Verifica di Algoritmi Distribuiti Probabilistici

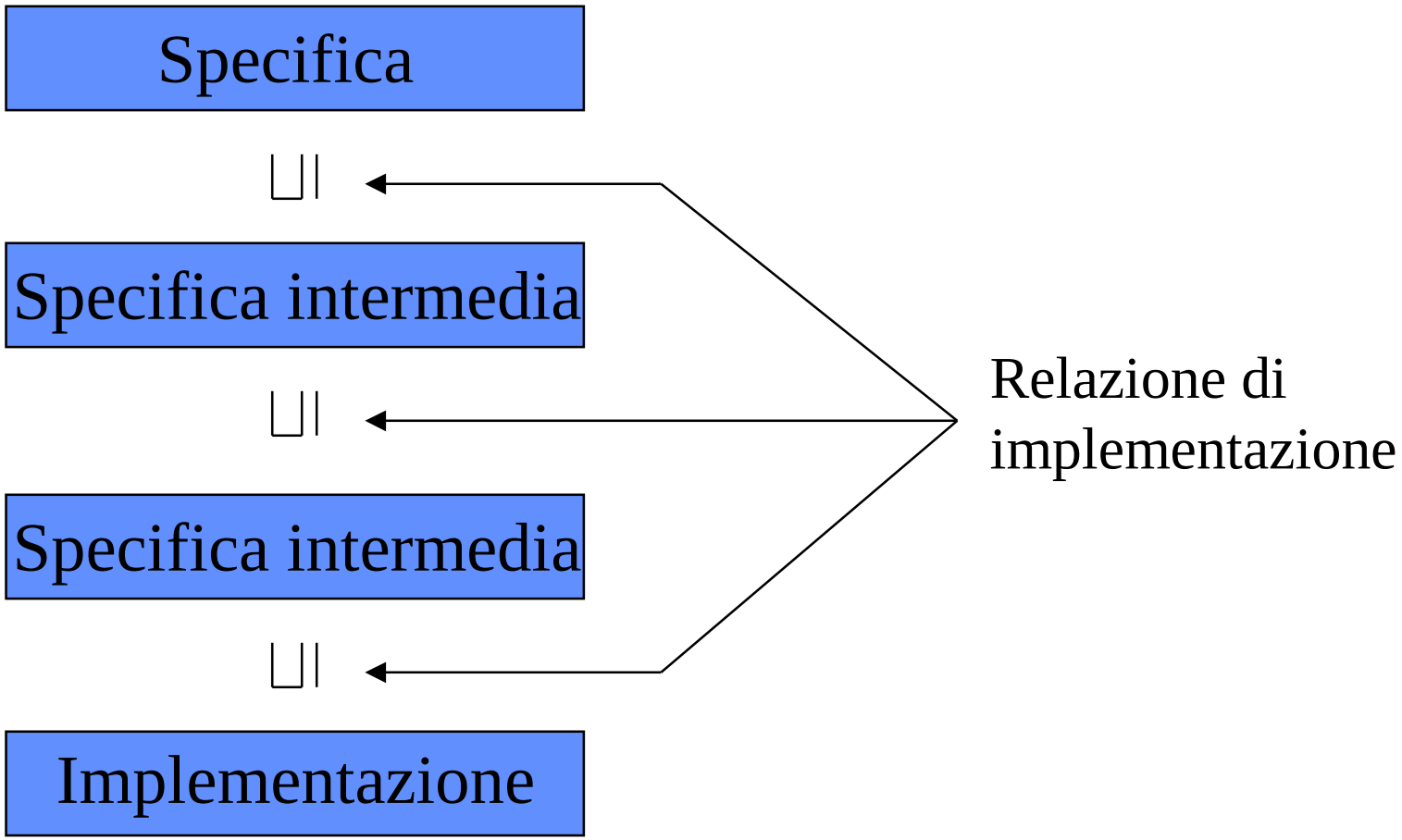
Roberto Segala

Università di Verona

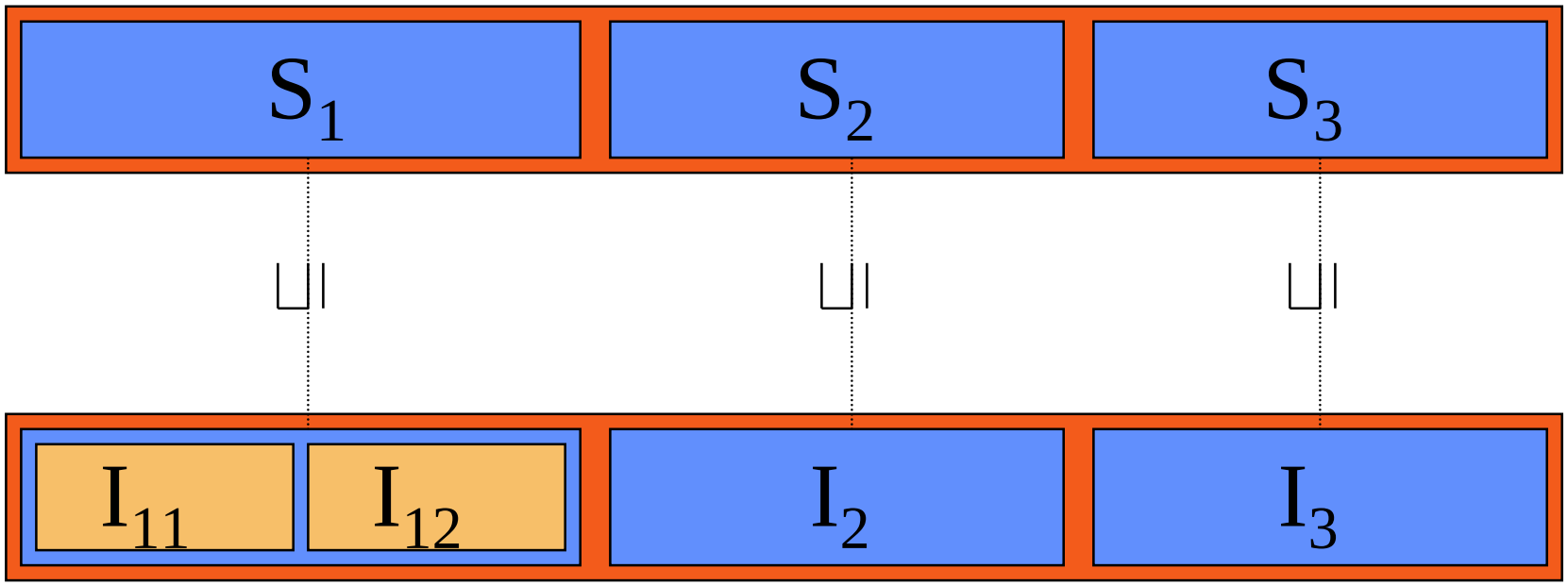
Teoria e Applicazioni

- Vasta ricerca in teoria della concorrenza
  - Process algebra
  - Relazioni di preordine ed equivalenza
  - Composizionalità
- Risultati di scarsa applicabilità
  - Il mondo è meno regolare di quanto si speri
  - Non tutti i fenomeni hanno buone proprietà algebriche
  - Non tutto si studia con la matematica discreta
- I problemi da risolvere ci sono
  - Sistemi distribuiti
  - Embedded systems

