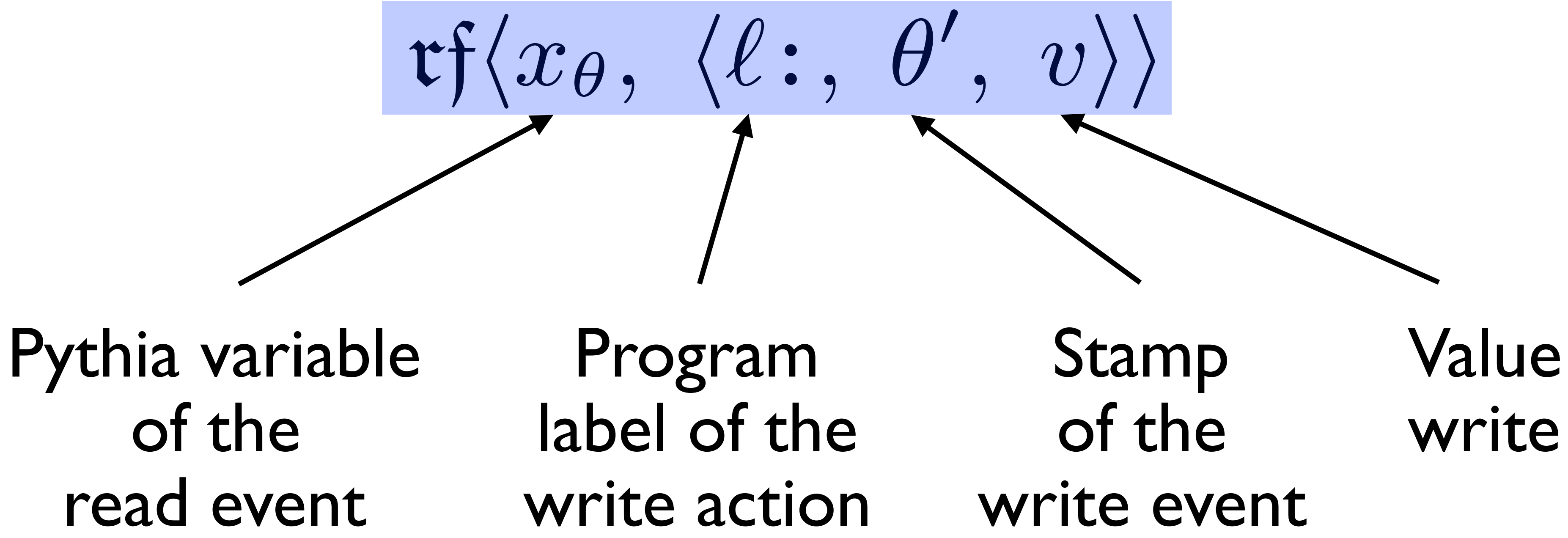
Local invariant



- Attached to each program point ℓ of each process p
- Depends on
  - Program points of all other processes κ
  - Stamps θ of all processes
  - Local registers of all processes ρ
  - Pythia variables ν
  - Communications (rf)

Communication relation rf

- rf: relation between write and read events
- Each rf is encoded by $\mathit{\Gamma}$, a set of pairs



$$\mathfrak{rf}\langle x_\theta,\ \langle \ell :,\ \theta',\ v\rangle\rangle$$

Pythia variable of the read event

Program label of the write action

Stamp of the write event

Value write

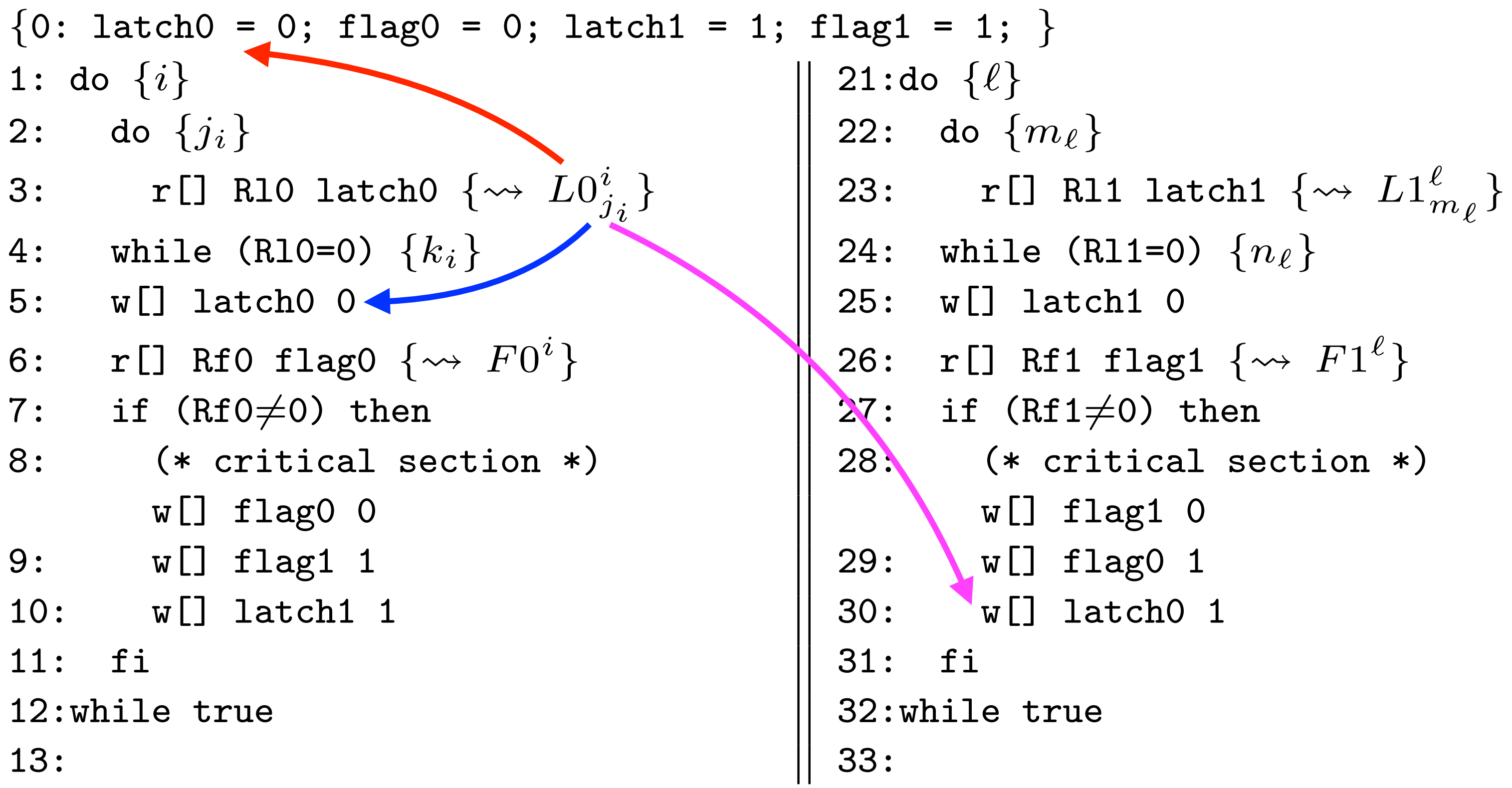
- $\mathit{\Gamma} \in \Gamma$ (the set of all possible communications rf)

Anarchic communications

Anarchic communications

- Any read can read from any write on the same shared variable (location)

$$\mathrm{RL0}^i_{j_i} \triangleq \{\mathfrak{rf}\langle L0^i_{j_i},\ \langle \mathtt{0:},\ \_,\ 0\rangle\rangle, \mathfrak{rf}\langle L0^i_{j_i},\ \langle \mathtt{5:},\ i_5,\ 0\rangle\rangle, \mathfrak{rf}\langle L0^i_{j_i},\ \langle \mathtt{30:},\ \ell_{30},\ 1\rangle\rangle \mid i_5 \in \mathbb{N} \wedge \ell_{30} \in \mathbb{N}\}$$



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}                                 || 21:do {$\ell$}
2:   do {$j_i$}                             || 22:   do {$m_\ell$}
3:     r[] Rl0 latch0 {$\rightsquigarrow L0^i_{j_i}$}   || 23:     r[] Rl1 latch1 {$\rightsquigarrow L1^\ell_{m_\ell}$}
4:   while (Rl0=0) {$k_i$}                  || 24:   while (Rl1=0) {$n_\ell$}
5:   w[] latch0 0                           || 25:   w[] latch1 0
6:   r[] Rf0 flag0 {$\rightsquigarrow F0^i$}           || 26:   r[] Rf1 flag1 {$\rightsquigarrow F1^\ell$}
7:   if (Rf0≠0) then                        || 27:   if (Rf1≠0) then
8:     (* critical section *)               || 28:     (* critical section *)
       w[] flag0 0                          ||         w[] flag1 0
9:     w[] flag1 1                          || 29:     w[] flag0 1
10:    w[] latch1 1                         || 30:     w[] latch0 1
11:  fi                                     || 31:   fi
12:while true                               || 32:while true
13:                                         || 33:
```

Anarchic communications

- Possible communications for each read at each stamp (point in the execution):

$$\begin{aligned}
\mathrm{RL0}^i_{j_i} &\triangleq \{\mathfrak{rf}\langle L0^i_{j_i},\ \langle \texttt{0:},\ \_,\ 0\rangle\rangle, \mathfrak{rf}\langle L0^i_{j_i},\ \langle \texttt{5:},\ i_5,\ 0\rangle\rangle, \mathfrak{rf}\langle L0^i_{j_i},\ \langle \texttt{30:},\ \ell_{30},\ 1\rangle\rangle \mid i_5 \in \mathbb{N} \wedge \ell_{30} \in \mathbb{N}\} \\
\mathrm{RF0}^i &\triangleq \{\mathfrak{rf}\langle F0^i,\ \langle \texttt{0:},\ \_,\ 0\rangle\rangle, \mathfrak{rf}\langle F0^i,\ \langle \texttt{8:},\ i_8,\ 0\rangle\rangle, \mathfrak{rf}\langle F0^i,\ \langle \texttt{29:},\ \ell_{29},\ 1\rangle\rangle \mid i_8 \in \mathbb{N} \wedge \ell_{29} \in \mathbb{N}\} \\
\mathrm{RL1}^\ell_{m_\ell} &\triangleq \{\mathfrak{rf}\langle L1^\ell_{m_\ell},\ \langle \texttt{0:},\ \_,\ 1\rangle\rangle, \mathfrak{rf}\langle L1^\ell_{m_\ell},\ \langle \texttt{25:},\ \ell_{25},\ 0\rangle\rangle, \mathfrak{rf}\langle L1^\ell_{m_\ell},\ \langle \texttt{10:},\ i_{10},\ 1\rangle\rangle \mid \ell_{25} \in \mathbb{N} \wedge i_{10} \in \mathbb{N}\} \\
\mathrm{RF1}^\ell &\triangleq \{\mathfrak{rf}\langle F1^\ell,\ \langle \texttt{0:},\ \_,\ 1\rangle\rangle, \mathfrak{rf}\langle F1^\ell,\ \langle \texttt{28:},\ \ell_{28},\ 0\rangle\rangle, \mathfrak{rf}\langle F1^\ell,\ \langle \texttt{9:},\ i_9,\ 1\rangle\rangle \mid \ell_{28} \in \mathbb{N} \wedge i_9 \in \mathbb{N}\}
\end{aligned}$$

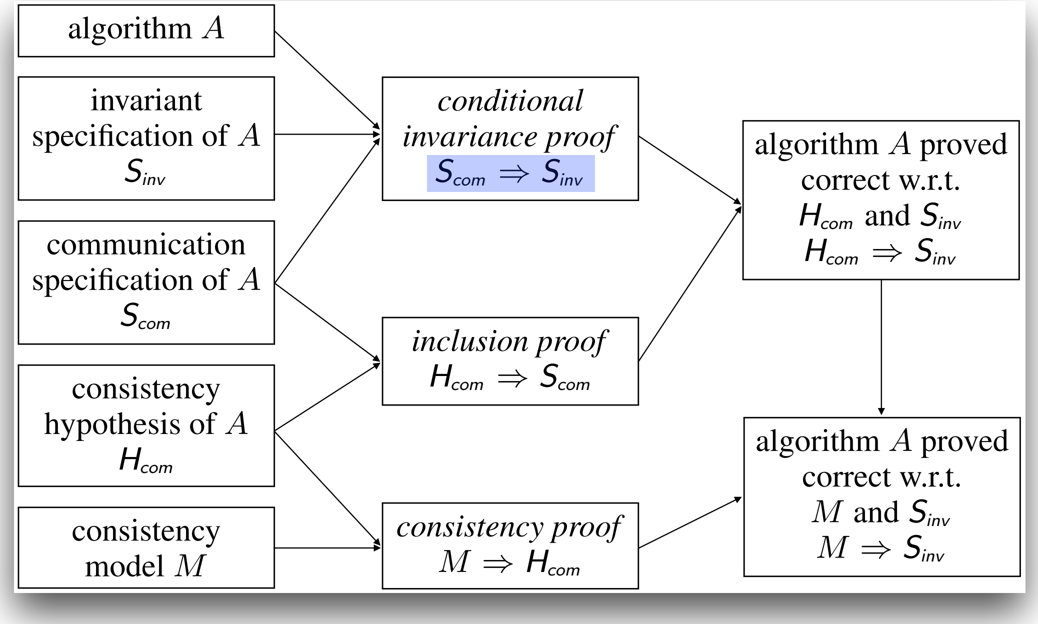
- Anarchic communications:

$$\begin{aligned}
\overline{\Gamma} = \{\{&\mathrm{rl0}^i_{j_i}, \mathrm{rf0}^i, \mathrm{rl1}^\ell_{m_\ell}, \mathrm{rf1}^\ell \mid i \in \mathbb{N} \wedge j_i \in [0, k_i] \wedge \ell \in \mathbb{N} \wedge j \in [0, n_\ell]\} \mid \forall i \in \mathbb{N}\ .\ \forall j_i \in [1, k_i]\ . \\
&\mathrm{rl0}^i_{j_i} \in \mathrm{RL0}^i_{j_i} \wedge \mathrm{rf0}^i \in \mathrm{RF0}^i \wedge \forall \ell \in \mathbb{N}\ .\ \forall m_\ell \in [1, m_\ell]\ .\ \mathrm{rl1}^\ell_{m_\ell} \in \mathrm{RL1}^\ell_{m_\ell} \wedge \mathrm{rf1}^\ell \in \mathrm{RF1}^\ell\}
\end{aligned}$$

- Anarchic semantics: $\mathit{\Gamma} \in \overline{\Gamma}$

- WCM semantics: $\mathit{\Gamma} \in \Gamma, \Gamma \subseteq \overline{\Gamma}$

Inductive invariant S_{ind}



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Inductive invariant

- S_{ind} is inductive under hypothesis S_{com} iff, assuming S_{com}, we have:
  - S_{ind} is true at the beginning of an execution
  - If S_{ind} is true during execution is remains true after one more computation or communication step
- $$\frac{S_{inv} \text{ holds under hypothesis } S_{com} \quad S_{ind} \Rightarrow S_{inv}}{S_{com} \Rightarrow S_{inv}}$$

Inductive invariant

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\mathit{\Gamma} \in \Gamma\}$
   do $\{i\}$
2:   $\{\mathit{\Gamma} \in \Gamma\}$
     do $\{j_i\}$
3:     $\{\mathit{\Gamma} \in \Gamma\}$
       r[] Rl0 latch0 $\{\leadsto L0^i_{j_i}\}$
4:     $\{\mathit{\Gamma} \in \Gamma \wedge \texttt{Rl0} = L0^i_{j_i} \wedge (\mathrm{r0Rl0}^i_{j_i}[\mathit{\Gamma}] \vee \mathrm{r1Rl0}^i_{j_i}[\mathit{\Gamma}])\}$
     while (Rl0=0) $\{k_i\}$
5:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
     w[] latch0 0
6:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
     r[] Rf0 flag0 $\{\leadsto F0^i\}$
7:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \texttt{Rf0} = F0^i$
                              $\wedge (\mathrm{r0Rf0}^i[\mathit{\Gamma}] \vee \mathrm{r1Rf0}^i[\mathit{\Gamma}])\}$
     if (Rf0≠0) then
8:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       (* critical section *)
       w[] flag0 0
9:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       w[] flag1 1
10:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       w[] latch1 1
11:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
     fi
12:  $\{\mathit{\Gamma} \in \Gamma\}$
   while true
13:$\{\mathsf{false}\}$
```

```
21:$\{\mathit{\Gamma} \in \Gamma\}$
   do $\{\ell\}$
22:  $\{\mathit{\Gamma} \in \Gamma\}$
     do $\{m_\ell\}$
23:    $\{\mathit{\Gamma} \in \Gamma\}$
       r[] Rl1 latch1 $\{\leadsto L1^\ell_{m_\ell}\}$
24:    $\{\mathit{\Gamma} \in \Gamma \wedge \texttt{Rl1} = L1^\ell_{m_\ell} \wedge (\mathrm{r0Rl1}^\ell_{m_\ell}[\mathit{\Gamma}] \vee \mathrm{r1Rl1}^\ell_{m_\ell}[\mathit{\Gamma}])\}$
     while (Rl1=0) $\{n_\ell\}$
25:  $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}]\}$
     w[] latch1 0
26:  $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}]\}$
     r[] Rf1 flag1 $\{\leadsto F1^\ell\}$
27:  $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \texttt{Rf1} = F1^\ell$
                              $\wedge (\mathrm{r0Rf1}^\ell[\mathit{\Gamma}] \vee \mathrm{r1Rf1}^\ell[\mathit{\Gamma}])\}$
     if (Rf1≠0) then
28:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       (* critical section *)
       w[] flag1 0
29:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       w[] flag0 1
30:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       w[] latch0 1
31:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
     fi
32:  $\{\mathit{\Gamma} \in \Gamma\}$
   while true
33:$\{\mathsf{false}\}$
```

Inductive invariant

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  $\{\mathit{\Gamma} \in \Gamma\}$
    do $\{i\}$
2:    $\{\mathit{\Gamma} \in \Gamma\}$
      do $\{j_i\}$
3:      $\{\mathit{\Gamma} \in \Gamma\}$
        r[] R10 latch0 $\{\leadsto L0^i_{j_i}\}$
4:      $\{\mathit{\Gamma} \in \Gamma \wedge \texttt{R10} = L0^i_{j_i} \wedge (\mathrm{r0Rl0}^i_{j_i}[\mathit{\Gamma}] \vee \mathrm{r1}$
      while (R10=0) $\{k_i\}$
5:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      w[] latch0 0
6:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      r[] Rf0 flag0 $\{\leadsto F0^i\}$
7:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \texttt{Rf0} = F0^i$
                              $\wedge (\mathrm{r0Rf0}^i[\mathit{\Gamma}] \vee \mathrm{r1Rf0}^i[\mathit{\Gamma}])\}$
      if (Rf0$\neq$0) then
8:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
        (* critical section *)
        w[] flag0 0
9:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
        w[] flag1 1
10:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
        w[] latch1 1
11:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
      fi
12:   $\{\mathit{\Gamma} \in \Gamma\}$
    while true
13:$\{\mathsf{false}\}$
```

```
21:$\{\mathit{\Gamma} \in \Gamma\}$
    do $\{\ell\}$
22:   $\{\mathit{\Gamma} \in \Gamma\}$
                                   $\mathrm{Rl1}^\ell_{m_\ell}[\mathit{\Gamma}])\}$
      w[] latch1 0
26:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}]\}$
      r[] Rf1 flag1 $\{\leadsto F1^\ell\}$
27:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \texttt{Rf1} = F1^\ell$
                              $\wedge (\mathrm{r0Rf1}^\ell[\mathit{\Gamma}] \vee \mathrm{r1Rf1}^\ell[\mathit{\Gamma}])\}$
      if (Rf1$\neq$0) then
28:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
        (* critical section *)
        w[] flag1 0
29:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
        w[] flag0 1
30:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
        w[] latch0 1
31:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
      fi
32:   $\{\mathit{\Gamma} \in \Gamma\}$
    while true
33:$\{\mathsf{false}\}$
```

Possible communications

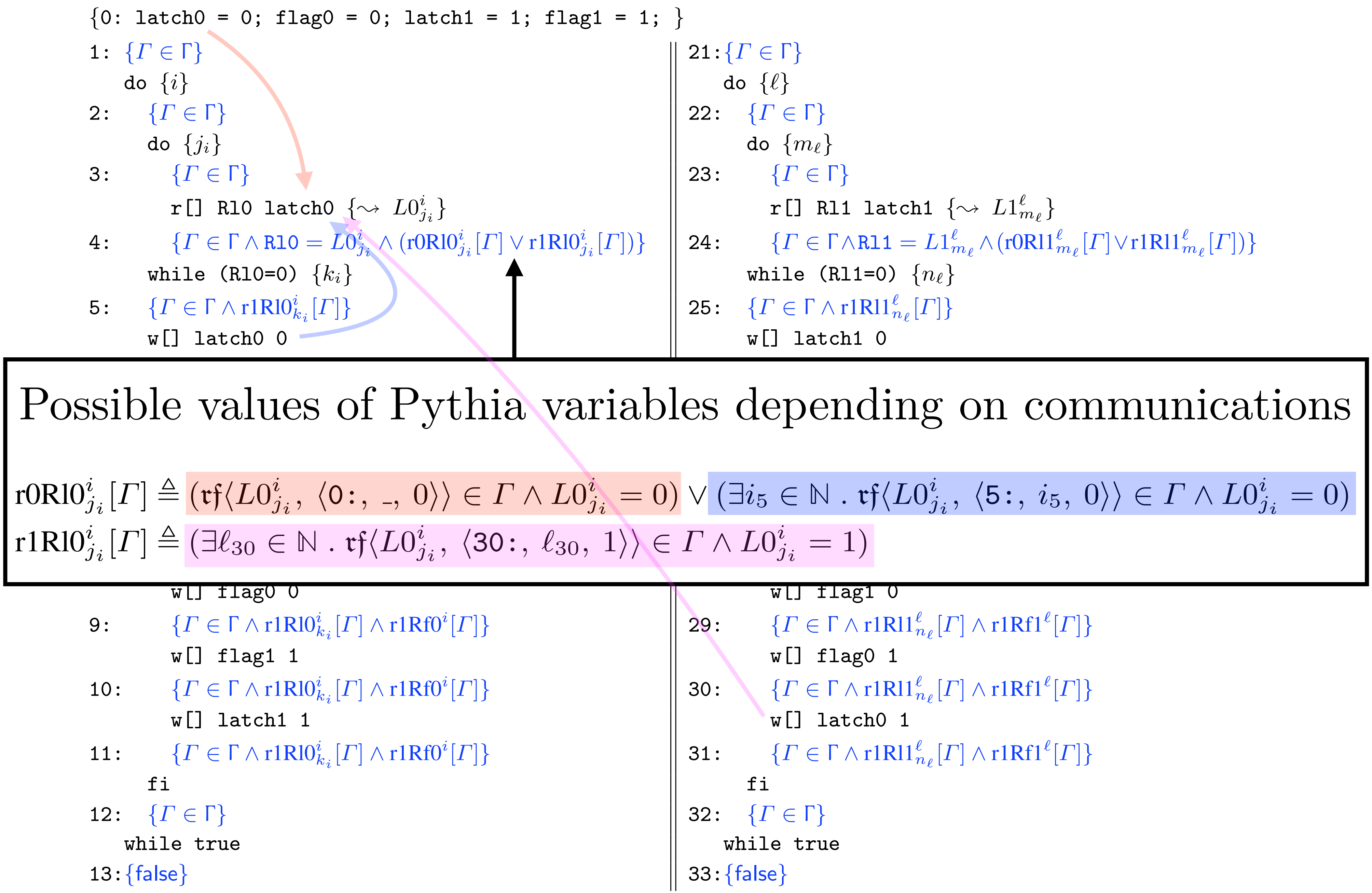
Inductive invariant

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\varGamma \in \Gamma\}$
   do $\{i\}$
2:    $\{\varGamma \in \Gamma\}$
      do $\{j_i\}$
3:       $\{\varGamma \in \Gamma\}$
         r[] Rl0 latch0 $\{\leadsto L0^i_{j_i}\}$
4:       $\{\varGamma \in \Gamma \wedge \texttt{Rl0} = L0^i_{j_i} \wedge (\mathrm{r0Rl0}^i_{j_i}[\varGamma] \vee \mathrm{r1Rl0}^i_{j_i}[\varGamma])\}$
      while (Rl0=0) $\{k_i\}$
5:    $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\varGamma]\}$
      w[] latch0 0
6:    $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\varGamma]\}$
      r[] Rf0 flag0 $\{\leadsto F0^i\}$
7:    $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\varGamma] \wedge \texttt{Rf0} = F0^i$
                      $\wedge (\mathrm{r0Rf0}^i[\varGamma] \vee \mathrm{r1Rf0}^i[\varGamma])\}$
      if (Rf0≠0) then
8:       $\{\varGamma \in$
         (* c
         w[]
9:       $\{\varGamma \in$
         w[]
10:      $\{\varGamma \in$
         w[]
11:      $\{\varGamma \in$
      fi
12:   $\{\varGamma \in \Gamma\}$
   while true
13:{false}
```

```
21:$\{\varGamma \in \Gamma\}$
   do $\{\ell\}$
22:   $\{\varGamma \in \Gamma\}$
      do $\{m_\ell\}$
23:      $\{\varGamma \in \Gamma\}$
         r[] Rl1 latch1 $\{\leadsto L1^\ell_{m_\ell}\}$
24:      $\{\varGamma \in \Gamma \wedge \texttt{Rl1} = L1^\ell_{m_\ell} \wedge (\mathrm{r0Rl1}^\ell_{m_\ell}[\varGamma] \vee \mathrm{r1Rl1}^\ell_{m_\ell}[\varGamma])\}$
      while (Rl1=0) $\{n_\ell\}$
25:   $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma]\}$
      w[] latch1 0
26:   $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma]\}$
      r[] Rf1 flag1 $\{\leadsto F1^\ell\}$
27:   $\{\varGamma \in \Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma] \wedge \texttt{Rf1} = F1^\ell$
                      $\wedge (\mathrm{r0Rf1}^\ell[\varGamma] \vee \mathrm{r1Rf1}^\ell[\varGamma])\}$
      if (Rf1≠0) then
         $\Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^\ell[\varGamma]\}$
         itical section *)
         lag1 0
         $\Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^\ell[\varGamma]\}$
         lag0 1
         $\Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^\ell[\varGamma]\}$
         atch0 1
         $\Gamma \wedge \mathrm{r1Rl1}^\ell_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^\ell[\varGamma]\}$
      fi
32:   $\{\varGamma \in \Gamma\}$
   while true
33:{false}
```

