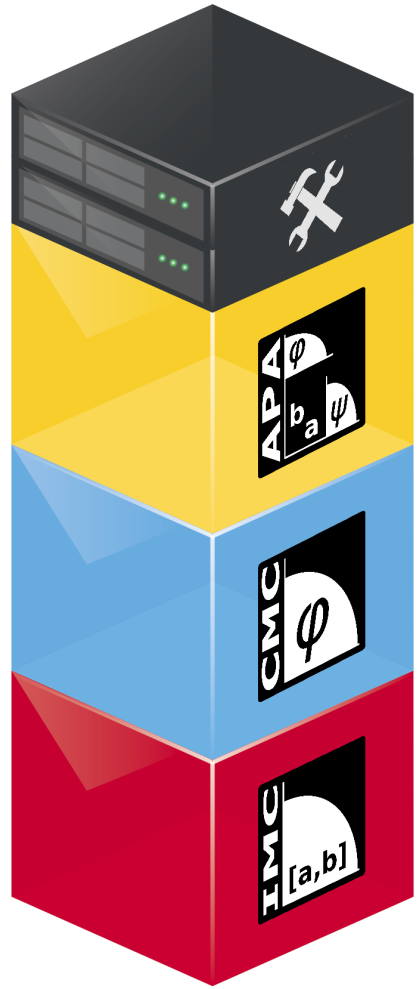
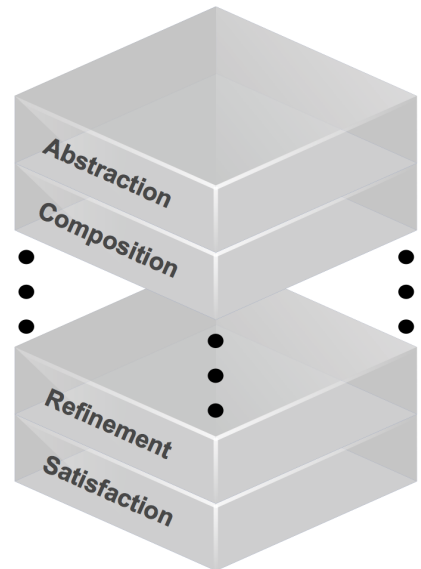
Specification Theories for Probabilistic Systems

Mikkel Larsen Pedersen
Aalborg University, Denmark

PhD defense, Dec 15 2011





AALBORG UNIVERSITY

Cassiopeia
House of Computer Science

MT-LAB
A VKR CENTRE OF EXCELLENCE

Motivation

Context

Embedded systems

- "An embedded system is an engineering artifact involving computation that is subject to physical constraints" Henzinger & Sifakis 2006

- Examples: Braking systems in cars, GPS systems in planes etc.

Motivation

Initiating problem

How do we eliminate errors in embedded systems?

Motivation

Initiating problem

How do we eliminate errors in embedded systems?

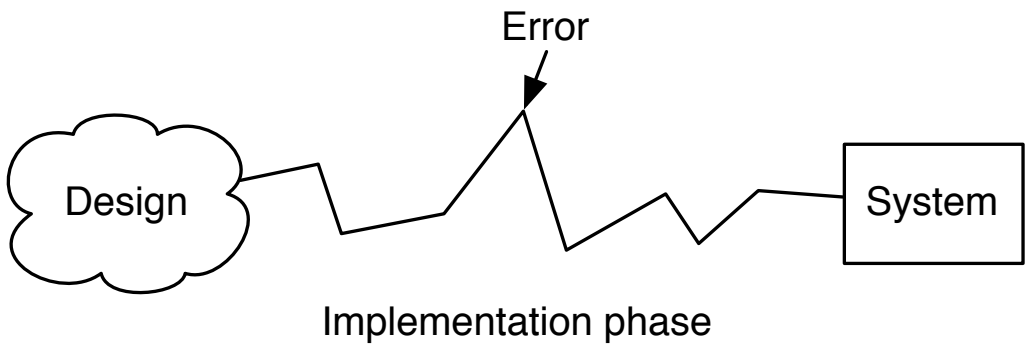
- Errors are expensive (Ariane 5 501, $ 500 million)
- Errors can be life-threatening (Therac-25, 3 persons deceased)

Motivation

Initiating problem

How do we eliminate errors in embedded systems?