Specifiche e Implementazioni



Modularità



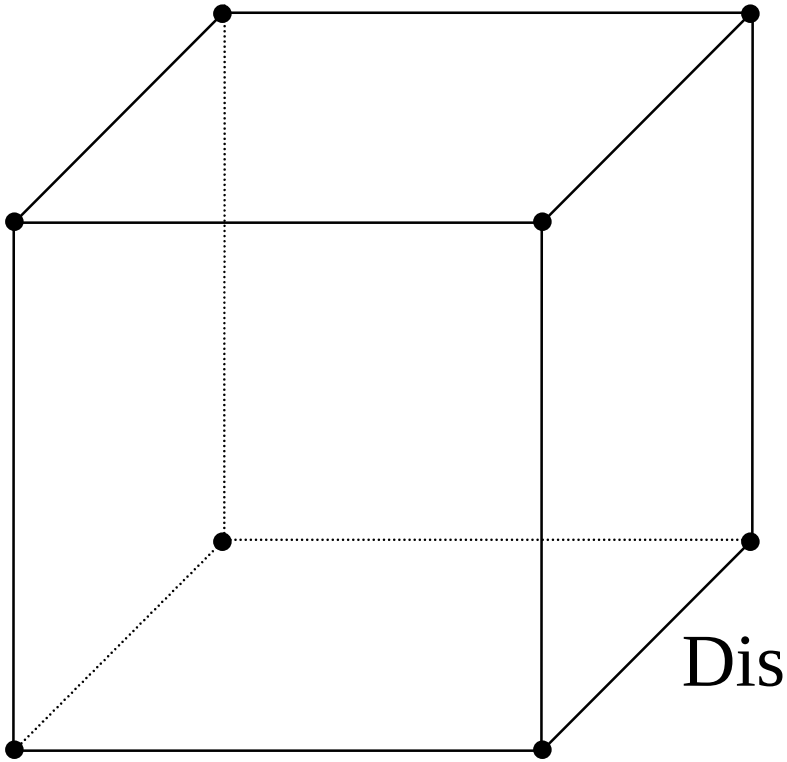
Paradigmi

- Real-Time
  - Entità che assumono valori reali
  - Alcuni sistemi sono corretti solo in presenza di condizioni temporali
- Comportamento continuo
  - Equazioni differenziali in presenza di controllori discreti
- Comportamento stocastico
  - Alcune scelte sono governate da probabilità

Modularità di Paradigmi

- Sistema da composizione di sistemi eterogenei
  - Solo alcuni componenti sono basati su real-time
  - Solo alcuni componenti sono stocastici o ibridi
- Principio di riduzione
  - Riduzione di una proprietà stocastica ad una proprietà non stocastica
  - Suddivisione di una proprietà di un sistema ibrido
    - Proprietà del discreto
    - Proprietà del continuo