Register assignment of the Pythia variable after read event

Inductive invariant



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  {$\mathit{\Gamma} \in \Gamma$}
    do {$i$}
2:     {$\mathit{\Gamma} \in \Gamma$}
       do {$j_i$}
3:        {$\mathit{\Gamma} \in \Gamma$}
          r[] R10 latch0 {$\leadsto L0^i_{j_i}$}
4:        {$\mathit{\Gamma} \in \Gamma \wedge \texttt{R10} = L0^i_{j_i} \wedge (\mathrm{r0R10}^i_{j_i}[\mathit{\Gamma}] \vee \mathrm{r1R10}^i_{j_i}[\mathit{\Gamma}])$}
       while (R10=0) {$k_i$}
5:     {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}]$}
       w[] latch0 0
          w[] flag0 0
9:        {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]$}
          w[] flag1 1
10:       {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]$}
          w[] latch1 1
11:       {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]$}
       fi
12:    {$\mathit{\Gamma} \in \Gamma$}
    while true
13:{false}
```

```
21:{$\mathit{\Gamma} \in \Gamma$}
    do {$\ell$}
22:    {$\mathit{\Gamma} \in \Gamma$}
       do {$m_\ell$}
23:       {$\mathit{\Gamma} \in \Gamma$}
          r[] R11 latch1 {$\leadsto L1^\ell_{m_\ell}$}
24:       {$\mathit{\Gamma} \in \Gamma \wedge \texttt{R11} = L1^\ell_{m_\ell} \wedge (\mathrm{r0R11}^\ell_{m_\ell}[\mathit{\Gamma}] \vee \mathrm{r1R11}^\ell_{m_\ell}[\mathit{\Gamma}])$}
       while (R11=0) {$n_\ell$}
25:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}]$}
       w[] latch1 0
          w[] flag1 0
29:       {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]$}
          w[] flag0 1
30:       {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]$}
          w[] latch0 1
31:       {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]$}
       fi
32:    {$\mathit{\Gamma} \in \Gamma$}
    while true
33:{false}
```

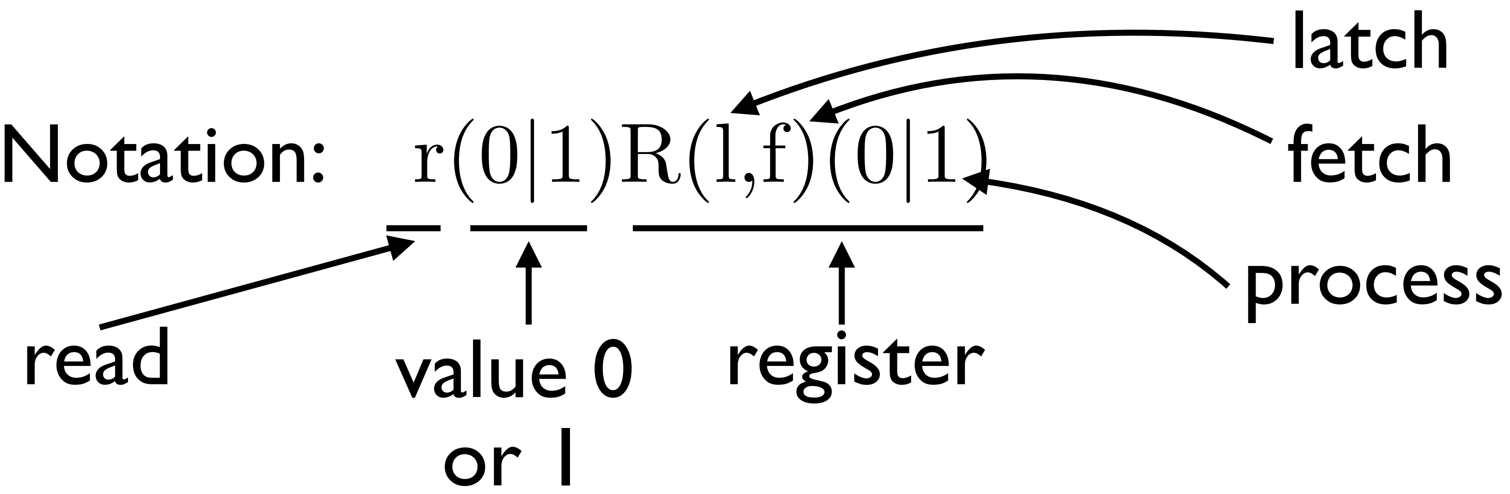
Possible values of Pythia variables depending on communications

$$\mathrm{r0R10}^i_{j_i}[\mathit{\Gamma}] \triangleq (\mathfrak{rf}\langle L0^i_{j_i},\ \langle 0:,\ \_,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge L0^i_{j_i} = 0) \vee (\exists i_5 \in \mathbb{N}\ .\ \mathfrak{rf}\langle L0^i_{j_i},\ \langle 5:,\ i_5,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge L0^i_{j_i} = 0)$$

$$\mathrm{r1R10}^i_{j_i}[\mathit{\Gamma}] \triangleq (\exists \ell_{30} \in \mathbb{N}\ .\ \mathfrak{rf}\langle L0^i_{j_i},\ \langle 30:,\ \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge L0^i_{j_i} = 1)$$

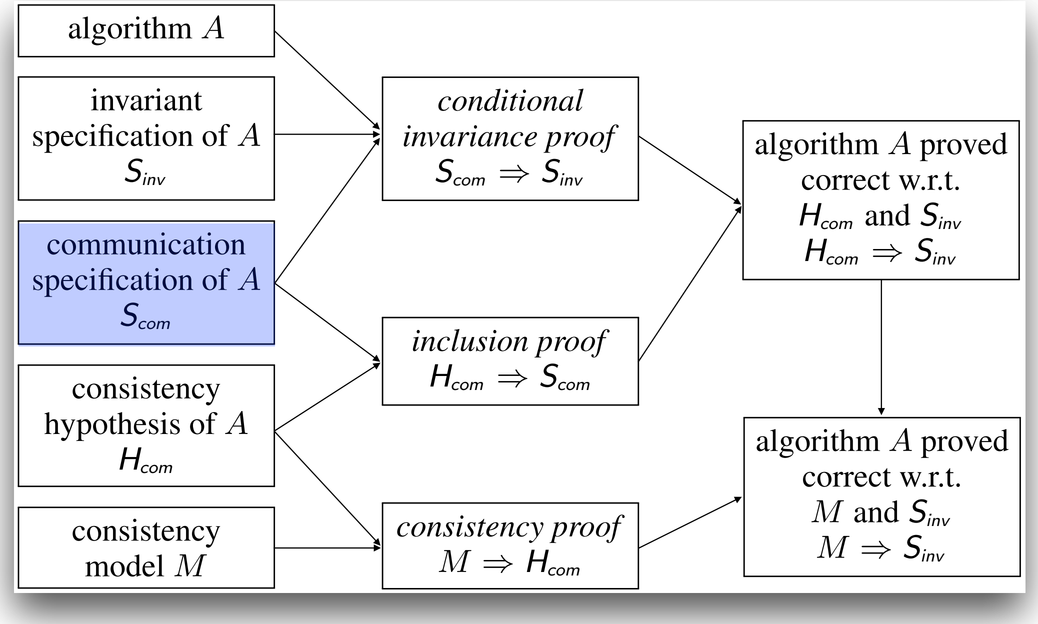
Communicated values

- Notation: r(0|1)R(l,f)(0|1)



$$
\begin{aligned}
\mathrm{r0Rl0}^{i}_{j_i}[\mathit{\Gamma}] &\triangleq (\mathfrak{rf}\langle L0^{i}_{j_i},\ \langle \texttt{0:},\ \_,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge L0^{i}_{j_i} = 0) \vee (\exists i_5 \in \mathbb{N}\ .\ \mathfrak{rf}\langle L0^{i}_{j_i},\ \langle \texttt{5:},\ i_5,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge L0^{i}_{j_i} = 0)\\
\mathrm{r1Rl0}^{i}_{j_i}[\mathit{\Gamma}] &\triangleq (\exists \ell_{30} \in \mathbb{N}\ .\ \mathfrak{rf}\langle L0^{i}_{j_i},\ \langle \texttt{30:},\ \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge L0^{i}_{j_i} = 1)\\
\mathrm{r0Rf0}^{i}[\mathit{\Gamma}] &\triangleq (\mathfrak{rf}\langle F0^{i},\ \langle \texttt{0:},\ \_,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge F0^{i} = 0) \vee (\exists i_8 \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^{i},\ \langle \texttt{8:},\ i_8,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge F0^{i} = 0)\\
\mathrm{r1Rf0}^{i}[\mathit{\Gamma}] &\triangleq (\exists \ell_{29} \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^{i},\ \langle \texttt{29:},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge F0^{i} = 1)\\
\mathrm{r0Rl1}^{\ell}_{m_\ell}[\mathit{\Gamma}] &\triangleq (\exists \ell_{25} \in \mathbb{N}\ .\ \mathfrak{rf}\langle L1^{\ell}_{m_\ell},\ \langle \texttt{25:},\ \ell_{25},\ 0\rangle\rangle \in \mathit{\Gamma} \wedge L1^{\ell}_{m_\ell} = 0)\\
\mathrm{r1Rl1}^{\ell}_{m_\ell}[\mathit{\Gamma}] &\triangleq (\mathfrak{rf}\langle L1^{\ell}_{m_\ell},\ \langle \texttt{0:},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma} \wedge L1^{\ell}_{m_\ell} = 1) \vee (\exists i_{10} \in \mathbb{N}\ .\ \mathfrak{rf}\langle L1^{\ell}_{m_\ell},\ \langle \texttt{10:},\ i_{10},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge L1^{\ell}_{m_\ell} = 1)\\
\mathrm{r0Rf1}^{\ell}[\mathit{\Gamma}] &\triangleq (\exists m_{28} \in \mathbb{N}\ .\ \mathfrak{rf}\langle F1^{\ell},\ \langle \texttt{28:},\ m_{28},\ 0\rangle\rangle \in \mathit{\Gamma} \wedge F1^{\ell} = 0)\\
\mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}] &\triangleq (\mathfrak{rf}\langle F1^{\ell},\ \langle \texttt{0:},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma} \wedge F1^{\ell} = 1) \vee (\exists i_9 \in \mathbb{N}\ .\ \mathfrak{rf}\langle F1^{\ell},\ \langle \texttt{9:},\ i_9,\ 1\rangle\rangle \in \mathit{\Gamma} \wedge F1^{\ell} = 1)
\end{aligned}
$$

Communication specification



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Calculational design of the communication specification

$(\neg\mathsf{S}_{inv}(\mathit{\Gamma},\Gamma)) \wedge \mathsf{S}_{ind}(\mathit{\Gamma},\Gamma)$

$\triangleq$ $\mathsf{at}\{8\} \wedge \mathsf{at}\{28\} \wedge \mathsf{S}_{ind}(\mathit{\Gamma},\Gamma)$ ⦃def. invariance specification S_{inv}⦄

$\Rightarrow$ $\mathsf{at}\{8\} \wedge \mathsf{at}\{28\} \wedge (\exists i, k_i, \ell, n_\ell \in \mathbb{N} \;.\; \mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^{i}[\mathit{\Gamma}] \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}])$ ⦃by invariant $\mathsf{S}_{ind}(\mathit{\Gamma},\Gamma)$⦄

$\Rightarrow$ $\mathsf{at}\{8\} \wedge \mathsf{at}\{28\} \wedge \big(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29} \in \mathbb{N} \;.\; \mathit{\Gamma} \in \Gamma \wedge (\mathfrak{rf}\langle L0^{i}_{k_i}, \langle \mathsf{30}\!:, \ell_{30}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F0^{i}, \langle \mathsf{29}\!:, \ell_{29}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle L1^{\ell}_{n_\ell}, \langle \mathsf{0}\!:, \_, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F1^{\ell}, \langle \mathsf{0}\!:, \_, 1\rangle\rangle \in \mathit{\Gamma})\big) \vee$
$\big(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_9 \in \mathbb{N} \;.\; \mathit{\Gamma} \in \Gamma \wedge (\mathfrak{rf}\langle L0^{i}_{k_i}, \langle \mathsf{30}\!:, \ell_{30}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F0^{i}, \langle \mathsf{29}\!:, \ell_{29}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle L1^{\ell}_{n_\ell}, \langle \mathsf{0}\!:, \_, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F1^{\ell}, \langle \mathsf{9}\!:, i_9, 1\rangle\rangle \in \mathit{\Gamma})\big) \vee$
$\big(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10} \in \mathbb{N} \;.\; \mathit{\Gamma} \in \Gamma \wedge (\mathfrak{rf}\langle L0^{i}_{k_i}, \langle \mathsf{30}\!:, \ell_{30}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F0^{i}, \langle \mathsf{29}\!:, \ell_{29}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle L1^{\ell}_{n_\ell}, \langle \mathsf{10}\!:, i_{10}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F1^{\ell}, \langle \mathsf{0}\!:, \_, 1\rangle\rangle \in \mathit{\Gamma})\big) \vee$
$\big(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10}, i_9 \in \mathbb{N} \;.\; \mathit{\Gamma} \in \Gamma \wedge (\mathfrak{rf}\langle L0^{i}_{k_i}, \langle \mathsf{30}\!:, \ell_{30}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F0^{i}, \langle \mathsf{29}\!:, \ell_{29}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle L1^{\ell}_{n_\ell}, \langle \mathsf{10}\!:, i_{10}, 1\rangle\rangle \in \mathit{\Gamma}) \wedge (\mathfrak{rf}\langle F1^{\ell}, \langle \mathsf{9}\!:, i_9, 1\rangle\rangle \in \mathit{\Gamma})\big)$

⦃def. $\mathrm{r1Rl0}^{i}_{k_i}[\mathit{\Gamma}]$, $\mathrm{r1Rf0}^{i}[\mathit{\Gamma}]$, $\mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}]$, and $\mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]$, $\mathfrak{rf}\langle x_\theta, \langle \ell\!:, \theta', v\rangle\rangle$ implies that $x_\theta = v$, $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$, $\exists$ distributes over $\vee$, and $(\exists x \;.\; A(x)) \wedge B = \exists x \;.\; (A(x) \wedge B)$ when x is not free in B⦄

$\Rightarrow$ $\mathsf{at}\{8\} \wedge \mathsf{at}\{28\} \wedge (\neg\mathsf{S}_{com_1}(\mathit{\Gamma},\Gamma) \vee \neg\mathsf{S}_{com_2}(\mathit{\Gamma},\Gamma) \vee \neg\mathsf{S}_{com_3}(\mathit{\Gamma},\Gamma) \vee \neg\mathsf{S}_{com_4}(\mathit{\Gamma},\Gamma))$

$\Rightarrow$ $\neg\mathsf{S}_{com}(\mathit{\Gamma},\Gamma)$

Calculational design of the communication specification

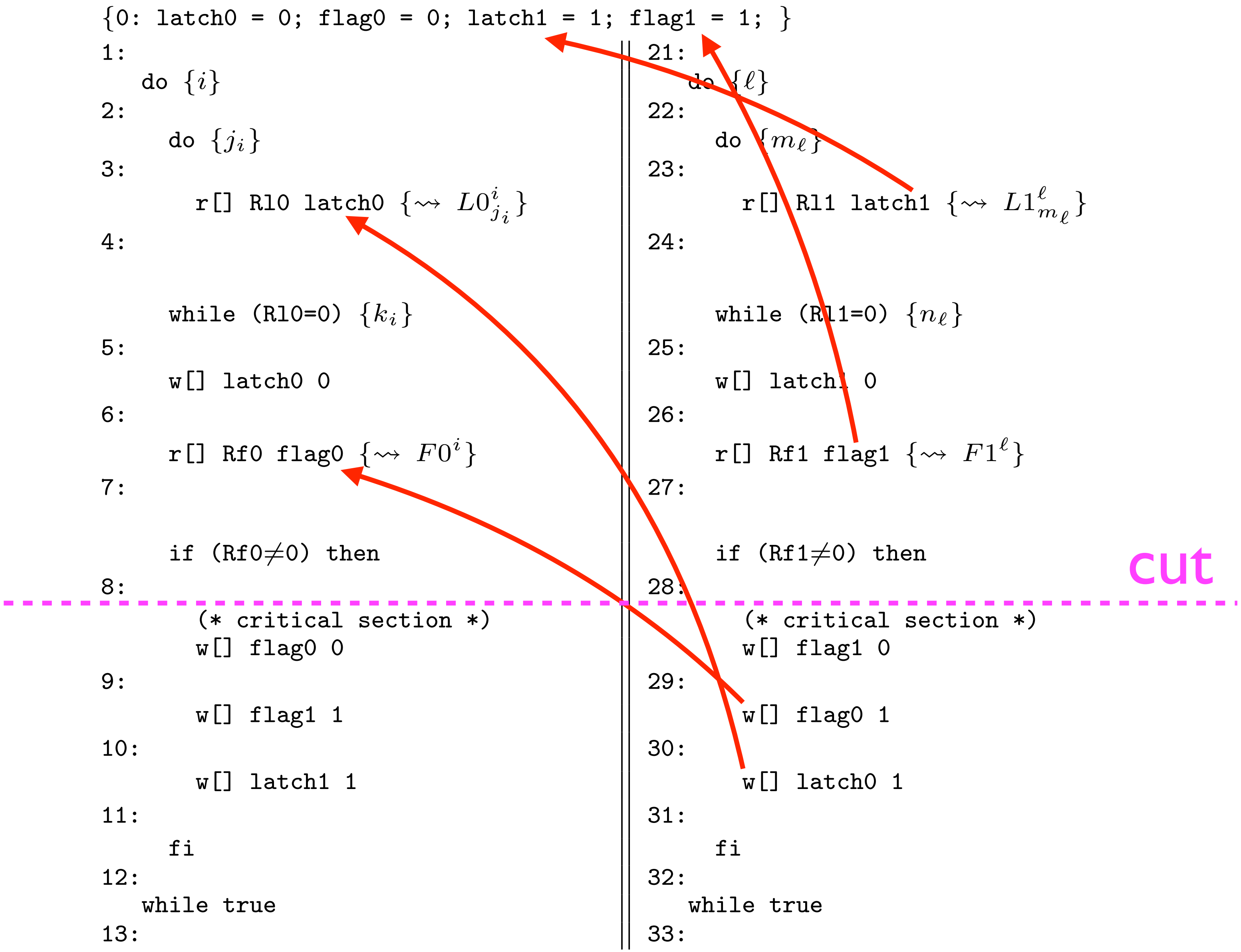
- where

$$\mathcal{S}_{com}(\Gamma,\overline{\Gamma}) \triangleq (\mathsf{at}\{8\} \wedge \mathsf{at}\{28\}) \Longrightarrow (\mathcal{S}_{com_1}(\Gamma,\overline{\Gamma}) \wedge \mathcal{S}_{com_2}(\Gamma,\overline{\Gamma}) \wedge \mathcal{S}_{com_3}(\Gamma,\overline{\Gamma}) \wedge \mathcal{S}_{com_4}(\Gamma,\overline{\Gamma}))$$

$$\begin{aligned}
\mathcal{S}_{com_1} \triangleq{} & \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29} \in \mathbb{N} \,.\, \mathit{\Gamma} \in \Gamma \wedge \mathfrak{rf}\langle L0^{i}_{k_i},\ \langle \mathsf{30{:}},\\
& \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F0^{i},\ \langle \mathsf{29{:}},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle L1^{\ell}_{n_\ell},\\
& \langle \mathsf{0{:}},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F1^{\ell},\ \langle \mathsf{0{:}},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma}\\
\mathcal{S}_{com_2} \triangleq{} & \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_9 \in \mathbb{N} \,.\, \mathit{\Gamma} \in \Gamma \wedge \mathfrak{rf}\langle L0^{i}_{k_i},\ \langle \mathsf{30{:}},\\
& \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F0^{i},\ \langle \mathsf{29{:}},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle L1^{\ell}_{n_\ell},\\
& \langle \mathsf{0{:}},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F1^{\ell},\ \langle \mathsf{9{:}},\ i_9,\ 1\rangle\rangle \in \mathit{\Gamma}\\
\mathcal{S}_{com_3} \triangleq{} & \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10} \in \mathbb{N} \,.\, \mathit{\Gamma} \in \Gamma \wedge \mathfrak{rf}\langle L0^{i}_{k_i},\ \langle \mathsf{30{:}},\\
& \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F0^{i},\ \langle \mathsf{29{:}},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle L1^{\ell}_{n_\ell},\\
& \langle \mathsf{10{:}},\ i_{10},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F1^{\ell},\ \langle \mathsf{0{:}},\ \_,\ 1\rangle\rangle \in \mathit{\Gamma}\\
\mathcal{S}_{com_4} \triangleq{} & \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10}, i_9 \in \mathbb{N} \,.\, \mathit{\Gamma} \in \Gamma \wedge \mathfrak{rf}\langle L0^{i}_{k_i},\\
& \langle \mathsf{30{:}},\ \ell_{30},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F0^{i},\ \langle \mathsf{29{:}},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge\\
& \mathfrak{rf}\langle L1^{\ell}_{n_\ell},\ \langle \mathsf{10{:}},\ i_{10},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge \mathfrak{rf}\langle F1^{\ell},\ \langle \mathsf{9{:}},\ i_9,\ 1\rangle\rangle \in \mathit{\Gamma}
\end{aligned}$$

- This proves $\mathcal{S}_{com}$ sufficient for correctness
- Counter-examples prove $\mathcal{S}_{com}$ necessary ⇒ $\mathcal{S}_{com}$ is the weakest WCM requirement for correctness

Example of counter-example to S_{com_1}



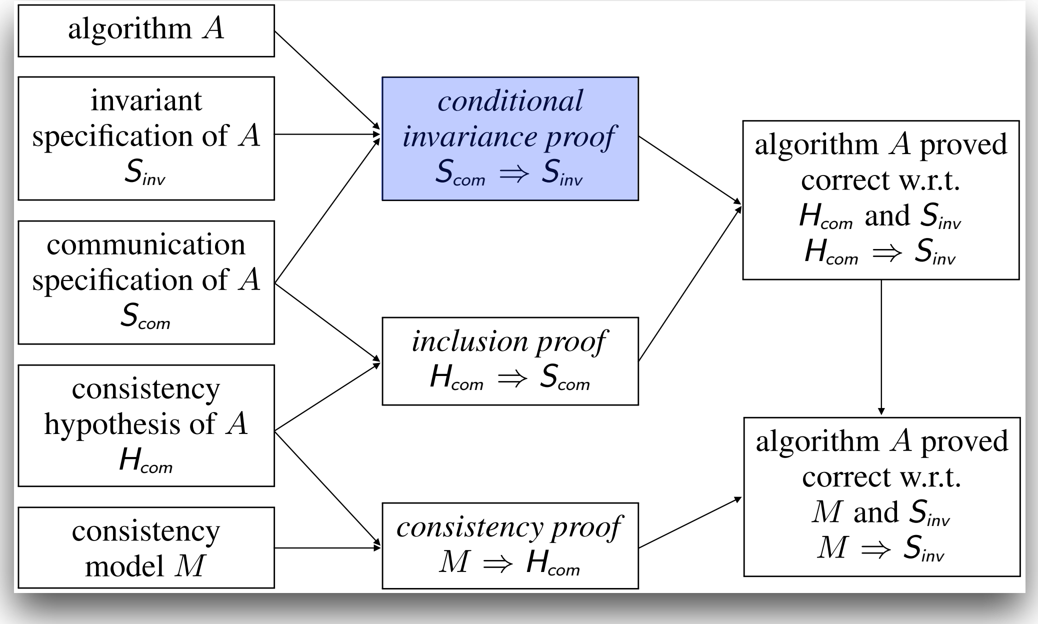
```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:                                        ||  21:
   do {$i$}                               ||     do {$\ell$}
2:                                        ||  22:
      do {$j_i$}                          ||        do {$m_\ell$}
3:                                        ||  23:
         r[] R10 latch0 {$\leadsto L0^i_{j_i}$}  ||  r[] R11 latch1 {$\leadsto L1^\ell_{m_\ell}$}
4:                                        ||  24:

