Reasoning on Labelled Petri Nets and Their Dynamics in a Stochastic Setting

Sander Leemans, Fabrizio Maggi, Marco Montali

RWTH AACHEN UNIVERSITY

unibz

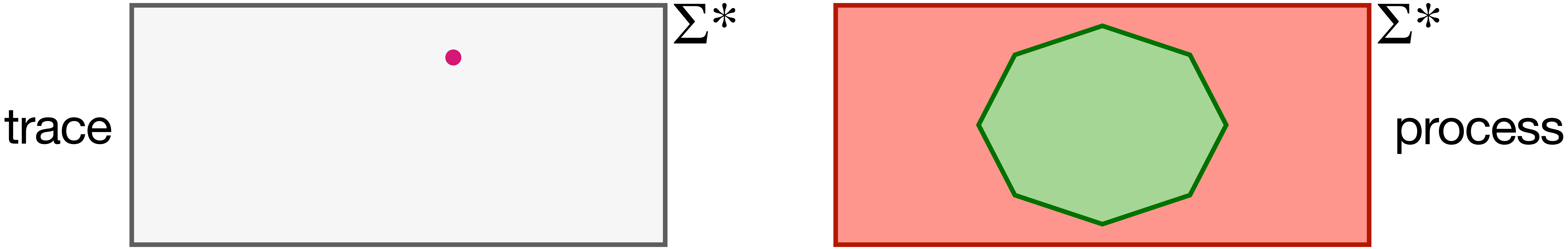
BPM 2022, Münster

analytics

analytics

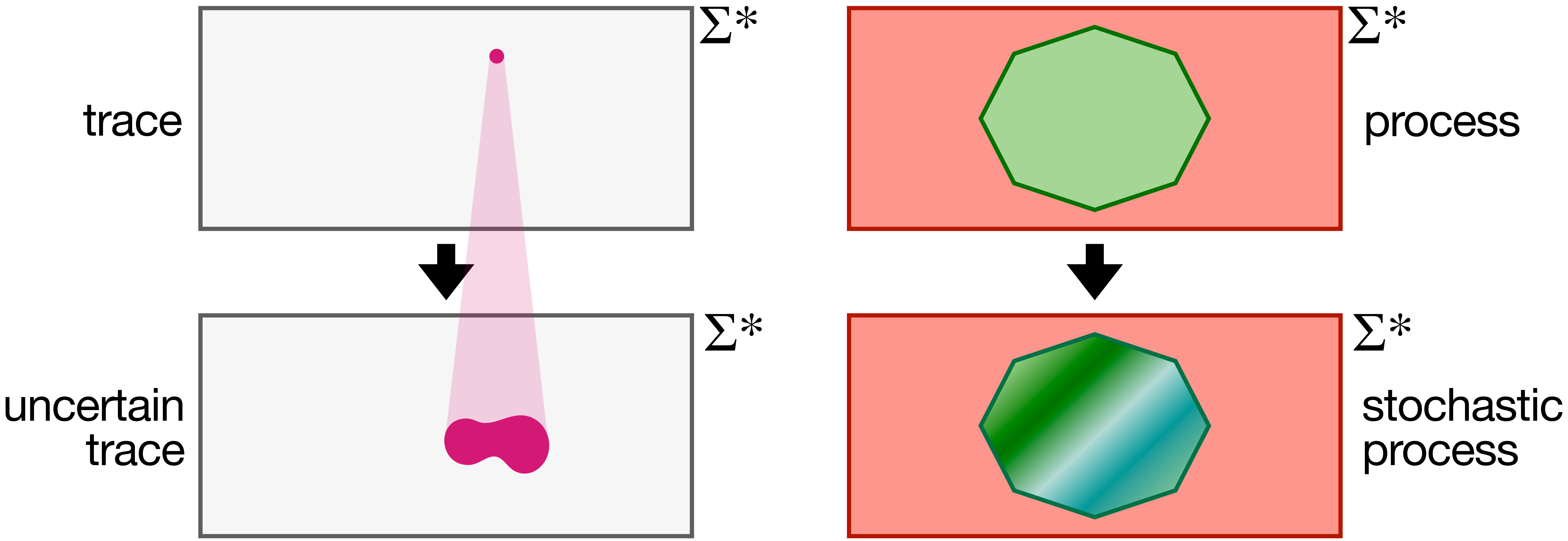
Stochastic process mining

Revived interest in stochastic processes



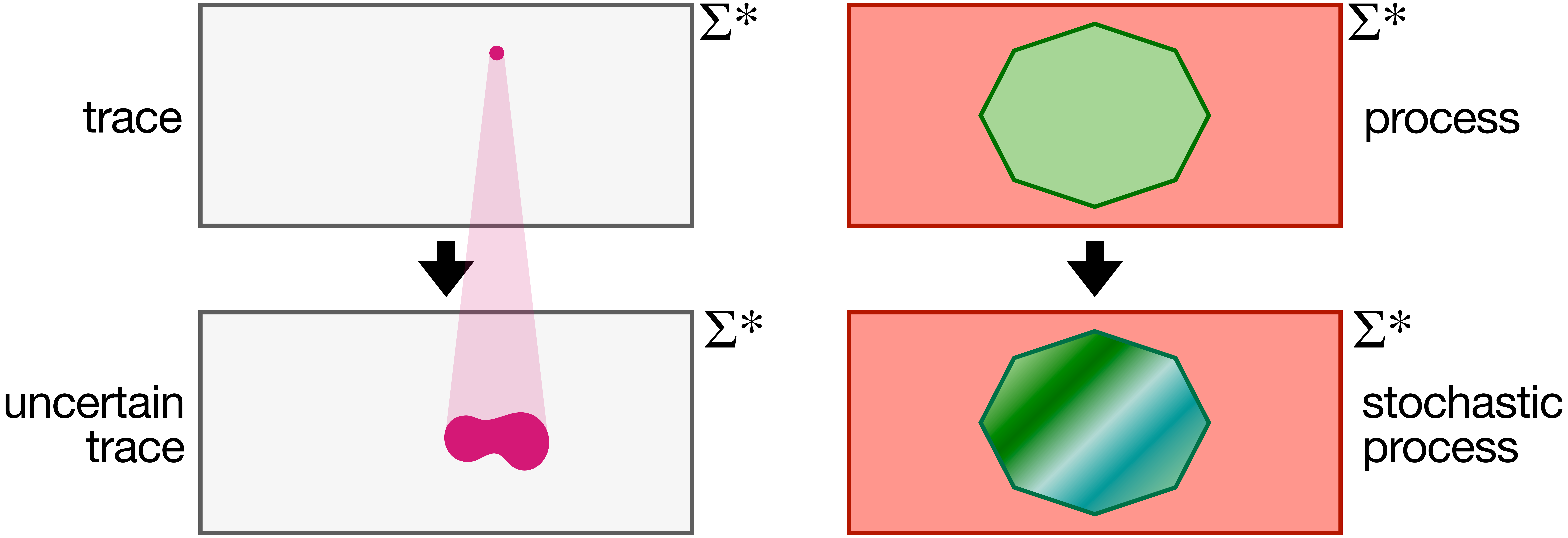
Stochastic process mining

Revived interest in stochastic processes



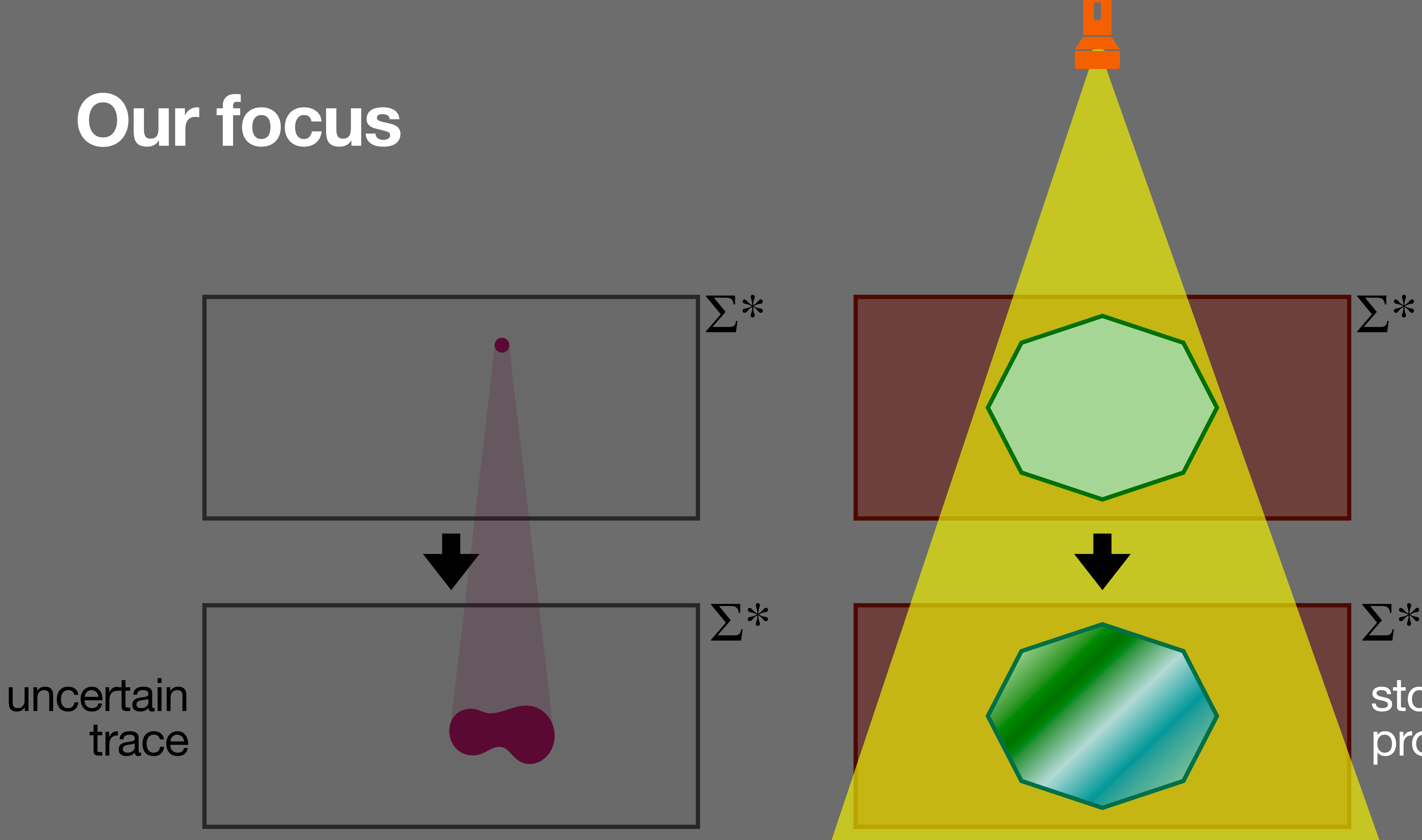
Stochastic process mining

Revived interest in stochastic processes



stochastic conformance checking, probabilistic trace alignment, probabilistic declarative process mining

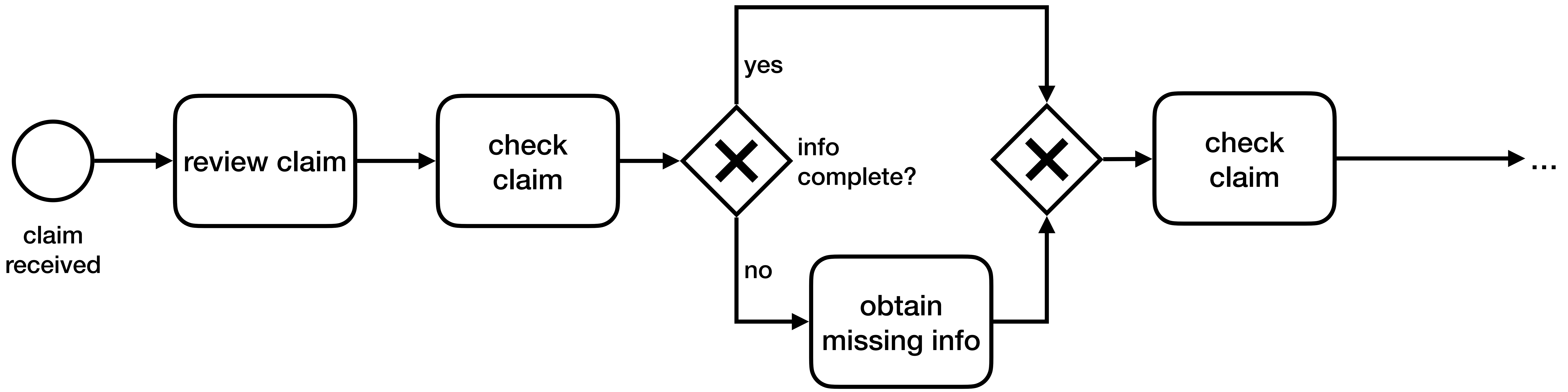
Our focus



stochastic conformance checking, probabilistic trace alignment, probabilistic declarative process mining