Ipercubo di Modelli



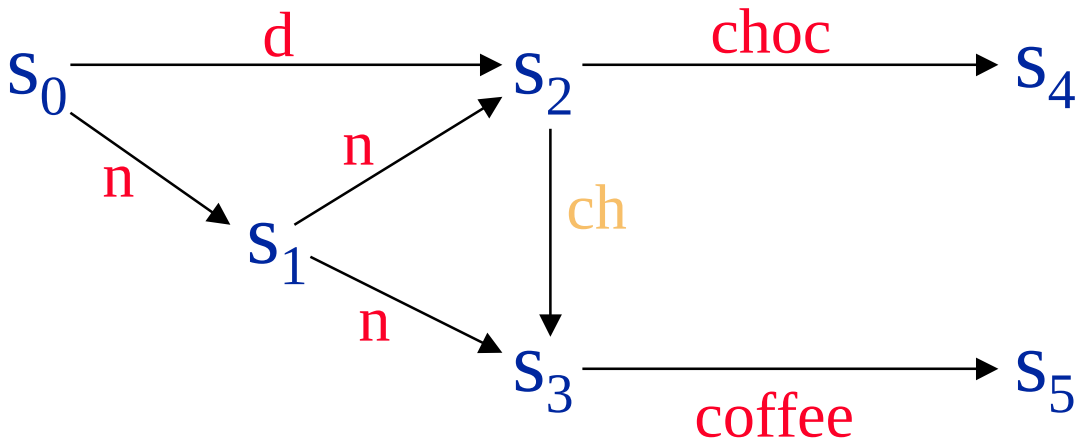
Determinismo / Probabilità

Discreto / Continuo

No real-time / real-time

Automi

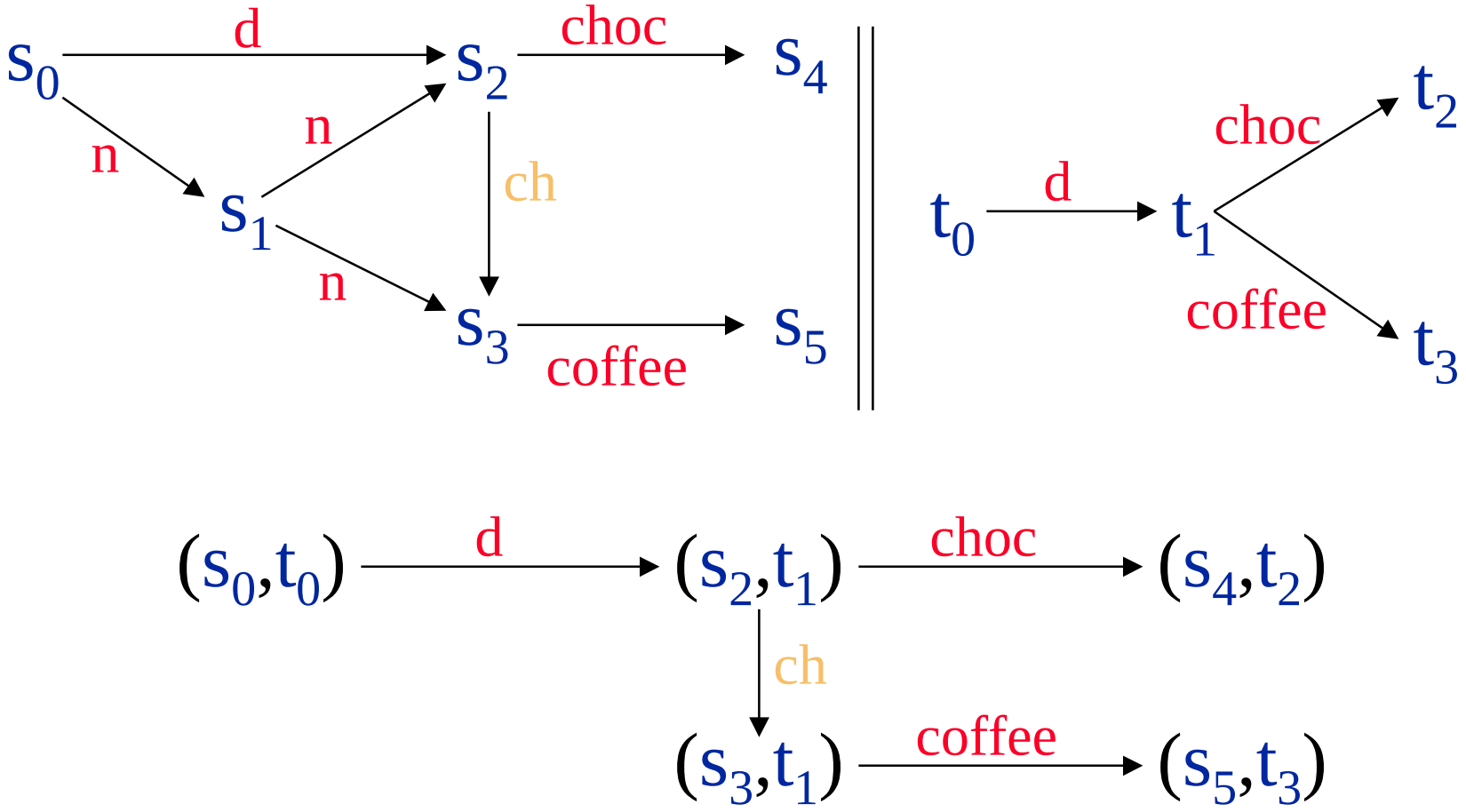
$$A=(Q,Q_0,S^{ext},S^{int},tran)$$



Esecuzione: s_0 n s_1 n s_2 ch s_3 coffee s_5

Traccia: n n coffee

Composizione Parallela



Proiezioni

Se

$\alpha = (s_0,t_0)\ d\ (s_2,t_1)\ ch\ (s_3,t_1)\ coffee\ (s_5,t_3)$

È un'esecuzione di $A \| B$, il contributo di A è

$\alpha \lceil A = s_0\ d\ s_2\ ch\ s_3\ coffee\ s_5$

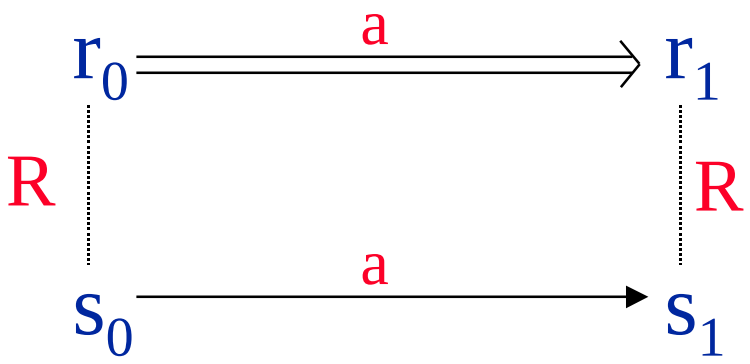
Il contributo di B è

$\alpha \lceil B = t_0\ d\ t_1\ coffee\ t_3$

Simulazioni

$A \leq B$: esiste $R \subseteq states(A)$ x $states(B)$