      while (R10=0) {$k_i$}               ||        while (R11=0) {$n_\ell$}
5:                                        ||  25:
      w[] latch0 0                        ||        w[] latch1 0
6:                                        ||  26:
      r[] Rf0 flag0 {$\leadsto F0^i$}     ||        r[] Rf1 flag1 {$\leadsto F1^\ell$}
7:                                        ||  27:

      if (Rf0$\neq$0) then                ||        if (Rf1$\neq$0) then
8:                                        ||  28:
         (* critical section *)           ||           (* critical section *)
         w[] flag0 0                      ||           w[] flag1 0
9:                                        ||  29:
         w[] flag1 1                      ||           w[] flag0 1
10:                                       ||  30:
         w[] latch1 1                     ||           w[] latch0 1
11:                                       ||  31:
      fi                                  ||        fi
12:                                       ||  32:
   while true                             ||     while true
13:                                       ||  33:
```

Proof of mutual exclusion

- S_{com} implies mutual exclusion (for any Γ)

$$\begin{array}{lll}
 & (\neg S_{inv}(\Gamma,\Gamma) \wedge S_{ind}(\Gamma,\Gamma)) \Longrightarrow \neg(S_{com}(\Gamma,\Gamma)) & \\
\Longrightarrow & S_{com}(\Gamma,\Gamma) \Longrightarrow (S_{inv}(\Gamma,\Gamma) \vee \neg S_{ind}(\Gamma,\Gamma)) & \wr\text{contraposition}\wr \\
\Longrightarrow & S_{com}(\Gamma,\Gamma) \Longrightarrow (S_{ind}(\Gamma,\Gamma) \Longrightarrow S_{inv}(\Gamma,\Gamma)) & \wr\text{implication}\wr \\
\Longrightarrow & (S_{com}(\Gamma,\Gamma) \wedge S_{ind}(\Gamma,\Gamma)) \Longrightarrow S_{inv}(\Gamma,\Gamma) & \wr\text{implication}\wr \\
\Longrightarrow & S_{com}(\Gamma,\overline{\Gamma}) \Rightarrow S_{inv}(\Gamma,\overline{\Gamma}) & \wr\text{since } S_{com}(\Gamma,\overline{\Gamma}) \Rightarrow S_{ind}(\Gamma,\overline{\Gamma})\wr
\end{array}$$

Conditional invariance proof


algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Sequential proof $\ell = \kappa$ and $p = q$

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  $\{\varGamma \in \Gamma\}$
    do $\{i\}$
2:    $\{\varGamma \in \Gamma\}$
      do $\{j_i\}$
3:      $\{\varGamma \in \Gamma\}$
        r[] R10 lat
4:      $\{\varGamma \in \Gamma \wedge$ R10
      while (R10=0)
5:    $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma]\}$
      w[] latch0 0
6:    $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma]\}$
      r[] Rf0 flag0 $\{\leadsto F0^{i}\}$
7:    $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma] \wedge \mathtt{Rf0} = F0^{i}$
                    $\wedge (\mathrm{r0Rf0}^{i}[\varGamma] \vee \mathrm{r1Rf0}^{i}[\varGamma])\}$
      if (Rf0≠0) then
8:      $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma] \wedge \mathrm{r1Rf0}^{i}[\varGamma]\}$
        (* critical section *)
        w[] flag0 0
9:      $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma] \wedge \mathrm{r1Rf0}^{i}[\varGamma]\}$
        w[] flag1 1
10:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma] \wedge \mathrm{r1Rf0}^{i}[\varGamma]\}$
        w[] latch1 1
11:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\varGamma] \wedge \mathrm{r1Rf0}^{i}[\varGamma]\}$
      fi
12:   $\{\varGamma \in \Gamma\}$
    while true
13:$\{\mathsf{false}\}$
```

```
||
21:$\{\varGamma \in \Gamma\}$
25:   $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma]\}$
      w[] latch1 0
26:   $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma]\}$
      r[] Rf1 flag1 $\{\leadsto F1^{\ell}\}$
27:   $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma] \wedge \mathtt{Rf1} = F1^{\ell}$
                    $\wedge (\mathrm{r0Rf1}^{\ell}[\varGamma] \vee \mathrm{r1Rf1}^{\ell}[\varGamma])\}$
      if (Rf1≠0) then
28:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^{\ell}[\varGamma]\}$
        (* critical section *)
        w[] flag1 0
29:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^{\ell}[\varGamma]\}$
        w[] flag0 1
30:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^{\ell}[\varGamma]\}$
        w[] latch0 1
31:     $\{\varGamma \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\varGamma] \wedge \mathrm{r1Rf1}^{\ell}[\varGamma]\}$
      fi
32:   $\{\varGamma \in \Gamma\}$
    while true
33:$\{\mathsf{false}\}$
```

For a ***read instruction*** $\kappa : \mathtt{r[}\mathit{ts}\mathtt{]\ R\ x}\ \kappa'$: (read)

$$\mathsf{PRE}^{\ell,\kappa}_{p,r}[\theta_r, \rho_r, \nu_r, \mathsf{rf}] \wedge \mathit{rf}[\mathfrak{w}(\langle q,\ \ell',\ \mathtt{w[}\mathit{ts}\mathtt{]\ x}\ \textit{r-value},\ \theta'\rangle, v),\ \mathfrak{r}(\langle r,\ \ell,\ \mathtt{r[}\mathit{ts}\mathtt{]\ R\ x},\ \theta_r\rangle, \mathtt{x}_{\theta_r})] \in \mathsf{rf}$$

$$\Rightarrow \mathsf{POST}^{\ell,\kappa'}_{p,r}[\rho_r \leftarrow \rho_r[\mathtt{R} := \mathtt{x}_{\theta_r}], \nu_r \leftarrow \nu_r[\mathtt{x}_{\theta_r} := v]]$$

Sequential proof $\ell = \kappa$ and $p = q$

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {$\mathit{\Gamma} \in \Gamma$}
   do {$i$}
2:   {$\mathit{\Gamma} \in \Gamma$}
     do {$j_i$}
3:     {$\mathit{\Gamma} \in \Gamma$}
       r[] R10 lat
4:     {$\mathit{\Gamma} \in \Gamma \wedge$ R1
     while (R10=0)
5:   {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}$
     w[] latch0 0
6:   {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}]$}
     r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:   {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathtt{Rf0} = F0^{i}$
                        $\wedge (\mathrm{r0Rf0}^{i}[\mathit{\Gamma}] \vee \mathrm{r1Rf0}^{i}[\mathit{\Gamma}])$}
     if (Rf0≠0) then
8:     {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^{i}[\mathit{\Gamma}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^{i}[\mathit{\Gamma}]$}
       w[] flag1 1
10:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^{i}[\mathit{\Gamma}]$}
       w[] latch1 1
11:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^{i}_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^{i}[\mathit{\Gamma}]$}
     fi
12:  {$\mathit{\Gamma} \in \Gamma$}
   while true
13:{false}
```

```
21:{$\mathit{\Gamma} \in \Gamma$}
   do {$\ell$}
22:  {$\mathit{\Gamma} \in \Gamma$}
     do {$m_\ell$}
26:  {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}]$}
     r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:  {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathtt{Rf1} = F1^{\ell}$
                        $\wedge (\mathrm{r0Rf1}^{\ell}[\mathit{\Gamma}] \vee \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}])$}
     if (Rf1≠0) then
28:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]$}
       w[] flag0 1
30:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]$}
       w[] latch0 1
31:    {$\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]$}
     fi
32:  {$\mathit{\Gamma} \in \Gamma$}
   while true
33:{false}
```

For a ***test instruction*** κ : `b[`*ts*`]` *operation* l_t κ': (test)

$$\mathsf{PRE}^{\ell,\kappa}_{p,r}[\rho_r,\nu_r] \wedge \mathsf{sat}(E[\![\mathit{operation}]\!](\rho_r,\nu_r) \neq 0) \Rightarrow \mathsf{POST}^{\ell,l_t}_{p,r}$$

$$\mathsf{PRE}^{\ell,\kappa}_{p,r}[\rho_r,\nu_r] \wedge \mathsf{sat}(E[\![\mathit{operation}]\!](\rho_r,\nu_r) = 0) \Rightarrow \mathsf{POST}^{\ell,\kappa'}_{p,r}$$

Sequential proof $\ell = \kappa$ and $p = q$

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\mathit{\Gamma} \in \Gamma\}$
   do $\{i\}$
2:   $\{\mathit{\Gamma} \in \Gamma\}$
     do $\{j_i\}$
3:     $\{\mathit{\Gamma} \in \Gamma\}$
       r[] R10 latch0 $\{\leadsto L0^i_{j_i}\}$
4:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{R10} = L0^i_{j_i} \wedge (\mathrm{r0R10}^i_{j_i}[\mathit{\Gamma}] \vee \mathrm{r1R10}^i_{j_i}[\mathit{\Gamma}])\}$
     while (R10=0)
5:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i$
     w[] latch0 0
6:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i$
     r[] Rf0 flag0
7:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{Rf0} = F0$
                                   $\wedge\, (\mathrm{r0Rf0}^i[\mathit{\Gamma}] \vee \mathrm{r1Rf0}^i[\mathit{\Gamma}])\}$
     if (Rf0≠0) then
8:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       (* critical section *)
       w[] flag0 0
9:     $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       w[] flag1 1
10:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
       w[] latch1 1
11:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R10}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
     fi
12:  $\{\mathit{\Gamma} \in \Gamma\}$
   while true
13:$\{\mathsf{false}\}$
```

```
21:$\{\mathit{\Gamma} \in \Gamma\}$
   do $\{\ell\}$
22:  $\{\mathit{\Gamma} \in \Gamma\}$
     do $\{m_\ell\}$
23:    $\{\mathit{\Gamma} \in \Gamma\}$
       r[] R11 latch1 $\{\leadsto L1^\ell_{m_\ell}\}$
24:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{R11} = L1^\ell_{m_\ell} \wedge (\mathrm{r0R11}^\ell_{m_\ell}[\mathit{\Gamma}] \vee \mathrm{r1R11}^\ell_{m_\ell}[\mathit{\Gamma}])\}$
27:  $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{Rf1} = F1$
                                   $\wedge\, (\mathrm{r0Rf1}^\ell[\mathit{\Gamma}] \vee \mathrm{r1Rf1}^\ell[\mathit{\Gamma}])\}$
     if (Rf1≠0) then
28:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       (* critical section *)
       w[] flag1 0
29:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       w[] flag0 1
30:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
       w[] latch0 1
31:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1R11}^\ell_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
     fi
32:  $\{\mathit{\Gamma} \in \Gamma\}$
   while true
33:$\{\mathsf{false}\}$
```

For local side-effect free ***marker instructions*** κ : *instr* κ' where *instr* = `f` [*ts*] $\left[\{l_1^0 \ldots l_1^m\}\ \{l_2^0 \ldots l_2^q\}\ \right]$, `w` [*ts*] `x` *r-value*, `beginrmw` [*ts*] `x`, `endrmw` [*ts*] `x`: (marker)

$$\mathsf{PRE}^{\ell,\kappa}_{p,r} \Rightarrow \mathsf{POST}^{\ell,\kappa'}_{p,r}$$

Non-interference proof

The local invariants of process p depend only on Γ and local registers or Pythia variables unchanged by a step in the other process

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\Gamma \in \Gamma\}$
   do $\{i\}$
2:    $\{\Gamma \in \Gamma\}$
      do $\{j_i\}$
3:       $\{\Gamma \in \Gamma\}$
         r[] R10 latch0 $\{\rightsquigarrow L0$
4:       $\{\Gamma \in \Gamma \land \mathtt{R10} = L0^i_{j_i} \land ($r
      while (R10=0) $\{k_i\}$
5:    $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma]\}$
      w[] latch0 0
6:    $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma]\}$
      r[] Rf0 flag0 $\{\rightsquigarrow F0^i\}$
7:    $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma] \land \mathtt{Rf0} = F0^i$
                           $\land (\mathrm{r0Rf0}^i[\Gamma] \lor \mathrm{r1Rf0}^i[\Gamma])\}$
      if (Rf0≠0) then
8:       $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma] \land \mathrm{r1Rf0}^i[\Gamma]\}$
         (* critical section *)
         w[] flag0 0
9:       $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma] \land \mathrm{r1Rf0}^i[\Gamma]\}$
         w[] flag1 1
10:      $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma] \land \mathrm{r1Rf0}^i[\Gamma]\}$
         w[] latch1 1
11:      $\{\Gamma \in \Gamma \land \mathrm{r1R10}^i_{k_i}[\Gamma] \land \mathrm{r1Rf0}^i[\Gamma]\}$
      fi
12:   $\{\Gamma \in \Gamma\}$
   while true
13:{false}
```

```
26:   $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma]\}$
      r[] Rf1 flag1 $\{\rightsquigarrow F1^\ell\}$
27:   $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma] \land \mathtt{Rf1} = F1^\ell$
                           $\land (\mathrm{r0Rf1}^\ell[\Gamma] \lor \mathrm{r1Rf1}^\ell[\Gamma])\}$
      if (Rf1≠0) then
28:      $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma] \land \mathrm{r1Rf1}^\ell[\Gamma]\}$
         (* critical section *)
         w[] flag1 0
29:      $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma] \land \mathrm{r1Rf1}^\ell[\Gamma]\}$
         w[] flag0 1
30:      $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma] \land \mathrm{r1Rf1}^\ell[\Gamma]\}$
         w[] latch0 1
31:      $\{\Gamma \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\Gamma] \land \mathrm{r1Rf1}^\ell[\Gamma]\}$
      fi
32:   $\{\Gamma \in \Gamma\}$
   while true
33:{false}
```

Communication proof

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\mathit{\Gamma} \in \Gamma\}$
   do $\{i\}$
2:    $\{\mathit{\Gamma} \in \Gamma\}$
      do $\{j_i\}$
3:       $\{\mathit{\Gamma} \in \Gamma\}$
         r[] R10 latch0 $\{\leadsto\ L0$
4:       $\{\mathit{\Gamma} \in \Gamma \land \texttt{R10} = L0^i_{j_i} \land ($r
      while (R10=0) $\{k_i\}$
5:    $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      w[] latch0 0
6:    $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      r[] Rf0 flag0 $\{\leadsto\ F0^i\}$
7:    $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \land \texttt{Rf0} = F0^i$
                          $\land\ (\mathrm{r0Rf0}^i[\mathit{\Gamma}] \lor \mathrm{r1Rf0}^i[\mathit{\Gamma}])\}$
      if (Rf0≠0) then
8:       $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \land \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
         (* critical section *)
         w[] flag0 0
9:       $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \land \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
         w[] flag1 1
10:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \land \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
         w[] latch1 1
11:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \land \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
      fi
12:   $\{\mathit{\Gamma} \in \Gamma\}$
   while true
13:$\{\mathsf{false}\}$
```

• • •

```
26:   $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}]\}$
      r[] Rf1 flag1 $\{\leadsto\ F1^\ell\}$
27:   $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \land \texttt{Rf1} = F1^\ell$
                          $\land\ (\mathrm{r0Rf1}^\ell[\mathit{\Gamma}] \lor \mathrm{r1Rf1}^\ell[\mathit{\Gamma}])\}$
      if (Rf1≠0) then
28:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \land \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
         (* critical section *)
         w[] flag1 0
29:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \land \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
         w[] flag0 1
30:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \land \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
         w[] latch0 1
31:      $\{\mathit{\Gamma} \in \Gamma \land \mathrm{r1Rl1}^\ell_{n_\ell}[\mathit{\Gamma}] \land \mathrm{r1Rf1}^\ell[\mathit{\Gamma}]\}$
      fi
32:   $\{\mathit{\Gamma} \in \Gamma\}$
   while true
33:$\{\mathsf{false}\}$
```

- *Communication condition*

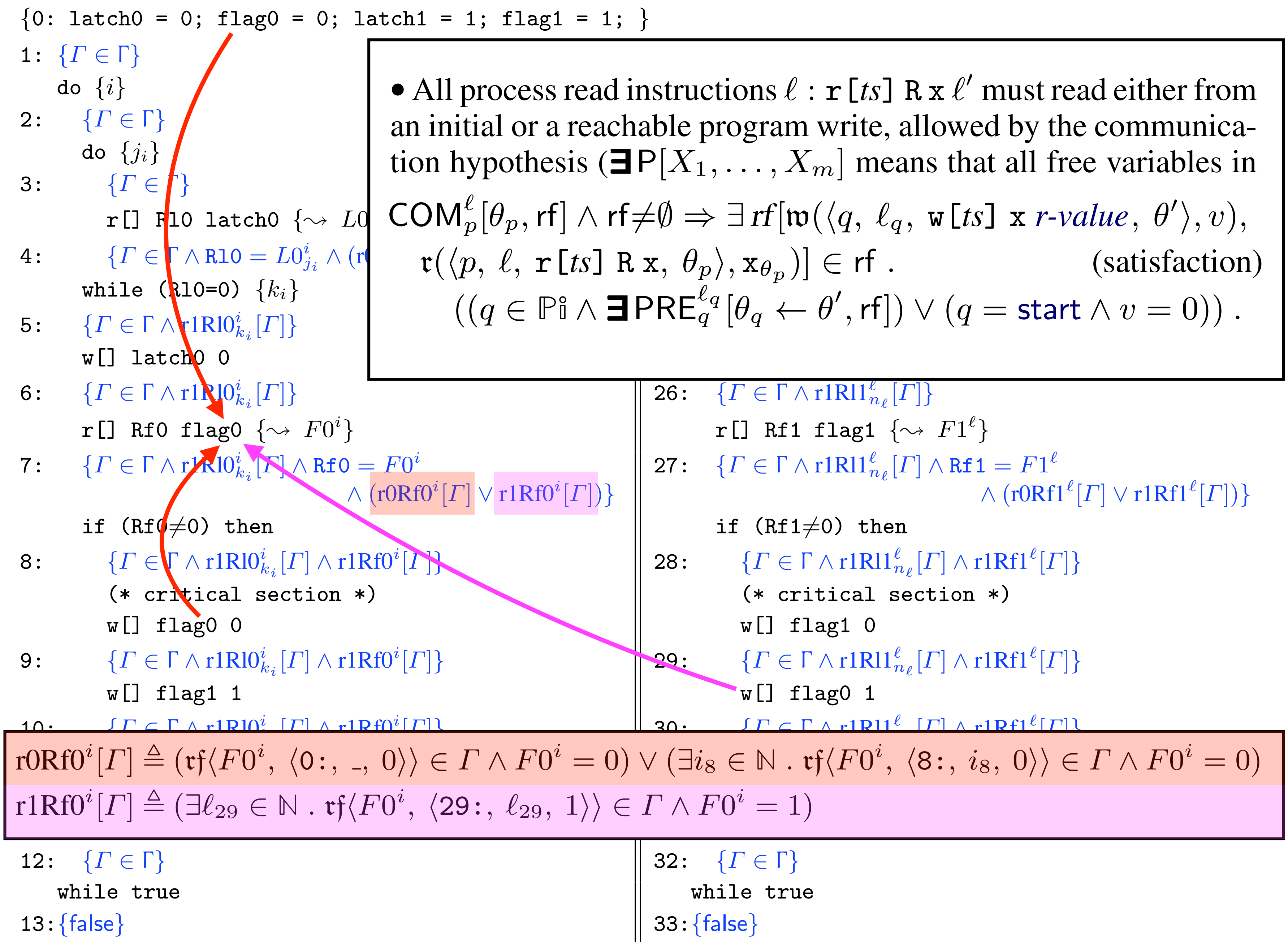
$$\mathsf{COM}^\ell_p[\mathsf{rf}] \quad \triangleq \quad \mathsf{S}_{\mathit{ind}\ p}(\ell)[\mathsf{rf}] \land \mathsf{S}_{\mathit{comp}_p}(\ell)[\mathsf{rf}]$$

- A read event can read from only one write event.