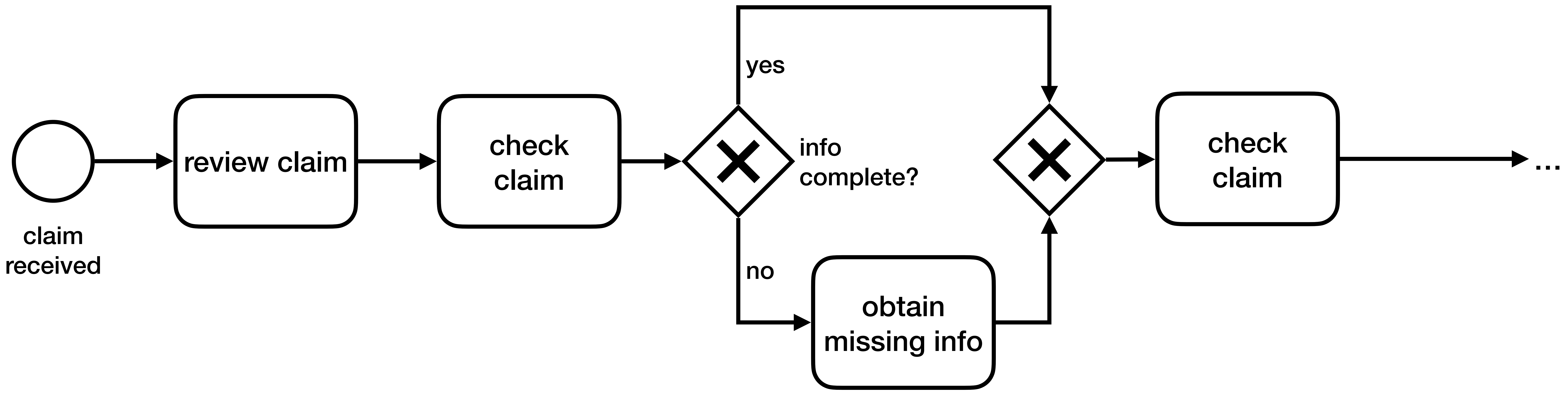
Process control-flow with Petri nets

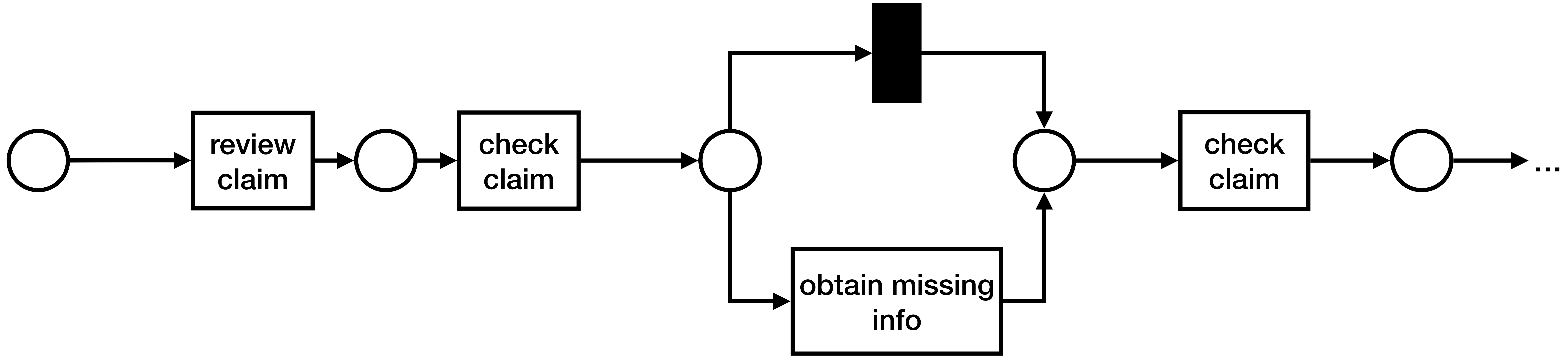
Characteristics



Process control-flow with Petri nets

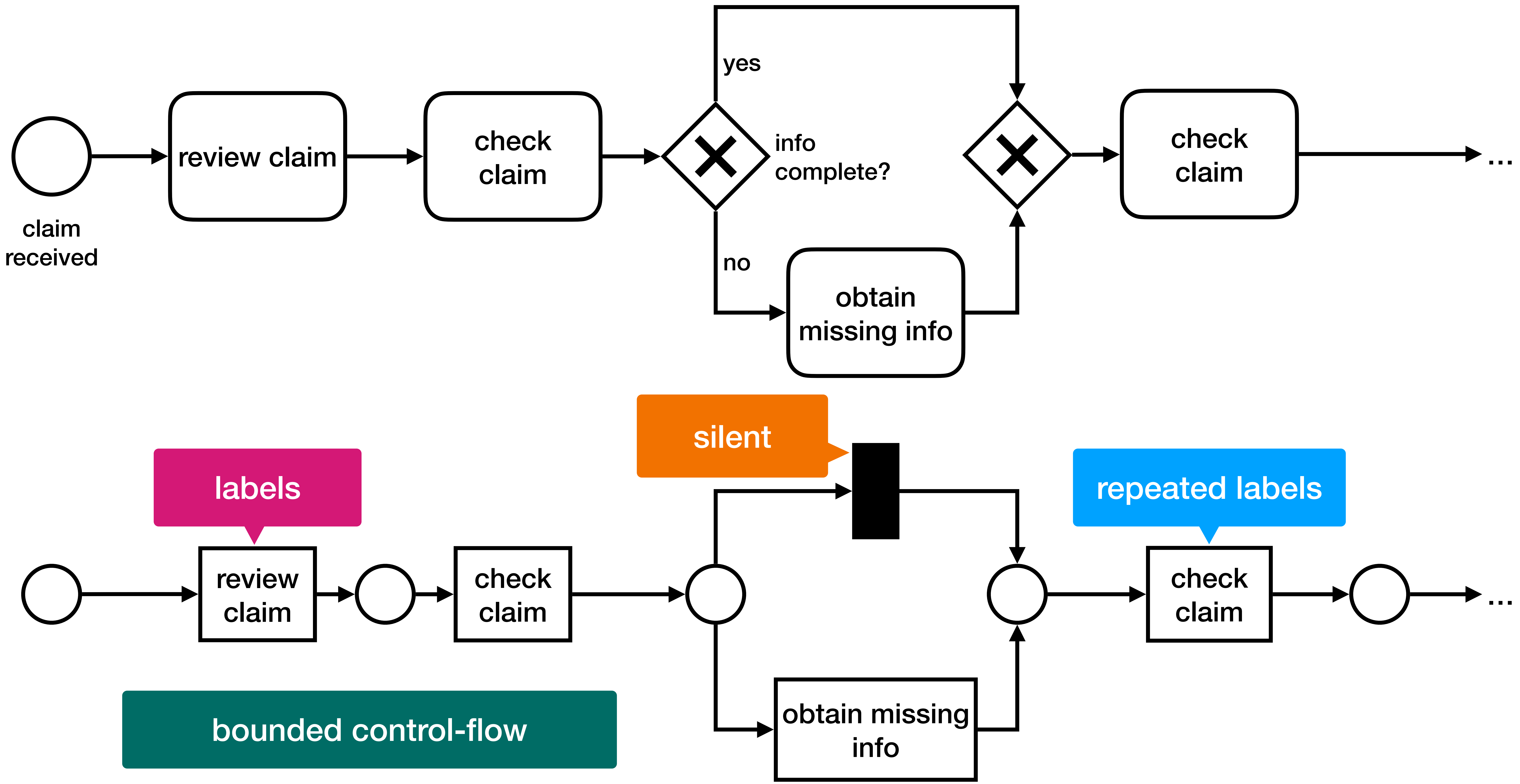
Characteristics



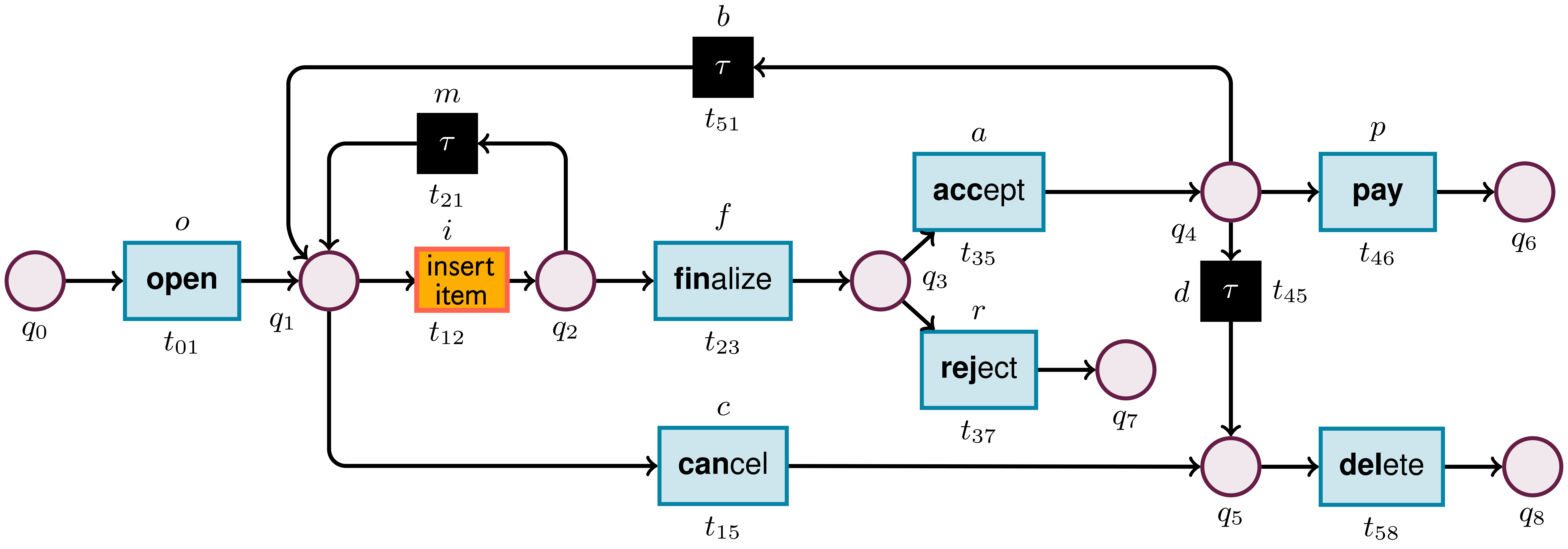


Process control-flow with Petri nets

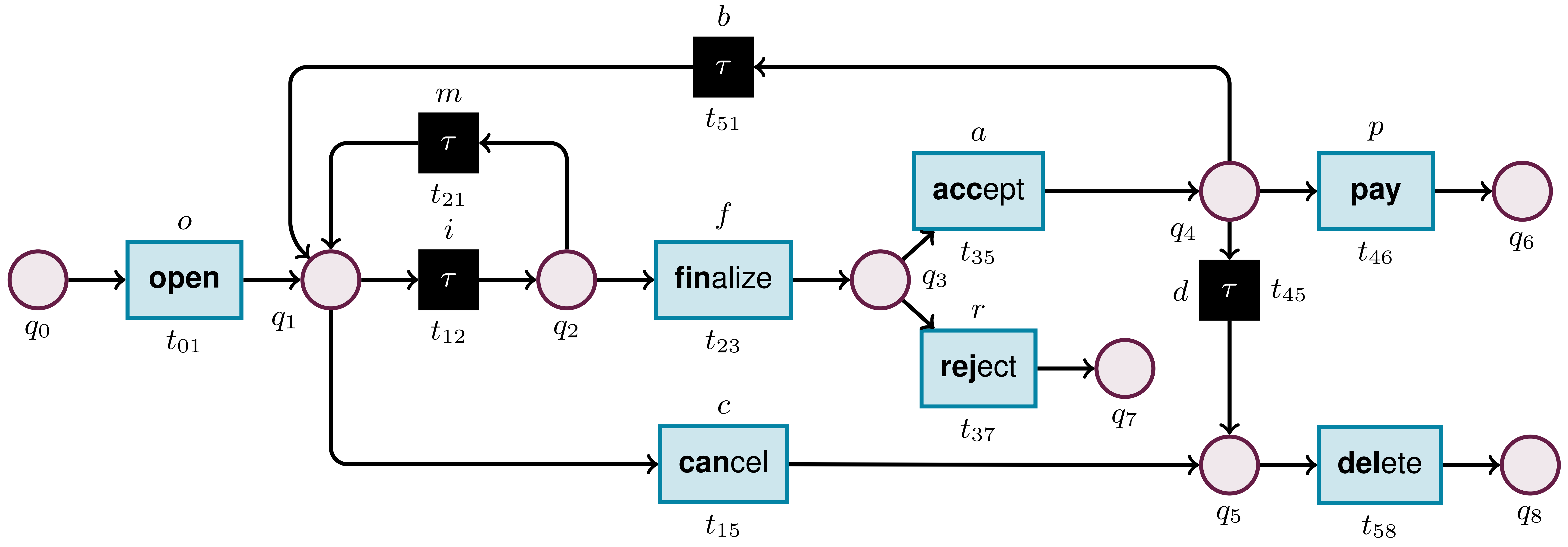
Characteristics



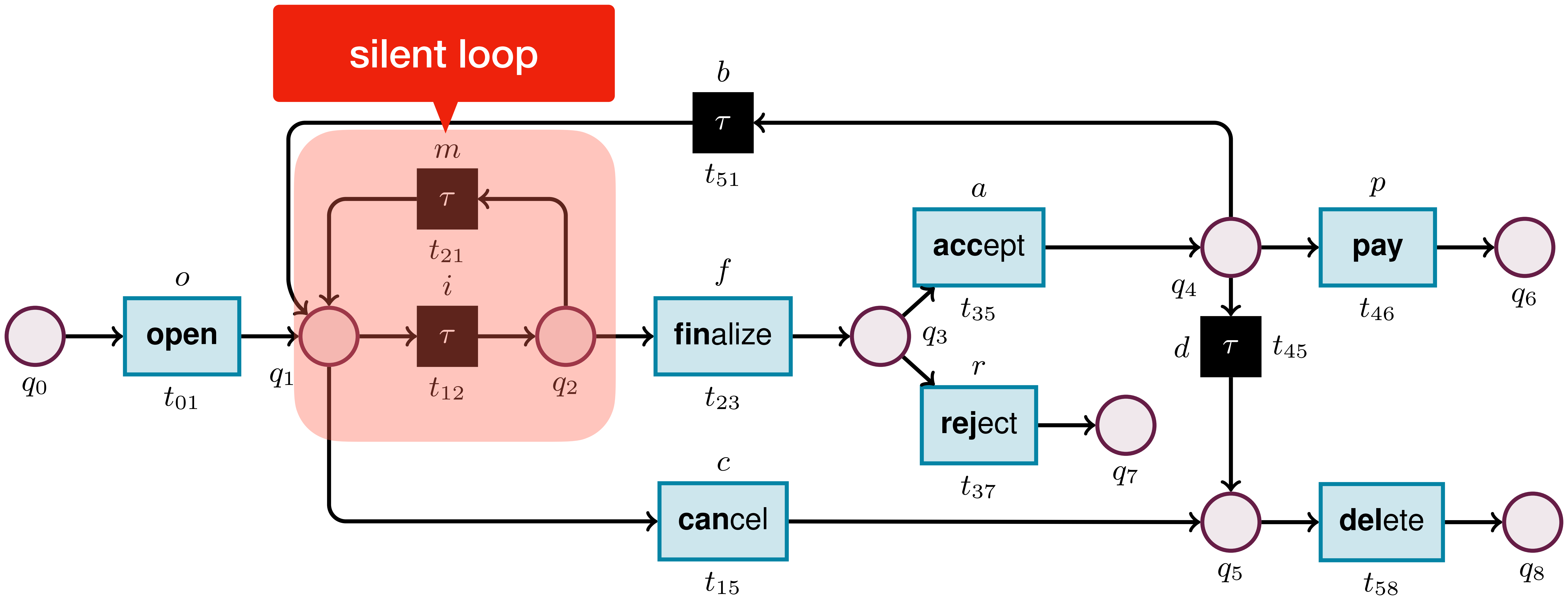
Unlogged tasks as silent transitions



Unlogged tasks as silent transitions



Unlogged tasks as silent transitions

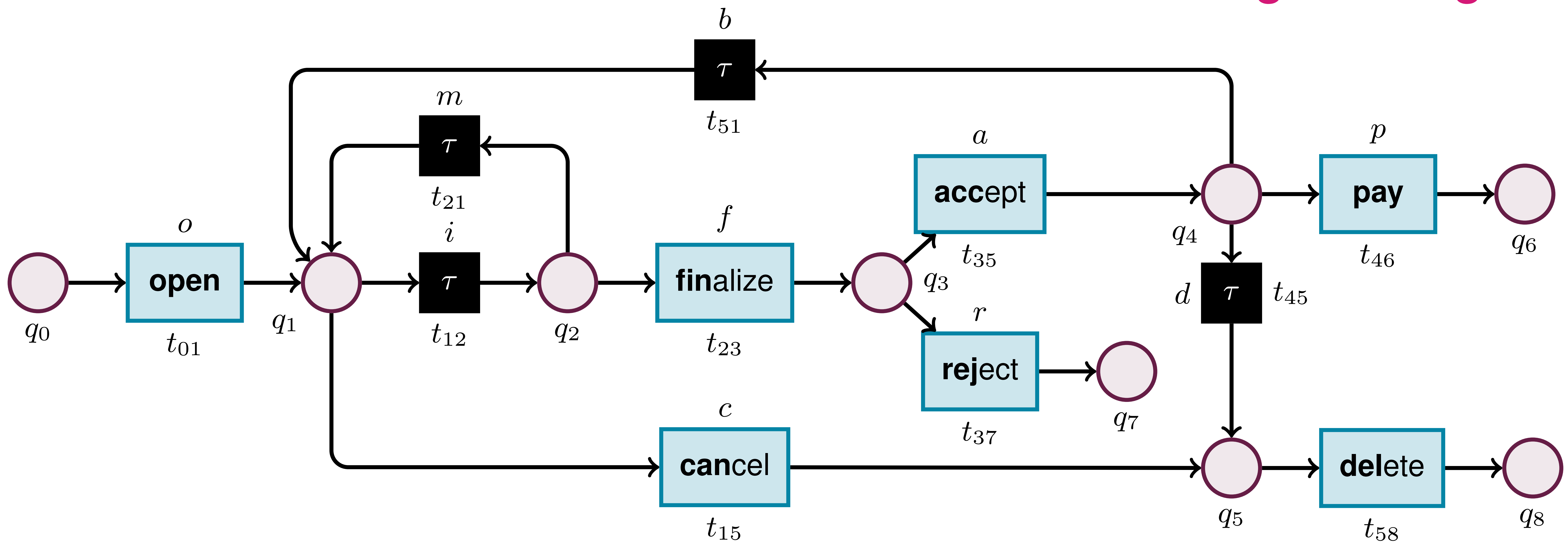


Semantics via finite traces

Start, end(s)

1 marking as **initial state**

1+ deadlocking markings as **final states**

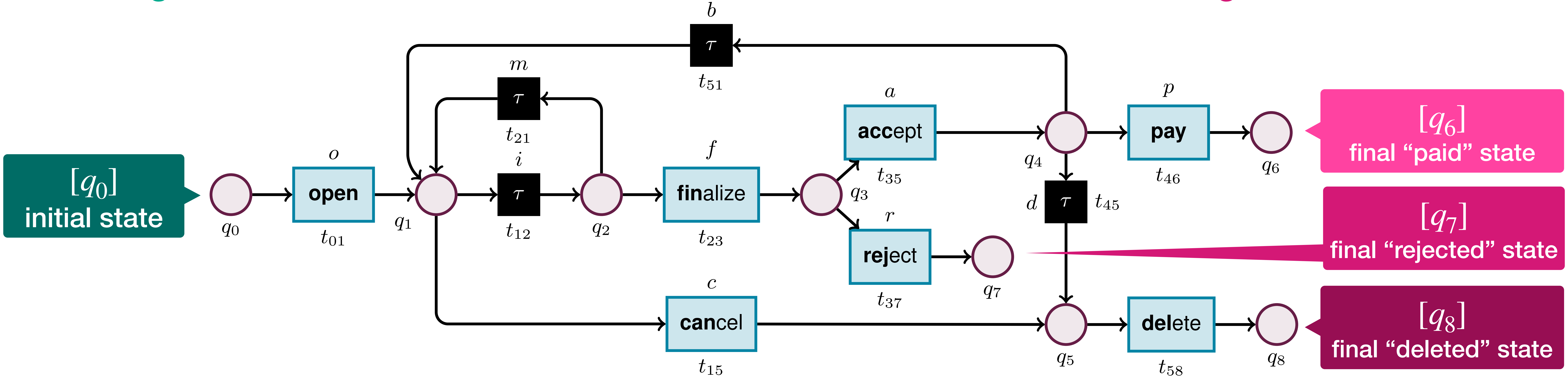


Semantics via finite traces

Start, end(s)

1 marking as **initial state**

1+ deadlock markings as **final states**

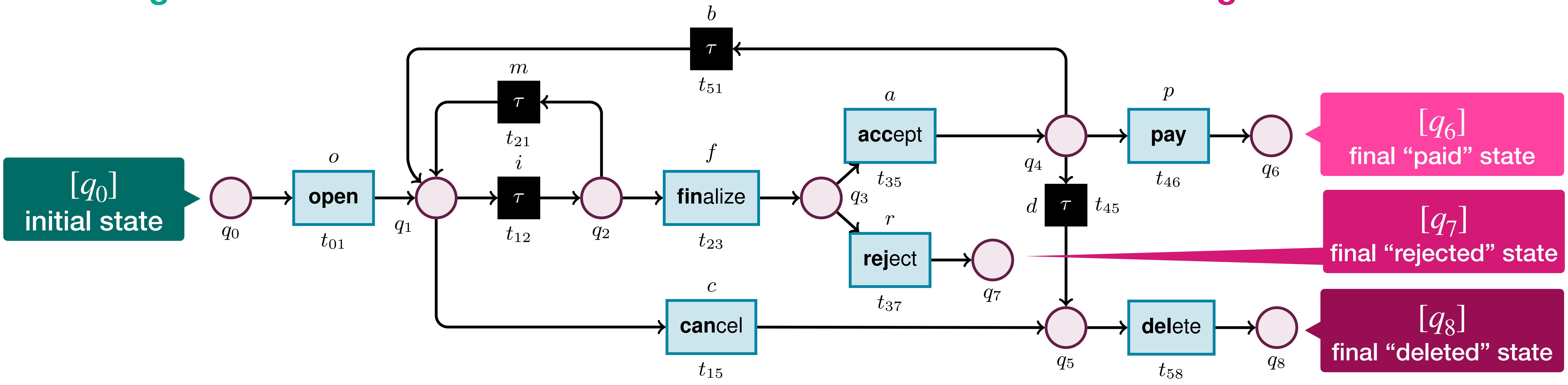


Semantics via finite traces

Runs and traces

1 marking as **initial state**

1+ deadlock markings as **final states**



Run: **valid sequence of transitions** from the initial state to some final state

Trace: **projection of the run** on labels of visible transitions

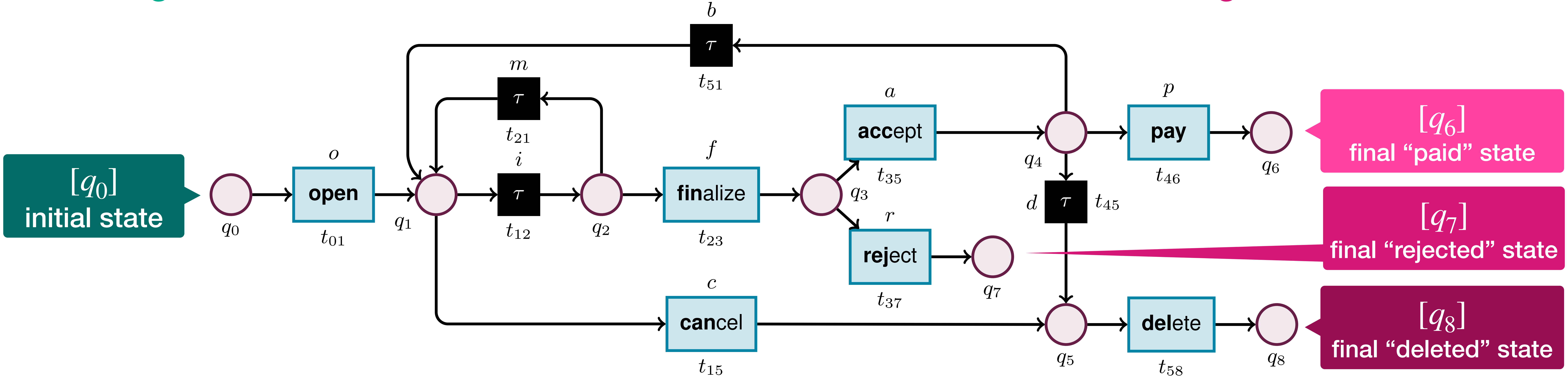
How many runs for the same trace?
Potentially infinitely many!

Semantics via finite traces

Runs and traces

1 marking as **initial state**

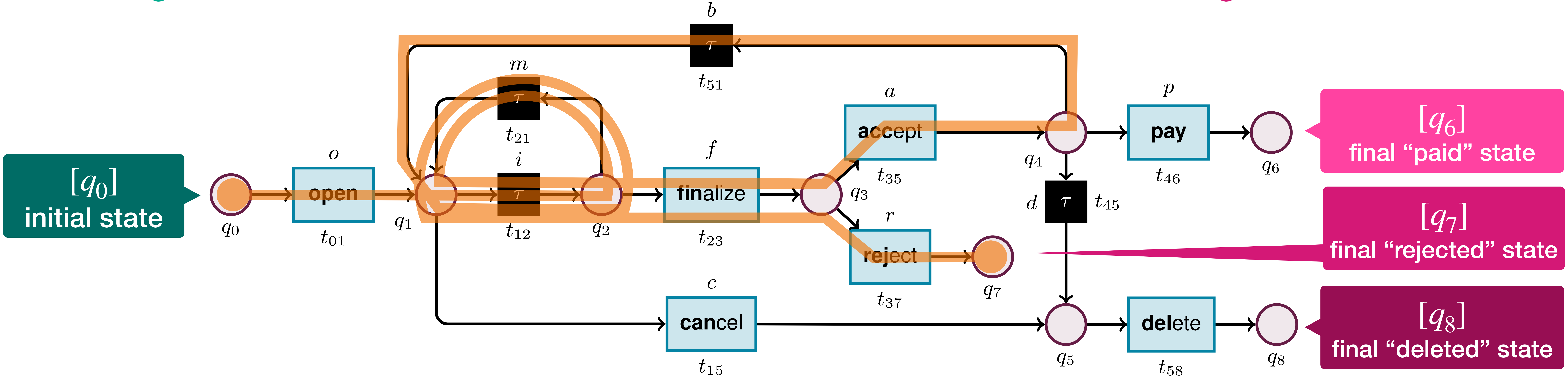
1+ deadlock markings as **final states**



Semantics via finite traces

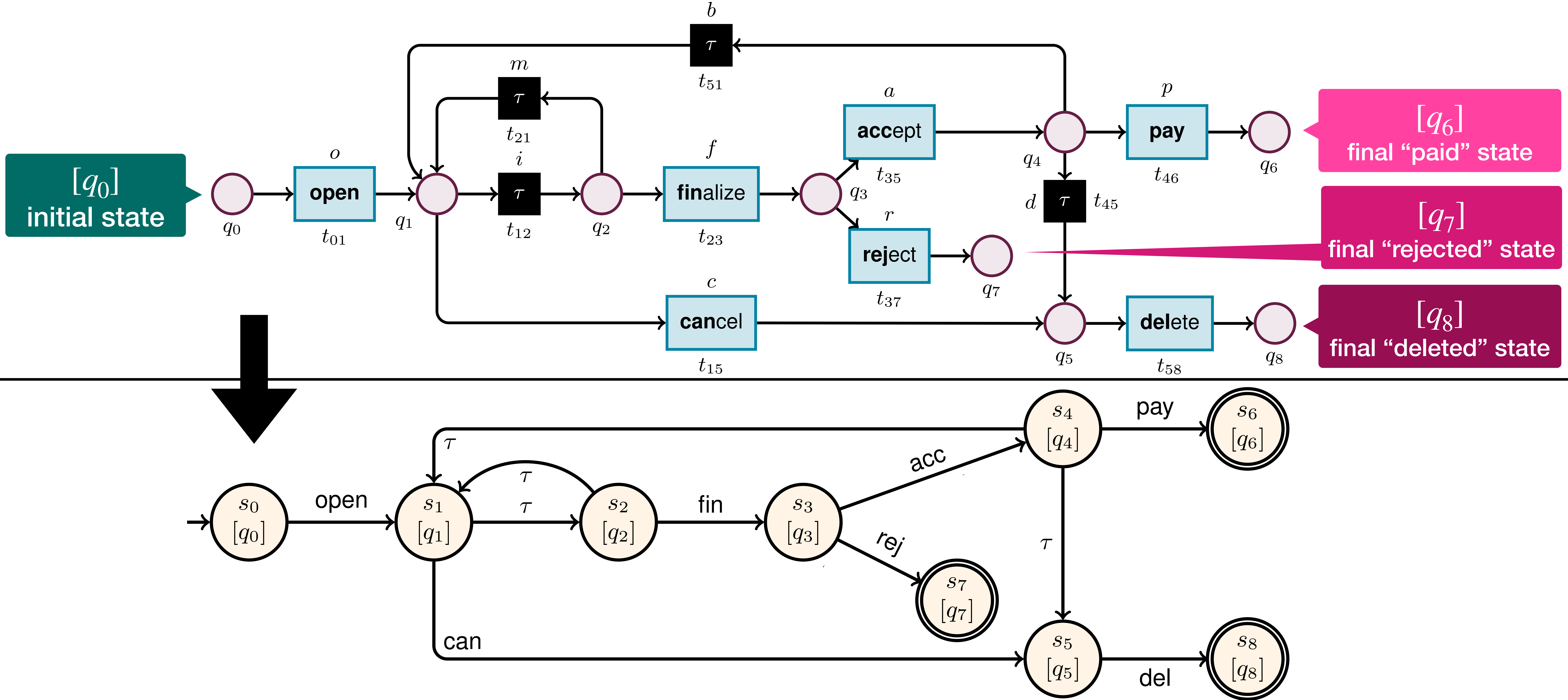
Runs and traces

1 marking as **initial state** **1+ deadlock markings** as **final states**

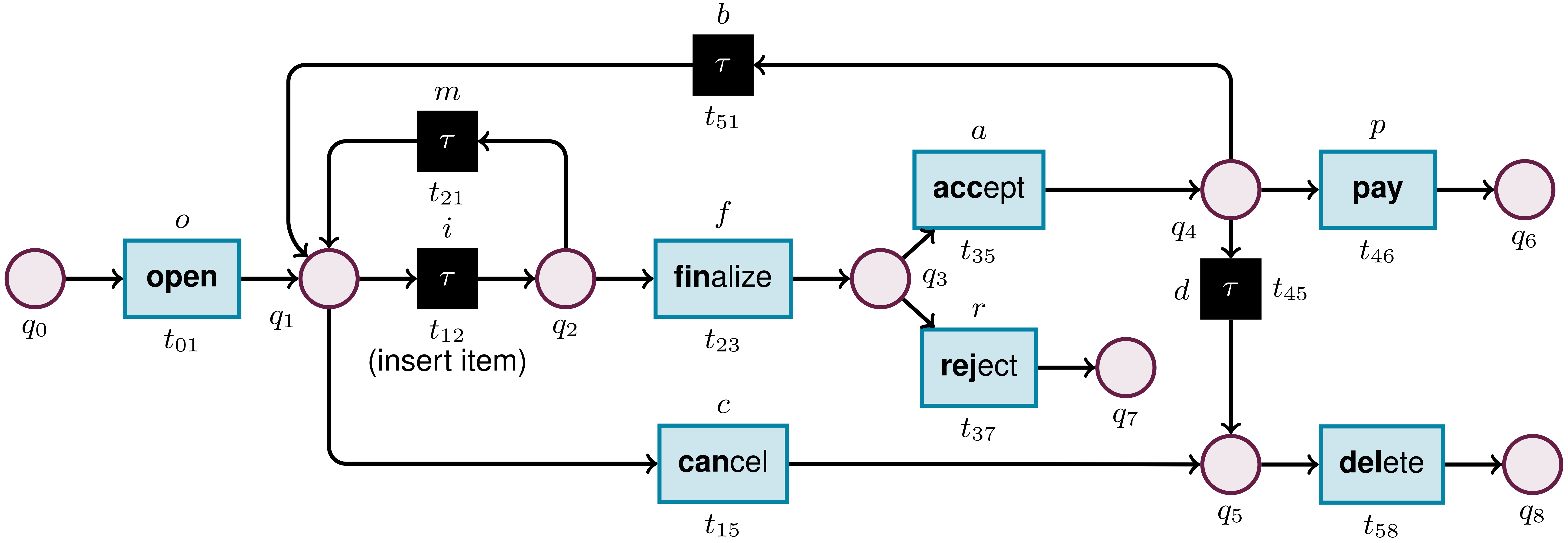


<open, finalize, accept, finalize, reject>

From net to transition system



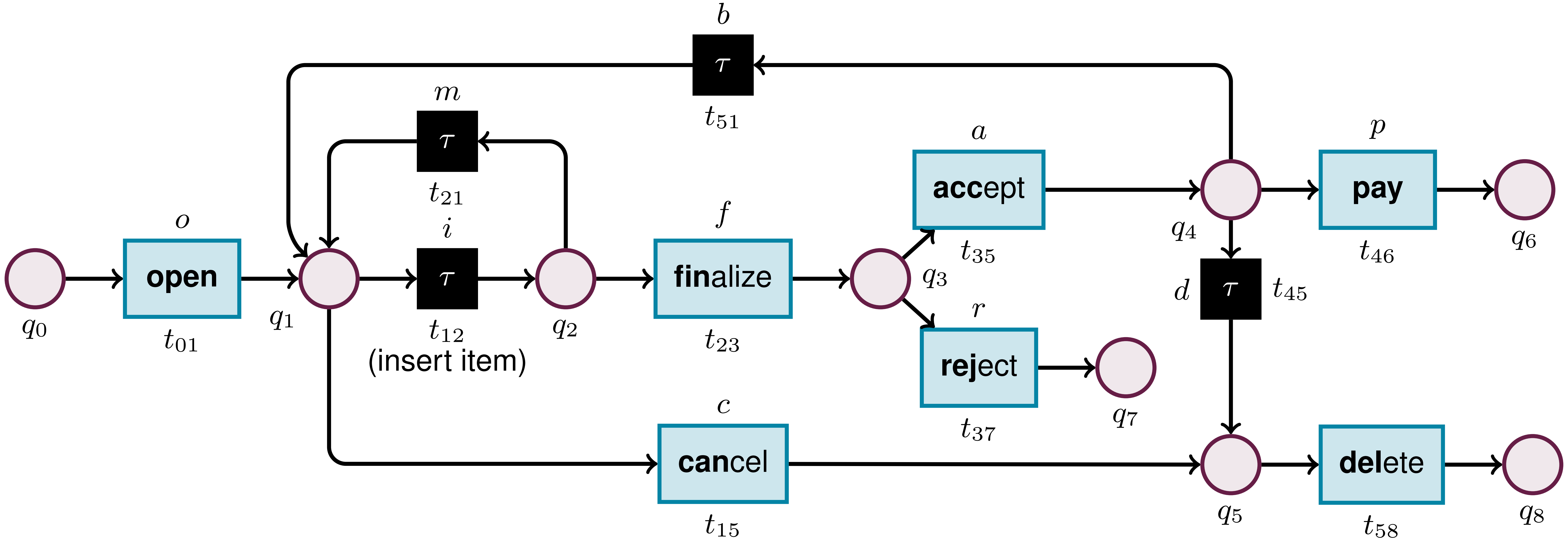
Stochastic Petri nets



Every transition gets a **weight**

- **Immediate** transition: **relative likelihood** to fire
- **Timed** transition: **rate/decay** of exponential distribution for waiting time

Stochastic Petri nets



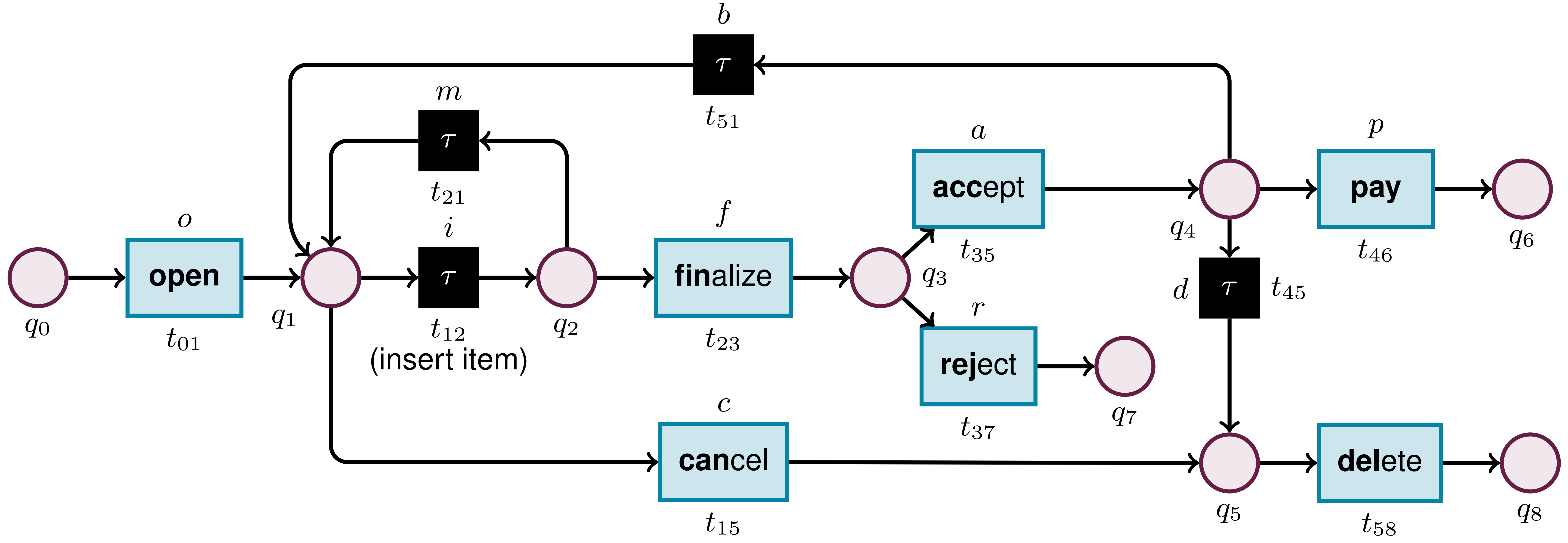
Every transition gets a **weight**

- **Immediate** transition: **relative likelihood** to fire
- **Timed** transition: **rate/decay** of exponential distribution for waiting time