per ogni s_0 R r_0, a, s_1, esiste r_1



Prop. $A \leq B$ implica $traces(A) \subseteq traces(B)$

e $A||C \leq B||C$

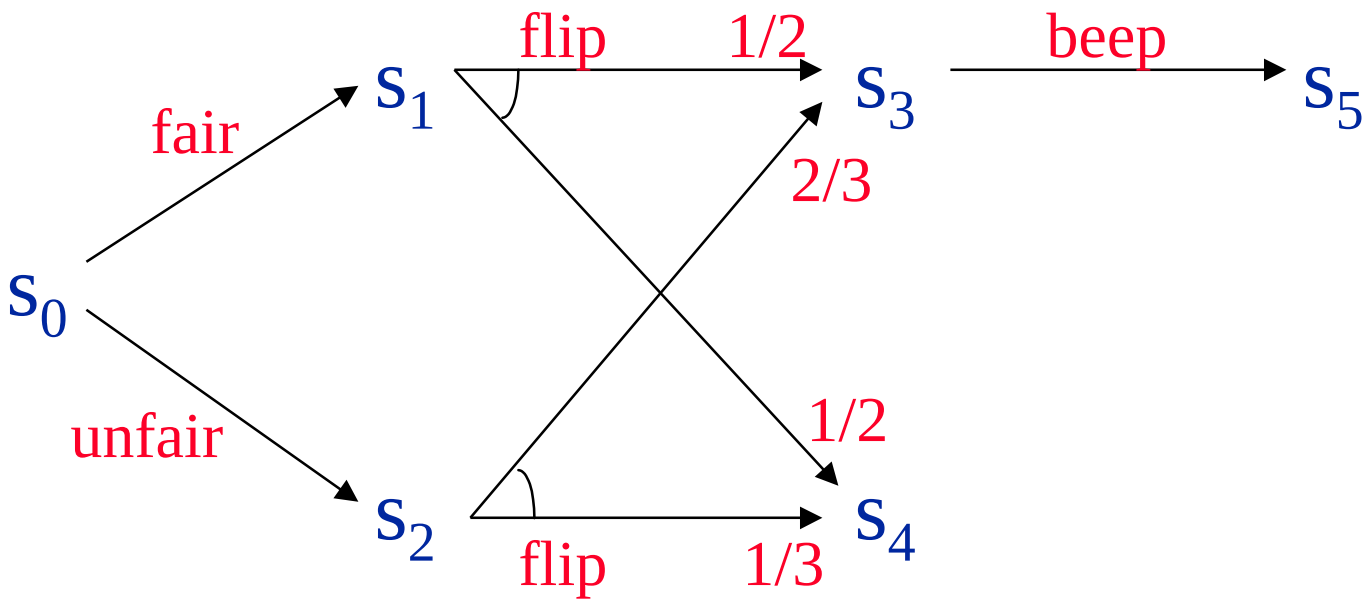
Estensione Real-Time

- Aggiungi una componente *.now* agli stati
- Aggiungi un'azione ν per il passaggio del tempo
- ν è visibile nelle tracce

Estensione Ibrida

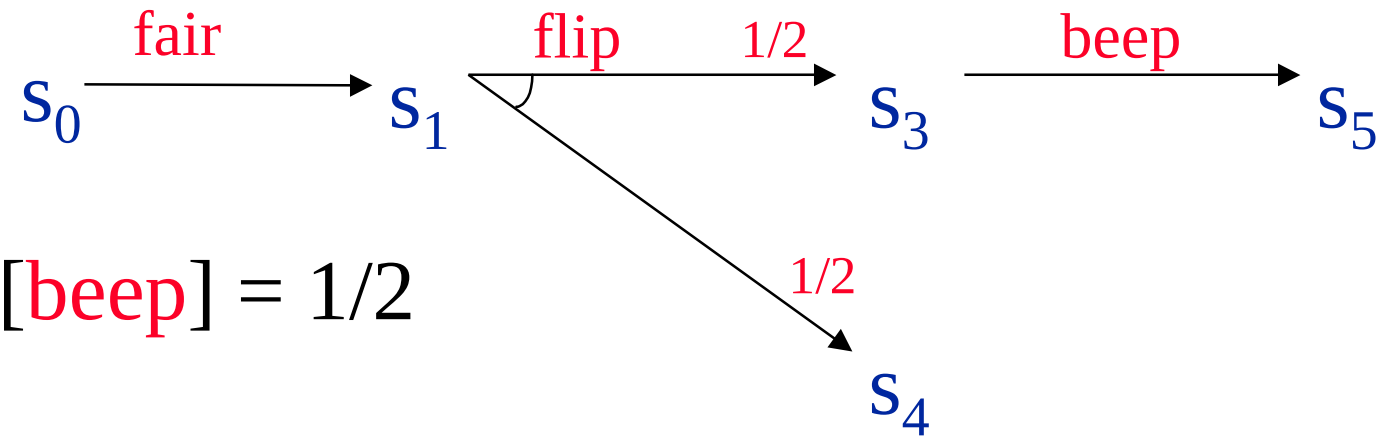
- Aggiungi variabili
- Aggiungi traiettorie
  - Funzione da tempo a valutazioni di variabili
  - Traiettorie descritte da equazioni differenziali
- Esecuzioni e tracce
  - Sequenze alterne di traiettorie e azioni

Estensione Probabilistica

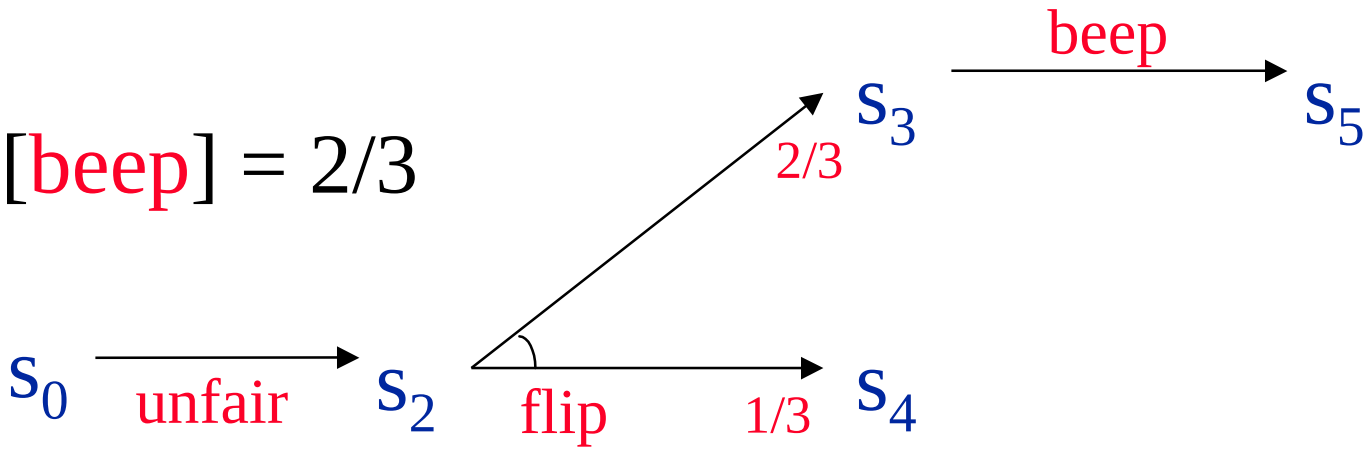


Probabilità di beep?