$$\mathsf{COM}^\ell_p[\mathsf{rf}] \land \mathit{rf}[r, w_1] \in \mathsf{rf} \land \mathit{rf}[r, w_2] \in \mathsf{rf} \qquad \text{(singleness)}$$

$$\Rightarrow w_1 = w_2\ .$$

Communication proof



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: ${\Gamma \in \Gamma}$
   do $\{i\}$
2:    ${\Gamma \in \Gamma}$
      do $\{j_i\}$
3:       ${\Gamma \in \Gamma}$
         r[] Rl0 latch0 $\{\rightsquigarrow L0$
4:       $\{\Gamma \in \Gamma \wedge \mathtt{Rl0} = L0^i_{j_i} \wedge (\mathrm{r}$
      while (Rl0=0) $\{k_i\}$
5:    $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma]\}$
      w[] latch0 0
6:    $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma]\}$
      r[] Rf0 flag0 $\{\rightsquigarrow F0^i\}$
7:    $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma] \wedge \mathtt{Rf0} = F0^i$
                            $\wedge (\mathrm{r0Rf0}^i[\Gamma] \vee \mathrm{r1Rf0}^i[\Gamma])\}$
      if (Rf0≠0) then
8:       $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma] \wedge \mathrm{r1Rf0}^i[\Gamma]\}$
         (* critical section *)
         w[] flag0 0
9:       $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma] \wedge \mathrm{r1Rf0}^i[\Gamma]\}$
         w[] flag1 1
10:      $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\Gamma] \wedge \mathrm{r1Rf0}^i[\Gamma]\}$
```

- All process read instructions $\ell : \mathtt{r}\,[ts]\,\mathtt{R}\,\mathtt{x}\,\ell'$ must read either from an initial or a reachable program write, allowed by the communication hypothesis ($\exists\,\mathsf{P}[X_1, \ldots, X_m]$ means that all free variables in

$$\mathsf{COM}^{\ell}_{p}[\theta_p, \mathsf{rf}] \wedge \mathsf{rf}\neq\emptyset \Rightarrow \exists\, rf[\mathfrak{w}(\langle q,\ \ell_q,\ \mathtt{w}\,[ts]\ \mathtt{x}\ \textit{r-value},\ \theta'\rangle, v),$$

$$\mathfrak{r}(\langle p,\ \ell,\ \mathtt{r}\,[ts]\ \mathtt{R}\ \mathtt{x},\ \theta_p\rangle, \mathtt{x}_{\theta_p})] \in \mathsf{rf}\ . \qquad \text{(satisfaction)}$$

$$((q \in \mathbb{P}\mathbb{i} \wedge \exists\,\mathsf{PRE}^{\ell_q}_{q}[\theta_q \leftarrow \theta', \mathsf{rf}]) \vee (q = \mathsf{start} \wedge v = 0))\ .$$

```
26:   $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma]\}$
      r[] Rf1 flag1 $\{\rightsquigarrow F1^{\ell}\}$
27:   $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma] \wedge \mathtt{Rf1} = F1^{\ell}$
                            $\wedge (\mathrm{r0Rf1}^{\ell}[\Gamma] \vee \mathrm{r1Rf1}^{\ell}[\Gamma])\}$
      if (Rf1≠0) then
28:      $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma]\}$
         (* critical section *)
         w[] flag1 0
29:      $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma]\}$
         w[] flag0 1
30:      $\{\Gamma \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma]\}$
```

$$\mathrm{r0Rf0}^i[\Gamma] \triangleq (\mathfrak{rf}\langle F0^i,\ \langle 0{:},\ \_,\ 0\rangle\rangle \in \Gamma \wedge F0^i = 0) \vee (\exists i_8 \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^i,\ \langle 8{:},\ i_8,\ 0\rangle\rangle \in \Gamma \wedge F0^i = 0)$$

$$\mathrm{r1Rf0}^i[\Gamma] \triangleq (\exists \ell_{29} \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^i,\ \langle 29{:},\ \ell_{29},\ 1\rangle\rangle \in \Gamma \wedge F0^i = 1)$$

```
12:   ${\Gamma \in \Gamma}$
   while true
13:{false}
```

```
32:   ${\Gamma \in \Gamma}$
   while true
33:{false}
```

Communication proof

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: $\{\mathit{\Gamma} \in \Gamma\}$
   do $\{i\}$
2:    $\{\mathit{\Gamma} \in \Gamma\}$
      do $\{j_i\}$
3:       $\{\mathit{\Gamma} \in \Gamma\}$
         r[] Rl0 latch0 $\{\leadsto L0$
4:       $\{\mathit{\Gamma} \in \Gamma \wedge \mathtt{Rl0} = L0^i_{j_i} \wedge (\mathrm{r}$
      while (Rl0=0) $\{k_i\}$
5:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      w[] latch0 0
6:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}]\}$
      r[] Rf0 flag0 $\{\leadsto F0^i\}$
7:    $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathtt{Rf0} = F0^i$
                 $\wedge (\mathrm{r0Rf0}^i[\mathit{\Gamma}] \vee \mathrm{r1Rf0}^i[\mathit{\Gamma}])\}$
      if (Rf0≠0) then
8:       $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
         (* critical section *)
         w[] flag0 0
9:       $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
         w[] flag1 1
10:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl0}^i_{k_i}[\mathit{\Gamma}] \wedge \mathrm{r1Rf0}^i[\mathit{\Gamma}]\}$
```

- The values v allowed to be read by the communication hypothesis must originate from reachable program write instructions ℓ : `w[`ts`] x` *r-value* ℓ':

$$\forall \mathsf{rf} \;.\; \forall\, rf[\mathfrak{w}(\langle q,\ \ell_q,\ \mathtt{w}[ts]\ \mathtt{x}\ \textit{r-value},\ \theta_p\rangle, v), r] \in \mathsf{rf}\ (\mathrm{match})$$

$$\mathsf{COM}^{\ell}_{p}[\theta_q, \rho_q, \nu_q, \mathsf{rf}] \Rightarrow v = \mathsf{E}[\![\textit{r-value}]\!](\rho_q, \nu_q)$$

```
26:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}]\}$
      r[] Rf1 flag1 $\{\leadsto F1^{\ell}\}$
27:   $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathtt{Rf1} = F1^{\ell}$
                 $\wedge (\mathrm{r0Rf1}^{\ell}[\mathit{\Gamma}] \vee \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}])\}$
      if (Rf1≠0) then
28:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]\}$
         (* critical section *)
         w[] flag1 0
29:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]\}$
         w[] flag0 1
30:      $\{\mathit{\Gamma} \in \Gamma \wedge \mathrm{r1Rl1}^{\ell}_{n_\ell}[\mathit{\Gamma}] \wedge \mathrm{r1Rf1}^{\ell}[\mathit{\Gamma}]\}$
```

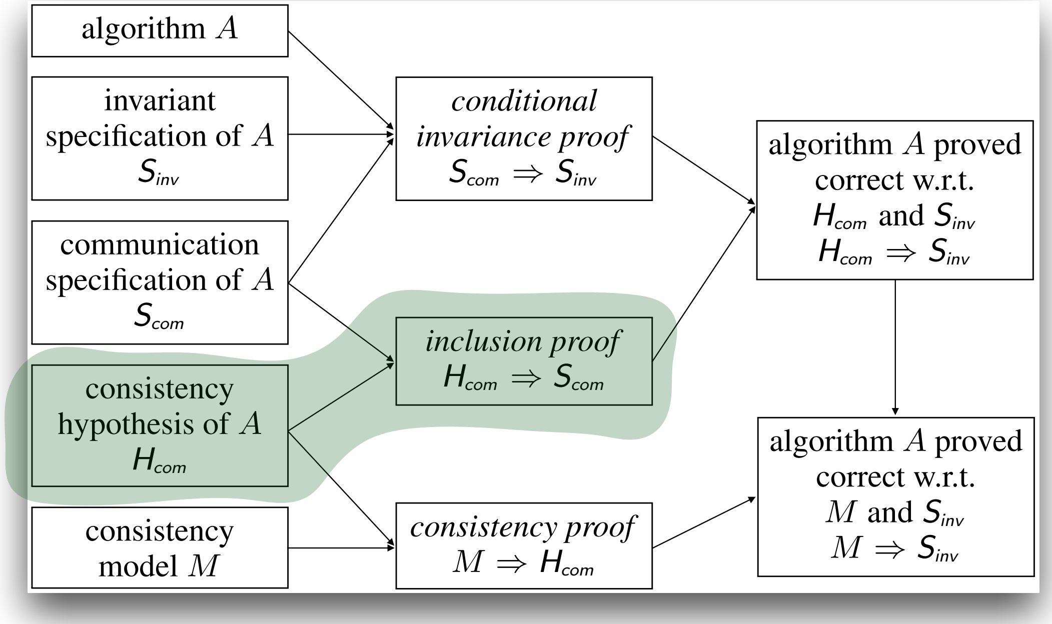
$$\mathrm{r0Rf0}^i[\mathit{\Gamma}] \triangleq (\mathfrak{rf}\langle F0^i,\ \langle 0{:},\ \_,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge F0^i = 0) \vee (\exists i_8 \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^i,\ \langle 8{:},\ i_8,\ 0\rangle\rangle \in \mathit{\Gamma} \wedge F0^i = 0)$$

$$\mathrm{r1Rf0}^i[\mathit{\Gamma}] \triangleq (\exists \ell_{29} \in \mathbb{N}\ .\ \mathfrak{rf}\langle F0^i,\ \langle 29{:},\ \ell_{29},\ 1\rangle\rangle \in \mathit{\Gamma} \wedge F0^i = 1)$$

```
12:   $\{\mathit{\Gamma} \in \Gamma\}$
   while true
13:$\{\mathsf{false}\}$
```

```
32:   $\{\mathit{\Gamma} \in \Gamma\}$
   while true
33:$\{\mathsf{false}\}$
```

Inclusion proof



algorithm A

invariant specification of A S_{inv}

communication specification of A S_{com}

consistency hypothesis of A H_{com}

consistency model M

conditional invariance proof $S_{com} \Rightarrow S_{inv}$

inclusion proof $H_{com} \Rightarrow S_{com}$

consistency proof $M \Rightarrow H_{com}$

algorithm A proved correct w.r.t. H_{com} and S_{inv} $H_{com} \Rightarrow S_{inv}$

algorithm A proved correct w.r.t. M and S_{inv} $M \Rightarrow S_{inv}$

Method

- The communication specification is

$$S_{com}(\Gamma,\overline{\Gamma}) \triangleq (\mathsf{at}\{8\} \wedge \mathsf{at}\{28\}) \Longrightarrow (S_{com_1}(\Gamma,\overline{\Gamma}) \wedge S_{com_2}(\Gamma,\overline{\Gamma}) \wedge S_{com_3}(\Gamma,\overline{\Gamma}) \wedge S_{com_4}(\Gamma,\overline{\Gamma}))$$

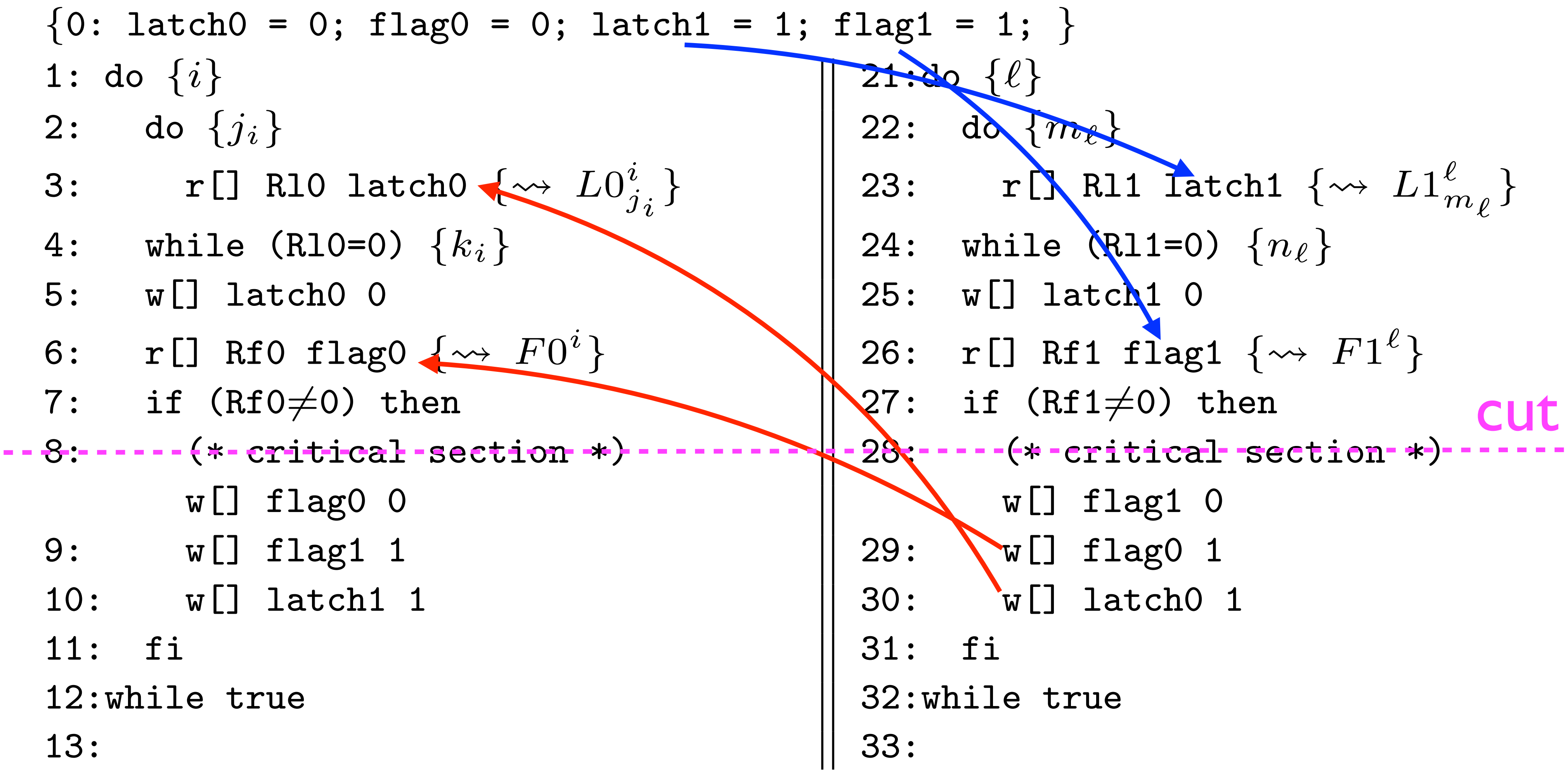
- The consistency specification must satisfy

$$H_{com}(\Gamma,\overline{\Gamma}) \Rightarrow S_{com}(\Gamma,\overline{\Gamma}) \qquad \text{i.e.} \qquad \neg S_{com}(\Gamma,\overline{\Gamma}) \Rightarrow \neg H_{com}(\Gamma,\overline{\Gamma})$$

- So the design of $H_{com}(\Gamma,\overline{\Gamma})$ must forbid the erroneous communications specified by the communication specification

$$\left(\mathsf{at}\{8\} \wedge \mathsf{at}\{28\} \wedge \bigvee_{i=1}^{4} \neg S_{com_i}(\Gamma,\overline{\Gamma})\right) \Longrightarrow \bigvee_{i=1}^{4} \neg H_{com\,i}(\Gamma,\overline{\Gamma})$$

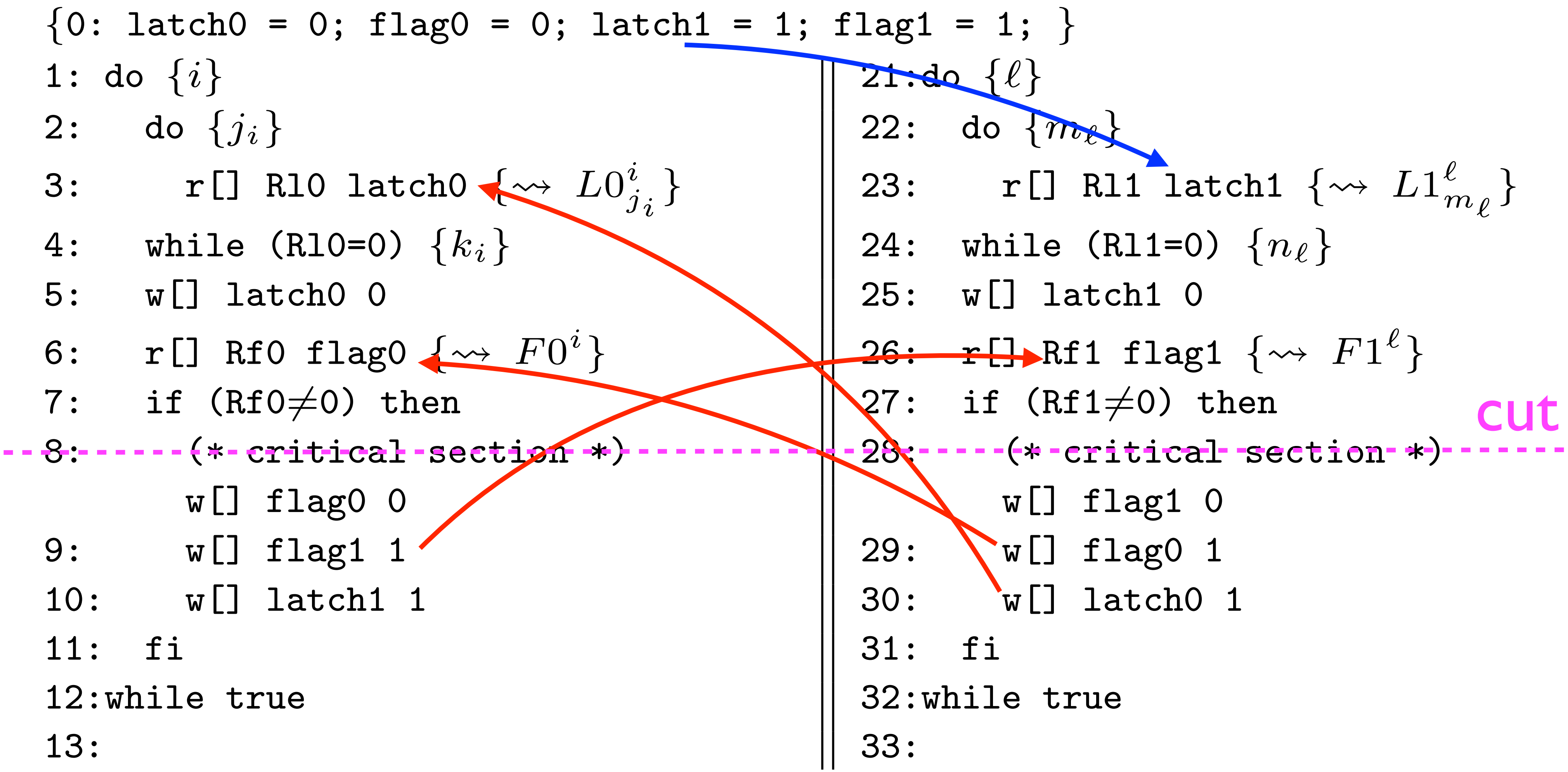
$$S_{com_1} \triangleq \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29} \in \mathbb{N} \;.\; \Gamma \in \Gamma \wedge \mathfrak{rf}\langle L0^i_{k_i},\ \langle 30\text{:},\ \ell_{30},\ 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F0^i,\ \langle 29\text{:},\ \ell_{29},\ 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle L1^\ell_{n_\ell},\ \langle 0\text{:},\ \_,\ 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F1^\ell,\ \langle 0\text{:},\ \_,\ 1\rangle\rangle \in \Gamma$$



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {i}                              || 21:do {ℓ}
2:   do {j_i}                          || 22:   do {m_ℓ}
3:     r[] Rl0 latch0 {⇝ L0^i_{j_i}}   || 23:     r[] Rl1 latch1 {⇝ L1^ℓ_{m_ℓ}}
4:   while (Rl0=0) {k_i}               || 24:   while (Rl1=0) {n_ℓ}
5:   w[] latch0 0                      || 25:   w[] latch1 0
6:   r[] Rf0 flag0 {⇝ F0^i}            || 26:   r[] Rf1 flag1 {⇝ F1^ℓ}
7:   if (Rf0≠0) then                   || 27:   if (Rf1≠0) then
8:     (* critical section *)          || 28:     (* critical section *)
       w[] flag0 0                     ||         w[] flag1 0
9:     w[] flag1 1                     || 29:     w[] flag0 1
10:    w[] latch1 1                    || 30:     w[] latch0 1
11:  fi                                || 31:   fi
12:while true                          || 32:while true
13:                                    || 33:
```

no prophecy beyond cut during execution

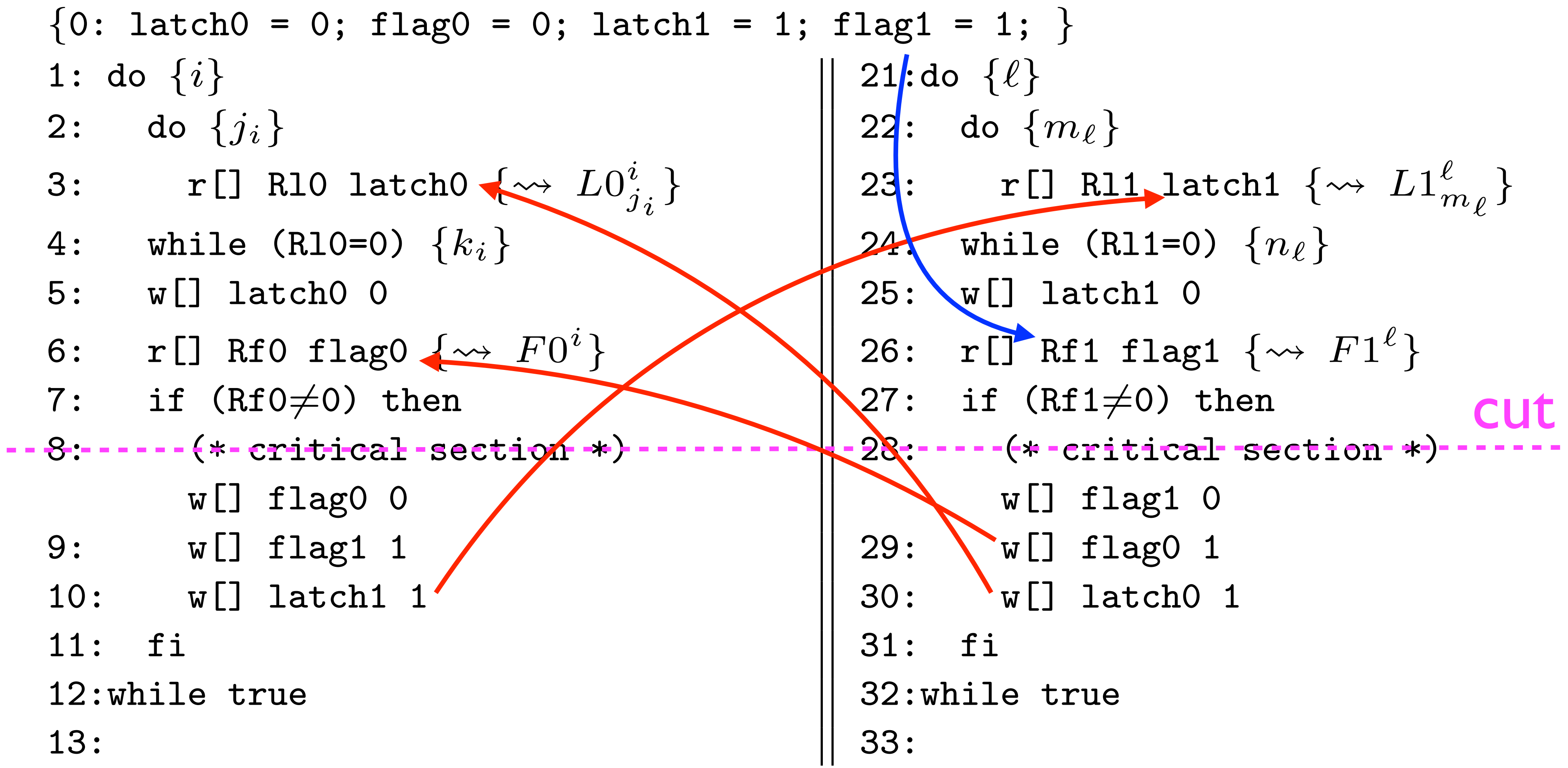
$$S_{com_2} \triangleq \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_9 \in \mathbb{N} \,.\, \Gamma \in \Gamma \wedge \mathfrak{rf}\langle L0^i_{k_i}, \langle 30\!:, \ell_{30}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F0^i, \langle 29\!:, \ell_{29}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle L1^\ell_{n_\ell}, \langle 0\!:, \_, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F1^\ell, \langle 9\!:, i_9, 1\rangle\rangle \in \Gamma$$



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}                                 || 21:do {$\ell$}
2:   do {$j_i$}                             || 22:  do {$m_\ell$}
3:     r[] Rl0 latch0 {$\leadsto L0^i_{j_i}$}   || 23:    r[] Rl1 latch1 {$\leadsto L1^\ell_{m_\ell}$}
4:   while (Rl0=0) {$k_i$}                  || 24:  while (Rl1=0) {$n_\ell$}
5:   w[] latch0 0                           || 25:  w[] latch1 0
6:   r[] Rf0 flag0 {$\leadsto F0^i$}          || 26:  r[] Rf1 flag1 {$\leadsto F1^\ell$}
7:   if (Rf0≠0) then                        || 27:  if (Rf1≠0) then
8:     (* critical section *)               || 28:    (* critical section *)
       w[] flag0 0                          ||        w[] flag1 0
9:     w[] flag1 1                          || 29:    w[] flag0 1
10:    w[] latch1 1                         || 30:    w[] latch0 1
11:  fi                                     || 31:  fi
12:while true                               || 32:while true
13:                                         || 33:
```

no prophecy beyond cut during execution

$$S_{com_3} \triangleq \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10} \in \mathbb{N} \,.\, \Gamma \in \Gamma \wedge \mathfrak{rf}\langle L0^i_{k_i}, \langle 30{:}, \ell_{30}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F0^i, \langle 29{:}, \ell_{29}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle L1^\ell_{n_\ell}, \langle 10{:}, i_{10}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F1^\ell, \langle 0{:}, \_, 1\rangle\rangle \in \Gamma$$