Our interest: **transition firings -> ordering without time**

Stochastic Petri nets

Focus: next transition firing/completion

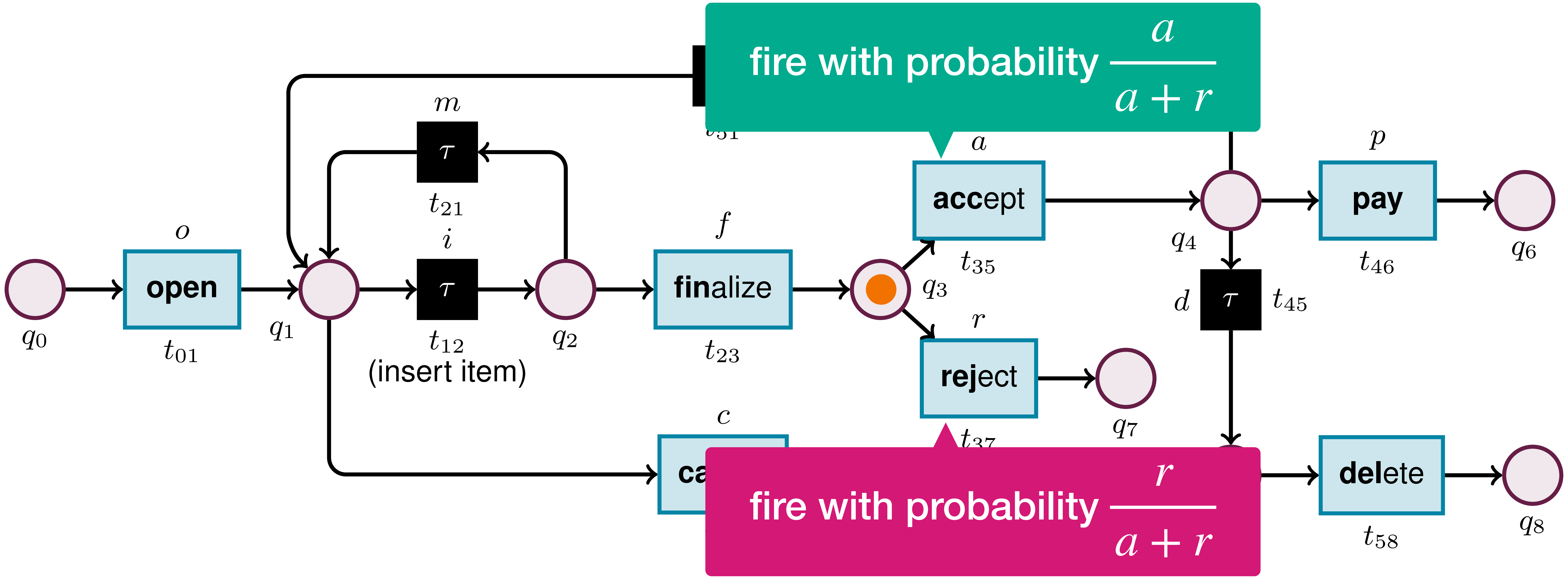


$$\text{P(fire } \mathbf{t} \text{ enabled in } \mathbf{m} \mid \text{marking } \mathbf{m}) = \frac{\text{weight of } t}{\sum_{t' \text{ enabled in } m} \text{weight of } t'}$$

Works both for set of immediate OR of timed transitions

Stochastic Petri nets

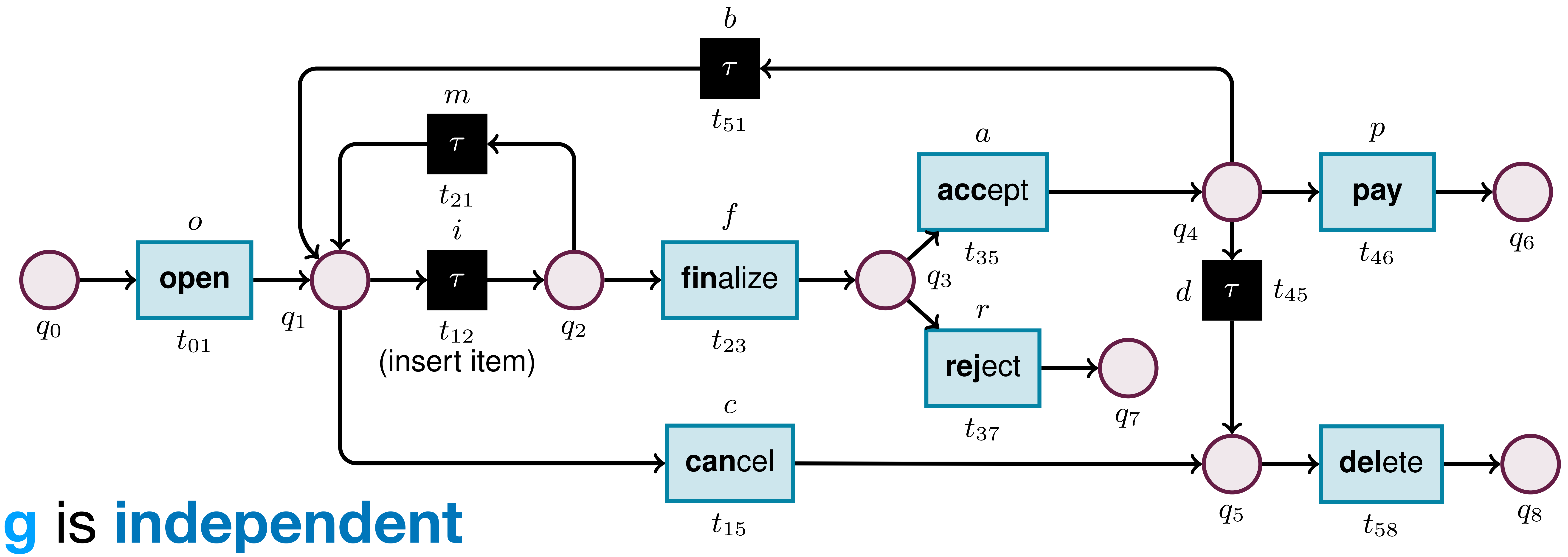
Focus: next transition firing/completion



$$P(\text{fire } \mathbf{t} \text{ enabled in } \mathbf{m} \mid \text{marking } \mathbf{m}) = \frac{\text{weight of } t}{\sum_{t' \text{ enabled in } m} \text{weight of } t'}$$

Stochastic Petri nets

Focus: next transition firing/completion

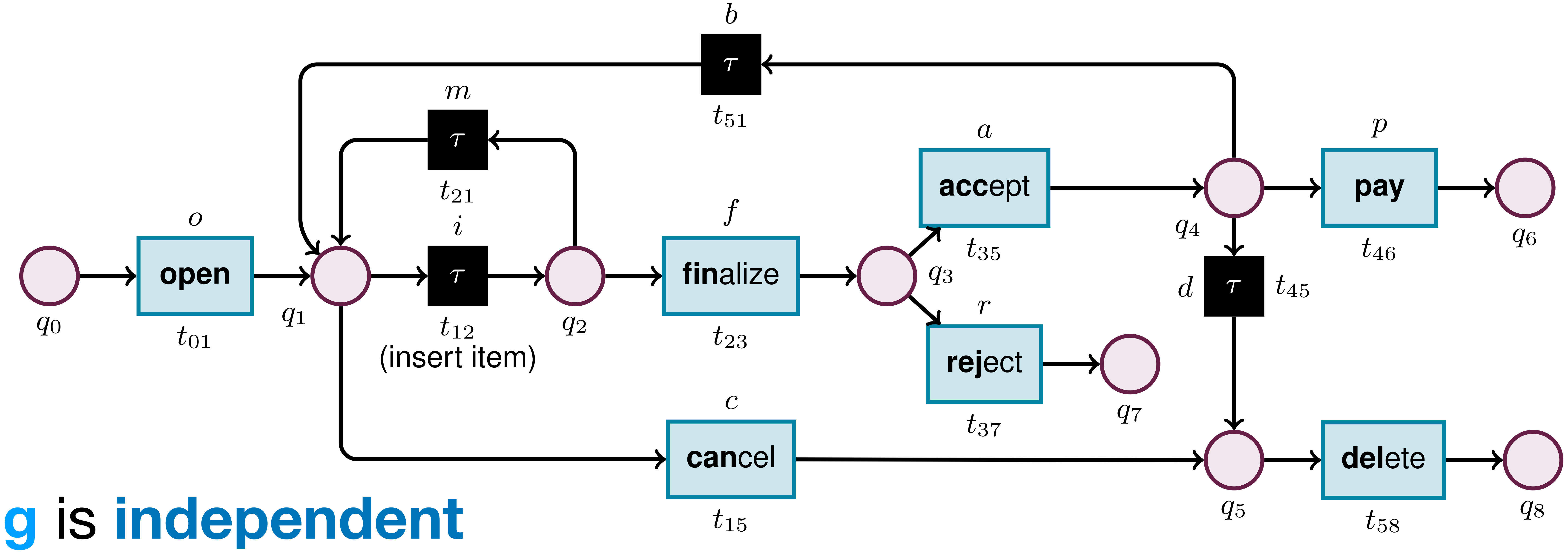


Every **firing** is **independent**

- **Probability** of a **run** = **product** of firing probabilities
- **Probability** of a **trace** = **(possibly infinite) sum** of probabilities of its runs

Stochastic Petri nets

Focus: next transition firing/completion



Every **firing** is **independent**

- **Probability** of a **run** = **product** of firing probabilities
- **Probability** of a **trace** = **(possibly infinite) sum** of probabilities of its runs

So far:

- approximate computation, or
- exact computation for restricted nets (no fully silent loops)

Semantics via stochastic transition systems

Looks familiar?

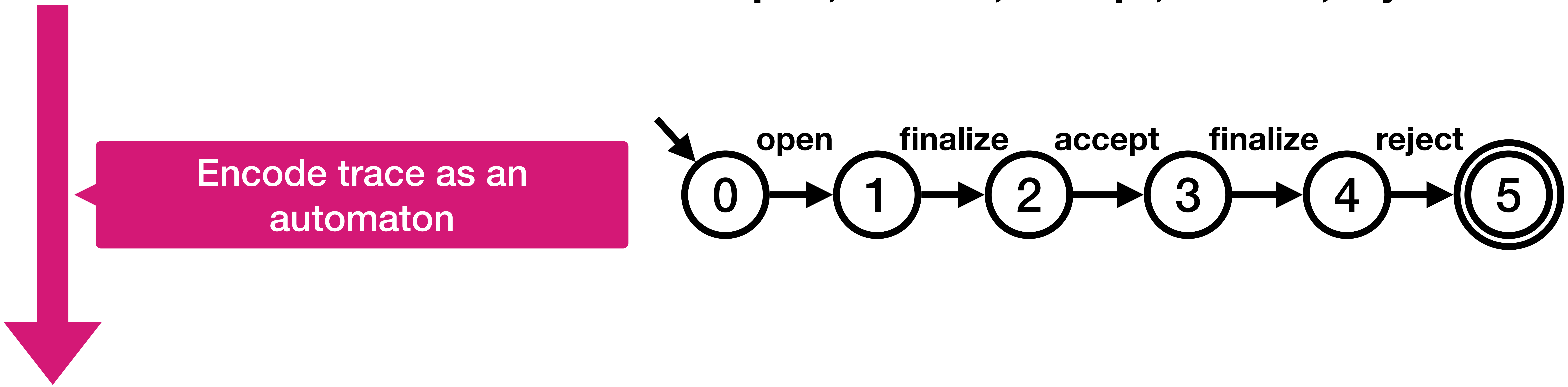
Key problems

1. **Probability of a trace**
2. **Probability of satisfying a qualitative property** expressed in temporal logics over finite traces, or as a finite-state automaton
3. **Conformance to a probabilistic Declare specification** that defines constraint scenarios, each coming with a different probability (or range of probabilities)

Observation

1. **Probability of a trace**

<open, finalize, accept, finalize, reject>



2. **Probability of satisfying a qualitative property** expressed in temporal logics over finite traces, or as a finite-state automaton

Attack strategy

Reasoning on states and probabilities

Reasoning on tasks and transitions

Attack strategy

Reasoning on states and probabilities

Markov chains

Elegant trick to deal with silent transitions

Qualitative model checking

Reasoning on tasks and transitions

Attack strategy

0. **Outcome probability**
   Probability of completing the process in some final states

---

1. **Probability of a trace**
2. **Probability of satisfying a qualitative property**
3. **Conformance to a probabilistic Declare specification**

Reasoning on states and probabilities

Markov chains

Elegant trick to deal with silent transitions

Qualitative model checking

Reasoning on tasks and transitions

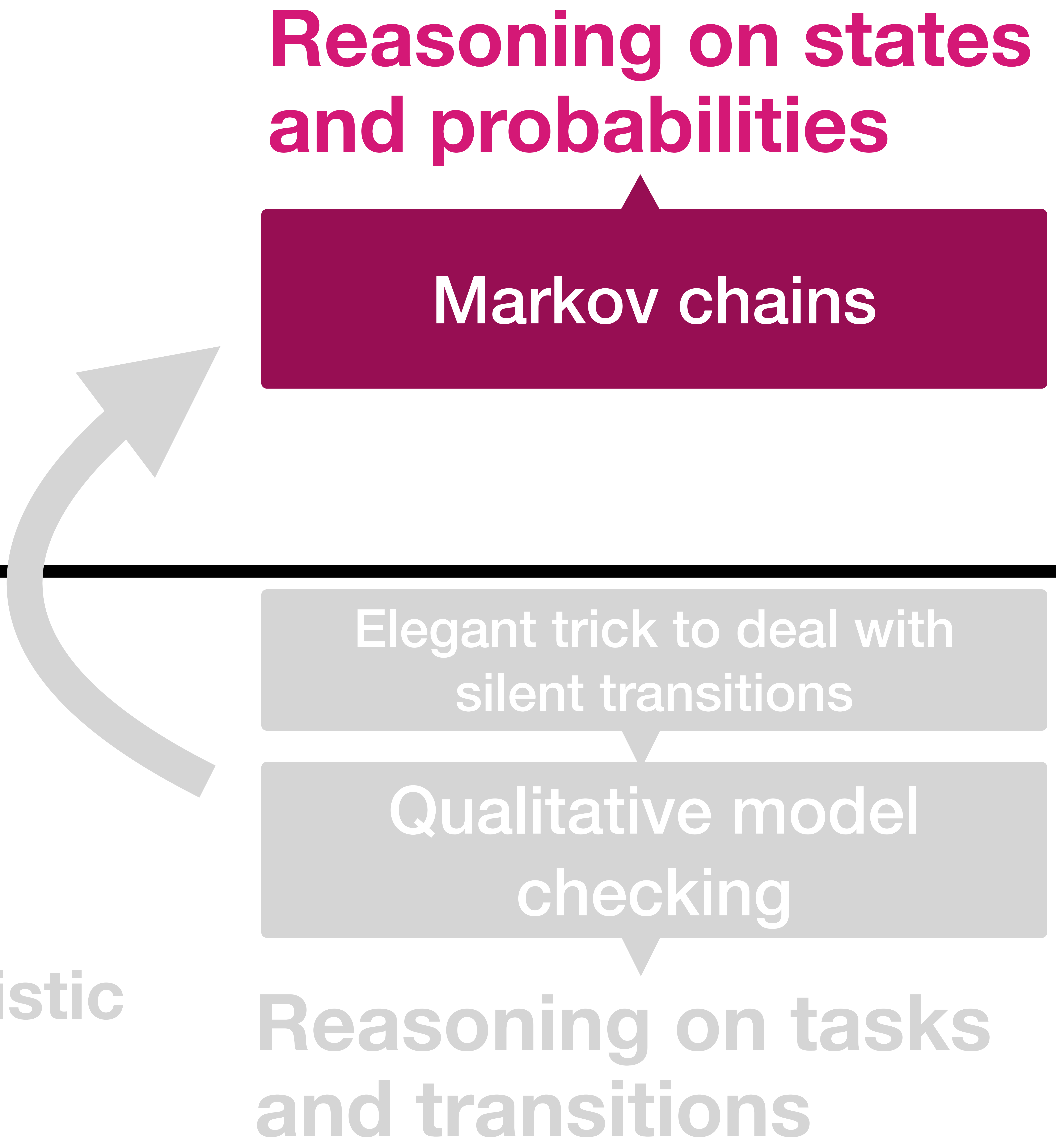
Attack strategy

0. **Outcome probability**
   Probability of completing the process in some final states

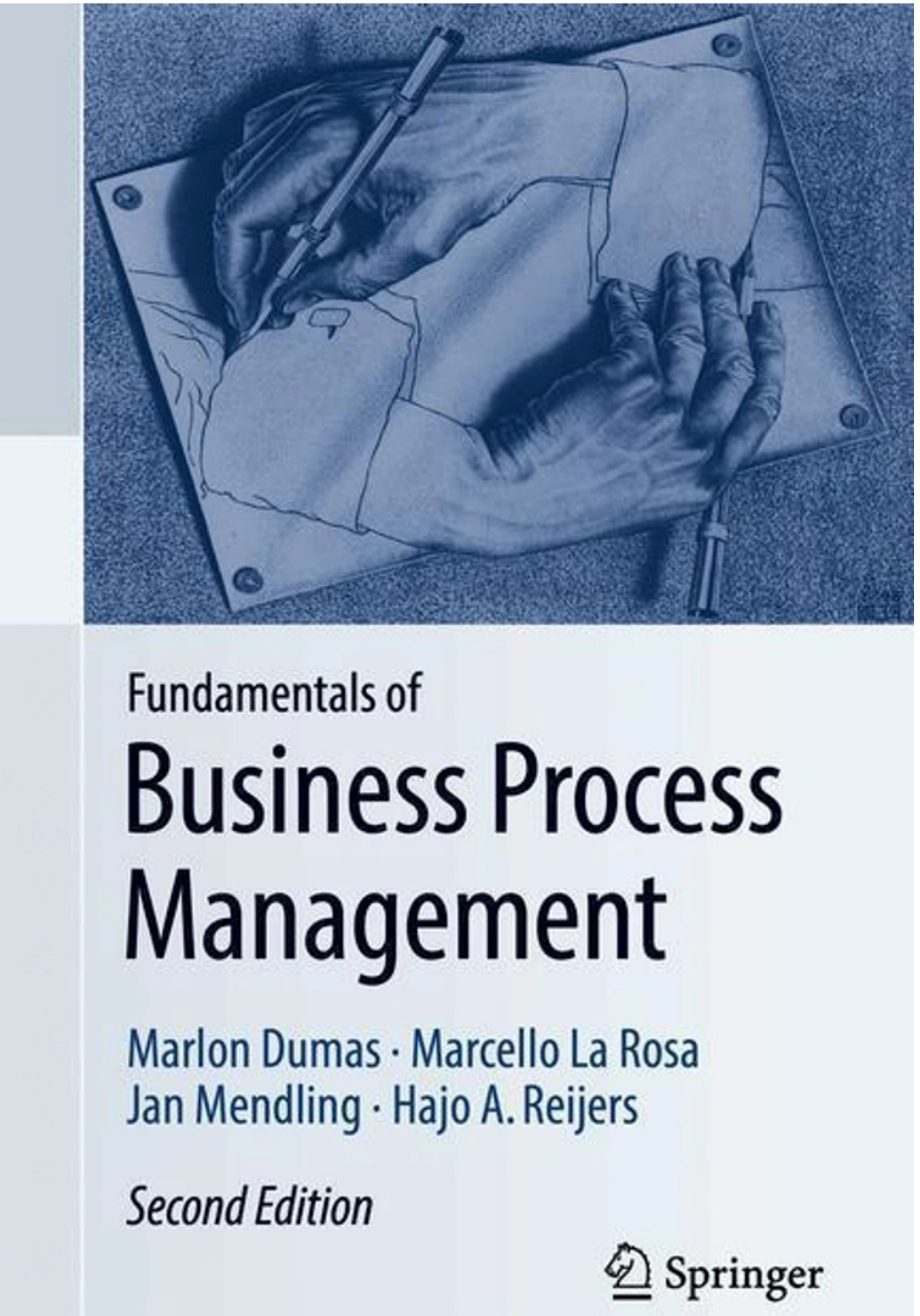
1. **Probability of a trace**

2. **Probability of satisfying a qualitative property**

3. **Conformance to a probabilistic Declare specification**



Recall the good-old days of studying?



Recall the good-old days of studying?

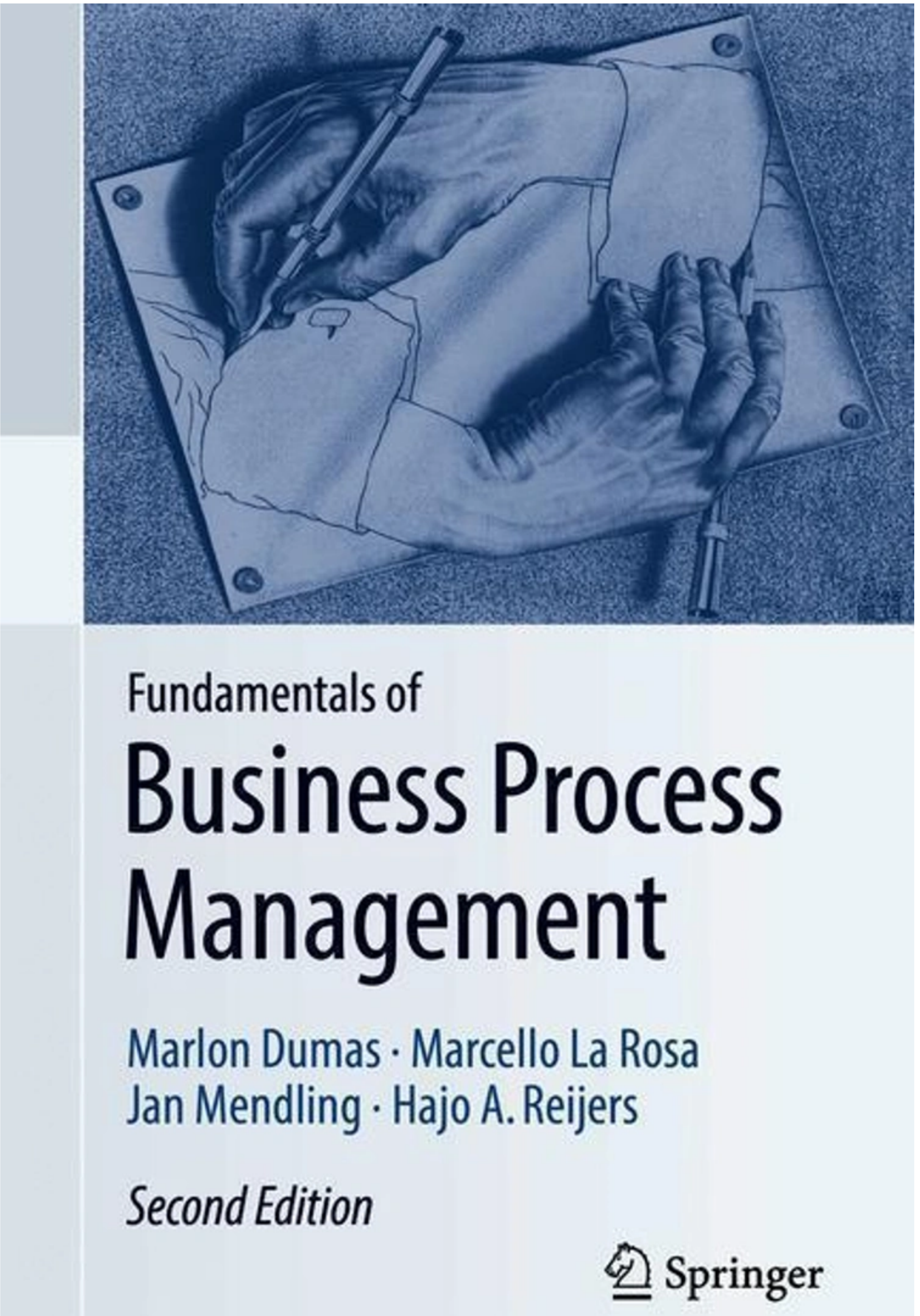


Fig. 7.8 Rework pattern

Hence, the average number of times that B is expected to be executed is $1/(1-0.2) = 1.25$. Now, if we multiply this expected number of instances of B times the cycle time of task B, we get $1.25 \times 20 = 25$. Thus the total cycle time of the process in Figure 7.7 is $10 + 25 = 35$.