- Errors are expensive (Ariane 5 501, $ 500 million)
- Errors can be life-threatening (Therac-25, 3 persons deceased)



Motivation

Initiating problem

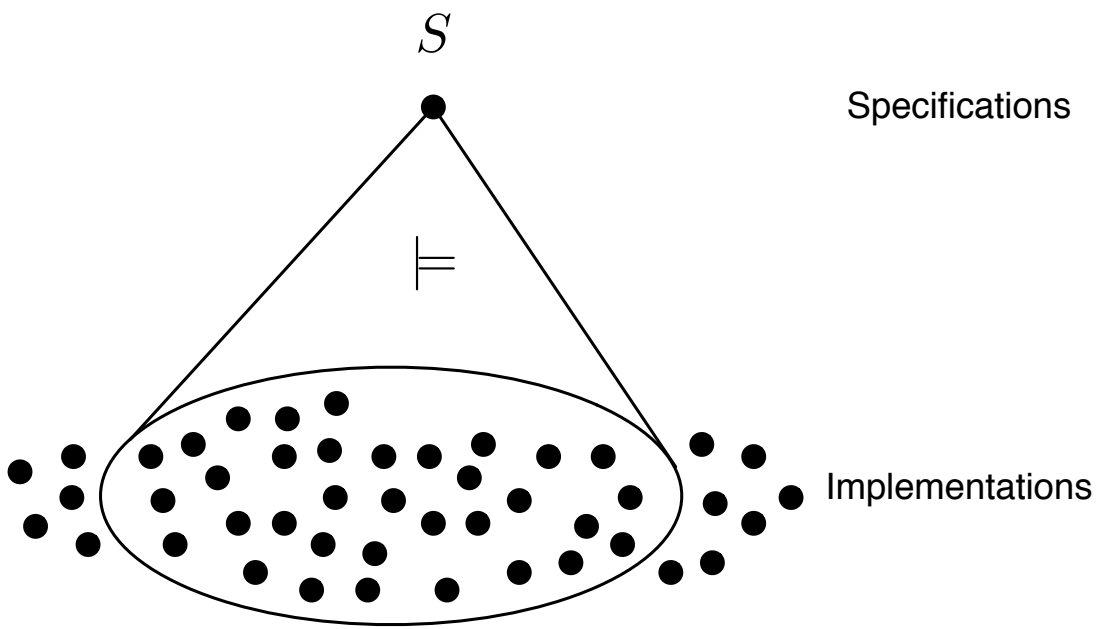
Solution:

- Component-based design
  - Reduced complexity in components
  - Stepwise refinement
  - Independent implementation