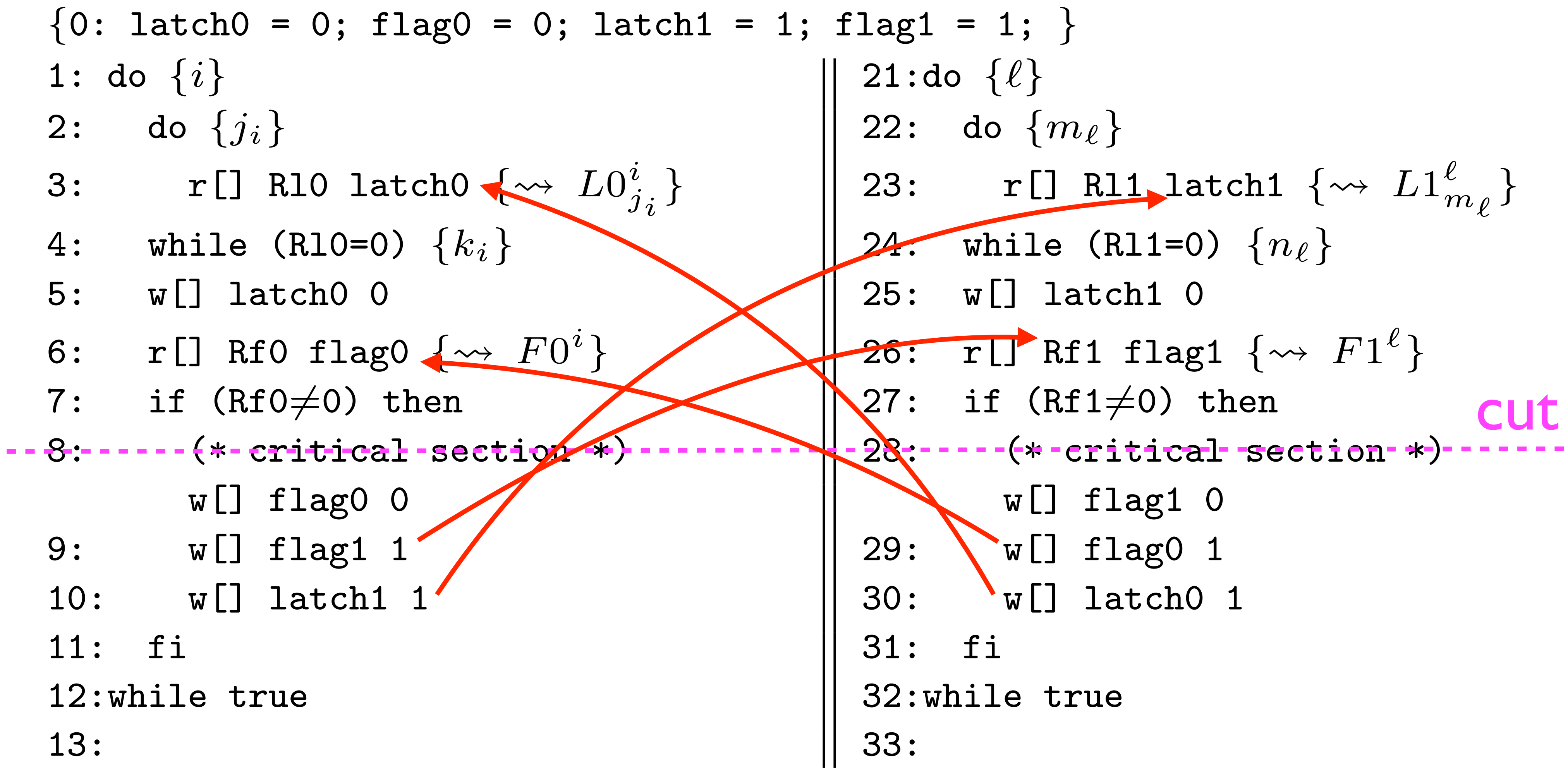


```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}
2:    do {$j_i$}
3:       r[] Rl0 latch0 {$\rightsquigarrow L0^i_{j_i}$}
4:    while (Rl0=0) {$k_i$}
5:    w[] latch0 0
6:    r[] Rf0 flag0 {$\rightsquigarrow F0^i$}
7:    if (Rf0≠0) then
8:       (* critical section *)
         w[] flag0 0
9:       w[] flag1 1
10:      w[] latch1 1
11:   fi
12:while true
13:
||
21:do {$\ell$}
22:   do {$m_\ell$}
23:      r[] Rl1 latch1 {$\rightsquigarrow L1^\ell_{m_\ell}$}
24:   while (Rl1=0) {$n_\ell$}
25:   w[] latch1 0
26:   r[] Rf1 flag1 {$\rightsquigarrow F1^\ell$}
27:   if (Rf1≠0) then
28:      (* critical section *)
         w[] flag1 0
29:      w[] flag0 1
30:      w[] latch0 1
31:   fi
32:while true
33:
```

cut

no prophecy beyond cut during execution

$$S_{com_4} \triangleq \neg(\exists i, k_i, \ell, n_\ell, \ell_{30}, \ell_{29}, i_{10}, i_9 \in \mathbb{N} \,.\, \Gamma \in \Gamma \wedge \mathfrak{rf}\langle L0^i_{k_i}, \langle 30:, \ell_{30}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F0^i, \langle 29:, \ell_{29}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle L1^\ell_{n_\ell}, \langle 10:, i_{10}, 1\rangle\rangle \in \Gamma \wedge \mathfrak{rf}\langle F1^\ell, \langle 9:, i_9, 1\rangle\rangle \in \Gamma$$

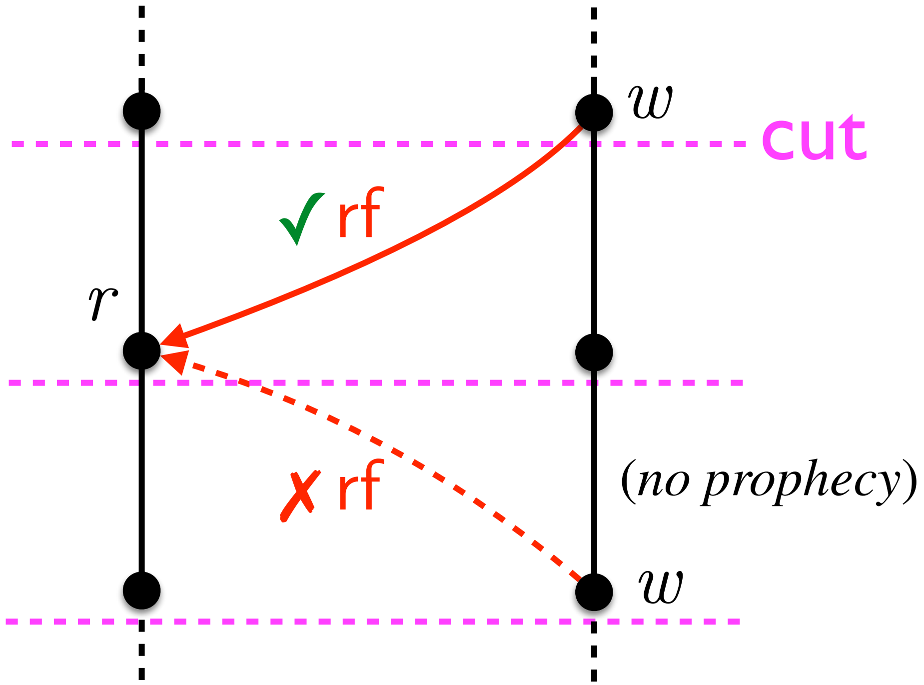


```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  do $\{i\}$
2:    do $\{j_i\}$
3:      r[] Rl0 latch0 $\{\leadsto L0^i_{j_i}\}$
4:    while (Rl0=0) $\{k_i\}$
5:    w[] latch0 0
6:    r[] Rf0 flag0 $\{\leadsto F0^i\}$
7:    if (Rf0≠0) then
8:      (* critical section *)
        w[] flag0 0
9:      w[] flag1 1
10:     w[] latch1 1
11:   fi
12:while true
13:
||
21:do $\{\ell\}$
22:   do $\{m_\ell\}$
23:     r[] Rl1 latch1 $\{\leadsto L1^\ell_{m_\ell}\}$
24:   while (Rl1=0) $\{n_\ell\}$
25:   w[] latch1 0
26:   r[] Rf1 flag1 $\{\leadsto F1^\ell\}$
27:   if (Rf1≠0) then
28:     (* critical section *)
        w[] flag1 0
29:     w[] flag0 1
30:     w[] latch0 1
31:   fi
32:while true
33:
```

no prophecy beyond cut during execution

Conclusion on mutual exclusion

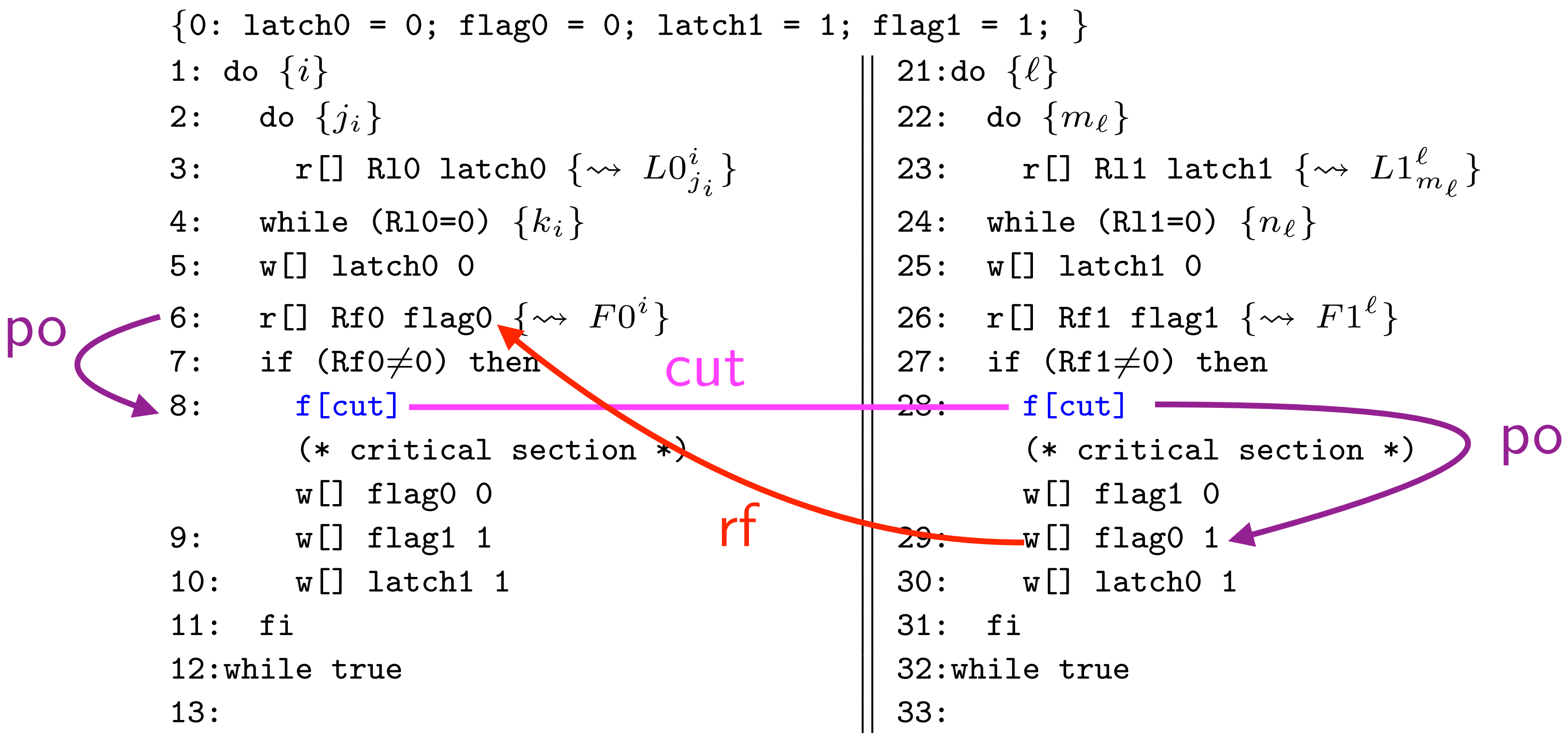
- PostgreSQL is correct on architectures satisfying the ``no prophecy beyond cut during execution'' property



- Intuition on necessity: when waiting for a spinlock, you should look at its current value, not at later ones!

in cat

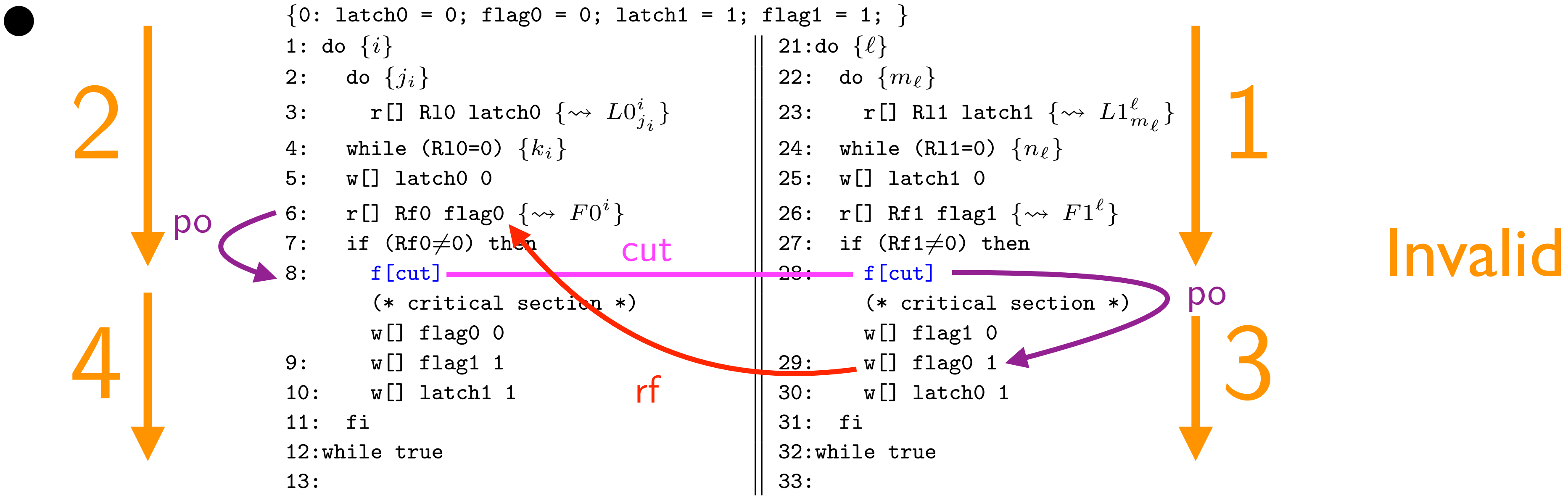
- A static condition to impose a dynamic condition:



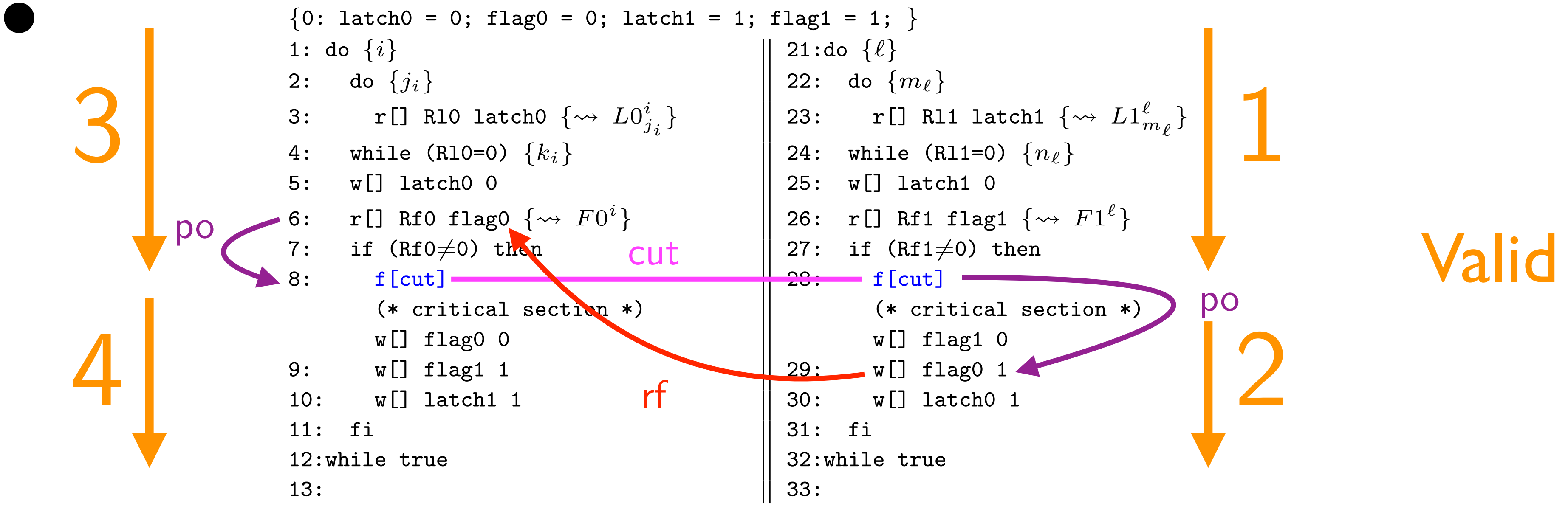
```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {$i$}
2:   do {$j_i$}
3:     r[] R10 latch0 {$\leadsto L0^i_{j_i}$}
4:   while (R10=0) {$k_i$}
5:   w[] latch0 0
6:   r[] Rf0 flag0 {$\leadsto F0^i$}
7:   if (Rf0$\neq$0) then
8:     f[cut]
       (* critical section *)
       w[] flag0 0
9:     w[] flag1 1
10:    w[] latch1 1
11:  fi
12:while true
13:
||
21:do {$\ell$}
22:  do {$m_\ell$}
23:    r[] R11 latch1 {$\leadsto L1^\ell_{m_\ell}$}
24:  while (R11=0) {$n_\ell$}
25:  w[] latch1 0
26:  r[] Rf1 flag1 {$\leadsto F1^\ell$}
27:  if (Rf1$\neq$0) then
28:    f[cut]
       (* critical section *)
       w[] flag1 0
29:    w[] flag0 1
30:    w[] latch0 1
31:  fi
32:while true
33:
```

```
enum fences = 'cut
instructions F[{'cut}]
let cut = (tag2events('cut) * tag2events('cut)) & ext
irreflexive rf; po; cut; po
```