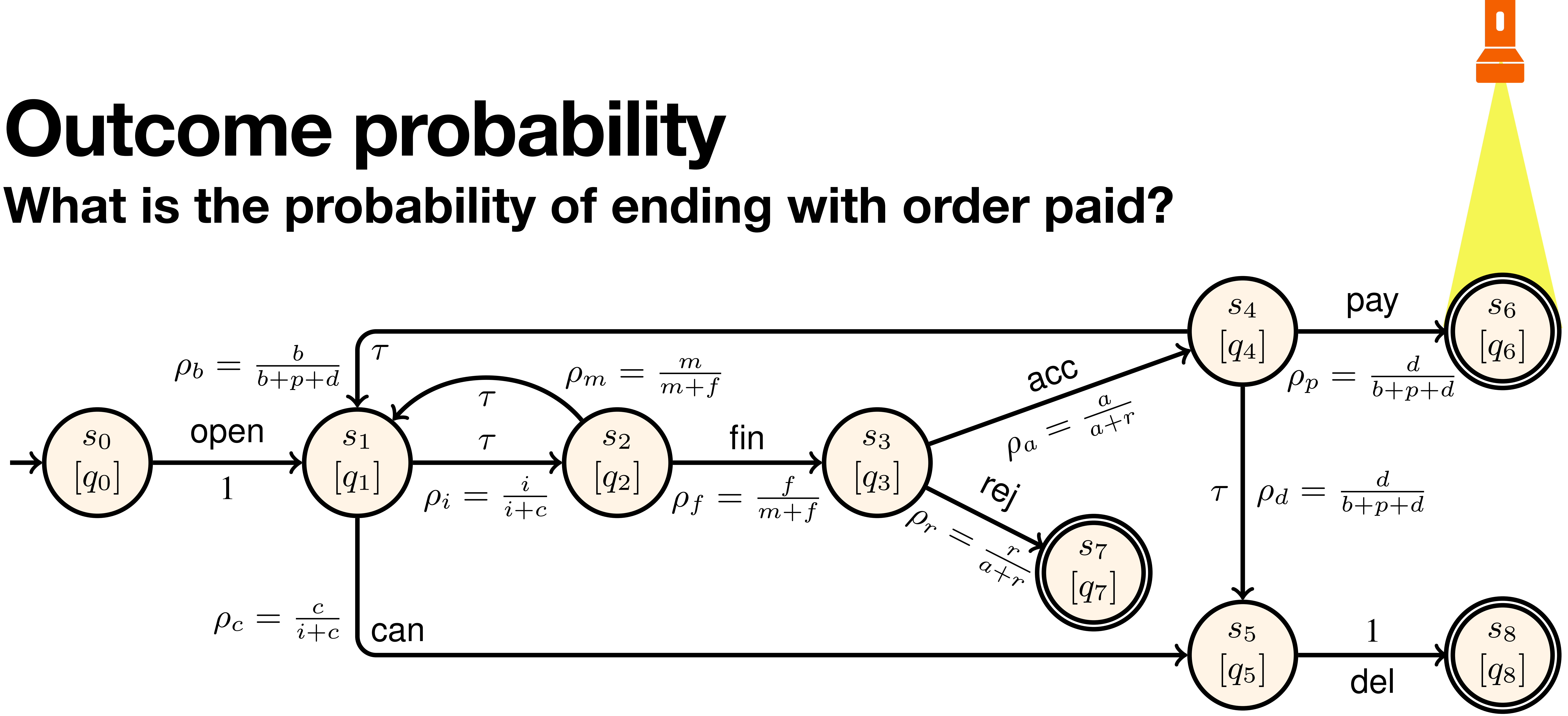
More generally, the cycle time of a fragment with the structure shown in Figure 7.8 is:

$$CT = \frac{T}{1-r}. \tag{7.4}$$

In this formula, the parameter r is called the *rework probability*, that is, the probability that the fragment inside the cycle will need to be reworked. This type of block is called a *rework block* or *repetition block*.

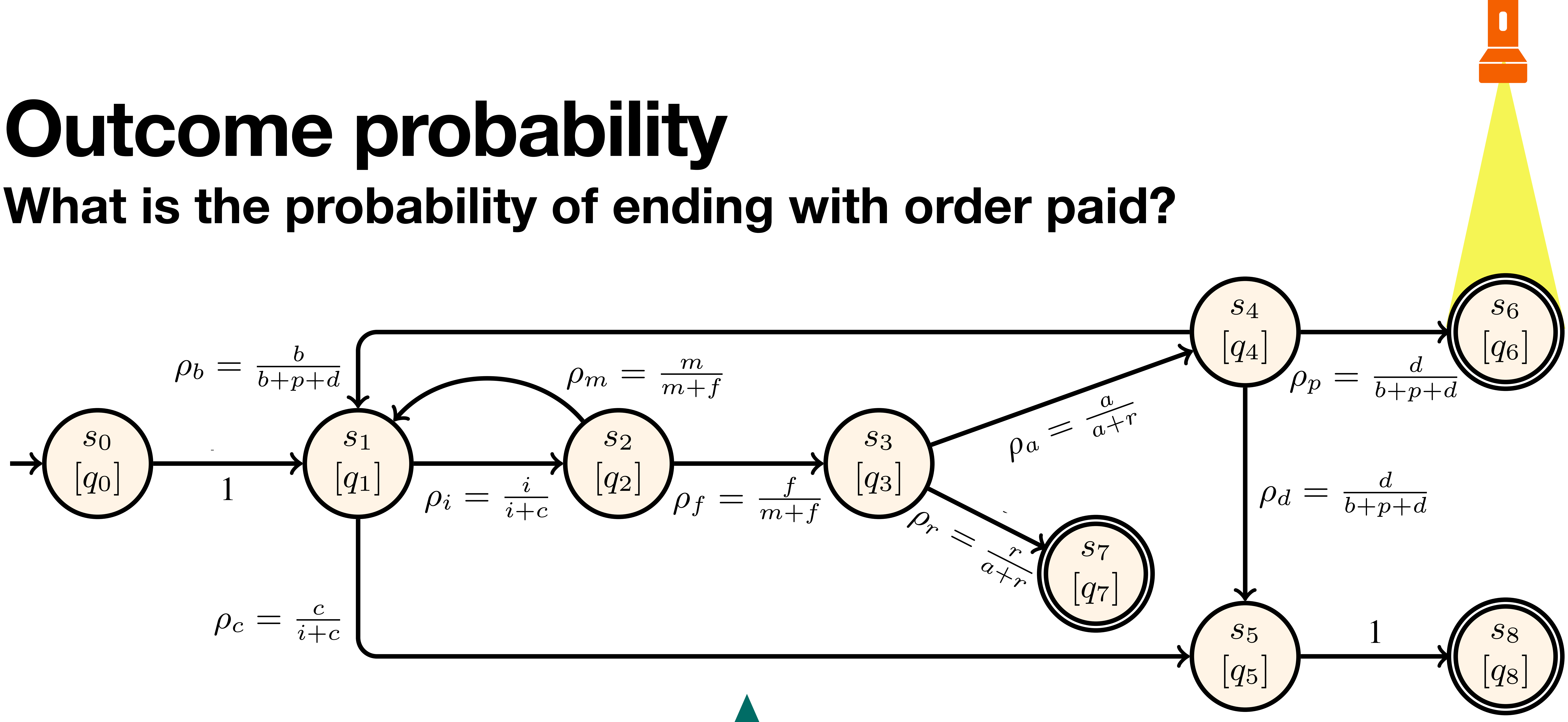
Outcome probability

What is the probability of ending with order paid?



Outcome probability

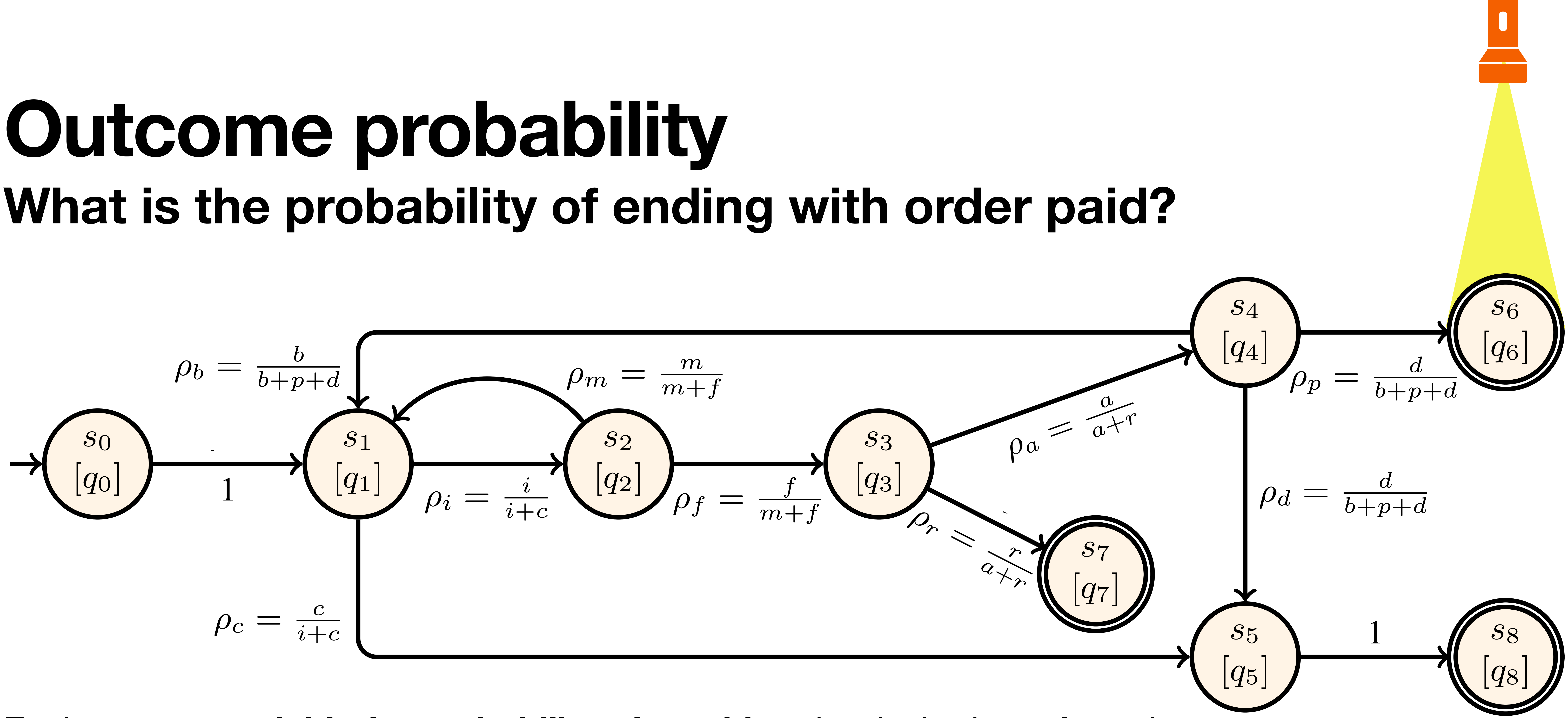
What is the probability of ending with order paid?



What matters: progression from state to state -> Markov chain

Outcome probability

What is the probability of ending with order paid?



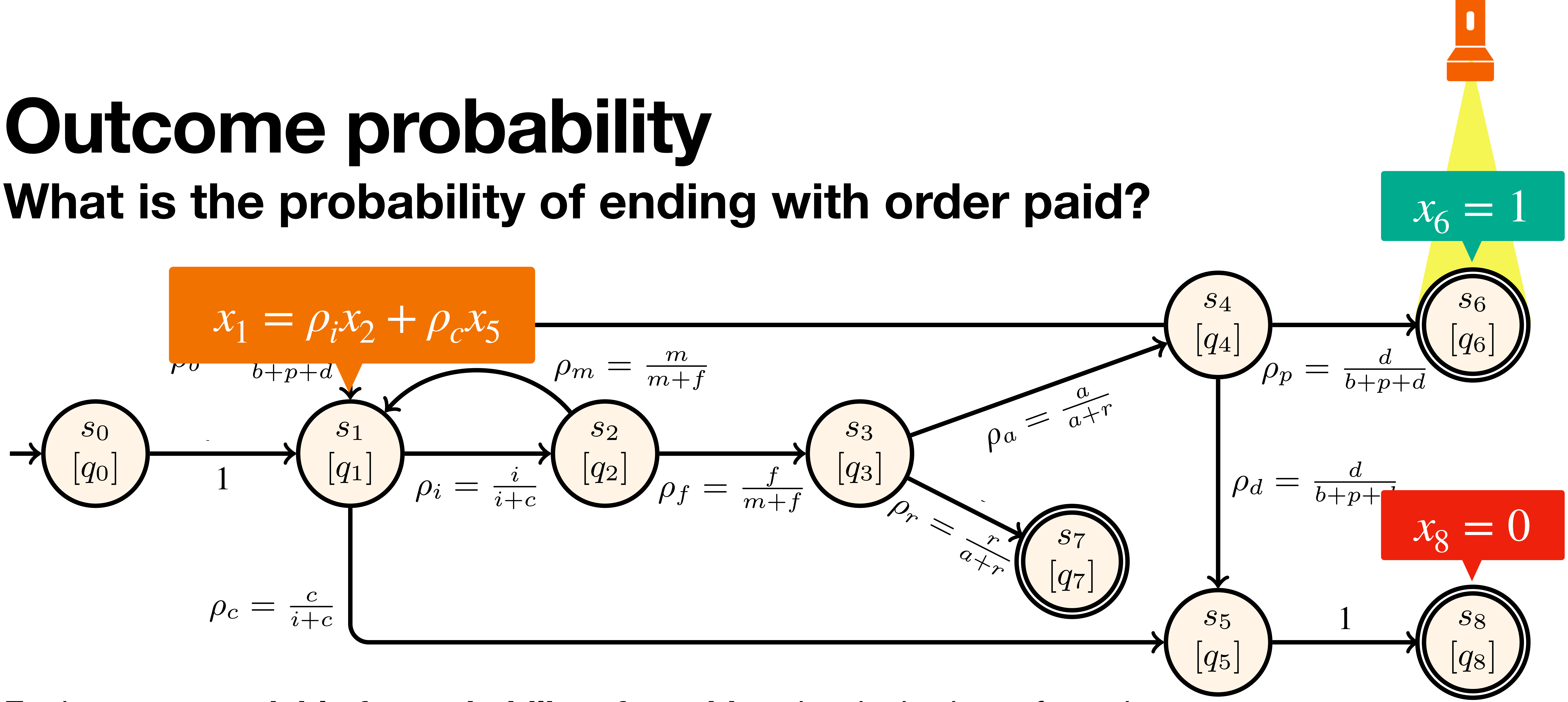
Each state -> **variable for probability of reaching** the desired one from there

3 cases:

- **final desired state** (good deadlock)
- **final non-desired state** (bad deadlock)
- **other states…**

Outcome probability

What is the probability of ending with order paid?



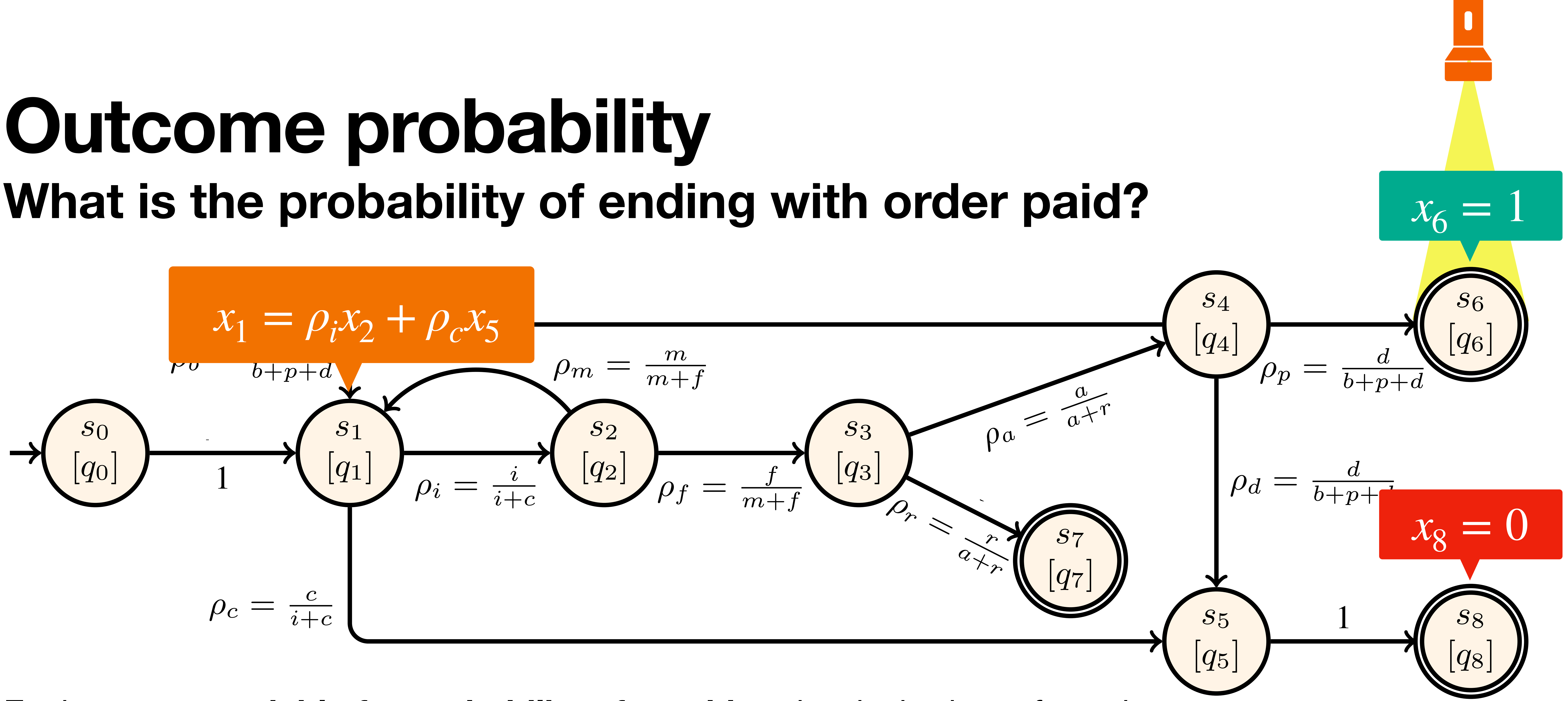
Each state -> **variable for probability of reaching** the desired one from there

3 cases:

- **final desired state** (good deadlock) **-> 1**
- **final non-desired state** (bad deadlock) **-> 0**
- **other states… -> recursive definition via linear equations**

Outcome probability

What is the probability of ending with order paid?



Each state -> **variable for probability of reaching** the desired one from there

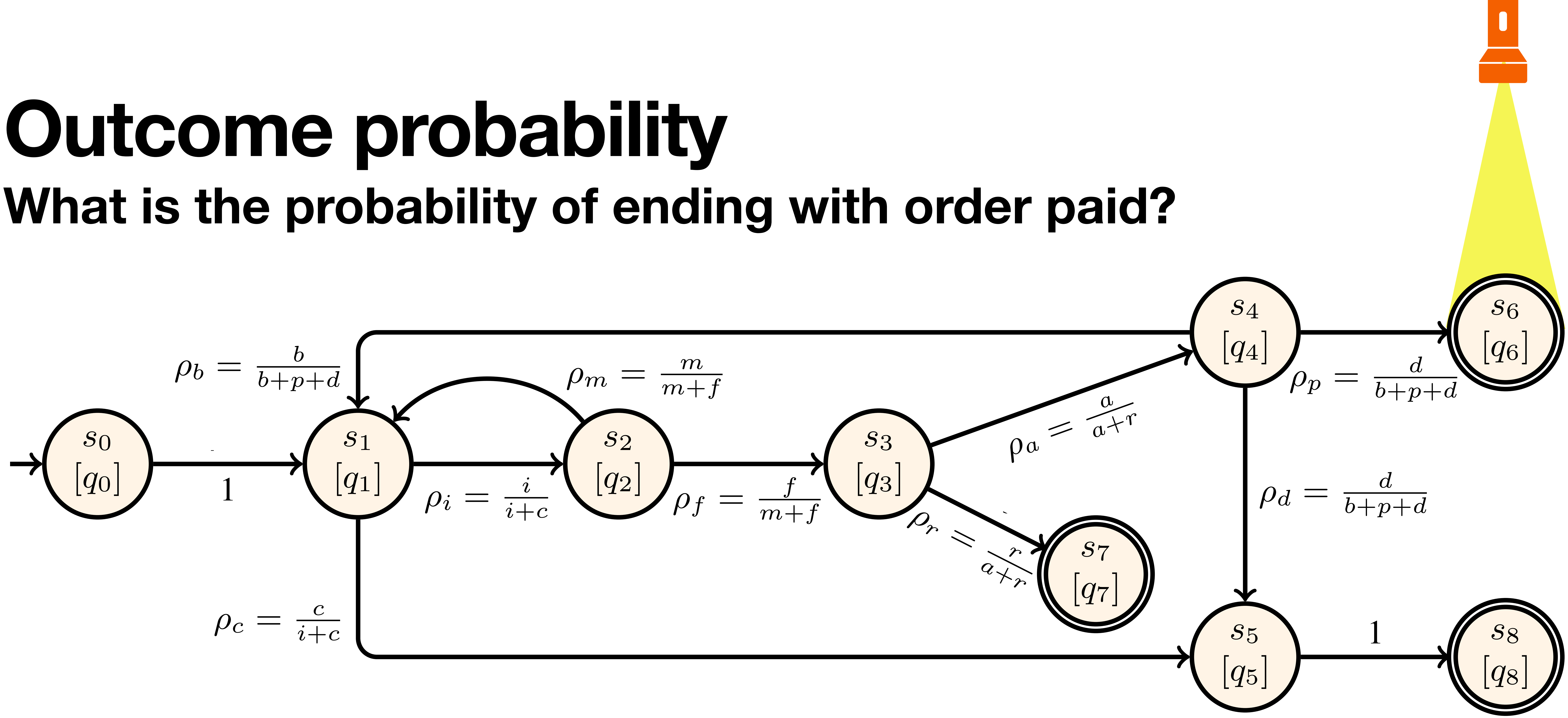
3 cases:

- **final desired state** (good deadlock) **-> 1**
- **final non-desired state** (bad deadlock) **-> 0**
- **other states… -> recursive definition via linear equations**

Solve for initial state variable!

Outcome probability

What is the probability of ending with order paid?



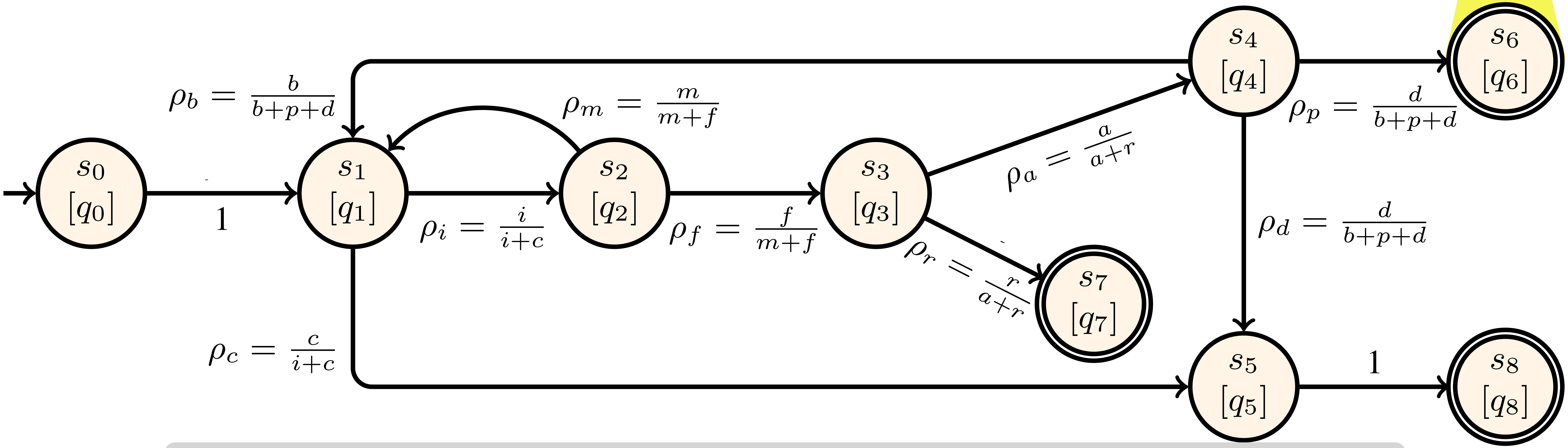
$$x_{s_8} = 0 \qquad x_{s_5} = x_{s_8} \qquad x_{s_2} = \rho_m x_{s_1} + \rho_f x_{s_3}$$

$$x_{s_7} = 0 \qquad x_{s_4} = \rho_b x_{s_1} + \rho_d x_{s_5} + \rho_p x_{s_6} \qquad x_{s_1} = \rho_i x_{s_2} + \rho_c x_{s_5}$$

$$x_{s_6} = 1 \qquad x_{s_3} = \rho_a x_{s_4} + \rho_r x_{s_7} \qquad x_{s_0} = x_{s_1}$$