Esecuzioni Probabilistiche

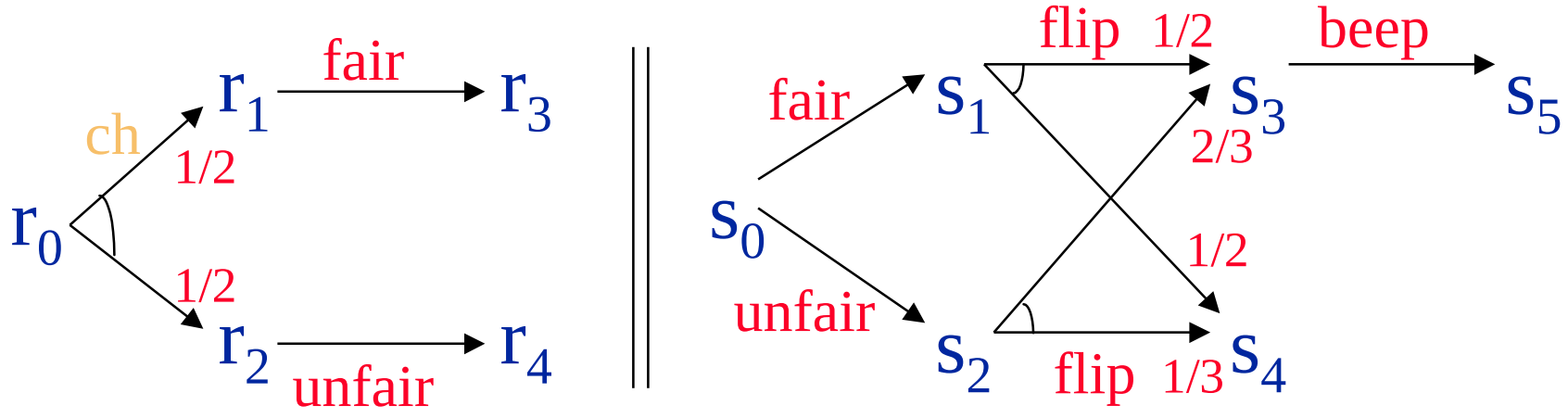


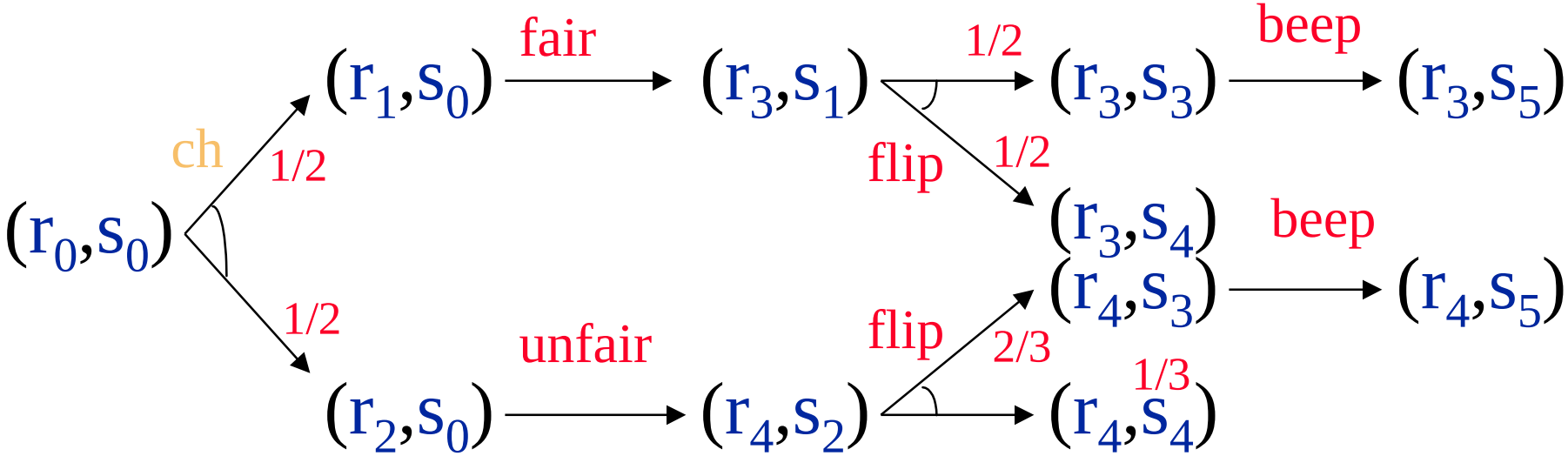
Pr[beep] = 1/2



Pr[beep] = 2/3

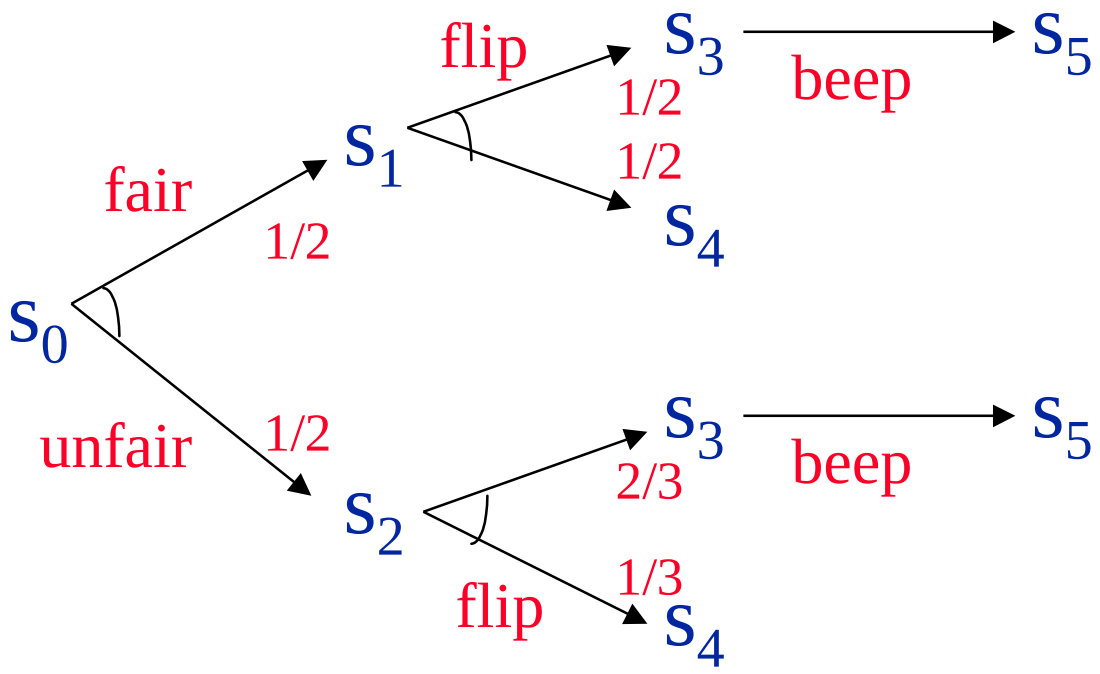
Composizione Parallela





Proiezioni

Proiettando sul componente destro otteniamo



•Lo schedulatore puo’ essere casuale

Proprietà delle Proiezioni

Sia H una esecuzione probabilistica di $A \| B$.