Prevents valid executions



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {i}                                 || 21:do {ℓ}
2:   do {j_i}                             || 22:  do {m_ℓ}
3:     r[] R10 latch0 {⇝ L0^i_{j_i}}      || 23:    r[] R11 latch1 {⇝ L1^ℓ_{m_ℓ}}
4:   while (R10=0) {k_i}                  || 24:  while (R11=0) {n_ℓ}
5:   w[] latch0 0                         || 25:  w[] latch1 0
6:   r[] Rf0 flag0 {⇝ F0^i}               || 26:  r[] Rf1 flag1 {⇝ F1^ℓ}
7:   if (Rf0≠0) then                      || 27:  if (Rf1≠0) then
8:     f[cut]                             || 28:    f[cut]
       (* critical section *)             ||        (* critical section *)
       w[] flag0 0                        ||        w[] flag1 0
9:     w[] flag1 1                        || 29:    w[] flag0 1
10:    w[] latch1 1                       || 30:    w[] latch0 1
11:  fi                                   || 31:  fi
12:while true                             || 32:while true
13:                                       || 33:
```



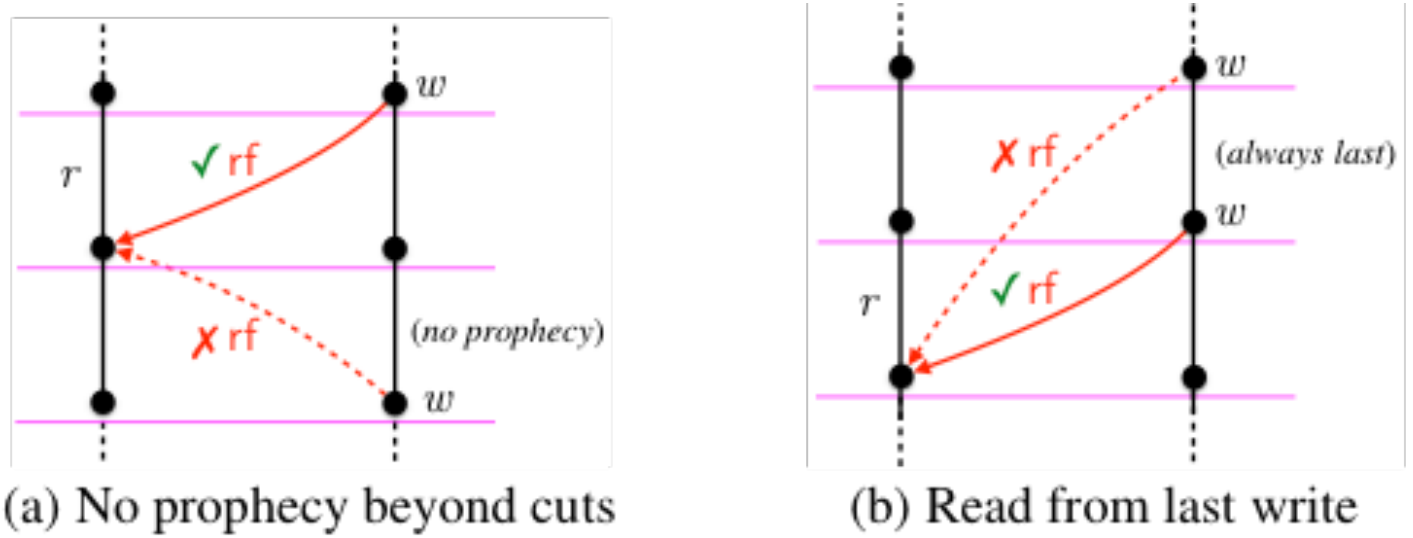
```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: do {i}                                 || 21:do {ℓ}
2:   do {j_i}                             || 22:  do {m_ℓ}
3:     r[] R10 latch0 {⇝ L0^i_{j_i}}      || 23:    r[] R11 latch1 {⇝ L1^ℓ_{m_ℓ}}
4:   while (R10=0) {k_i}                  || 24:  while (R11=0) {n_ℓ}
5:   w[] latch0 0                         || 25:  w[] latch1 0
6:   r[] Rf0 flag0 {⇝ F0^i}               || 26:  r[] Rf1 flag1 {⇝ F1^ℓ}
7:   if (Rf0≠0) then                      || 27:  if (Rf1≠0) then
8:     f[cut]                             || 28:    f[cut]
       (* critical section *)             ||        (* critical section *)
       w[] flag0 0                        ||        w[] flag1 0
9:     w[] flag1 1                        || 29:    w[] flag0 1
10:    w[] latch1 1                       || 30:    w[] latch0 1
11:  fi                                   || 31:  fi
12:while true                             || 32:while true
13:                                       || 33:
```

`irreflexive rf; po; cut; po`

Non-starvation

Difference with Lamport/Owicki-Gries

- The communications in L/O-G are fixed in the semantics (SC) for all executions:



(a) No prophecy beyond cuts (b) Read from last write

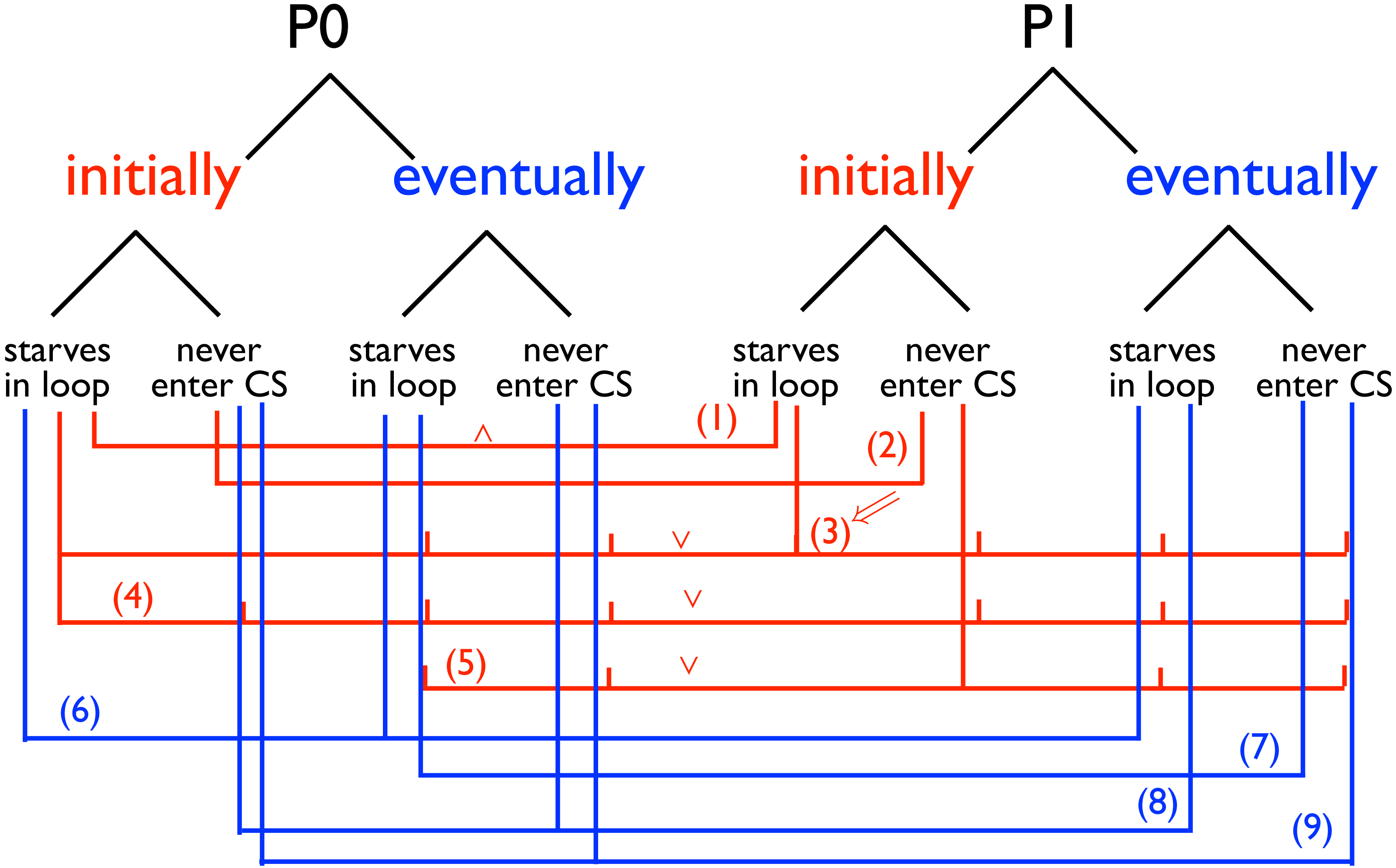
⟹ entangled with the verification conditions

⟹ impossible to reason on one execution trace only

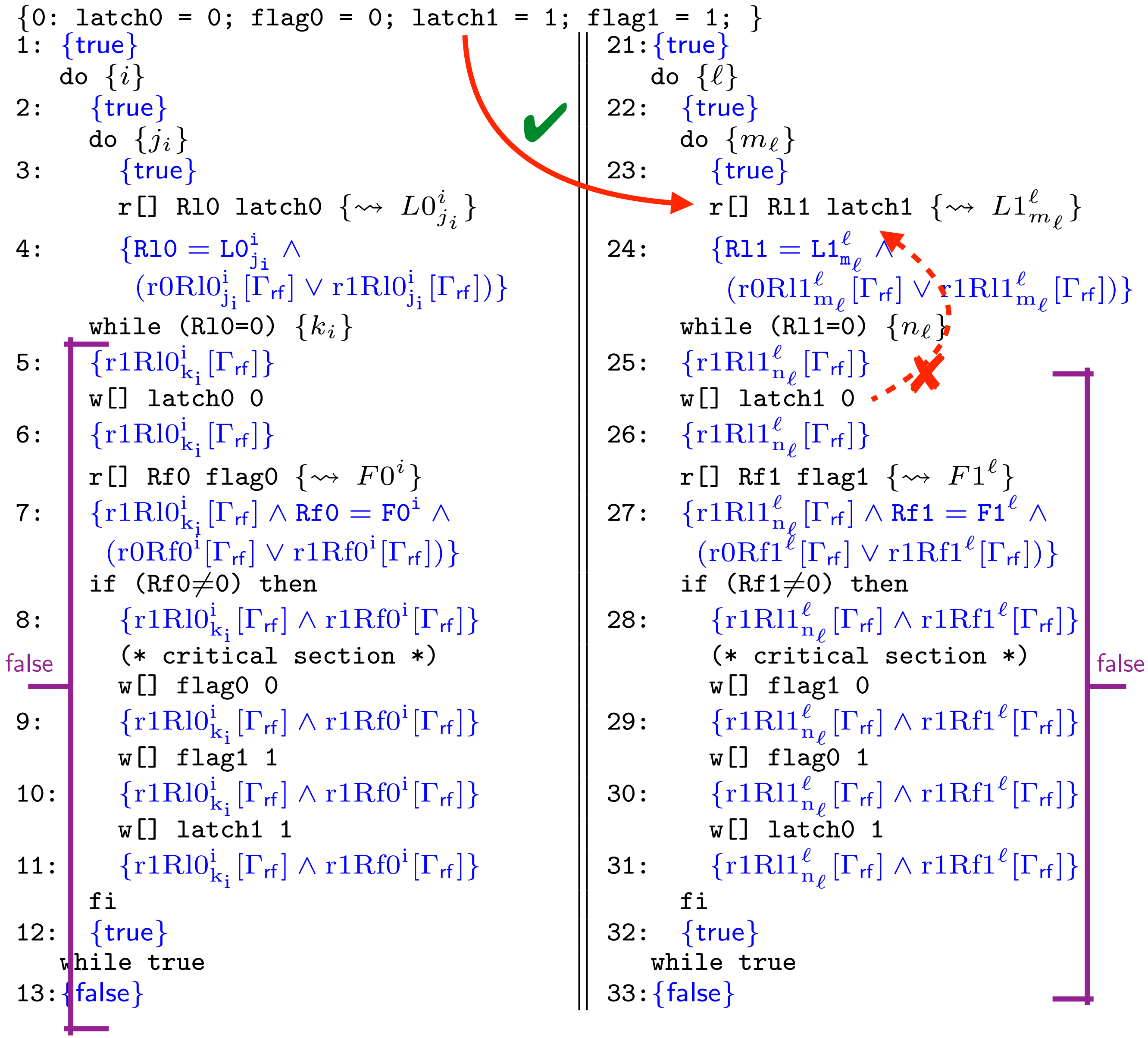
Reasoning on only one execution

- An execution is entirely determined by its read-from relation rf
- The verification conditions depend on a set Γ of verification conditions
- By choosing Γ = {rf}, we can reason on this execution
- This execution satisfies the inductive invariant $S_{ind}(\{rf\})$
- To prove that this execution is impossible it is sufficient to prove that $S_{ind}(\{rf\})$ cannot hold (according to the verification conditions)
- Since the method is sound, if the verification conditions are not satisfied, the execution is excluded by the semantics

9 cases of starvation



(1) Both processes starve in spin loops



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] R10 latch0 {$\rightsquigarrow L0^{i}_{j_i}$}
4:     {$\texttt{R10} = \texttt{L0}^{i}_{j_i} \wedge$
         $(\text{r0R10}^{i}_{j_i}[\Gamma_{rf}] \vee \text{r1R10}^{i}_{j_i}[\Gamma_{rf}])$}
     while (R10=0) {$k_i$}
5:   {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:   {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \texttt{Rf0} = \texttt{F0}^{i} \wedge$
      $(\text{r0Rf0}^{i}[\Gamma_{rf}] \vee \text{r1Rf0}^{i}[\Gamma_{rf}])$}
     if (Rf0≠0) then
8:     {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] flag1 1
10:    {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$\text{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] R11 latch1 {$\rightsquigarrow L1^{\ell}_{m_\ell}$}
24:    {$\texttt{R11} = \texttt{L1}^{\ell}_{m_\ell} \wedge$
         $(\text{r0R11}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \text{r1R11}^{\ell}_{m_\ell}[\Gamma_{rf}])$}
     while (R11=0) {$n_\ell$}
25:  {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:  {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \texttt{Rf1} = \texttt{F1}^{\ell} \wedge$
      $(\text{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \text{r1Rf1}^{\ell}[\Gamma_{rf}])$}
     if (Rf1≠0) then
28:    {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] flag0 1
30:    {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$\text{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- invariant false after both spin loops
- so `latch1` in 23: can only be read from initialization
- so `latch1` is 1 not 0, a contradiction

(2) Both processes never enter their critical section

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] Rl0 latch0 {$\leadsto L0^{i}_{j_i}$}
4:     {$\texttt{Rl0} = \texttt{L0}^{i}_{j_i} \wedge$
        $(\text{r0Rl0}^{i}_{j_i}[\Gamma_{rf}] \vee \text{r1Rl0}^{i}_{j_i}[\Gamma_{rf}])$}
     while (Rl0=0) {$k_i$}
5:   {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\leadsto F0^{i}$}
7:   {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \texttt{Rf0} = \texttt{F0}^{i} \wedge$
      $(\text{r0Rf0}^{i}[\Gamma_{rf}] \vee \text{r1Rf0}^{i}[\Gamma_{rf}])$}
     if (Rf0≠0) then
8:     {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] flag1 1
10:    {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$\text{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \text{r1Rf0}^{i}[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] Rl1 latch1 {$\leadsto L1^{\ell}_{m_\ell}$}
24:    {$\texttt{Rl1} = \texttt{L1}^{\ell}_{m_\ell} \wedge$
        $(\text{r0Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \text{r1Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}])$}
     while (Rl1=0) {$n_\ell$}
25:  {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\leadsto F1^{\ell}$}
27:  {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \texttt{Rf1} = \texttt{F1}^{\ell} \wedge$
      $(\text{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \text{r1Rf1}^{\ell}[\Gamma_{rf}])$}
     if (Rf1≠0) then
28:    {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] flag0 1
30:    {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$\text{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \text{r1Rf1}^{\ell}[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})

(2) Both processes never enter their critical section

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] Rl0 latch0 {$\rightsquigarrow L0^{i}_{j_i}$}
4:     {$\mathtt{Rl0} = \mathtt{L0}^{i}_{j_i} \wedge$
        $(\mathrm{r0Rl0}^{i}_{j_i}[\Gamma_{rf}] \vee \mathrm{r1Rl0}^{i}_{j_i}[\Gamma_{rf}])$}
     while (Rl0=0) {$k_i$}
5:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathtt{Rf0} = \mathtt{F0}^{i} \wedge$
      $(\mathrm{r0Rf0}^{i}[\Gamma_{rf}] \vee \mathrm{r1Rf0}^{i}[\Gamma_{rf}])$}
     if (Rf0≠0) then
8:     {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
 false
       w[] flag1 1
10:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13:{false}
||
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] Rl1 latch1 {$\rightsquigarrow L1^{\ell}_{m_\ell}$}
24:    {$\mathtt{Rl1} = \mathtt{L1}^{\ell}_{m_\ell} \wedge$
        $(\mathrm{r0Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \mathrm{r1Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}])$}
     while (Rl1=0) {$n_\ell$}
25:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
      $(\mathrm{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}])$}
     if (Rf1≠0) then
28:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
                                                  false
       w[] flag0 1
30:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant inside critical sections must be false

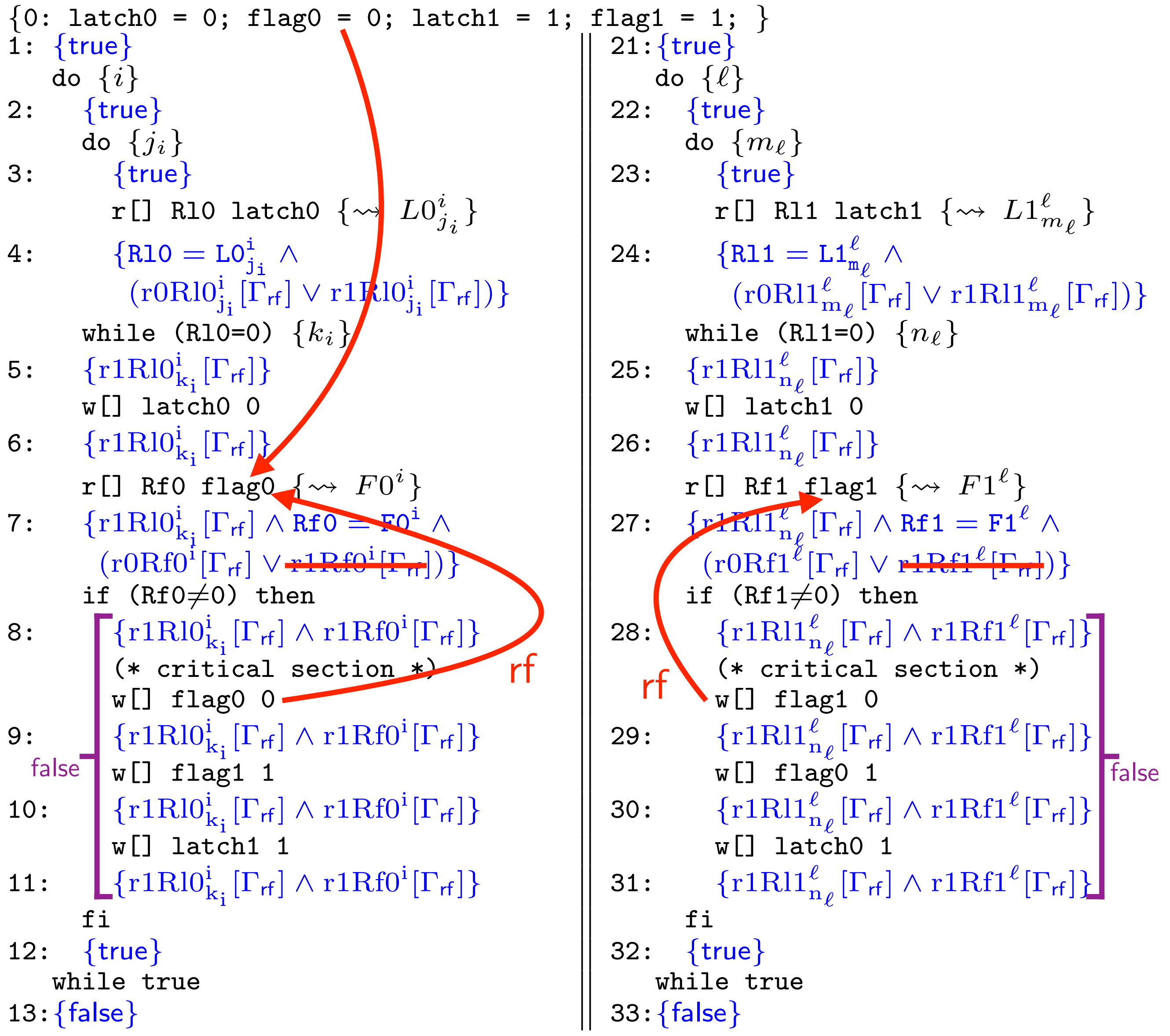
(2) Both processes never enter their critical section

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] R10 latch0 {$\leadsto L0^{i}_{j_i}$}
4:     {$\mathtt{R10} = \mathtt{L0}^{i}_{j_i} \wedge$
         $(\mathrm{r0R10}^{i}_{j_i}[\Gamma_{rf}] \vee \mathrm{r1R10}^{i}_{j_i}[\Gamma_{rf}])$}
     while (R10=0) {$k_i$}
5:   {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\leadsto F0^{i}$}
7:   {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathtt{Rf0} = \mathtt{F0}^{i} \wedge$
      $(\mathrm{r0Rf0}^{i}[\Gamma_{rf}] \vee \cancel{\mathrm{r1Rf0}^{i}[\Gamma_{rf}]})$}
     if (Rf0≠0) then
8:     {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
  false
       w[] flag1 1
10:    {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$\mathrm{r1R10}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13: {false}
```

```
21: {true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] R11 latch1 {$\leadsto L1^{\ell}_{m_\ell}$}
24:    {$\mathtt{R11} = \mathtt{L1}^{\ell}_{m_\ell} \wedge$
         $(\mathrm{r0R11}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \mathrm{r1R11}^{\ell}_{m_\ell}[\Gamma_{rf}])$}
     while (R11=0) {$n_\ell$}
25:  {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\leadsto F1^{\ell}$}
27:  {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
      $(\mathrm{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \cancel{\mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]})$}
     if (Rf1≠0) then
28:    {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] flag0 1                                   false
30:    {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$\mathrm{r1R11}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33: {false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant inside critical sections must be false
- tests ($\mathtt{Rf0}{\neq}0$) and ($\mathtt{Rf1}{\neq}0$) must be false (written ~~xxx~~)

(2) Both processes never enter their critical section

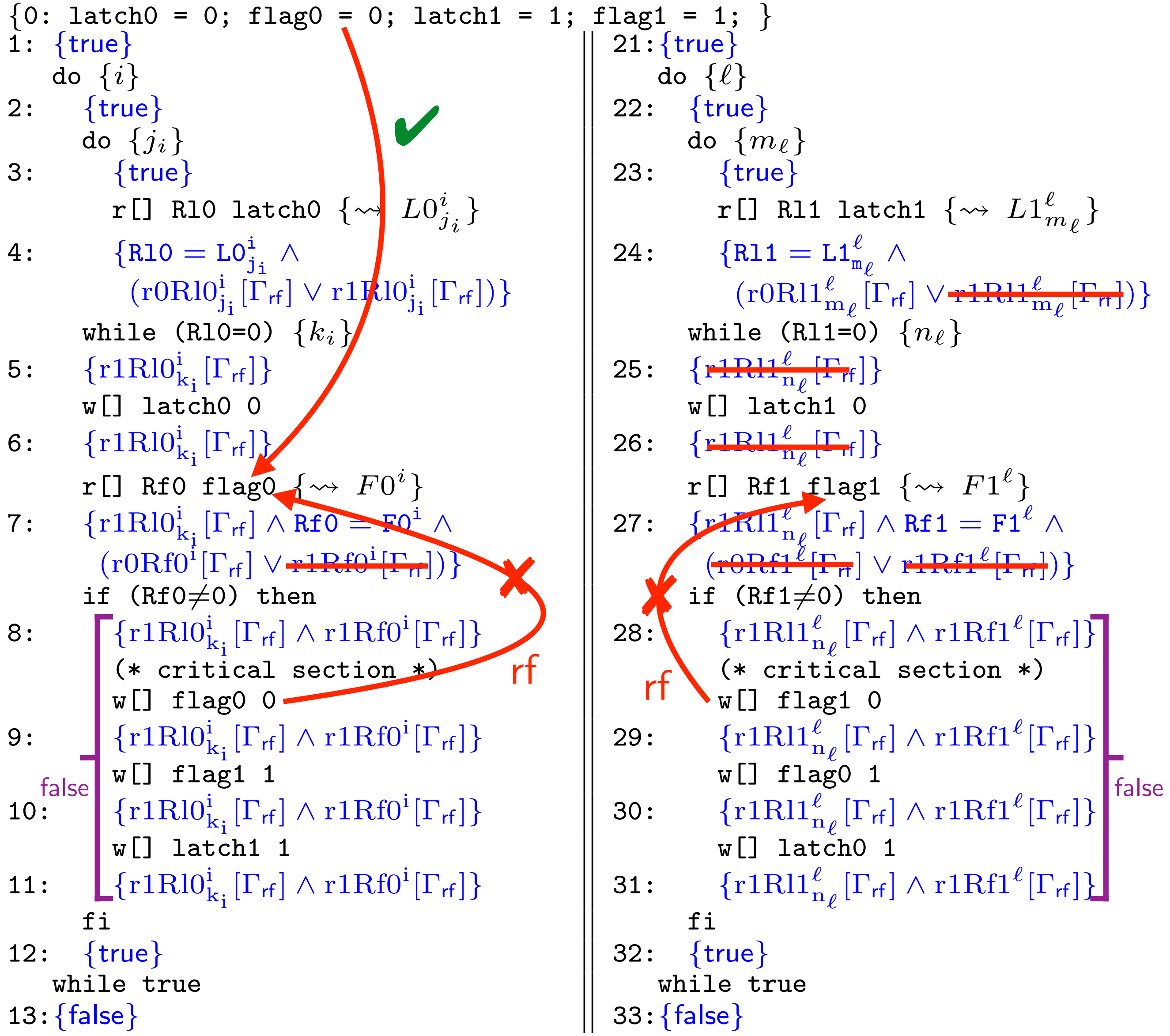


```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] R10 latch0 {$\leadsto L0^i_{j_i}$}
4:     {$R10 = L0^i_{j_i} \wedge$
         $(r0R10^i_{j_i}[\Gamma_{rf}] \vee r1R10^i_{j_i}[\Gamma_{rf}])$}
     while (R10=0) {$k_i$}
5:   {$r1R10^i_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$r1R10^i_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\leadsto F0^i$}
7:   {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge Rf0 = F0^i \wedge$
       $(r0Rf0^i[\Gamma_{rf}] \vee \cancel{r1Rf0^i[\Gamma_{rf}]})$}
     if (Rf0≠0) then
8:     {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
       w[] flag1 1
10:    {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] R11 latch1 {$\leadsto L1^\ell_{m_\ell}$}
24:    {$R11 = L1^\ell_{m_\ell} \wedge$
         $(r0R11^\ell_{m_\ell}[\Gamma_{rf}] \vee r1R11^\ell_{m_\ell}[\Gamma_{rf}])$}
     while (R11=0) {$n_\ell$}
25:  {$r1R11^\ell_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$r1R11^\ell_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\leadsto F1^\ell$}
27:  {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge Rf1 = F1^\ell \wedge$
       $(r0Rf1^\ell[\Gamma_{rf}] \vee \cancel{r1Rf1^\ell[\Gamma_{rf}]})$}
     if (Rf1≠0) then
28:    {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
       w[] flag0 1
30:    {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant inside critical sections must be false
- tests (`Rf0`≠`0`) and (`Rf1`≠`0`) must be false (written ~~xxx~~)
- so read of `Rf0` and `Rf1` is 0 from a reachable write