Outcome probability

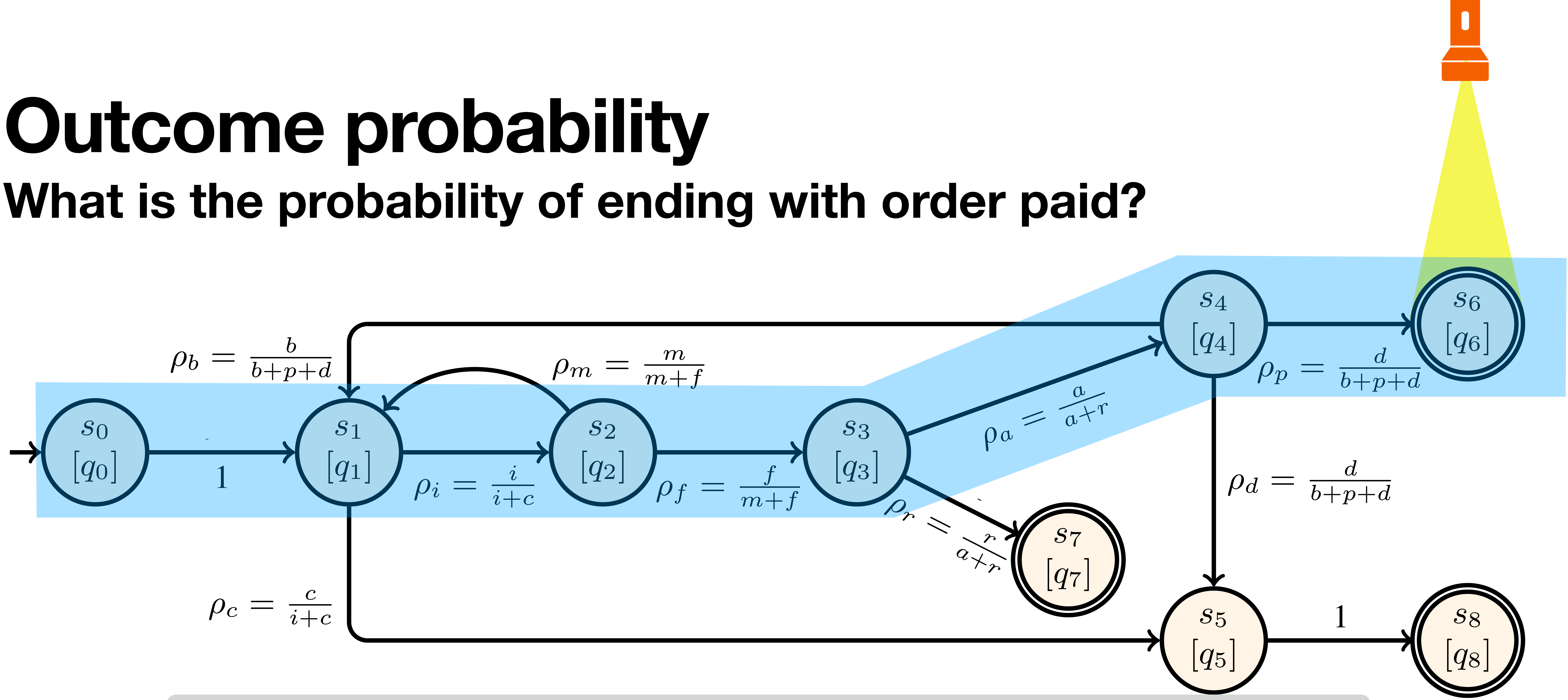
What is the probability of ending with order paid?



$$x_0 = \frac{\rho_i \rho_f \rho_a \rho_p}{1 - \rho_i \rho_m - \rho_i \rho_f \rho_a \rho_b}$$

Outcome probability

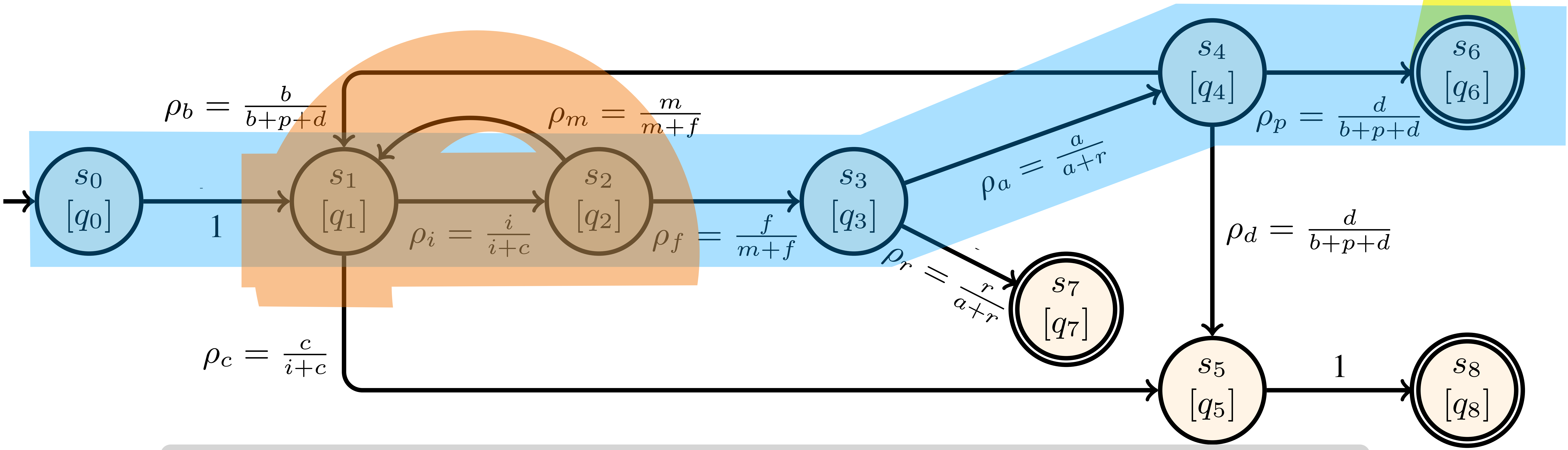
What is the probability of ending with order paid?



$$x_0 = \frac{\rho_i \rho_f \rho_a \rho_p}{1 - \rho_i \rho_m - \rho_i \rho_f \rho_a \rho_b}$$

Outcome probability

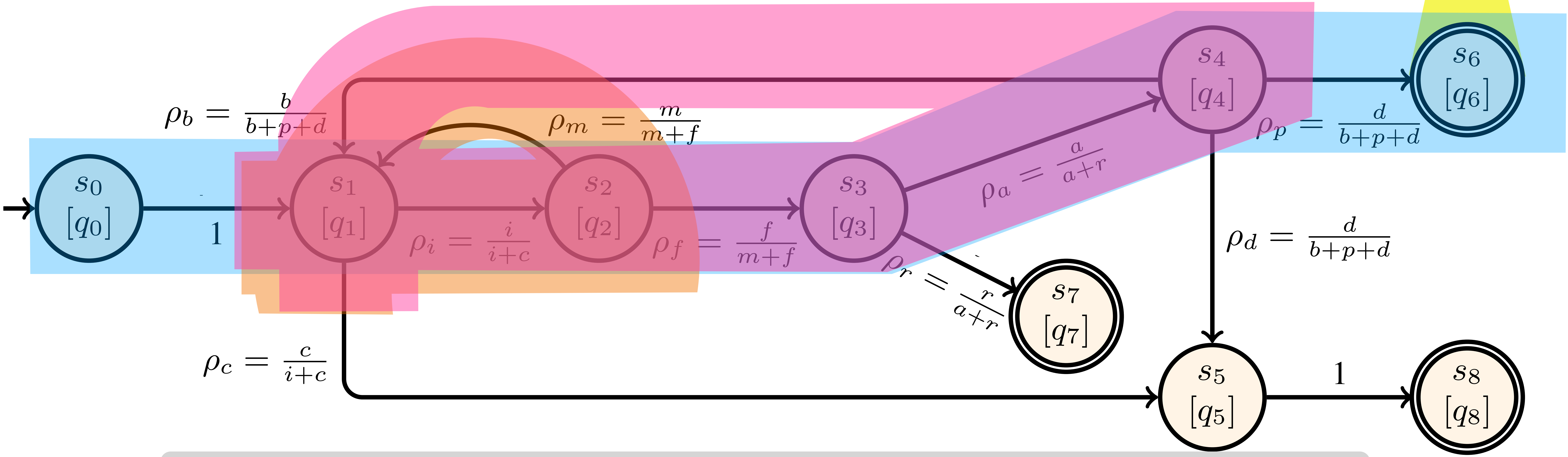
What is the probability of ending with order paid?



$$x_0 = \frac{\rho_i\rho_f\rho_a\rho_p}{1 - \rho_i\rho_m - \rho_i\rho_f\rho_a\rho_b}$$

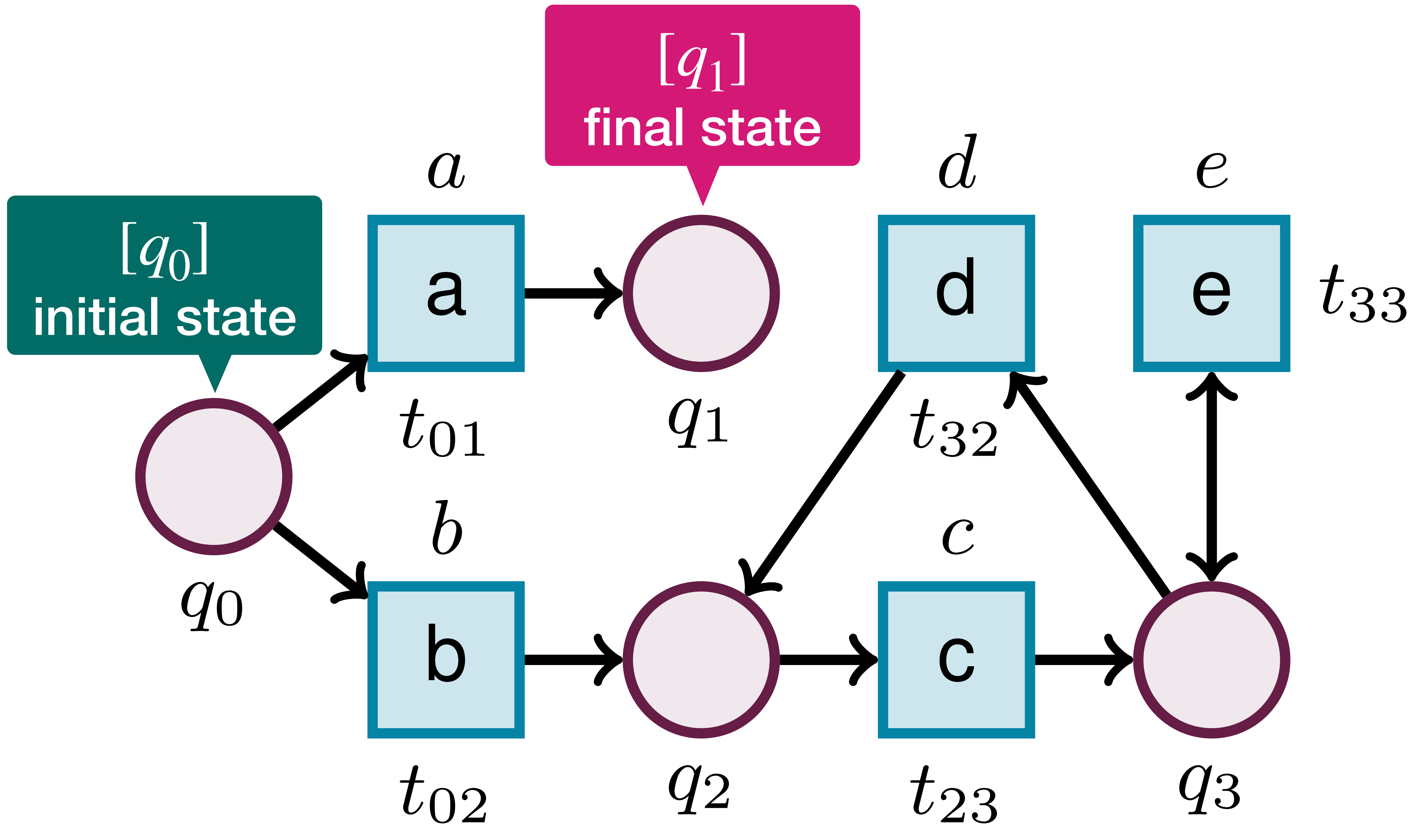
Outcome probability

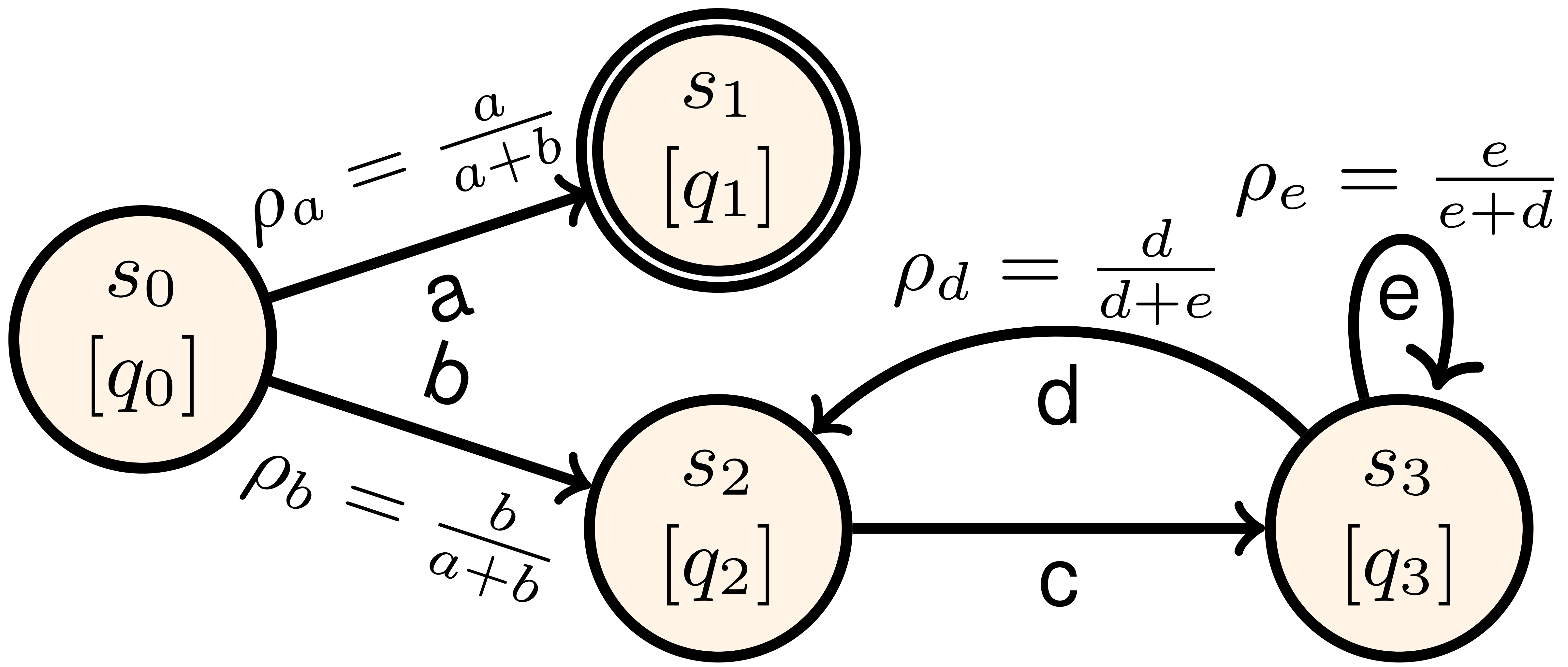
What is the probability of ending with order paid?



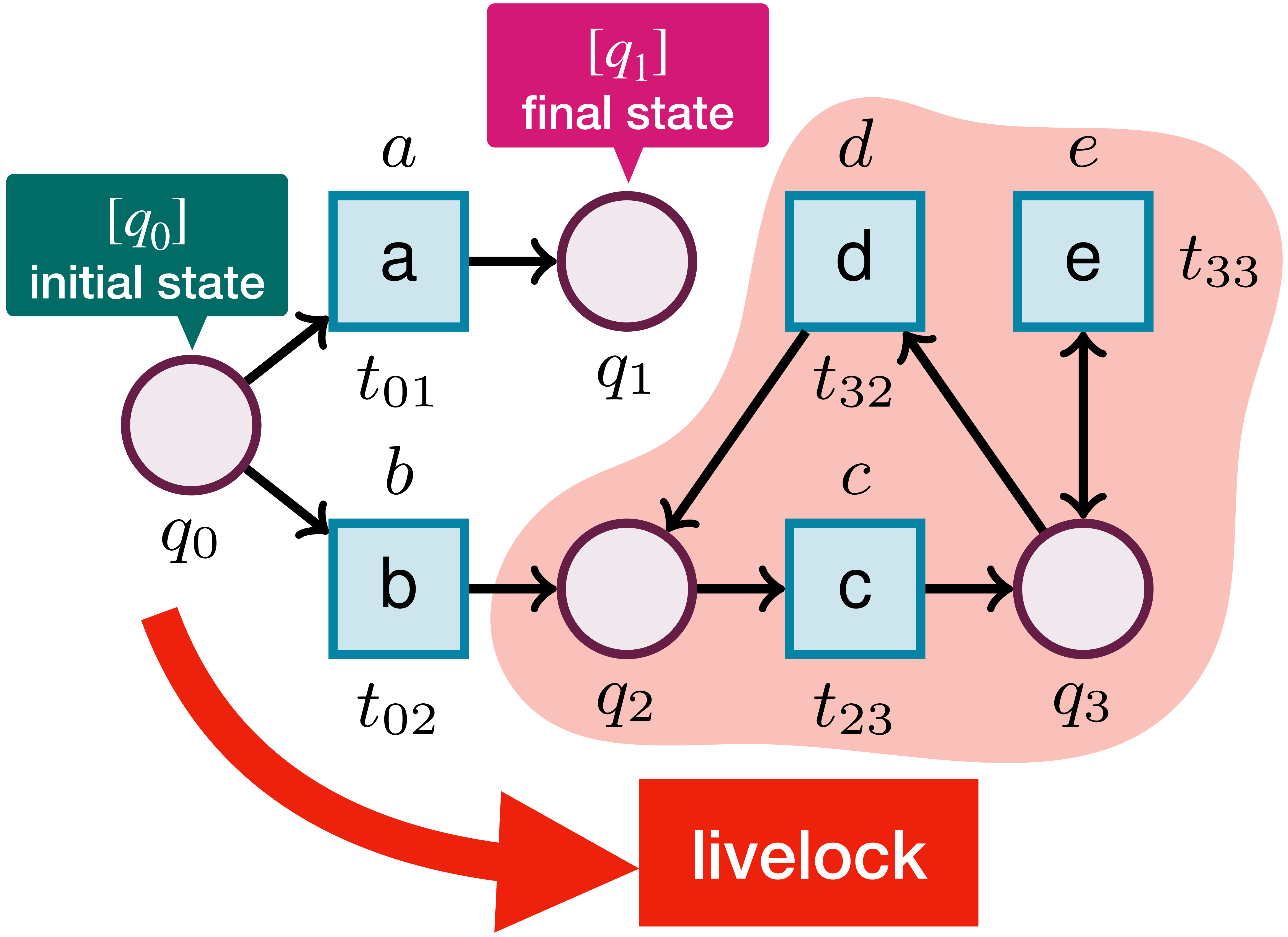
$$x_0 = \frac{\rho_i \rho_f \rho_a \rho_p}{1 - \rho_i \rho_m - \rho_i \rho_f \rho_a \rho_b}$$

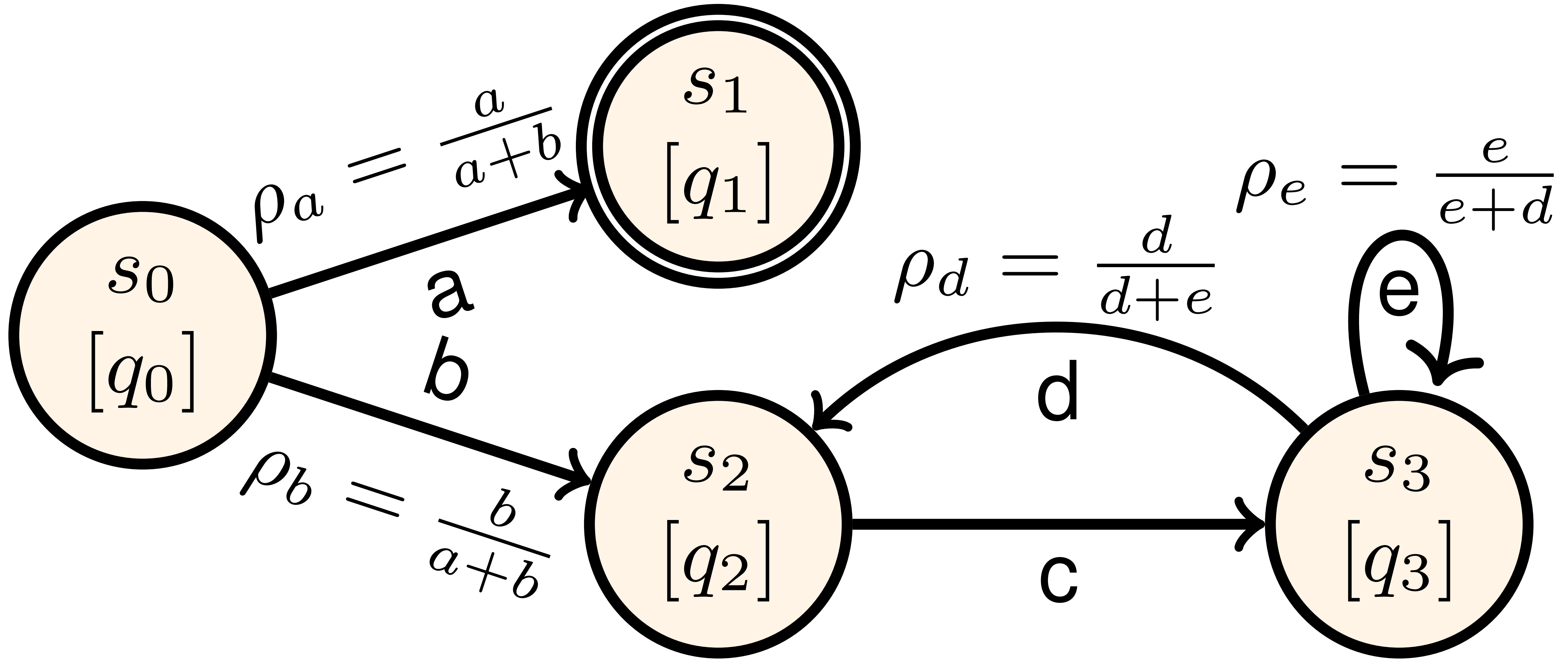
More care needed…



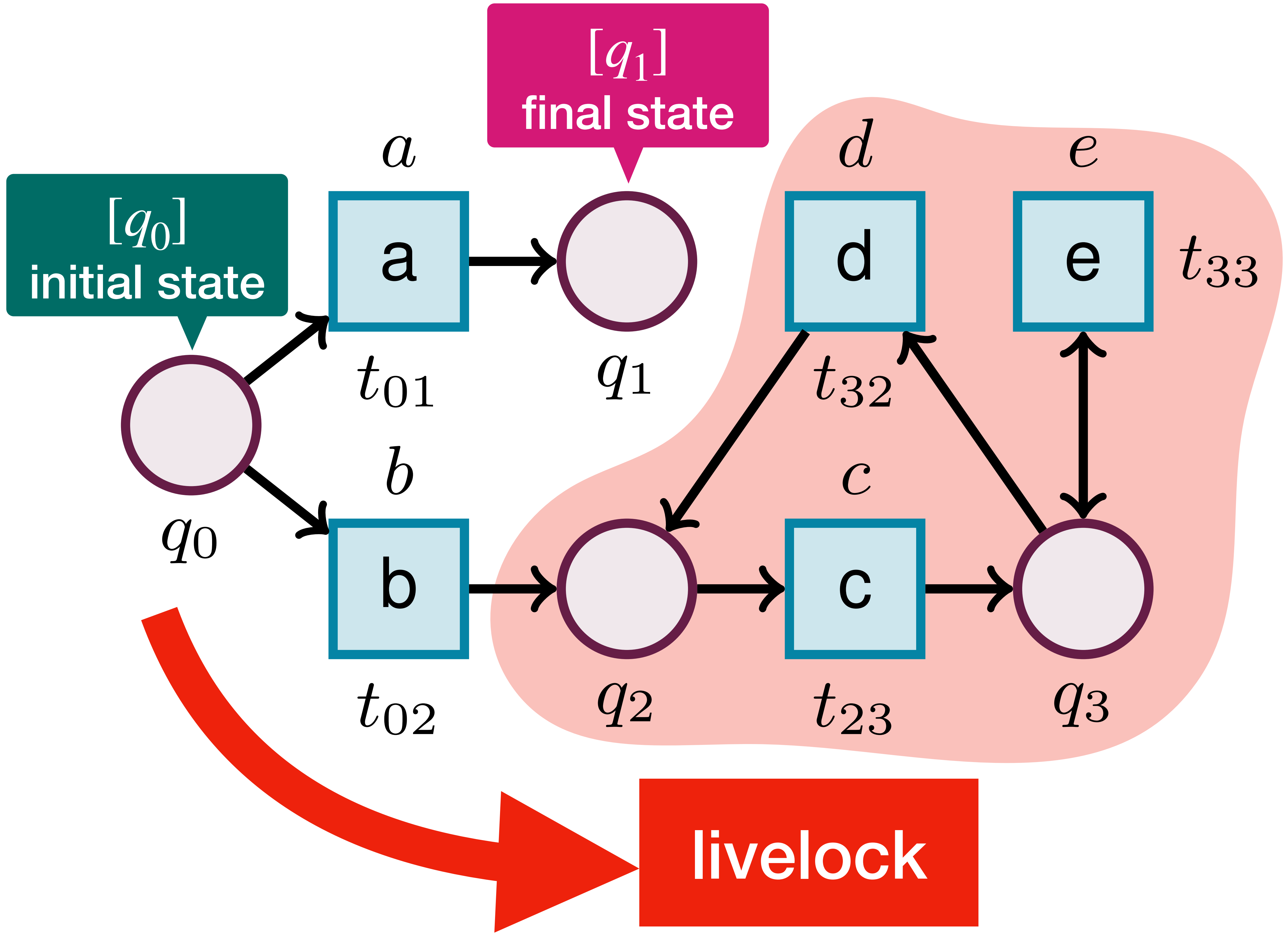


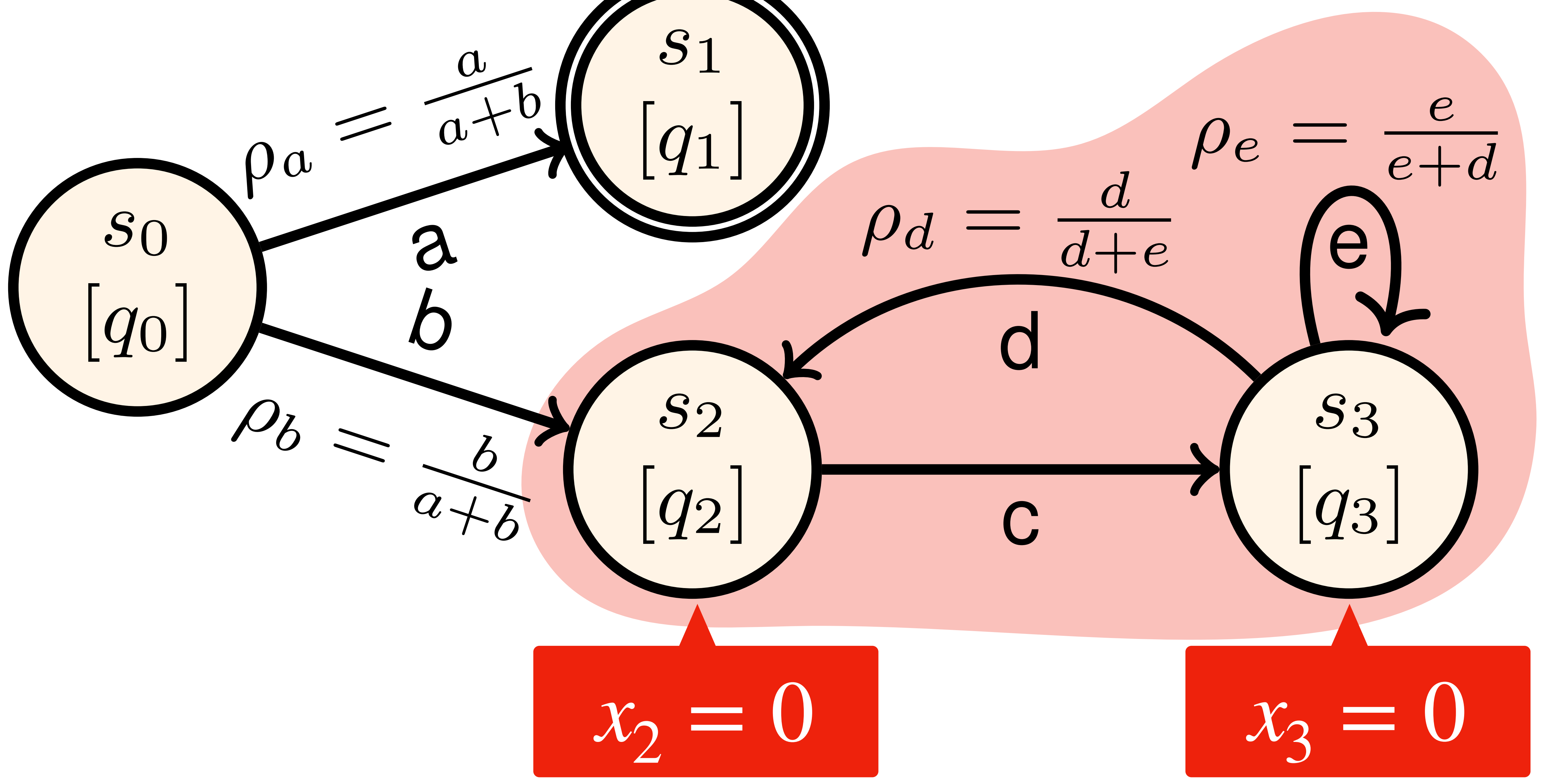
More care needed…





More care needed…





General case

Exit/absorption probability computation from Markov chains

$\text{OUTCOME-PROB}(\mathcal{N}, F)$ with $RG(\mathcal{N}) = \langle S, s_0, S_f, \varrho, p \rangle$