- $H \lceil A$ è una esecuzione probabilistica di A.
- $P_{H \lceil A} = P_H \lceil A$:

  per ogni evento θ di $P_{H \lceil A}$,

  $P_{H \lceil A}[\theta] = P_H[\{\alpha | \alpha \lceil A \in \theta\}]$

Trace Distributions

Traccia di una esecuzione probabilistica:

- Un'esecuzione probabilistica induce una distribuzione di probabilità su tracce.
- Chiamiamo tale distribuzione trace distribution.

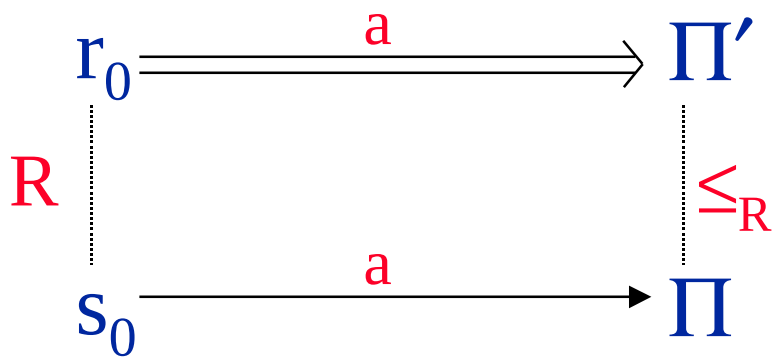
Osservazioni.

- Un'esecuzione è un caso speciale di esecuzione probabilistica.
- Una traccia è un caso speciale di trace distribution.

Simulazioni Probabilistiche

$A \leq B$: esiste $R \subseteq states(A) \times states(B)$

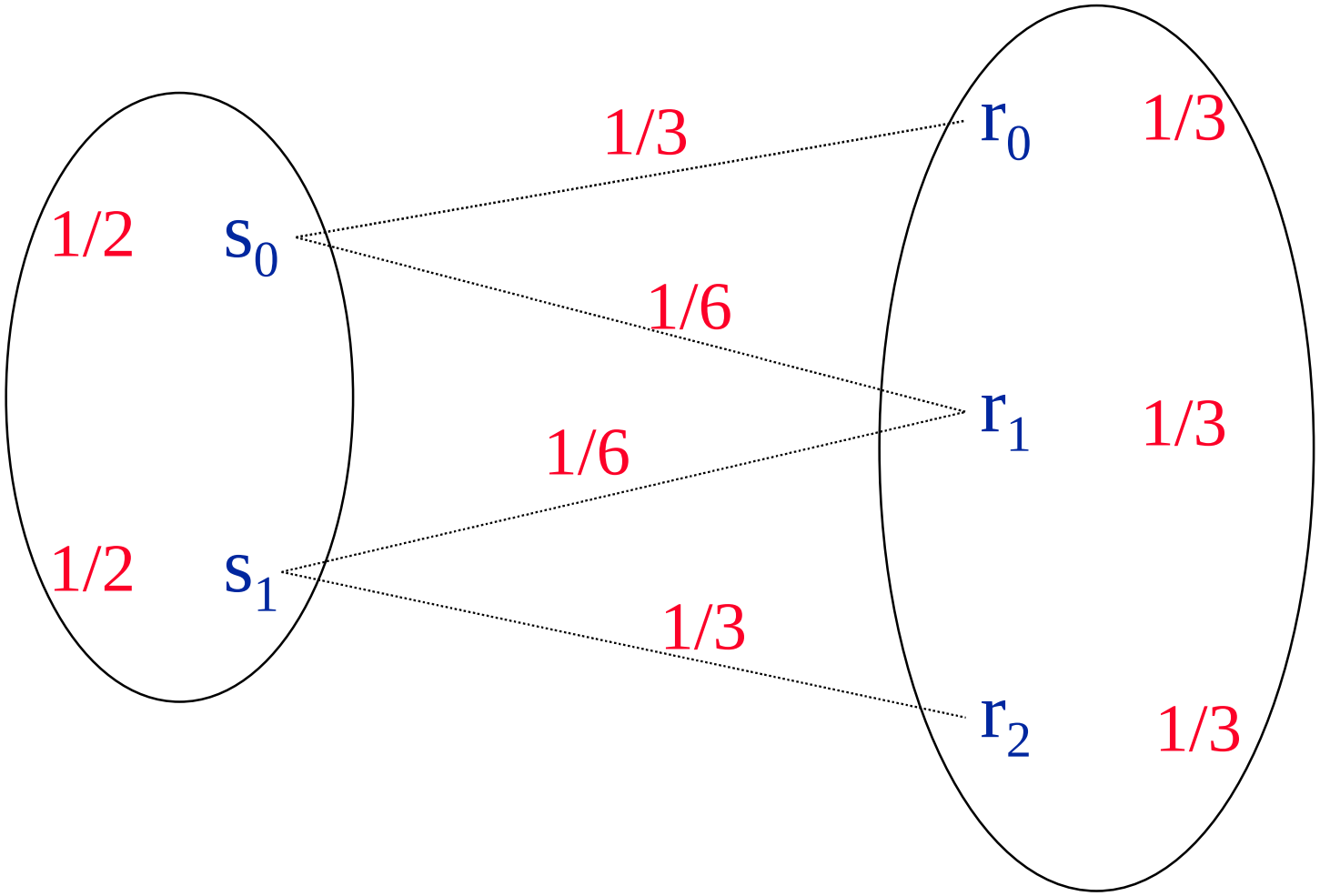
per ogni s_0 R r_0, a, Π, esiste Π'



Prop. $A \leq B$ implica $tdists(A) \subseteq tdists(B)$

e $A||C \leq B||C$

Lifting di una Relazione



Il Problema del Consenso

- Ogni processo propone un valore in {0,1}.
- Ogni processo puo' decidere un valore.
- I processi possono guastarsi e fermarsi.