(2) Both processes never enter their critical section



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  {true}
    do {$i$}
2:    {true}
      do {$j_i$}
3:      {true}
        r[] Rl0 latch0 {$\rightsquigarrow L0^{i}_{j_i}$}
4:      {$\mathtt{Rl0} = \mathtt{L0}^{\mathtt{i}}_{\mathtt{j_i}} \wedge$
         $(\mathrm{r0Rl0}^{\mathtt{i}}_{\mathtt{j_i}}[\Gamma_{\mathsf{rf}}] \vee \mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{j_i}}[\Gamma_{\mathsf{rf}}])$}
      while (Rl0=0) {$k_i$}
5:    {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}]$}
      w[] latch0 0
6:    {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}]$}
      r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:    {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}] \wedge \mathtt{Rf0} = \mathtt{F0}^{\mathtt{i}} \wedge$
       $(\mathrm{r0Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}] \vee \cancel{\mathrm{r1Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}]})$}
      if (Rf0≠0) then
8:      {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}]$}
        (* critical section *)
        w[] flag0 0
9:      {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}]$}
        w[] flag1 1
10:     {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}]$}
        w[] latch1 1
11:     {$\mathrm{r1Rl0}^{\mathtt{i}}_{\mathtt{k_i}}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf0}^{\mathtt{i}}[\Gamma_{\mathsf{rf}}]$}
      fi
12:   {true}
    while true
13: {false}

||

21: {true}
    do {$\ell$}
22:   {true}
      do {$m_\ell$}
23:     {true}
        r[] Rl1 latch1 {$\rightsquigarrow L1^{\ell}_{m_\ell}$}
24:     {$\mathtt{Rl1} = \mathtt{L1}^{\ell}_{\mathtt{m}_\ell} \wedge$
         $(\mathrm{r0Rl1}^{\ell}_{\mathrm{m}_\ell}[\Gamma_{\mathsf{rf}}] \vee \cancel{\mathrm{r1Rl1}^{\ell}_{\mathrm{m}_\ell}[\Gamma_{\mathsf{rf}}]})$}
      while (Rl1=0) {$n_\ell$}
25:   {$\cancel{\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}]}$}
      w[] latch1 0
26:   {$\cancel{\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}]}$}
      r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:   {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
       $(\cancel{\mathrm{r0Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]} \vee \cancel{\mathrm{r1Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]})$}
      if (Rf1≠0) then
28:     {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]$}
        (* critical section *)
        w[] flag1 0
29:     {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]$}
        w[] flag0 1
30:     {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]$}
        w[] latch0 1
31:     {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathsf{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathsf{rf}}]$}
      fi
32:   {true}
    while true
33: {false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant inside critical sections must be false
- tests (`Rf0`≠`0`) and (`Rf1`≠`0`) must be false (written ~~xxx~~)
- so read of `Rf0` and `Rf1` is 0 from a reachable write
- impossible for `Rf1` so loop 23—24 is never exited

⟹ we are in case (3), P1 stuck in spin loop

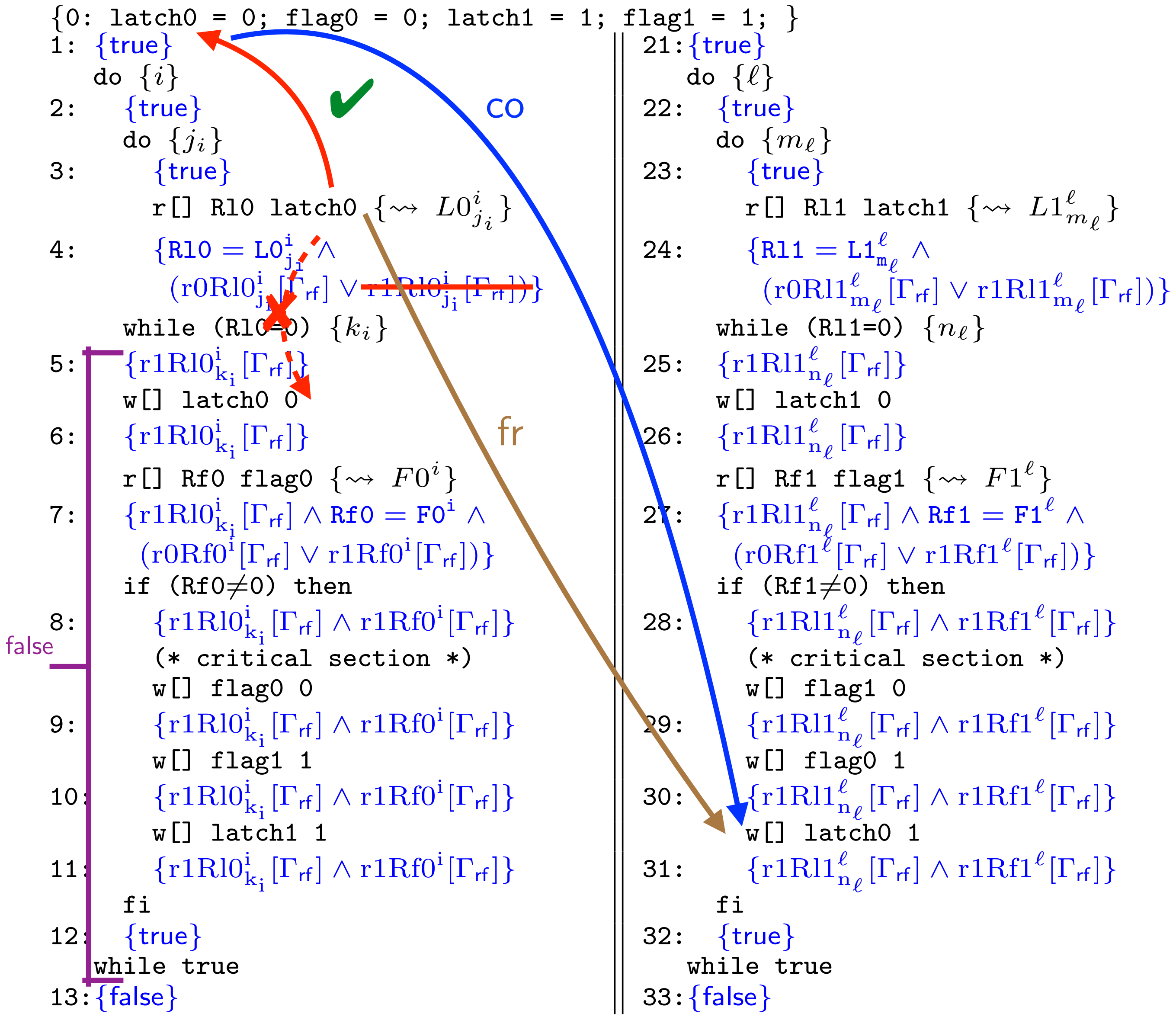
(3) Process `P1` stuck in spin loop (no hypothesis on `P0`)

```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:    {true}
      do {$j_i$}
3:       {true}
         r[] Rl0 latch0 {$\rightsquigarrow L0^{i}_{j_i}$}
4:       {$\mathtt{Rl0} = \mathtt{L0}^{i}_{j_i} \wedge$
          $(\mathrm{r0Rl0}^{i}_{j_i}[\Gamma_{rf}] \vee \mathrm{r1Rl0}^{i}_{j_i}[\Gamma_{rf}])$}
      while (Rl0=0) {$k_i$}
5:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
      w[] latch0 0
6:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
      r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathtt{Rf0} = \mathtt{F0}^{i} \wedge$
       $(\mathrm{r0Rf0}^{i}[\Gamma_{rf}] \vee \mathrm{r1Rf0}^{i}[\Gamma_{rf}])$}
      if (Rf0≠0) then
8:       {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
         (* critical section *)
         w[] flag0 0
9:       {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
  false
         w[] flag1 1
10:      {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
         w[] latch1 1
11:      {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
      fi
12:   {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:   {true}
      do {$m_\ell$}
23:      {true}
         r[] Rl1 latch1 {$\rightsquigarrow L1^{\ell}_{m_\ell}$}
24:      {$\mathtt{Rl1} = \mathtt{L1}^{\ell}_{m_\ell} \wedge$
          $(\mathrm{r0Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \cancel{\mathrm{r1Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}]})$}
      while (Rl1=0) {$n_\ell$}
25:   {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
      w[] latch1 0
26:   {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
      r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:   {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
       $(\mathrm{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}])$}
      if (Rf1≠0) then
28:      {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}   false
         (* critical section *)
         w[] flag1 0
29:      {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
         w[] flag0 1
30:      {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
         w[] latch0 1
31:      {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
      fi
32:   {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant after 25: must be false
- read of `latch1` in 23: must be a 0
- only possibility if from 25:
- A contradiction since 25: is unreachable

(4) Process `P0` starves in spin loop, no hypothesis on `P1`

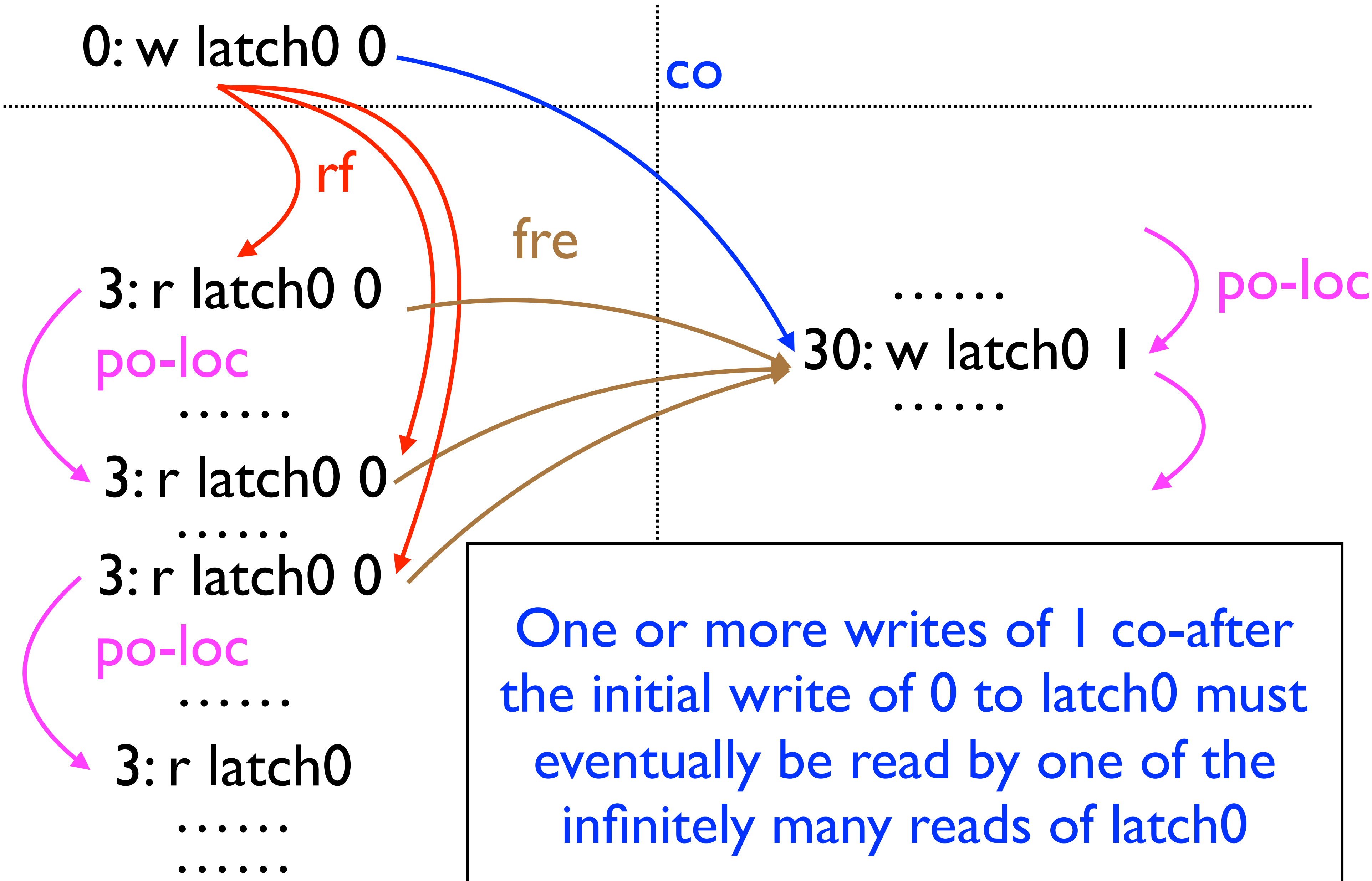


```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] Rl0 latch0 {$\leadsto L0^i_{j_i}$}
4:     {$\mathtt{Rl0} = \mathtt{L0}^{\mathrm{i}}_{\mathrm{j_i}} \wedge$
        $(\mathrm{r0Rl0}^{\mathrm{i}}_{\mathrm{j_i}}[\Gamma_{\mathrm{rf}}] \vee \cancel{\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{j_i}}[\Gamma_{\mathrm{rf}}]})$}
     while (Rl0=0) {$k_i$}
5:   {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}]$}
     w[] latch0 0
6:   {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}]$}
     r[] Rf0 flag0 {$\leadsto F0^i$}
7:   {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}] \wedge \mathtt{Rf0} = \mathtt{F0}^{\mathrm{i}} \wedge$
      $(\mathrm{r0Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}] \vee \mathrm{r1Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}])$}
     if (Rf0$\neq$0) then
8:     {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}]$}
       w[] flag1 1
10:    {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}]$}
       w[] latch1 1
11:    {$\mathrm{r1Rl0}^{\mathrm{i}}_{\mathrm{k_i}}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf0}^{\mathrm{i}}[\Gamma_{\mathrm{rf}}]$}
     fi
12:  {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] Rl1 latch1 {$\leadsto L1^\ell_{m_\ell}$}
24:    {$\mathtt{Rl1} = \mathtt{L1}^{\ell}_{\mathrm{m}_\ell} \wedge$
        $(\mathrm{r0Rl1}^{\ell}_{\mathrm{m}_\ell}[\Gamma_{\mathrm{rf}}] \vee \mathrm{r1Rl1}^{\ell}_{\mathrm{m}_\ell}[\Gamma_{\mathrm{rf}}])$}
     while (Rl1=0) {$n_\ell$}
25:  {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}]$}
     w[] latch1 0
26:  {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}]$}
     r[] Rf1 flag1 {$\leadsto F1^\ell$}
27:  {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
      $(\mathrm{r0Rf1}^{\ell}[\Gamma_{\mathrm{rf}}] \vee \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathrm{rf}}])$}
     if (Rf1$\neq$0) then
28:    {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathrm{rf}}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathrm{rf}}]$}
       w[] flag0 1
30:    {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathrm{rf}}]$}
       w[] latch0 1
31:    {$\mathrm{r1Rl1}^{\ell}_{\mathrm{n}_\ell}[\Gamma_{\mathrm{rf}}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{\mathrm{rf}}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- the invariant after 5: must be false so P0 never enters its critical section
- read of `latch0` in 3: must be a 0, with 2 possibilities
- cannot be from write at 5: which is unreachable
- so is from initial write 0:
- but P1 enters its critical section (otherwise see case 1)
- so `w[] latch0 1` will be executed later in co order
- so all `3:r[] Rl0 latch0` are fr to all 30: w[] latch0 1
- by fairness of communications, this write of 1 to latch0 will eventually be read at 3:
- in contradiction with always reading 0

(4) Process `P0` starves in spin loop, `P1` does not



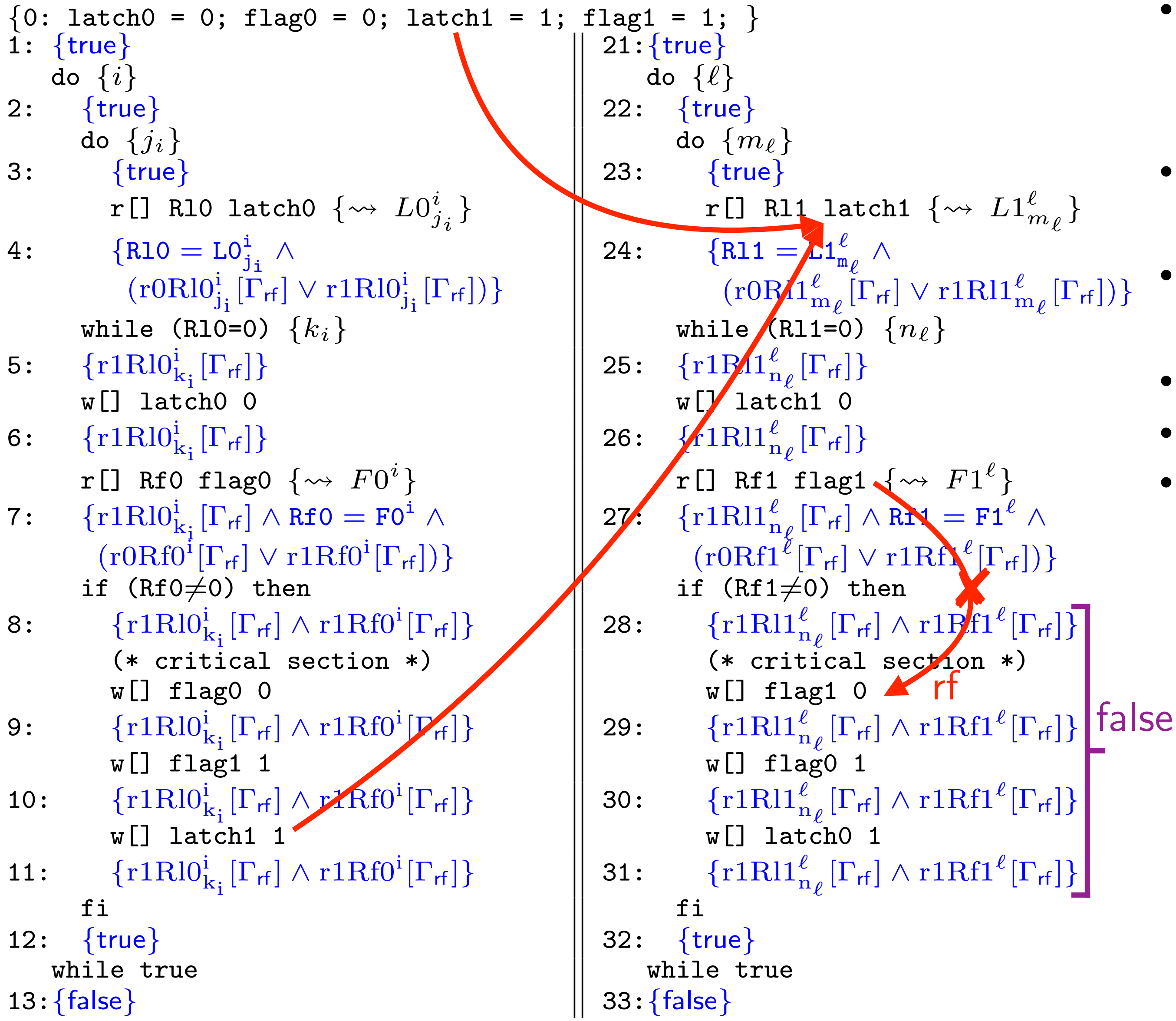
One or more writes of 1 co-after the initial write of 0 to latch0 must eventually be read by one of the infinitely many reads of latch0

Communication fairness hypothesis(*)

- All writes eventually hit the memory:
  - If, at a cut of the execution, all the processes infinitely often write the same value υ to a shared variable `x` and only that value υ
  - and from a later cut point of that execution, a process infinitely often repeats reads to that variable `x`
  - then the reads will end up reading that value υ

(*) The SPARC Architecture Manual, Version 8, Section K2, p. 283: ``if one processor does an *S* , and another processor repeatedly does *L* 's to the same location, then there is an *L* that will be after the *S*".

(5) Process P1 never enters its CS

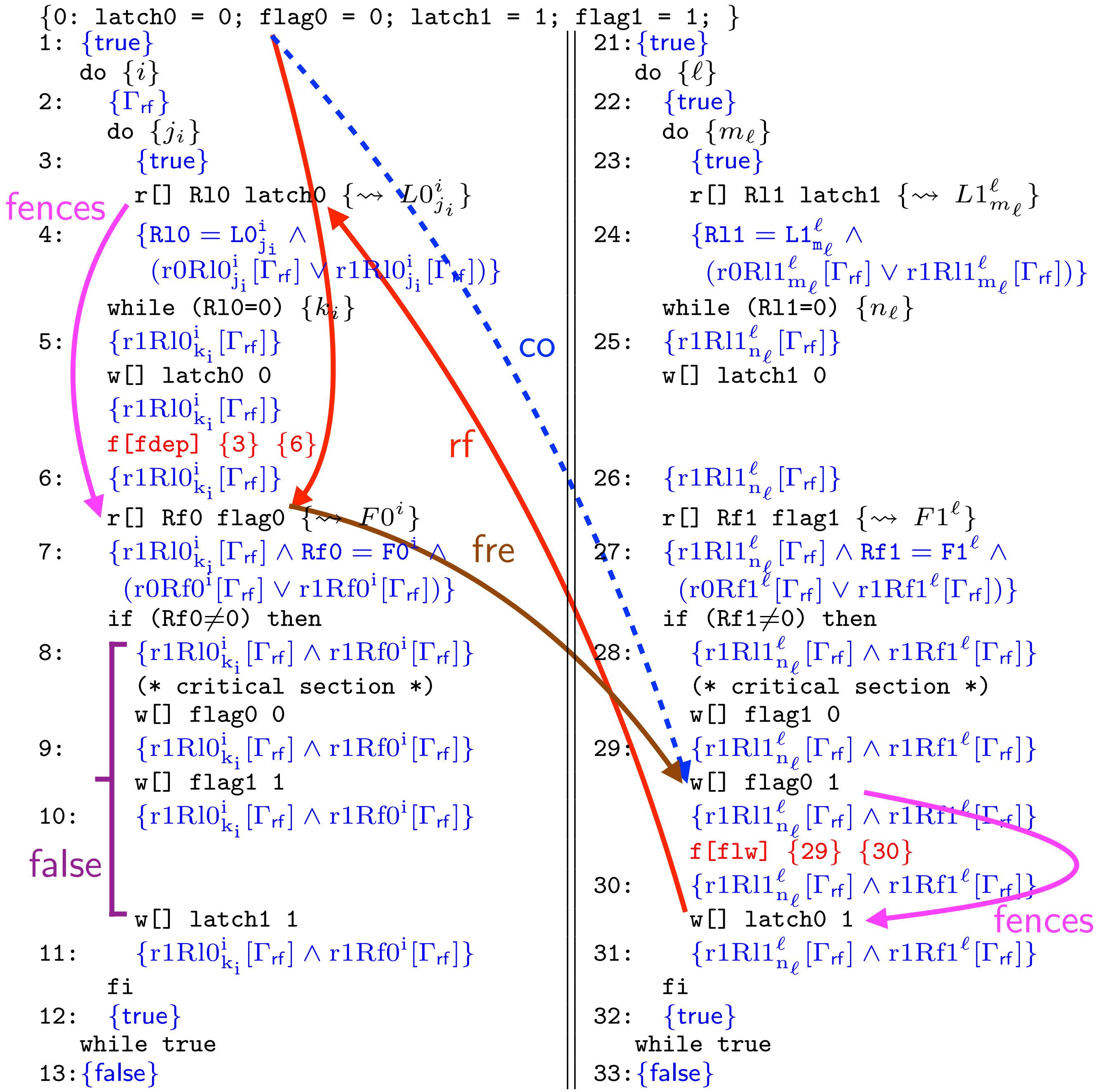


```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1: {true}
   do {$i$}
2:   {true}
     do {$j_i$}
3:     {true}
       r[] Rl0 latch0 {$\rightsquigarrow L0^{i}_{j_i}$}
4:     {$\mathtt{Rl0} = \mathtt{L0}^{i}_{j_i} \wedge$
         $(\mathrm{r0Rl0}^{i}_{j_i}[\Gamma_{rf}] \vee \mathrm{r1Rl0}^{i}_{j_i}[\Gamma_{rf}])$}
     while (Rl0=0) {$k_i$}
5:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     w[] latch0 0
6:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}]$}
     r[] Rf0 flag0 {$\rightsquigarrow F0^{i}$}
7:   {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathtt{Rf0} = \mathtt{F0}^{i} \wedge$
      $(\mathrm{r0Rf0}^{i}[\Gamma_{rf}] \vee \mathrm{r1Rf0}^{i}[\Gamma_{rf}])$}
     if (Rf0≠0) then
8:     {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag0 0
9:     {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] flag1 1
10:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
       w[] latch1 1
11:    {$\mathrm{r1Rl0}^{i}_{k_i}[\Gamma_{rf}] \wedge \mathrm{r1Rf0}^{i}[\Gamma_{rf}]$}
     fi
12:  {true}
   while true
13:{false}
```

```
21:{true}
   do {$\ell$}
22:  {true}
     do {$m_\ell$}
23:    {true}
       r[] Rl1 latch1 {$\rightsquigarrow L1^{\ell}_{m_\ell}$}
24:    {$\mathtt{Rl1} = \mathtt{L1}^{\ell}_{m_\ell} \wedge$
         $(\mathrm{r0Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}] \vee \mathrm{r1Rl1}^{\ell}_{m_\ell}[\Gamma_{rf}])$}
     while (Rl1=0) {$n_\ell$}
25:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     w[] latch1 0
26:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}]$}
     r[] Rf1 flag1 {$\rightsquigarrow F1^{\ell}$}
27:  {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathtt{Rf1} = \mathtt{F1}^{\ell} \wedge$
      $(\mathrm{r0Rf1}^{\ell}[\Gamma_{rf}] \vee \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}])$}
     if (Rf1≠0) then
28:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       (* critical section *)
       w[] flag1 0
29:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] flag0 1
30:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
       w[] latch0 1
31:    {$\mathrm{r1Rl1}^{\ell}_{n_\ell}[\Gamma_{rf}] \wedge \mathrm{r1Rf1}^{\ell}[\Gamma_{rf}]$}
     fi
32:  {true}
   while true
33:{false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- `P1` exits loop 23:–24: (else see cases (1) or (3))
- must read `Rl1` = 1 from 0: or 10:
- read of `Rf1` at 26: must be 0
- only possibility is from 28:
- impossible from unreachable code

(5) Process P0 leaves spin loop but always fails entering its CS