Return x_{s_0} from the minimal non-negative solution of

$x_{s_i} = 1$ for each $s_i \in F$

$x_{s_j} = 0$ for each $s_j \in S \setminus F$ s.t. $|succ_{RG(\mathcal{N})}(s_j)| = 0$

$x_{s_k} = \sum_{\langle s_k, l, s'_k \rangle \in succ_{RG(\mathcal{N})}(s_k)} p(\langle s_k, l, s'_k \rangle) \cdot x_{s'_k}$ for each $s_k \in S$ s.t. $|succ_{RG(\mathcal{N})}(s_k)| > 0$

General case

Exit/absorption probability computation from Markov chains

$\text{OUTCOME-PROB}(\mathcal{N}, F)$ with $RG(\mathcal{N}) = \langle S, s_0, S_f, \varrho, p \rangle$

Return x_{s_0} from the minimal non-negative solution of

$$x_{s_i} = 1 \qquad \text{for each } s_i \in F$$

$$x_{s_j} = 0 \qquad \text{for each } s_j \in S \setminus F \text{ s.t. } |succ_{RG(\mathcal{N})}(s_j)| = 0$$

$$x_{s_k} = \sum_{\langle s_k, l, s'_k \rangle \in succ_{RG(\mathcal{N})}(s_k)} p(\langle s_k, l, s'_k \rangle) \cdot x_{s'_k} \qquad \text{for each } s_k \in S \text{ s.t. } |succ_{RG(\mathcal{N})}(s_k)| > 0$$

⇕

$$x_{s_i} = 1 \qquad \text{for each deadlock marking } s_i \in F$$

$$x_{s_j} = 0 \qquad \text{for each deadlock marking } s_j \in S \setminus F$$

$$x_{s_k} = 0 \qquad \text{for each livelock marking } s_k \in S$$

$$x_{s_h} = \sum_{\langle s_h, l, s'_h \rangle \in succ_{RG(\mathcal{N})}(s_h)} p(\langle s_h, l, s'_h \rangle) \cdot x_{s'_h} \qquad \text{for each remaining marking } s_h \in S$$

Attack strategy

0. **Outcome probability**
   Probability of completing the process in some final states

1. **Probability of a trace**
2. **Probability of satisfying a qualitative property**
3. **Conformance to a probabilistic Declare specification**

Reasoning on states and probabilities

Markov chains

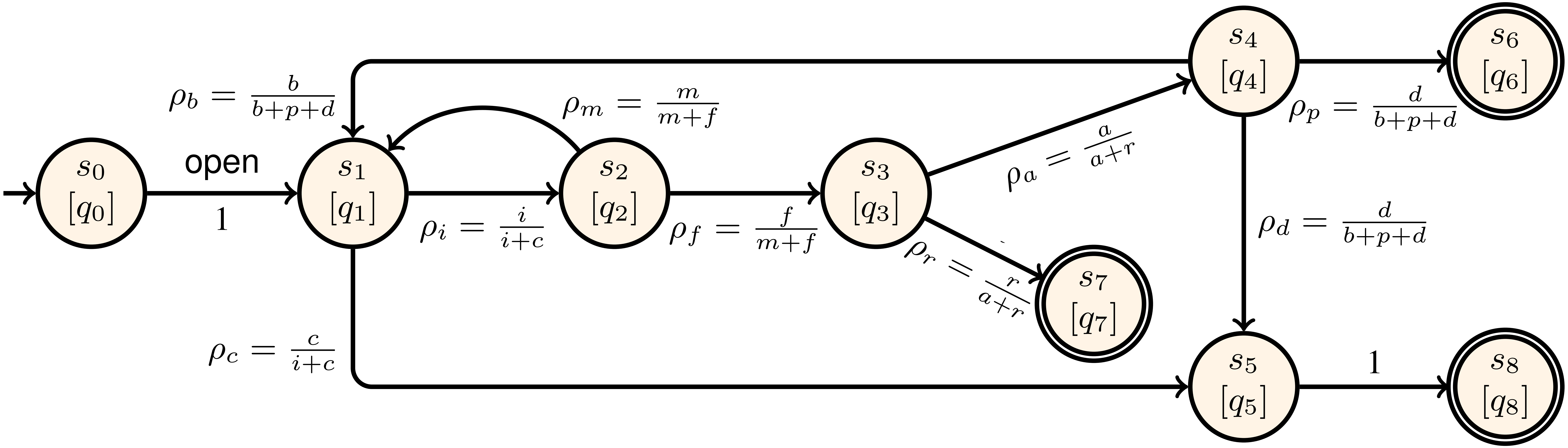
Elegant trick to deal with silent transitions

Qualitative model checking

Reasoning on tasks and transitions

Model checking

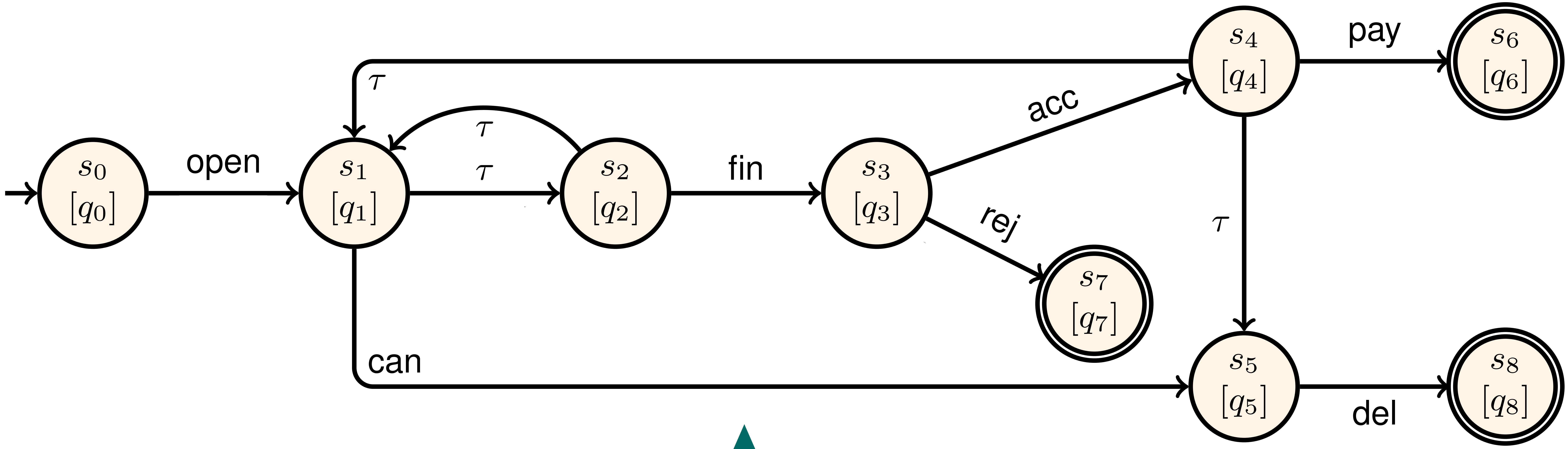
What are the traces of the net that satisfy my property?



property: good, finite traces (via an automaton)

Model checking

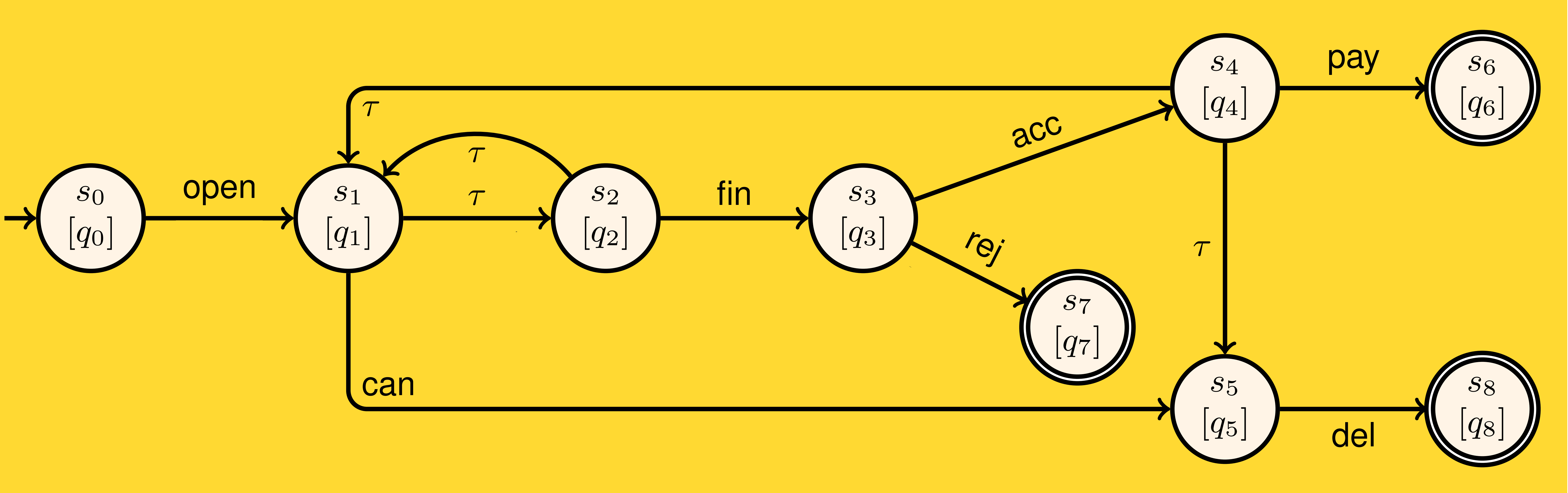
What are the traces of the net that satisfy my property?

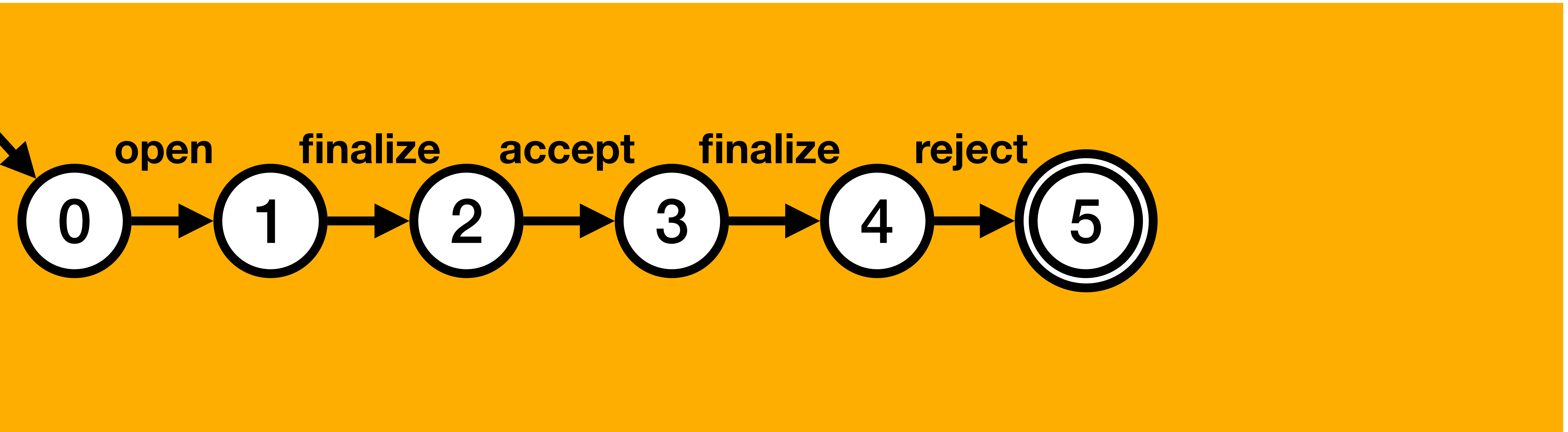


What matters: transitions and their labels -> transition system

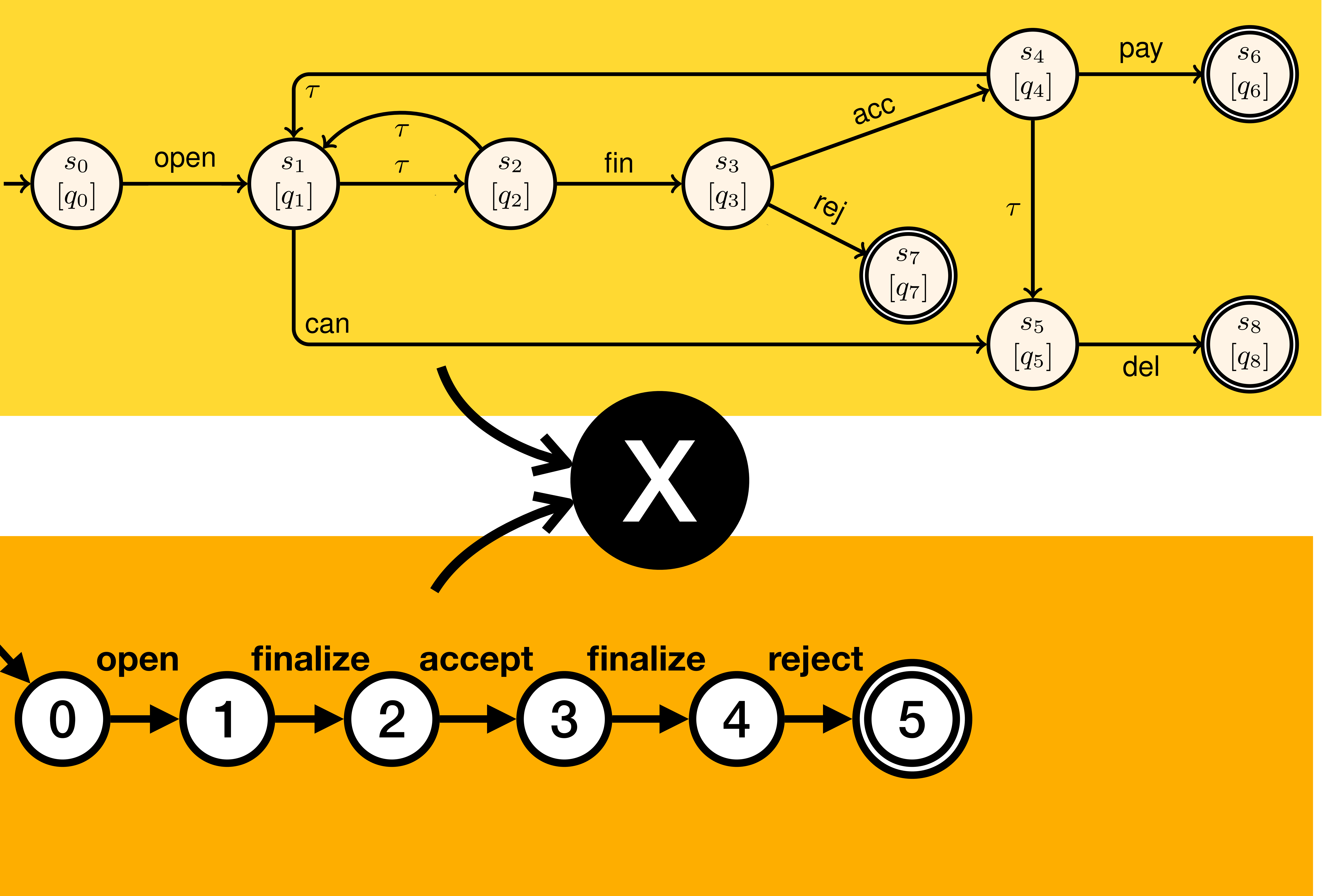
property: good, finite traces (via an automaton)

Automata-based product



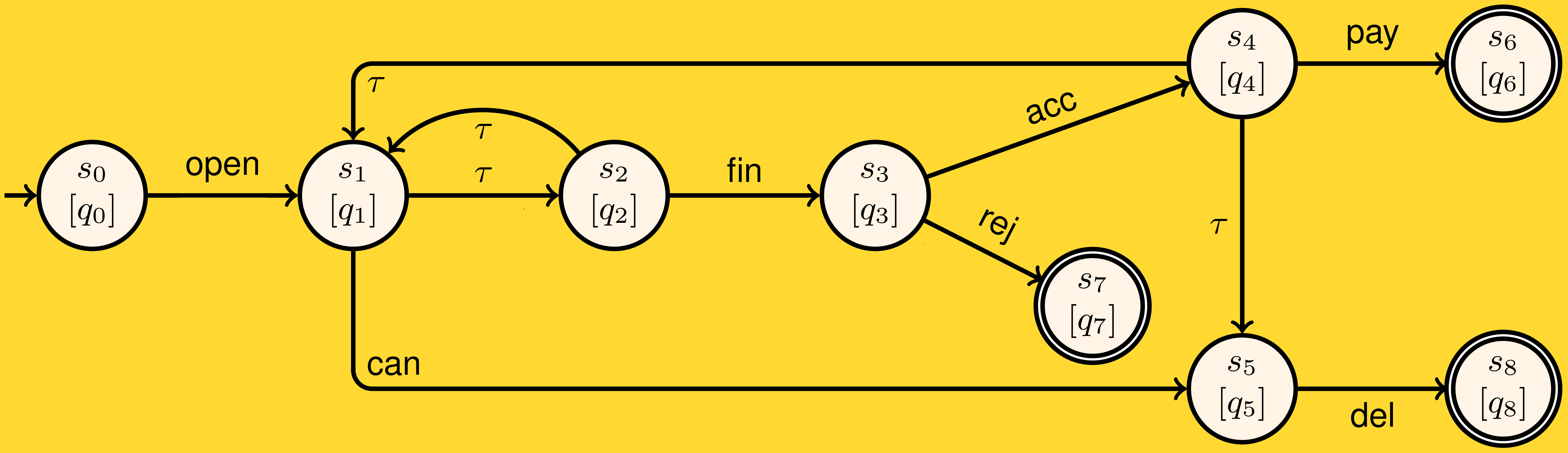


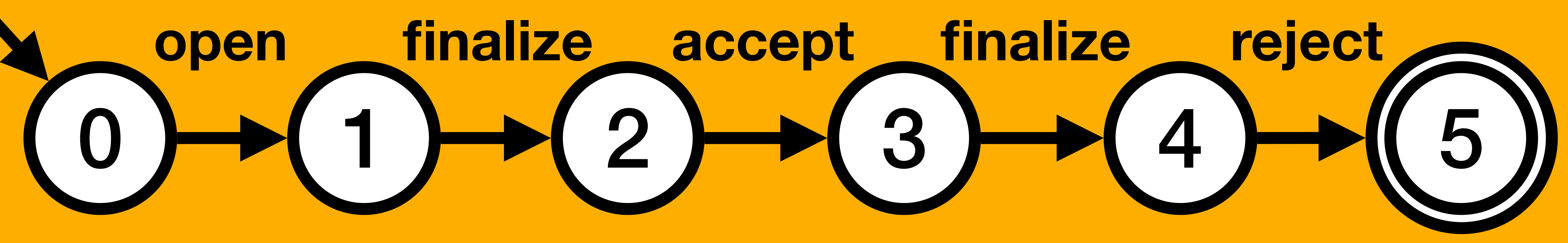
Automata-based product



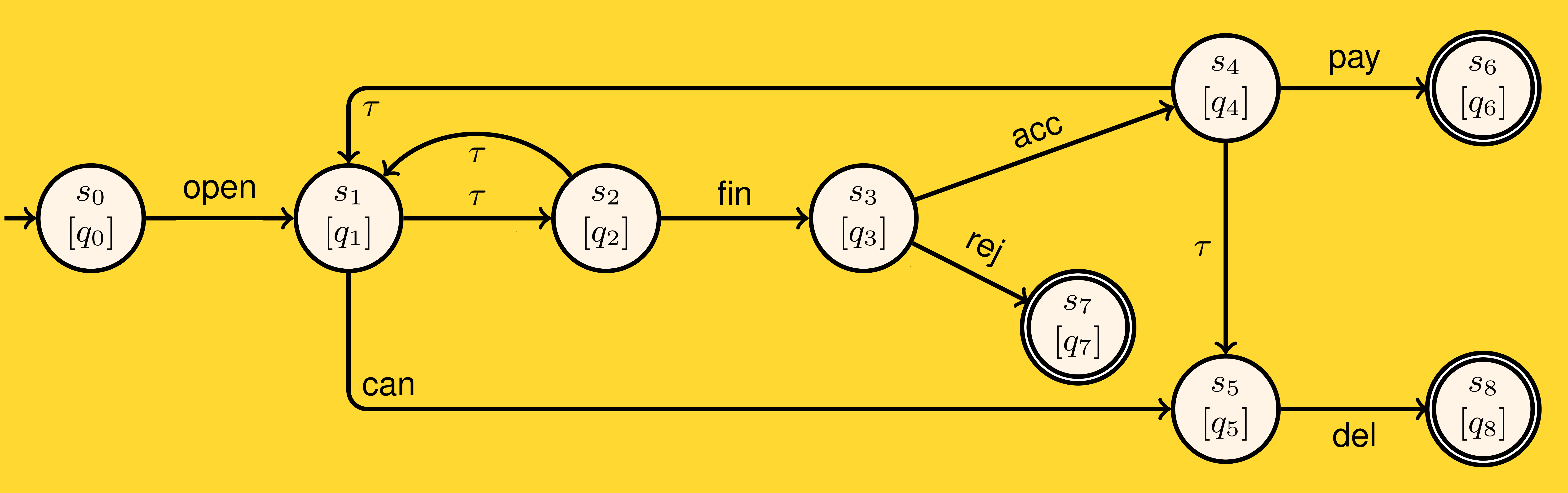
Automata-based product

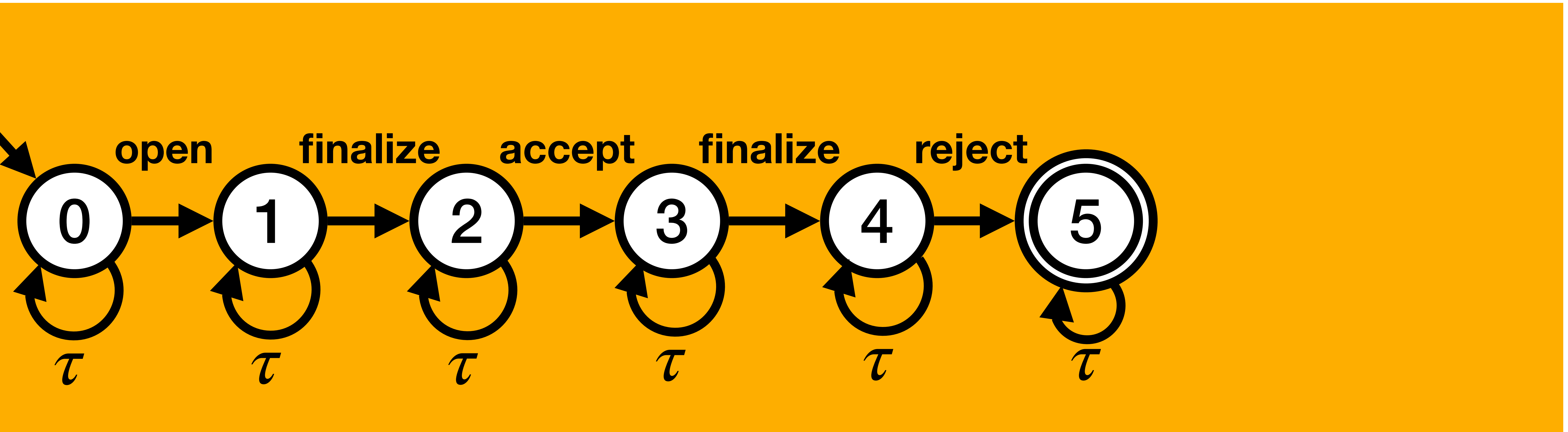
Properties must "enjoy the silence"