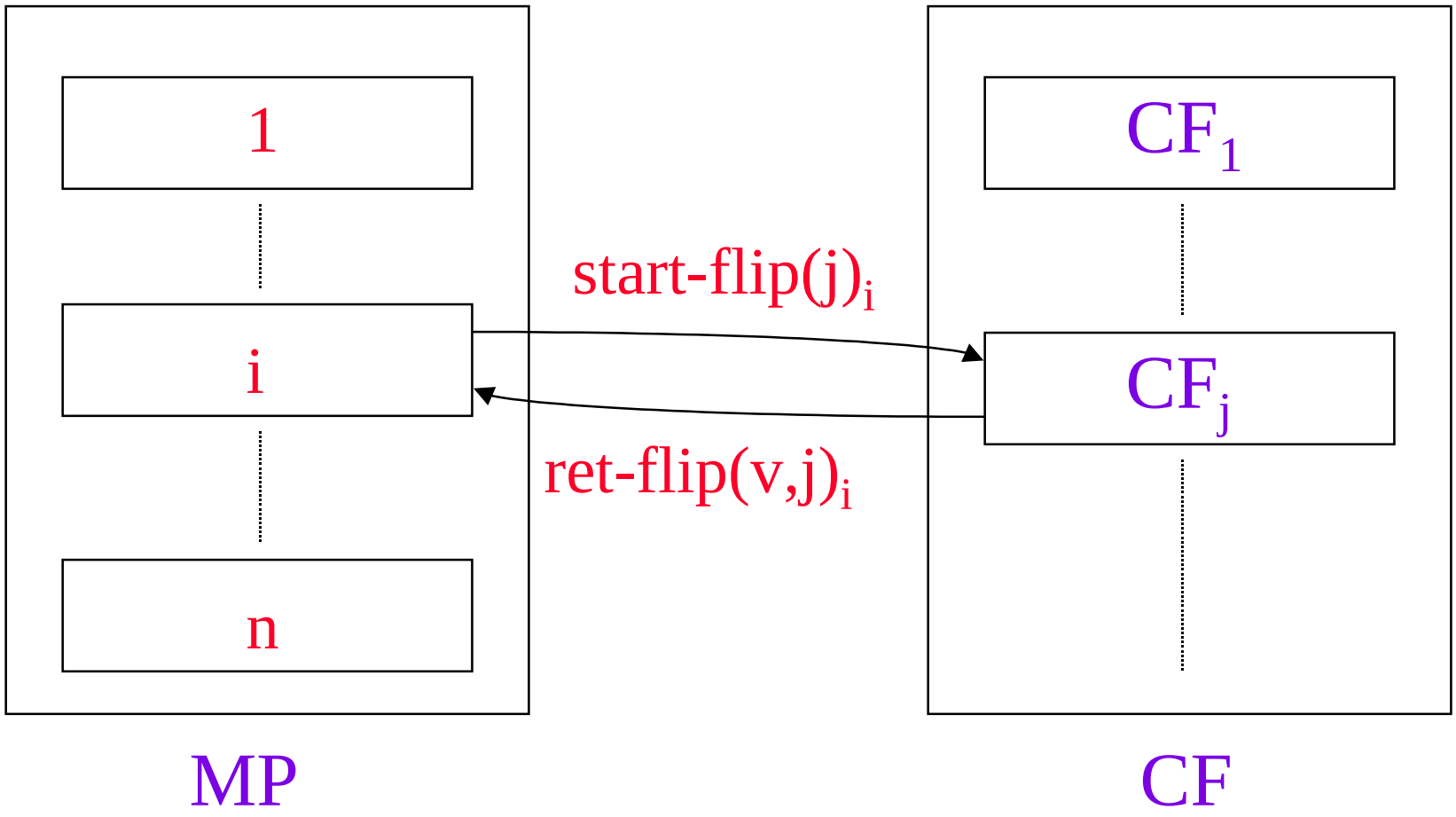
Proprietà richieste

Validity: decidere valori precedentemente proposti.

Agreement: no decisioni diverse.

Wait-Free Termination: ogni processo non guasto prima o poi decide.

Struttura del Protocollo



Correttezza: Progresso

Assumendo invocazioni a CF su porte non guaste ottiene risposte (M1)

$$R \xrightarrow{1} F_0 \cup F_1 \cup D$$

Assumendo risposte a round r sono 0 (dove, per $s \in F_0$, max-round(s) = r) (M2)

$$F_0 \xrightarrow{2} D$$

Assunzioni su CF

Ogni lanciatore di monete soddisfa:

C1: ogni invocazione su porta non guasta ottiene una risposta con probabilità 1.

C2: fissato v in {0,1} la probabilità che tutte le risposte siano v è almeno p.

Si dimostra che p non dipende da n.