```
{0: latch0 = 0; flag0 = 0; latch1 = 1; flag1 = 1; }
1:  {true}
    do {$i$}
2:    {$\Gamma_{rf}$}
      do {$j_i$}
3:      {true}
        r[] R10 latch0 {$\rightsquigarrow L0^i_{j_i}$}
4:      {$R10 = L0^i_{j_i} \wedge (r0R10^i_{j_i}[\Gamma_{rf}] \vee r1R10^i_{j_i}[\Gamma_{rf}])$}
      while (R10=0) {$k_i$}
5:    {$r1R10^i_{k_i}[\Gamma_{rf}]$}
      w[] latch0 0
      {$r1R10^i_{k_i}[\Gamma_{rf}]$}
      f[fdep] {3} {6}
6:    {$r1R10^i_{k_i}[\Gamma_{rf}]$}
      r[] Rf0 flag0 {$\rightsquigarrow F0^i$}
7:    {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge Rf0 = F0^i \wedge$
       $(r0Rf0^i[\Gamma_{rf}] \vee r1Rf0^i[\Gamma_{rf}])$}
      if (Rf0≠0) then
8:      {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
        (* critical section *)
        w[] flag0 0
9:      {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
        w[] flag1 1
10:     {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}

        w[] latch1 1
11:     {$r1R10^i_{k_i}[\Gamma_{rf}] \wedge r1Rf0^i[\Gamma_{rf}]$}
      fi
12:   {true}
    while true
13: {false}
```

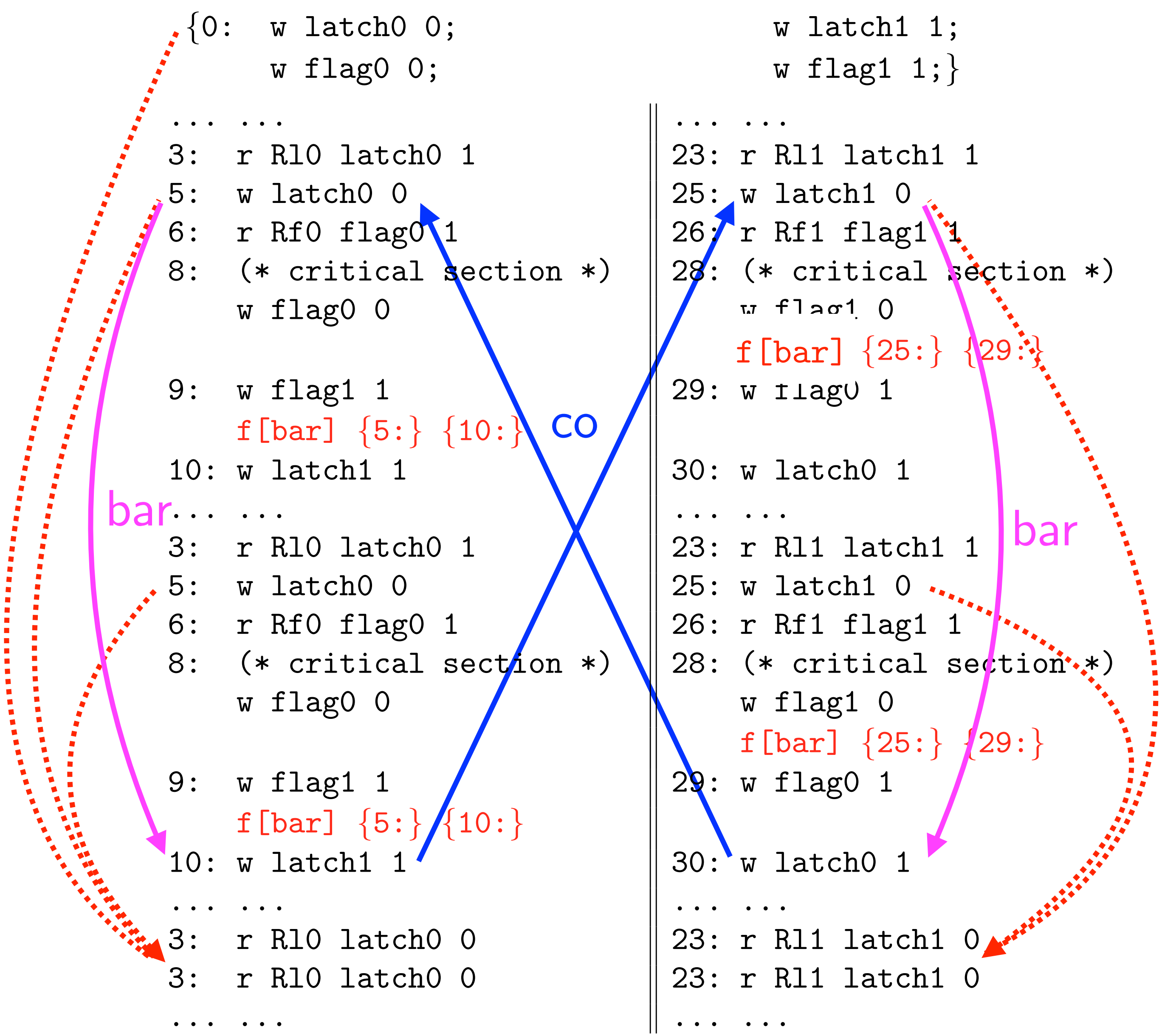
```
21: {true}
    do {$\ell$}
22:   {true}
      do {$m_\ell$}
23:     {true}
        r[] R11 latch1 {$\rightsquigarrow L1^\ell_{m_\ell}$}
24:     {$R11 = L1^\ell_{m_\ell} \wedge$
         $(r0R11^\ell_{m_\ell}[\Gamma_{rf}] \vee r1R11^\ell_{m_\ell}[\Gamma_{rf}])$}
      while (R11=0) {$n_\ell$}
25:   {$r1R11^\ell_{n_\ell}[\Gamma_{rf}]$}
      w[] latch1 0


26:   {$r1R11^\ell_{n_\ell}[\Gamma_{rf}]$}
      r[] Rf1 flag1 {$\rightsquigarrow F1^\ell$}
27:   {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge Rf1 = F1^\ell \wedge$
       $(r0Rf1^\ell[\Gamma_{rf}] \vee r1Rf1^\ell[\Gamma_{rf}])$}
      if (Rf1≠0) then
28:     {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
        (* critical section *)
        w[] flag1 0
29:     {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
        w[] flag0 1
        {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
        f[flw] {29} {30}
30:     {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
        w[] latch0 1
31:     {$r1R11^\ell_{n_\ell}[\Gamma_{rf}] \wedge r1Rf1^\ell[\Gamma_{rf}]$}
      fi
32:   {true}
    while true
33: {false}
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- loop 2:–4: exited
- read of `R10` = 1 at 3: is from 30:
- invariant false in critical section 8:–11:
- read of `Rf0` = 0 at 6: is from 0: (8: not reachable)

```
withco
let l-fencerel(S) =
        ((po&(_*S));po)&fromto(S)
let Fdep = F & tag2events('fdep)
let deps = l-fencerel(Fdep) & (R*_)
let Flw = F & tag2events('flw)
let flw = l-fencerel(Flw)
let fences = deps | flw
let fre = (rf^-1;co) & ext
irreflexive fre;fences;rfe;fences
```

In TSO there is no need for a fence since it is MP. For weaker than PSO, a fence is needed.

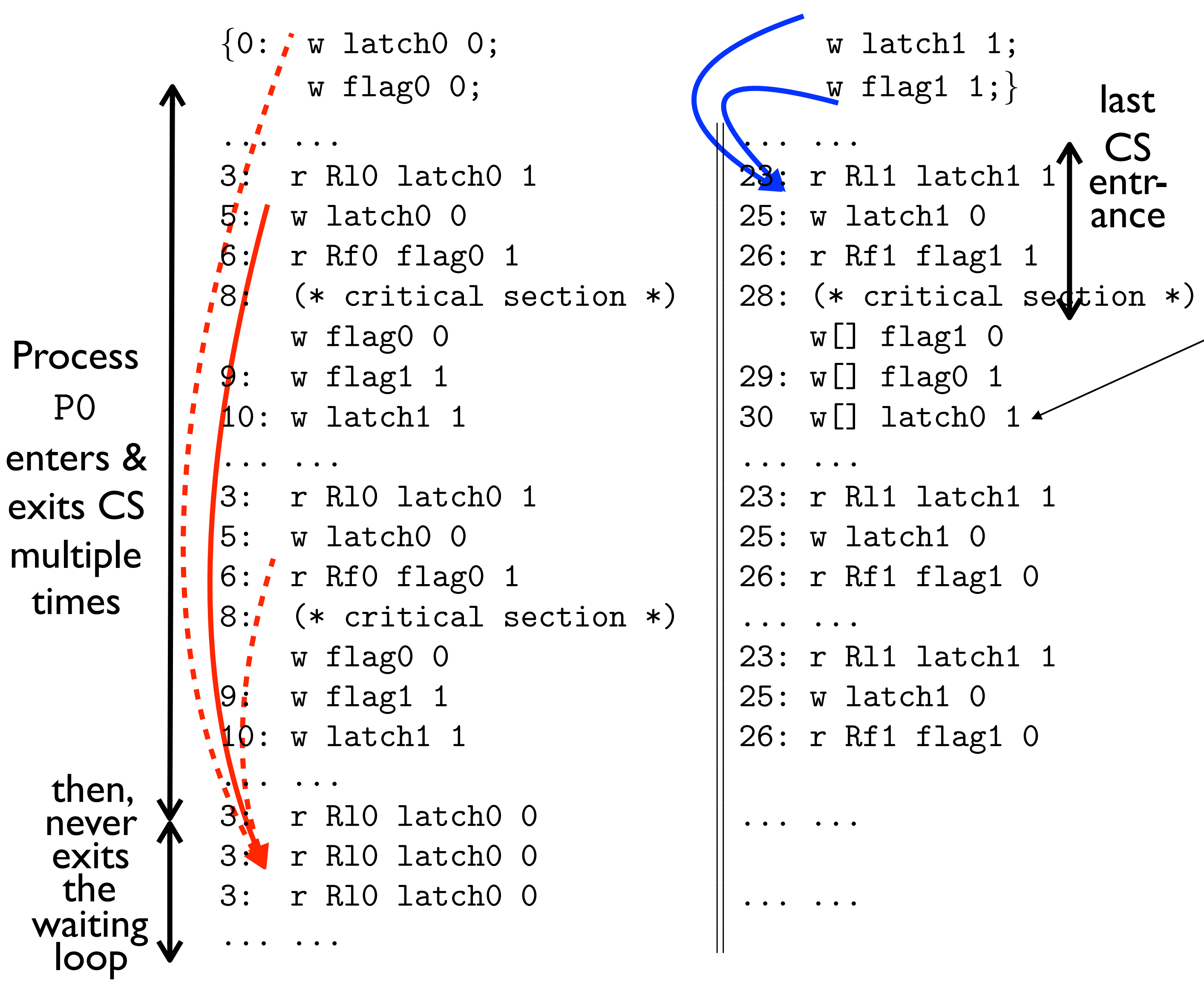
(6) Both processes eventually starve in spin loop



```
{0:   w latch0 0;                    w latch1 1;
      w flag0 0;                     w flag1 1;}
...  ...                        ...  ...
3:   r Rl0 latch0 1             23: r Rl1 latch1 1
5:   w latch0 0                 25: w latch1 0
6:   r Rf0 flag0 1              26: r Rf1 flag1 1
8:   (* critical section *)     28: (* critical section *)
     w flag0 0                      w flag1 0
                                    f[bar] {25:} {29:}
9:   w flag1 1                  29: w flag0 1
     f[bar] {5:} {10:}
10:  w latch1 1                 30: w latch0 1
...  ...                        ...  ...
3:   r Rl0 latch0 1             23: r Rl1 latch1 1
5:   w latch0 0                 25: w latch1 0
6:   r Rf0 flag0 1              26: r Rf1 flag1 1
8:   (* critical section *)     28: (* critical section *)
     w flag0 0                      w flag1 0
                                    f[bar] {25:} {29:}
9:   w flag1 1                  29: w flag0 1
     f[bar] {5:} {10:}
10:  w latch1 1                 30: w latch0 1
...  ...                        ...  ...
3:   r Rl0 latch0 0             23: r Rl1 latch1 0
3:   r Rl0 latch0 0             23: r Rl1 latch1 0
...  ...                        ...  ...
```

- let rf be the communication for such a trace (encoded in Γ_{rf})
- so `latch0` is always 0 and `latch1` is always 0
- so `latch0` in 23 is always read from 25:
- so `10: w latch1 1` was co-before (since otherwise by the communication hypothesis it would be eventually read)
- and `3: Rl0 latch0 0` is from 0: or 5:
- so `30: w latch0 1` is co-before them (since otherwise by the communication hypothesis it would be eventually read)
- impossible by fences
- irreflexive co; bar; co; bar

(7) Eventually, P0 starves in spin loop, P1 never enters its CS



Process P0 enters & exits CS multiple times

then, never exits the waiting loop

```
{0: w latch0 0;
    w flag0 0;
... ...
3:  r Rl0 latch0 1
5:  w latch0 0
6:  r Rf0 flag0 1
8:  (* critical section *)
    w flag0 0
9:  w flag1 1
10: w latch1 1
... ...
3:  r Rl0 latch0 1
5:  w latch0 0
6:  r Rf0 flag0 1
8:  (* critical section *)
    w flag0 0
9:  w flag1 1
10: w latch1 1
... ...
3:  r Rl0 latch0 0
3:  r Rl0 latch0 0
3:  r Rl0 latch0 0
... ...
```

```
    w latch1 1;
    w flag1 1;}
... ...
23: r Rl1 latch1 1
25: w latch1 0
26: r Rf1 flag1 1
28: (* critical section *)
    w[] flag1 0
29: w[] flag0 1
30  w[] latch0 1
... ...
23: r Rl1 latch1 1
25: w latch1 0
26: r Rf1 flag1 0
... ...
23: r Rl1 latch1 1
25: w latch1 0
26: r Rf1 flag1 0

... ...

... ...
```

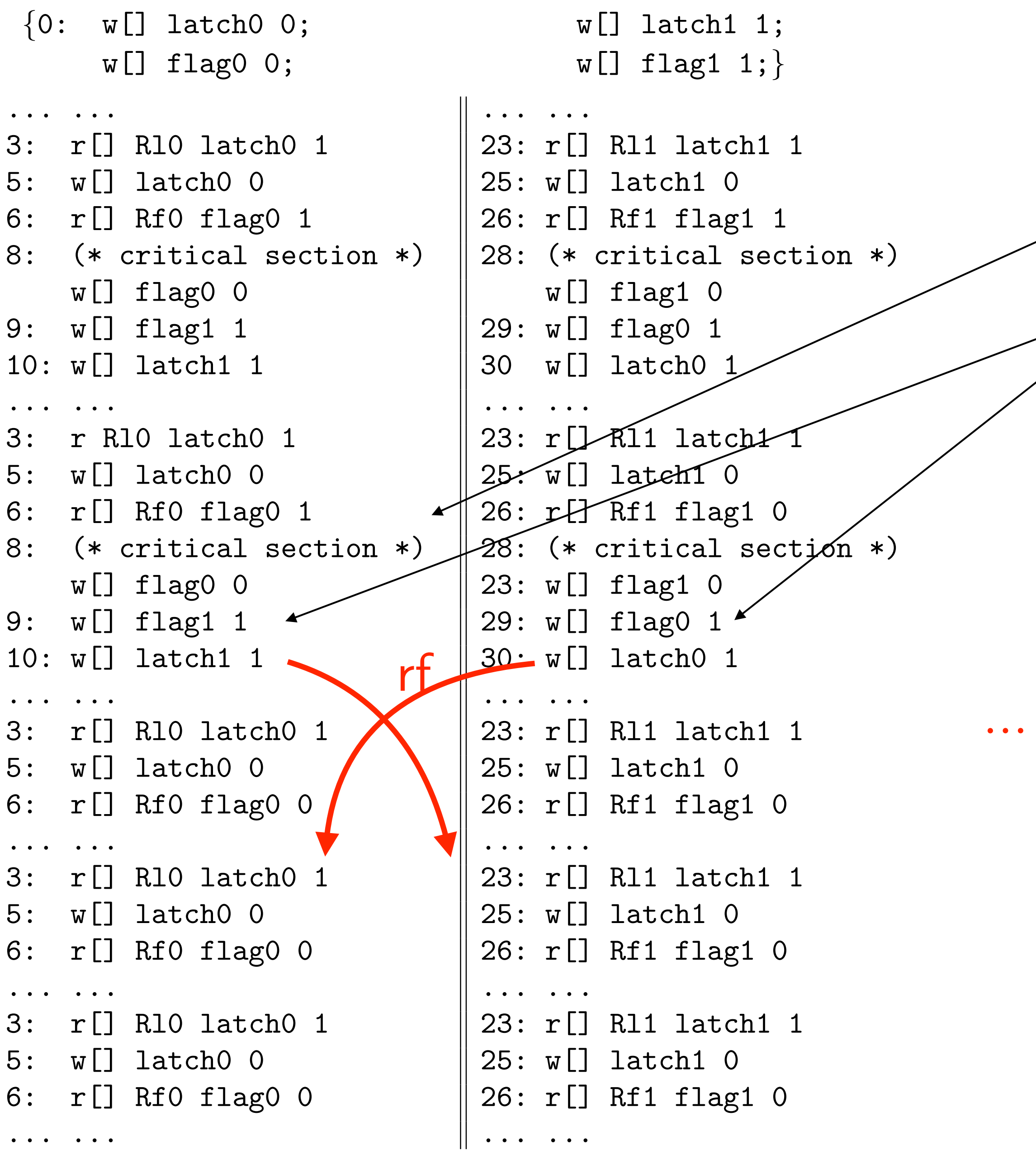
last CS entr-ance

- P1 does not eventually starves in spin loop (otherwise case 6)
- case P1 eventually never starves and never enters its critical section
- P1 then does a last write of 1 to latch0
- P0 eventually makes infinitely many reads of latch0
- A contradiction (since otherwise by the communication hypothesis, this 1 would be eventually read)

(8) Eventually, `P1` starves in spin loop, `P0` never enters its CS

symmetric of (7)

(9) P0 and P1 always leave spin loop and never enter their CS



```
 {0:  w[] latch0 0;                 w[] latch1 1;
      w[] flag0 0;                  w[] flag1 1;}
...  ...                        ...  ...
3:   r[] Rl0 latch0 1           23: r[] Rl1 latch1 1
5:   w[] latch0 0               25: w[] latch1 0
6:   r[] Rf0 flag0 1            26: r[] Rf1 flag1 1
8:   (* critical section *)     28: (* critical section *)
     w[] flag0 0                    w[] flag1 0
9:   w[] flag1 1                29: w[] flag0 1
10:  w[] latch1 1               30  w[] latch0 1
...  ...                        ...  ...
3:   r Rl0 latch0 1             23: r[] Rl1 latch1 1
5:   w[] latch0 0               25: w[] latch1 0
6:   r[] Rf0 flag0 1            26: r[] Rf1 flag1 0
8:   (* critical section *)     28: (* critical section *)
     w[] flag0 0                23: w[] flag1 0
9:   w[] flag1 1                29: w[] flag0 1
10:  w[] latch1 1               30: w[] latch0 1
...  ...                        ...  ...
3:   r[] Rl0 latch0 1           23: r[] Rl1 latch1 1
5:   w[] latch0 0               25: w[] latch1 0
6:   r[] Rf0 flag0 0            26: r[] Rf1 flag1 0
...  ...                        ...  ...
3:   r[] Rl0 latch0 1           23: r[] Rl1 latch1 1
5:   w[] latch0 0               25: w[] latch1 0
6:   r[] Rf0 flag0 0            26: r[] Rf1 flag1 0
...  ...                        ...  ...
3:   r[] Rl0 latch0 1           23: r[] Rl1 latch1 1
5:   w[] latch0 0               25: w[] latch1 0
6:   r[] Rf0 flag0 0            26: r[] Rf1 flag1 0
...  ...                        ...  ...
```

- P0 and P1 eventually never starve and never enter their critical sections
- They both have a last entrance in their critical sections
- This last write of 1 to the latches will, by communication fairness, eventually reach the memory
- Then we only have infinitely many writes of 0 to the latches
- So the read of the latches in the spin loops will eventually always read 0
- So from then on, by communication fairness, all the reads will be from 0, in reads of the latch will be zero
- In contradiction with the fact that the spin loop is always exited
- The barrier prevents infinitely postponing the write 0 actions

Conclusion

Conclusion

- The proof method is parameterized by consistency hypotheses, expressed in
  - Invariance form: S_{com}
  - Consistency form: H_{com} (e.g. in `cat`)
- Program not logic/architecture/consistency model dependent (hence the proof is portable)
- Can reason on *arbitrary* subsets of anarchic executions (hence flexible e.g. non-starvation)

Proposed design methodology

1. Design the algorithm A and its specification S_{inv} (e.g. in the sequential consistency model of parallelism)
2. Consider the anarchic semantics of algorithm A
3. Add communication specifications S_{com} to restrict anarchic communications and ensure the correctness of A with respect to specification S_{inv}
4. Do the invariance proof under WCM with S_{com}
5. Infer H_{com} (in `cat`) from invariant S_{com}
6. Prove that the machine memory model M in `cat` implies H_{cm}

Challenges

- Modern machines have complex memory models

⇒ portability has a price (refencing)

⇒ debugging is very hard/quasi-impossible

⇒ proofs are much harder than with sequential consistency (but still feasible?, mechanically?)

⇒ static analysis parameterized by a WCM will be a challenge

⇒ but we can start with S_{com}

Thanks

- *Patrick Cousot thanks Luc Maranget for his precious help at Dagstuhl on the non-starvation part.*

The End, Thank You