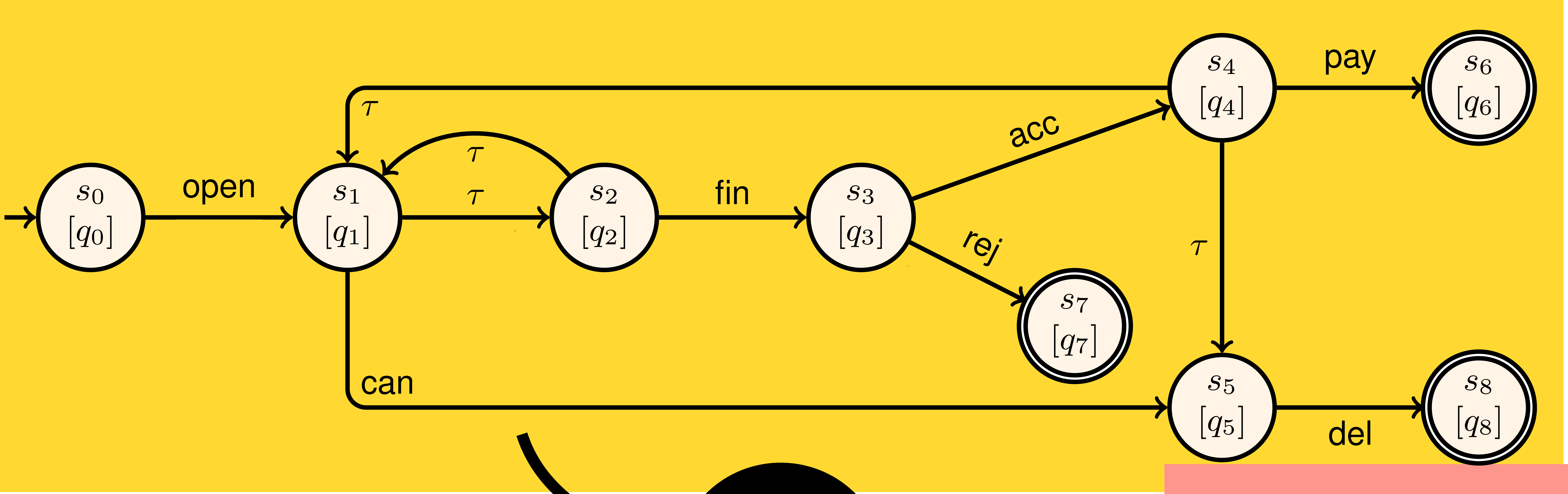
Properties must "enjoy the silence"



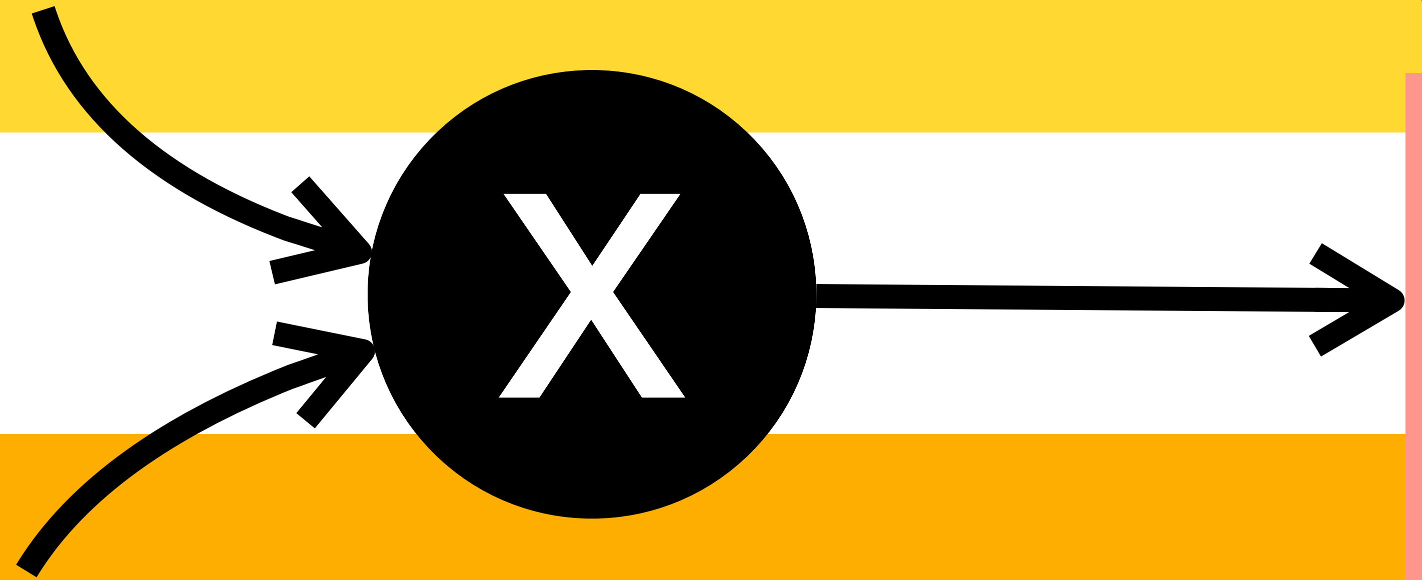


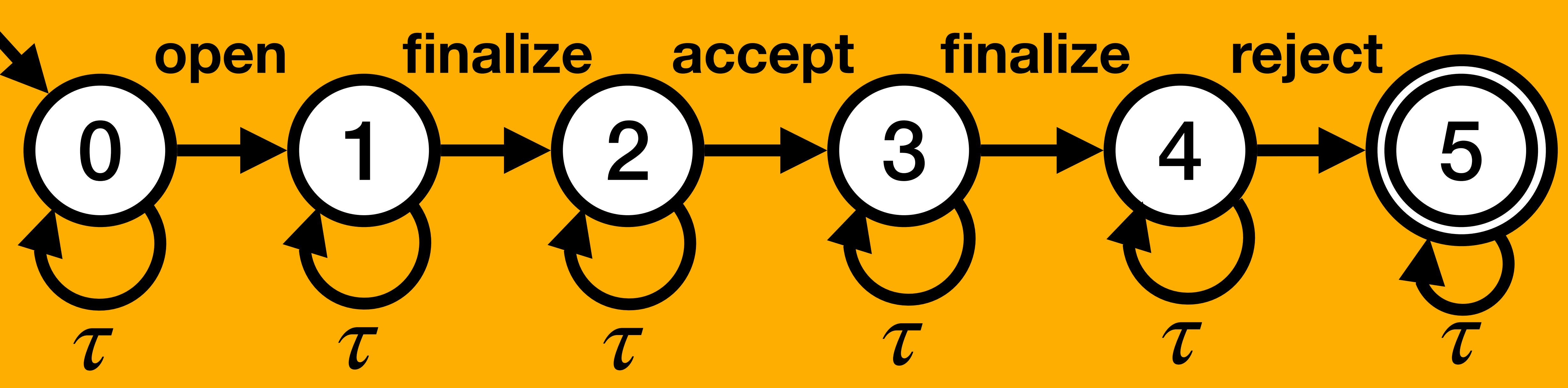
Silence-preserving cross-product

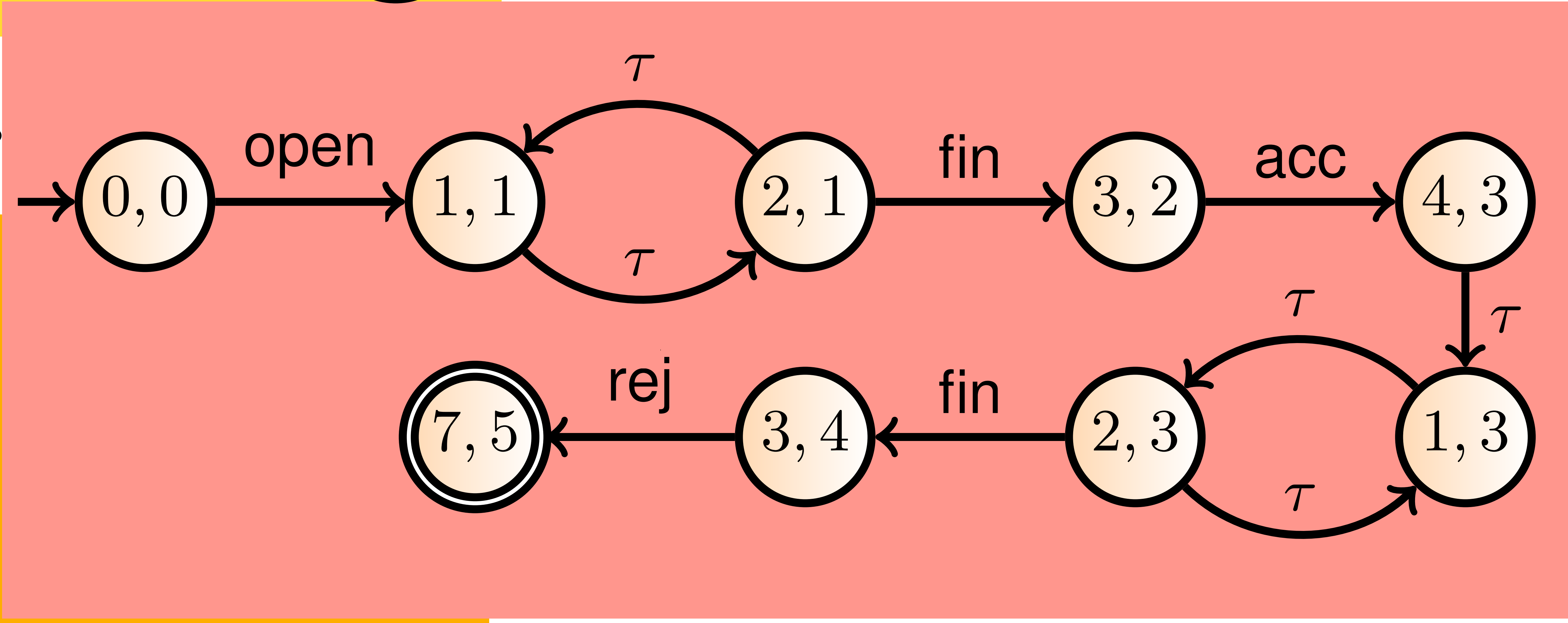
Silence-preserving cross-product



How to compute the probability that the specification is satisfied by a net trace?







Attack strategy

Reasoning on states and probabilities

Markov chains

0. **Outcome probability**
   Probability of completing the process in some final states

1. **Probability of a trace**

2. **Probability of satisfying a qualitative property**

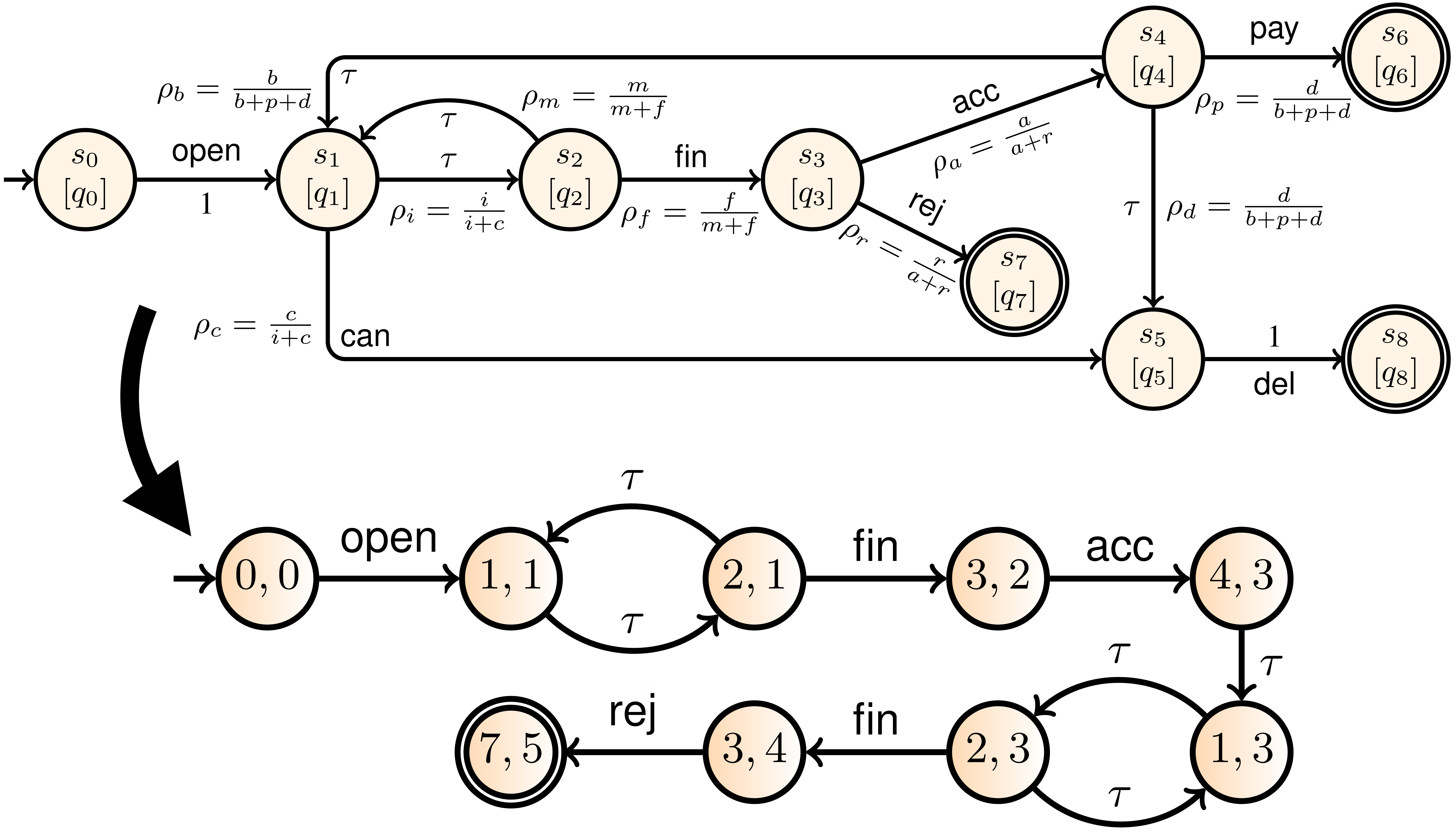
3. **Conformance to a probabilistic Declare specification**

Elegant trick to deal with silent transitions

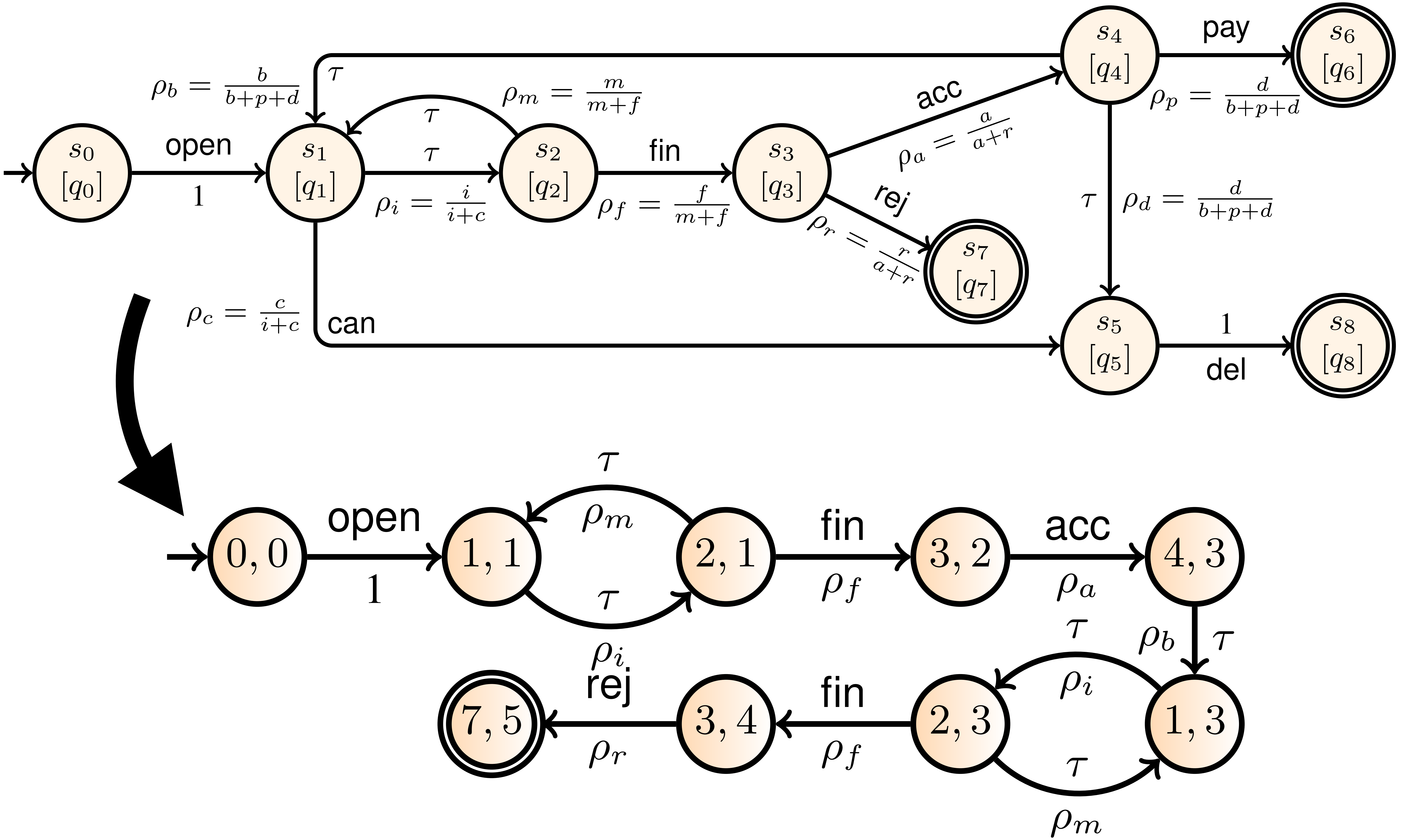
Qualitative model checking

Reasoning on tasks and transitions

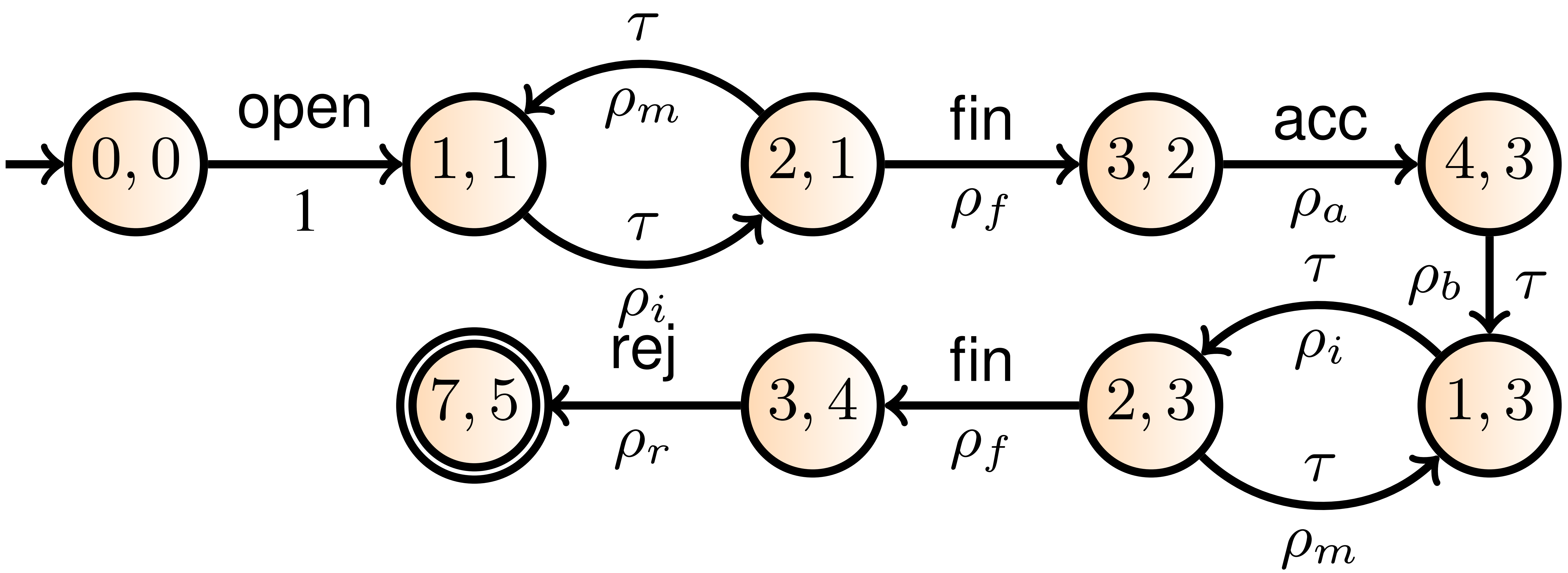
1. Infuse the cross-product with net probabilities