Combinazione delle Proprietà

Da C1 e C2 e M1 e M2,

$$R \xrightarrow[1]{1} F_0 \cup F_1 \cup D$$

$$F_0 \xrightarrow[p]{2\ 1} D$$

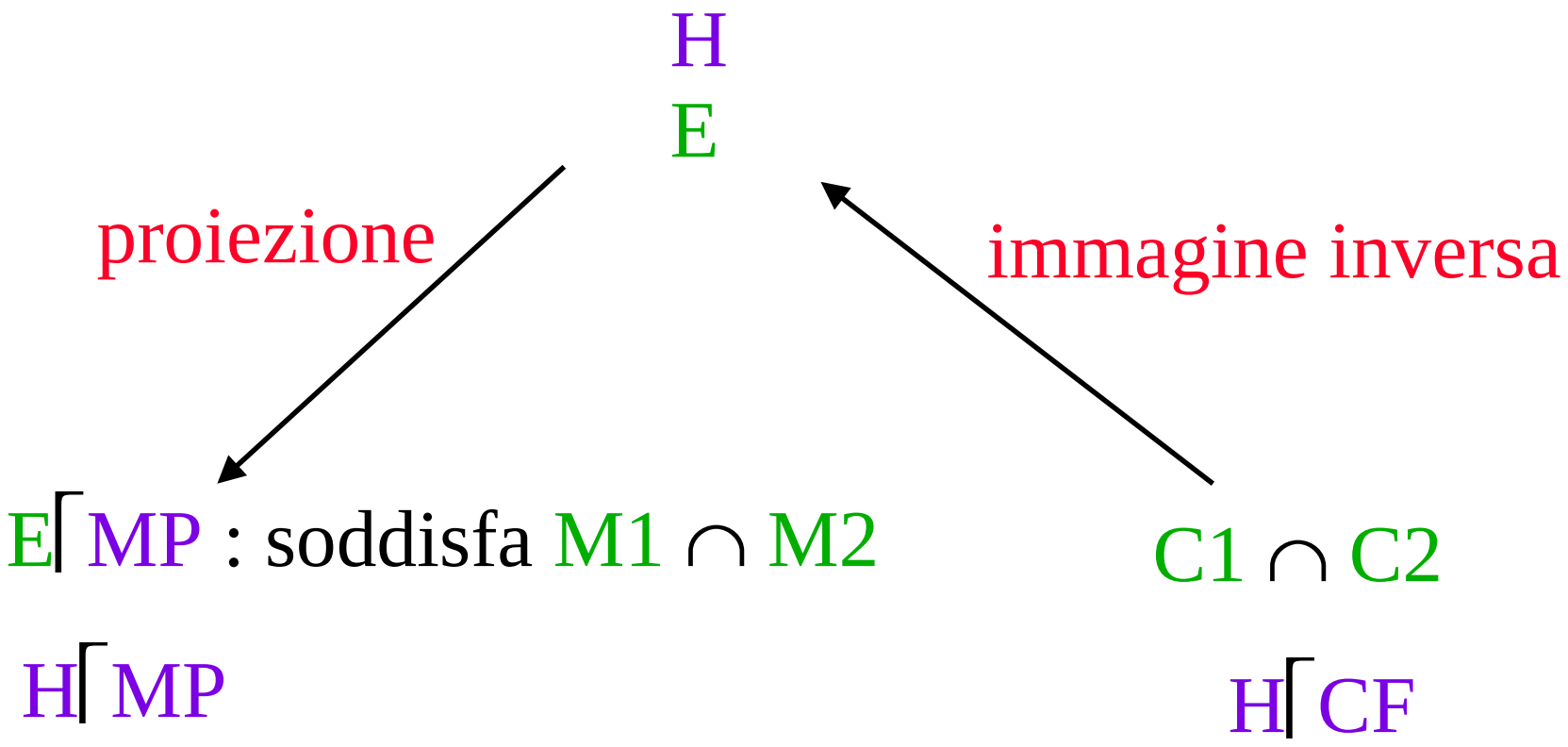
Combinando le proprietà di cui sopra

$$R \xrightarrow[p]{3} D$$

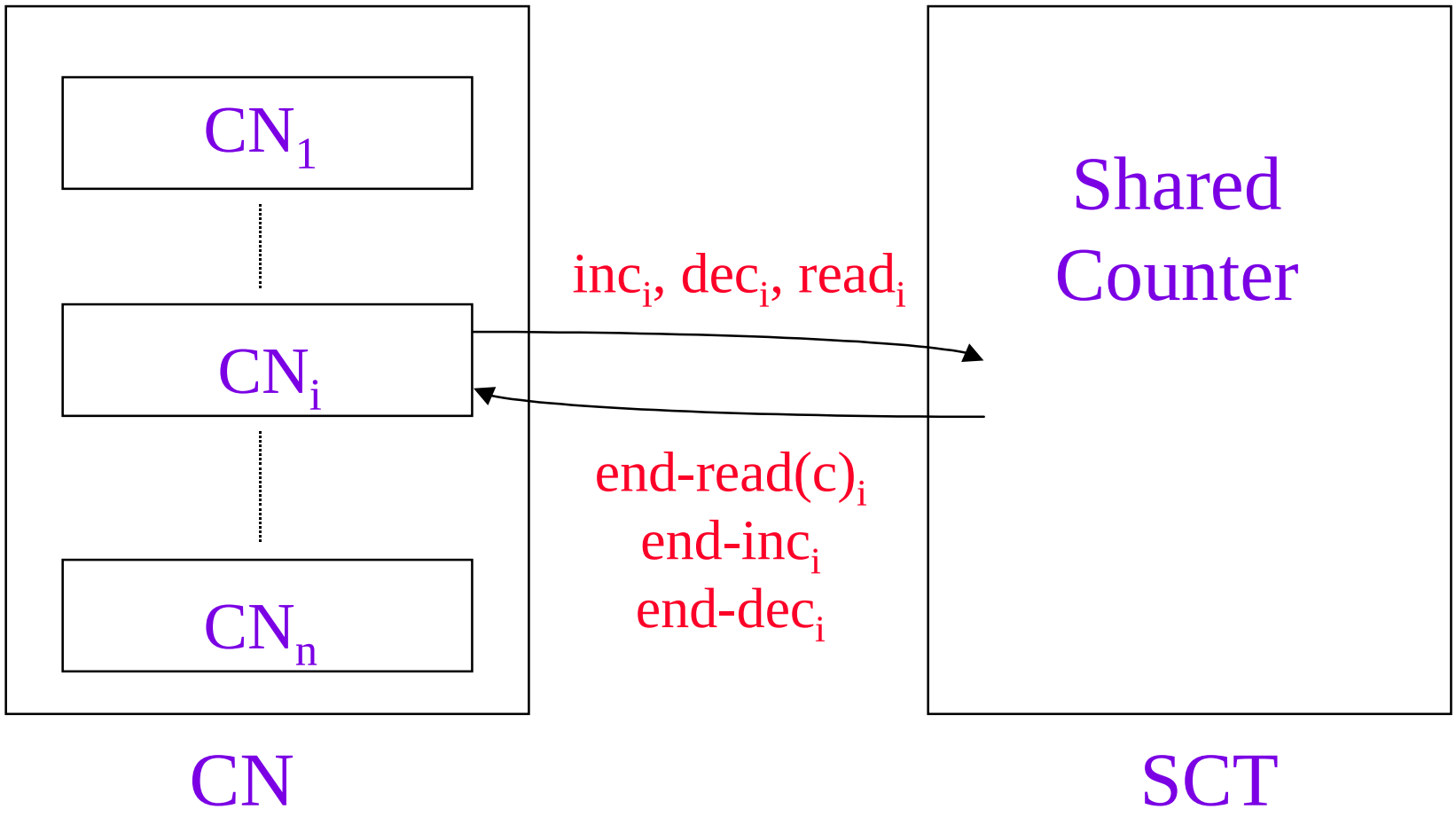
Quindi, con complessità attesa 3/p l'algoritmo termina.

Argomento Formale

Dato H con stato iniziale s di F_0.



Lanciatore di Monete



Uso delle Simulazioni

- Si dimostra la correttezza di CN||SCT
- Si costruisce un'implementazione distribuita di SCT (ACT)
- Utilizzando risultati noti su registri atomici si dimostra ACT ≤ SCT
- Per composizionalità esiste una simulazione probabilistica tra il protocollo atomico e il protocollo centralizzato
- Le proprietà del consenso sono esprimibili mediante tracce

Conclusioni

- Resta da fare
  - Integrazione modello ibrido e stocastico
  - Studio della liveness
  - Studio di linguaggi
  - Identificazione di altri casi di studio
  - Tecniche di verifica automatica
    - Model checking
    - Theorem proving
    - Integrazione di model checking e theorem proving
  - Altre aree
    - Valutazione della performance
    - Sicurezza