1. Infuse the cross-product with net probabilities

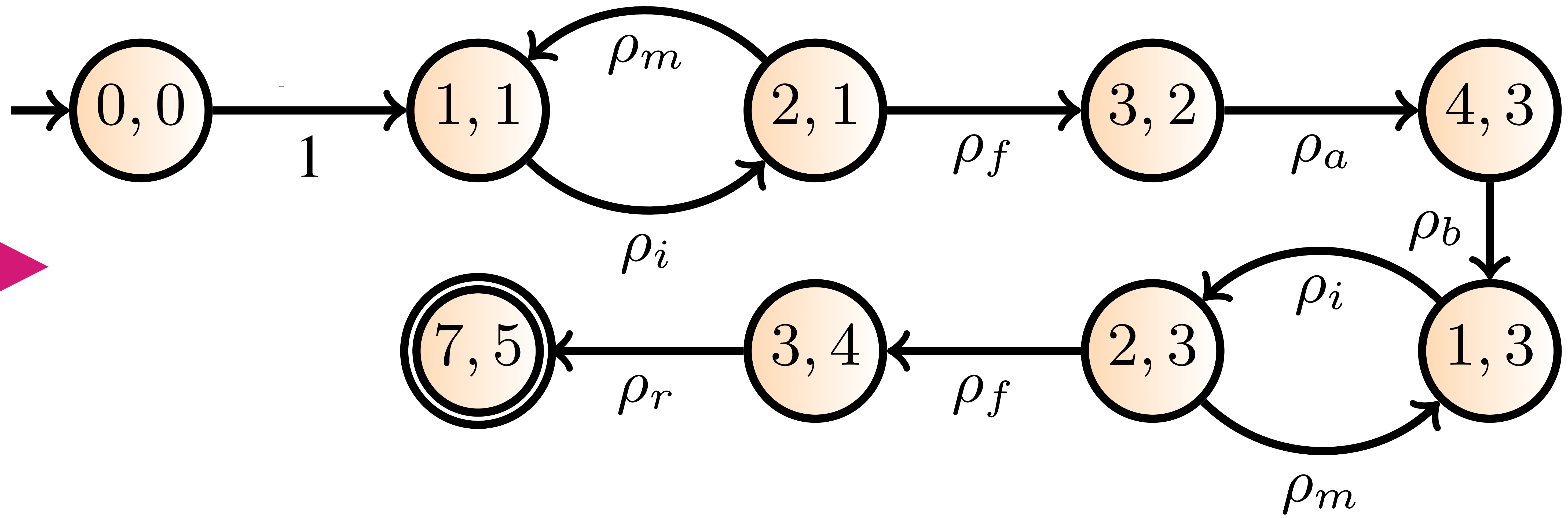


2. Focus on states



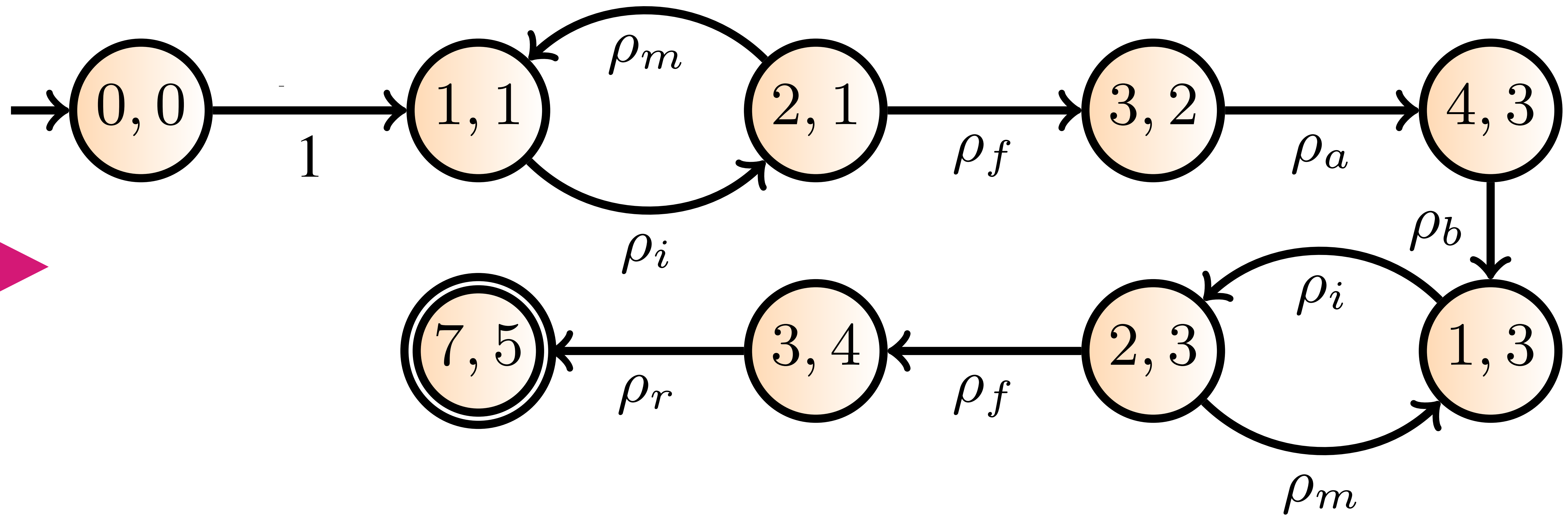
2. Focus on states

An "incomplete" Markov chain



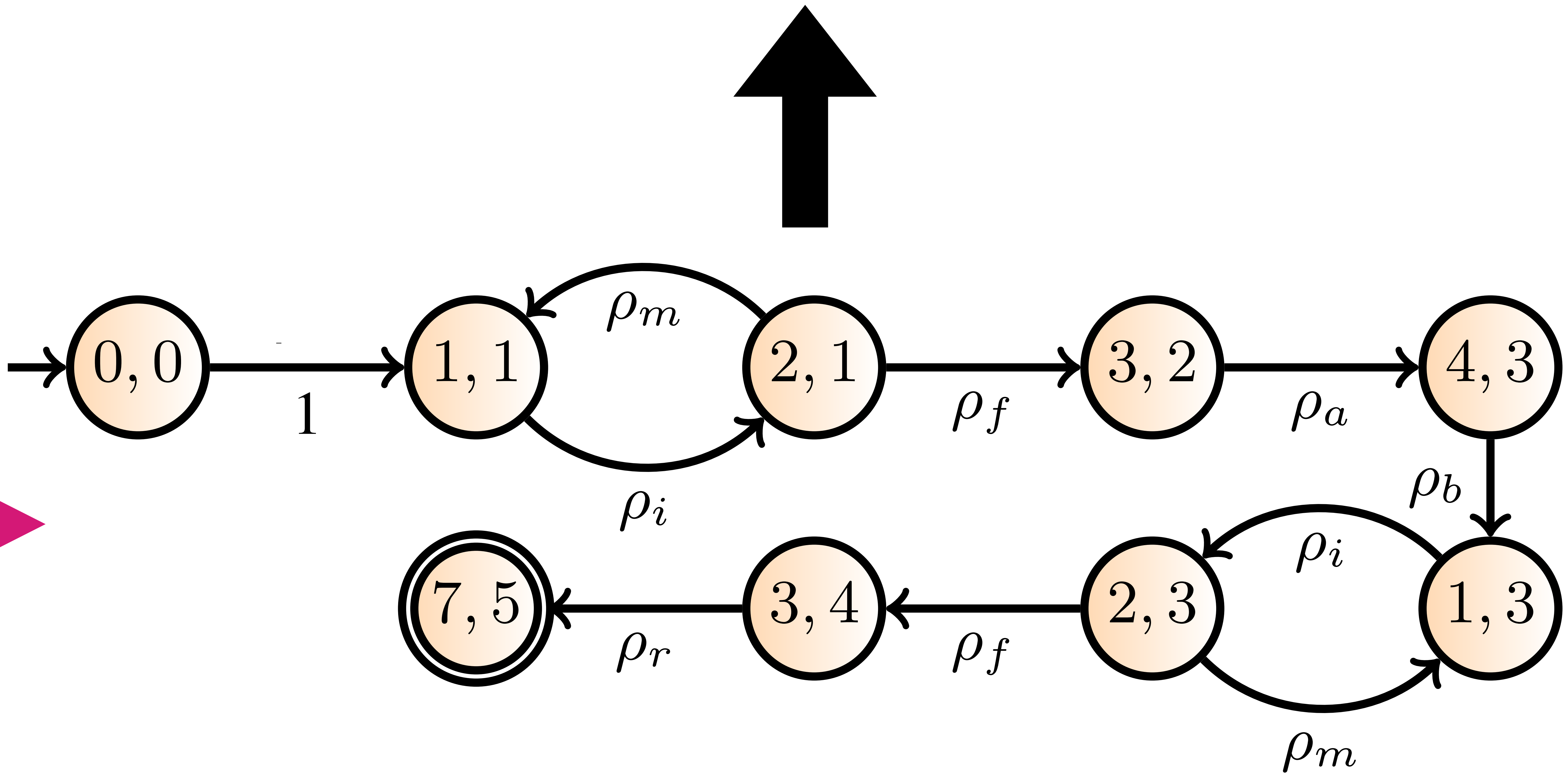
3. Solve the "outcome probability problem"

An "incomplete" Markov chain



3. Solve the “outcome probability problem”

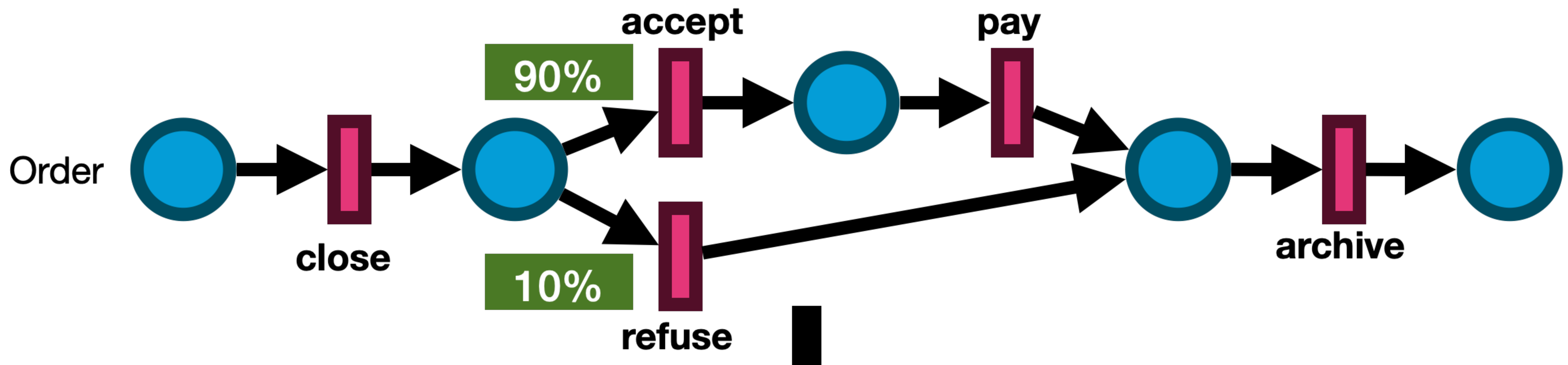
$$x_{00} = \frac{\rho_i \rho_f \rho_a \rho_b \rho_i \rho_f \rho_r}{1 - (\rho_i \rho_m)^2}$$



An “incomplete” Markov chain

Applications in stochastic process mining

Probabilistic trace alignment



P=0.9

close, accept, pay, archive

close	accept	pay	archive
close	>>	>>	archive

d=2

P=0.1

close, refuse, archive

close	refuse	archive
close	>>	archive

d=1

close, archive

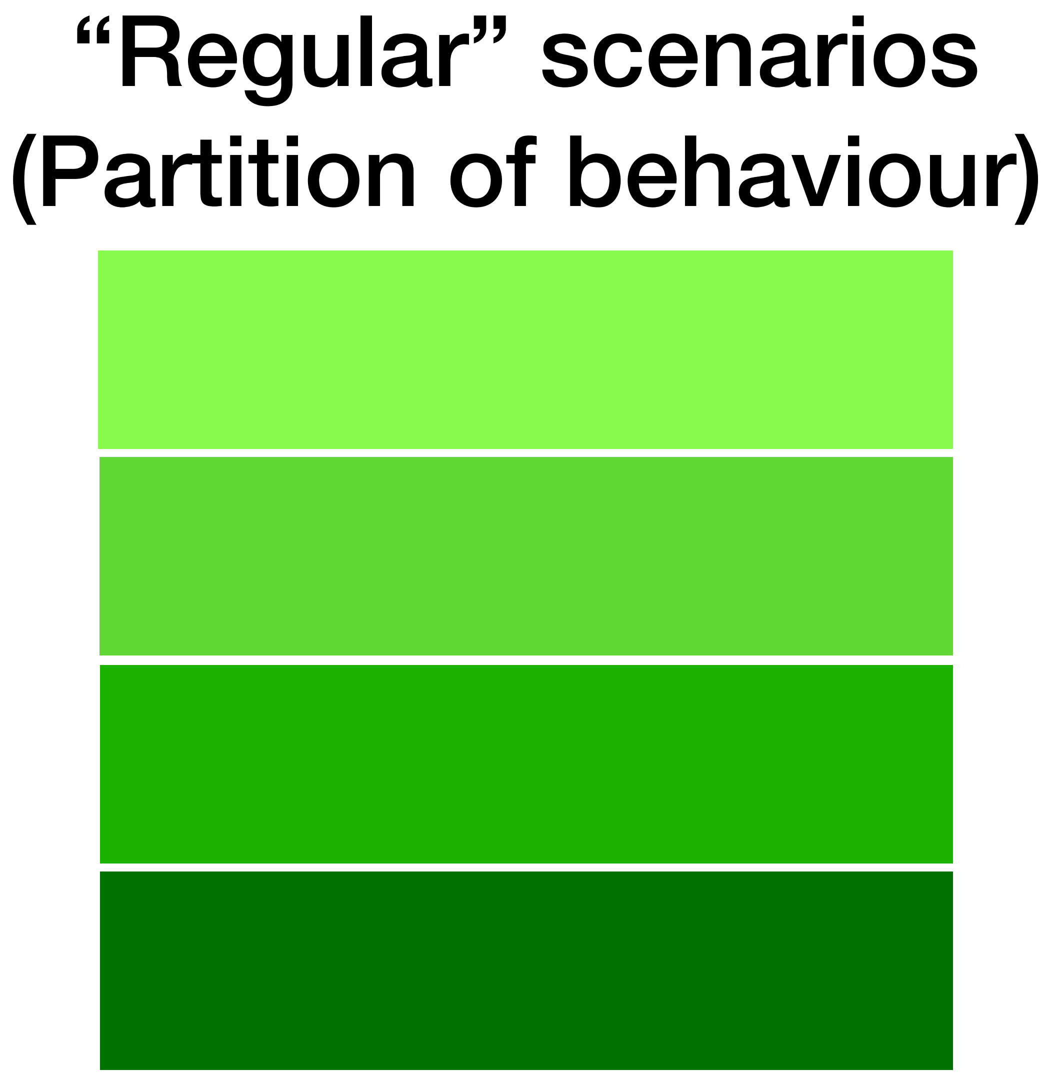
Compare with distributions over traces

Induced distribution via qualitative verification

Set of traces
(behaviour)

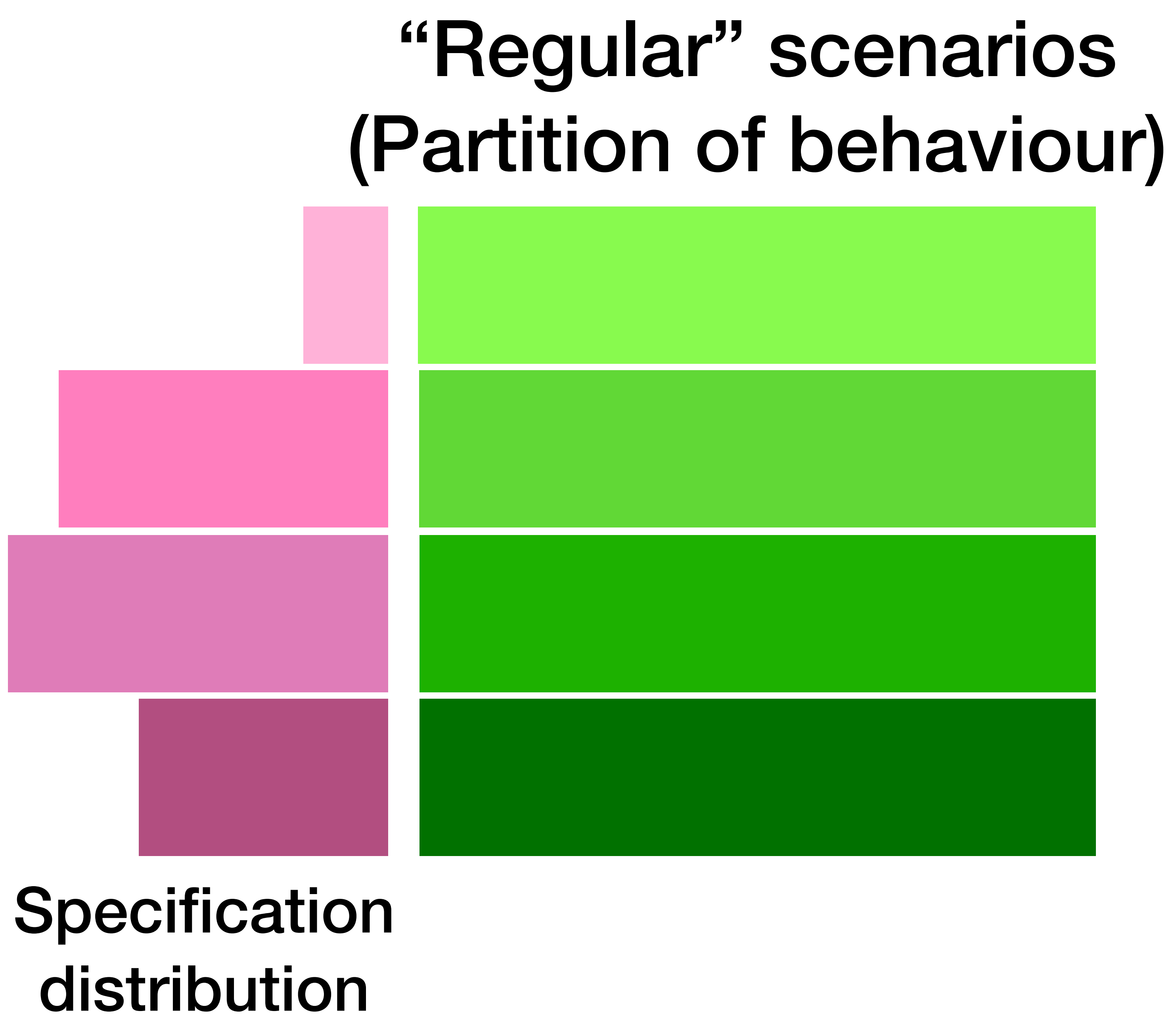
Compare with distributions over traces

Induced distribution via qualitative verification



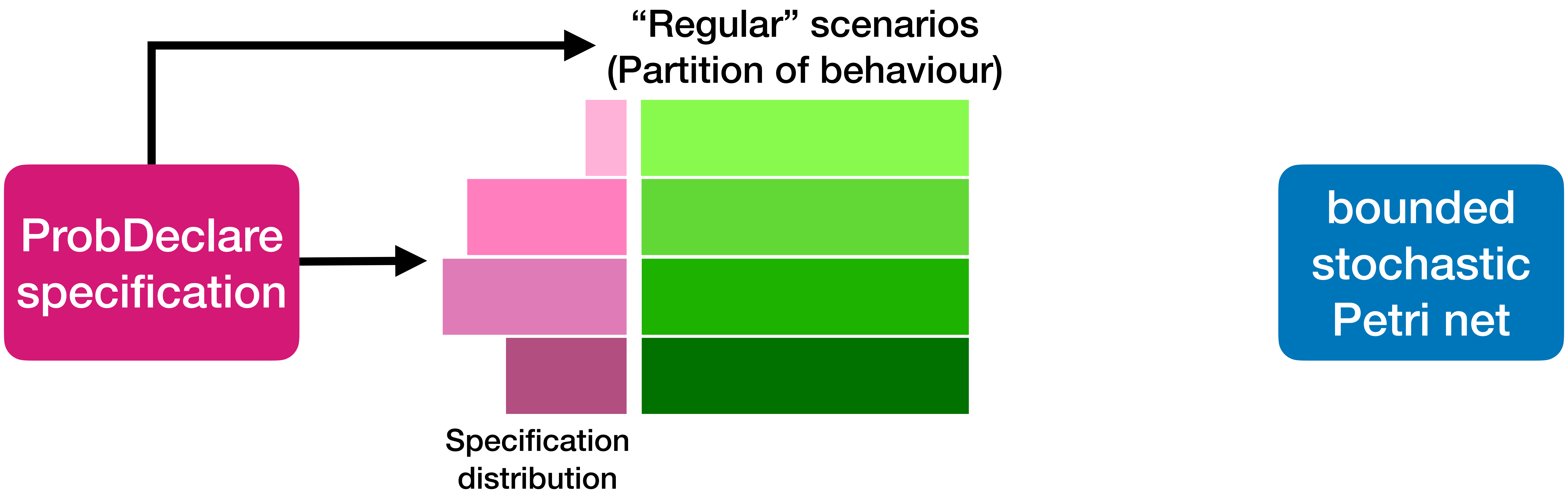
Compare with distributions over traces

Induced distribution via qualitative verification



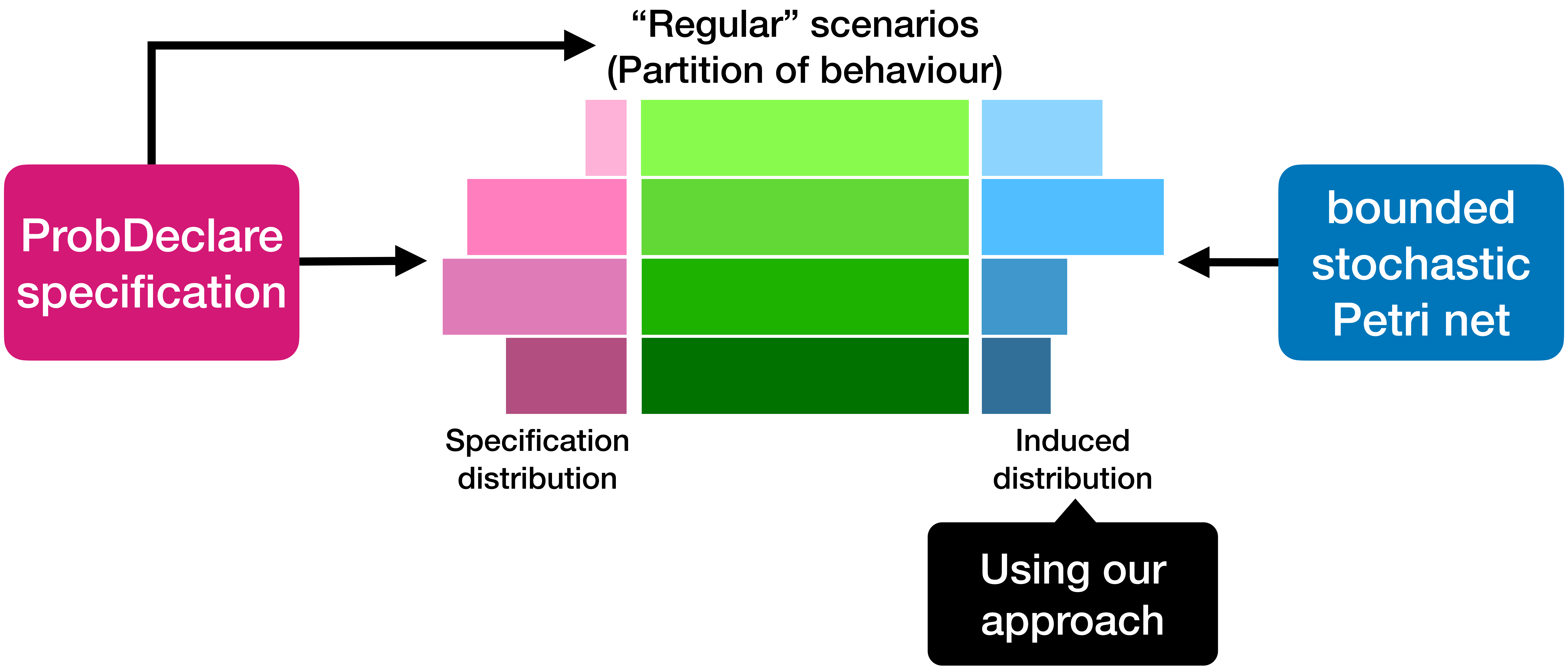
Compare with distributions over traces

Induced distribution via qualitative verification



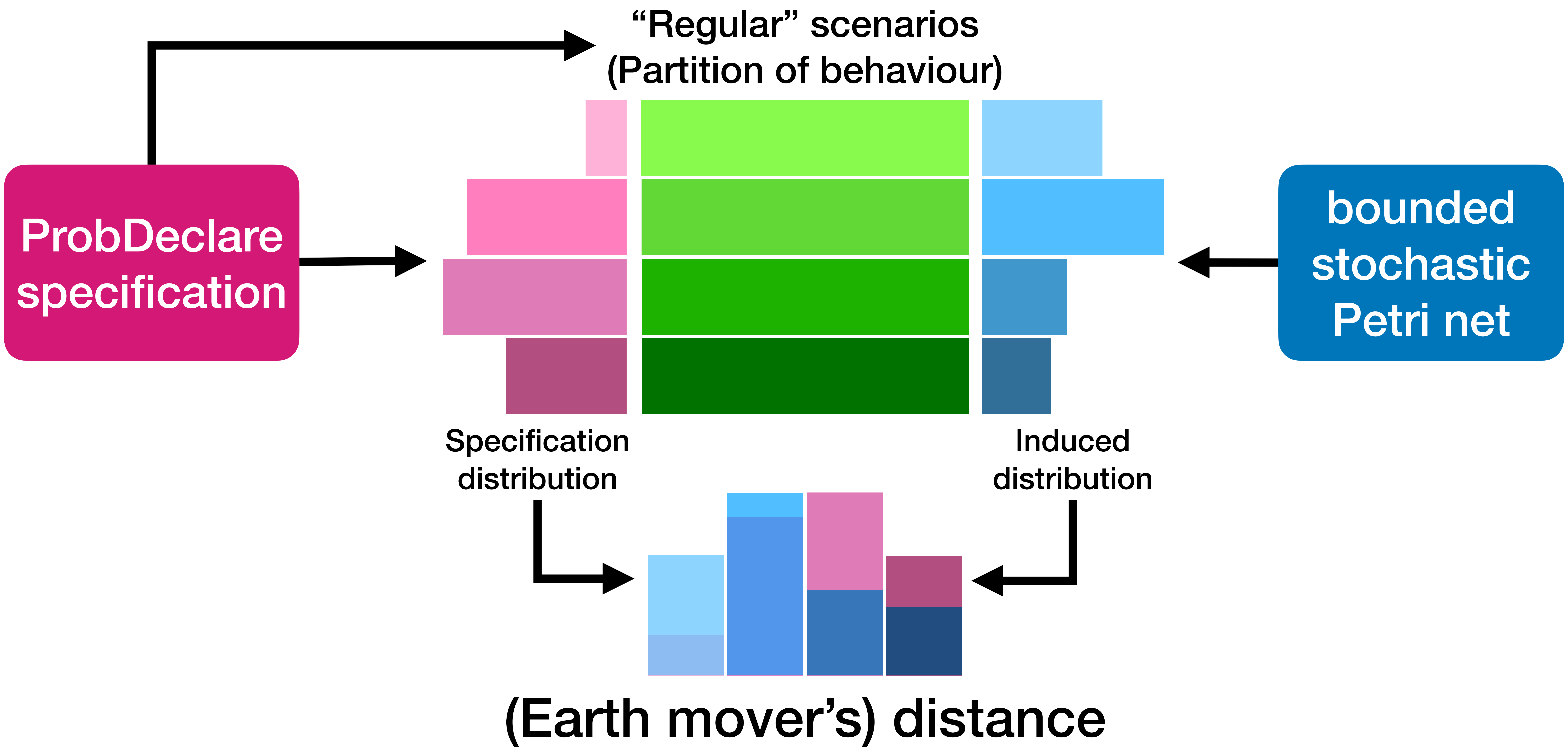
Compare with distributions over traces

Induced distribution via qualitative verification



Compare with distributions over traces

Induced distribution via qualitative verification



Conclusions

Stochastic process mining calls for techniques to **reason on stochastic process models**

Existing techniques do not readily apply due to the features of stochastic process models we are interested in (**silent transitions**)

Analytic solution to key problems related to **computing probabilities of behaviour**

- **Combination and extension** of techniques from **Markov chain analysis** and **qualitative model checking** of quantitative systems

Just the beginning:
timed analysis, discovery, more integration of mining&reasoning, …

Thank you!

montali@inf.unibz.it