Motivation

Specification theories

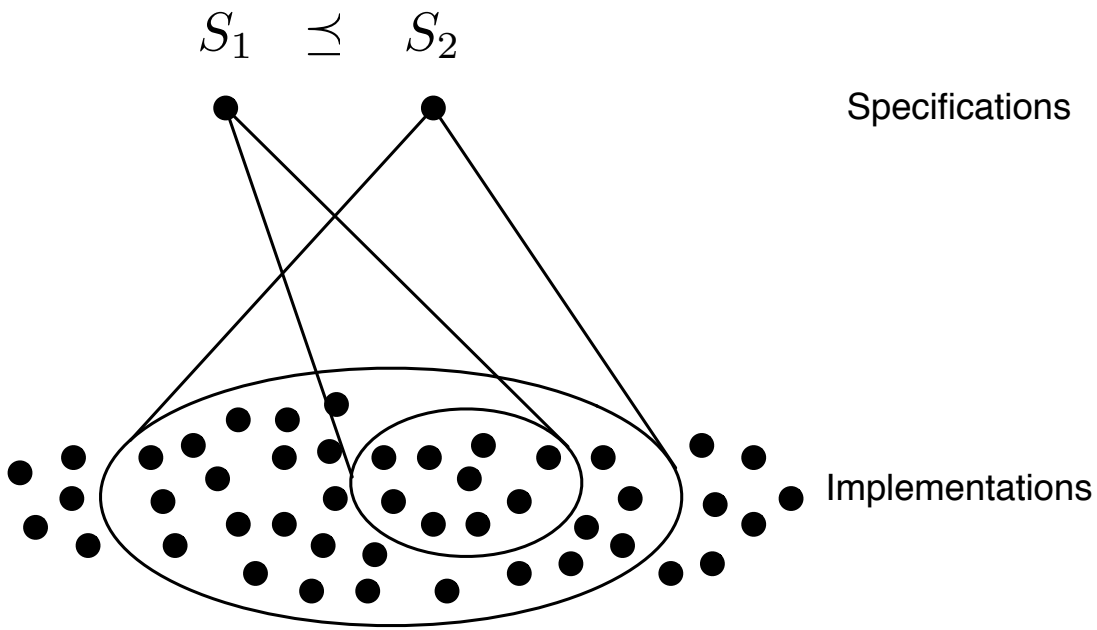
- Satisfaction



Motivation

Specification theories

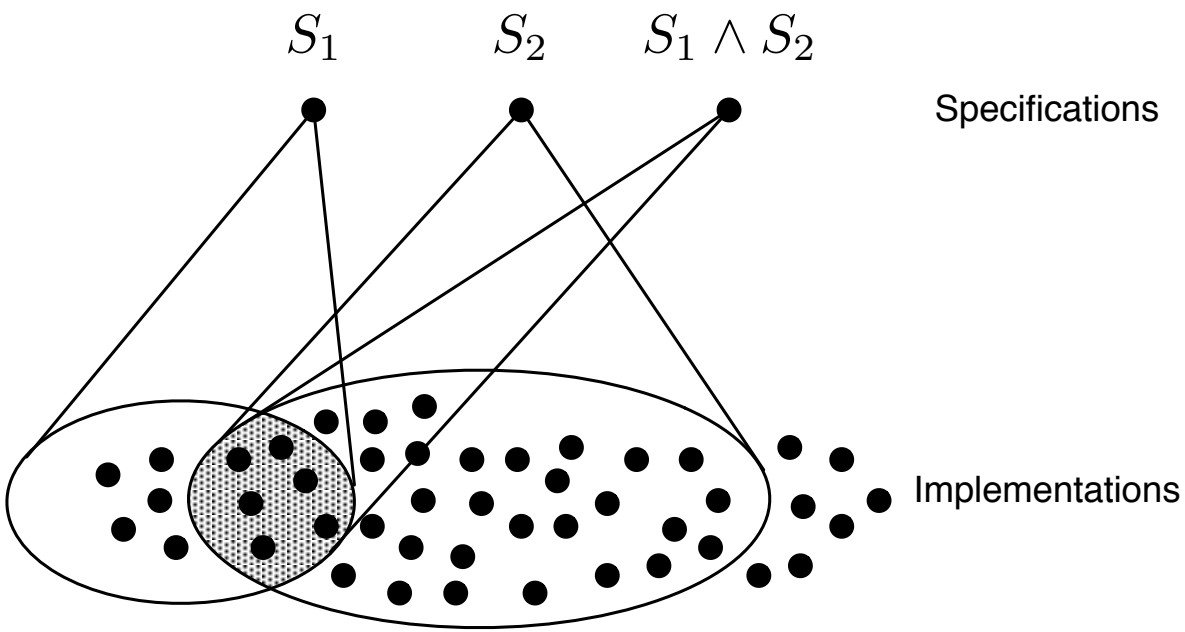
- Refinement



Motivation

Specification theories

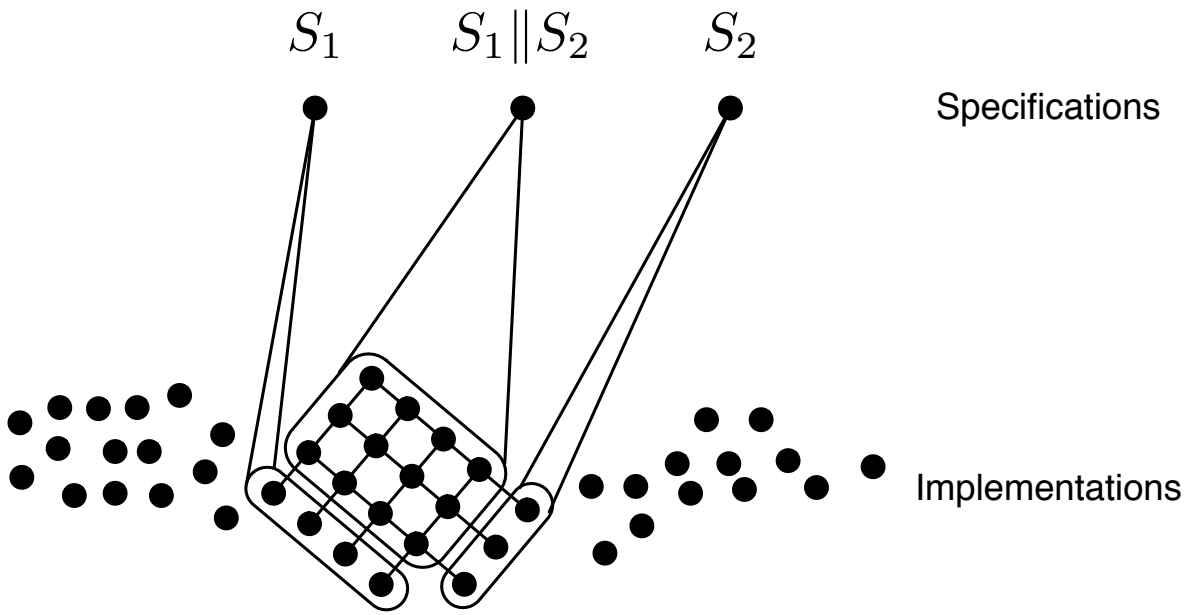
- Conjunction



Motivation

Specification theories

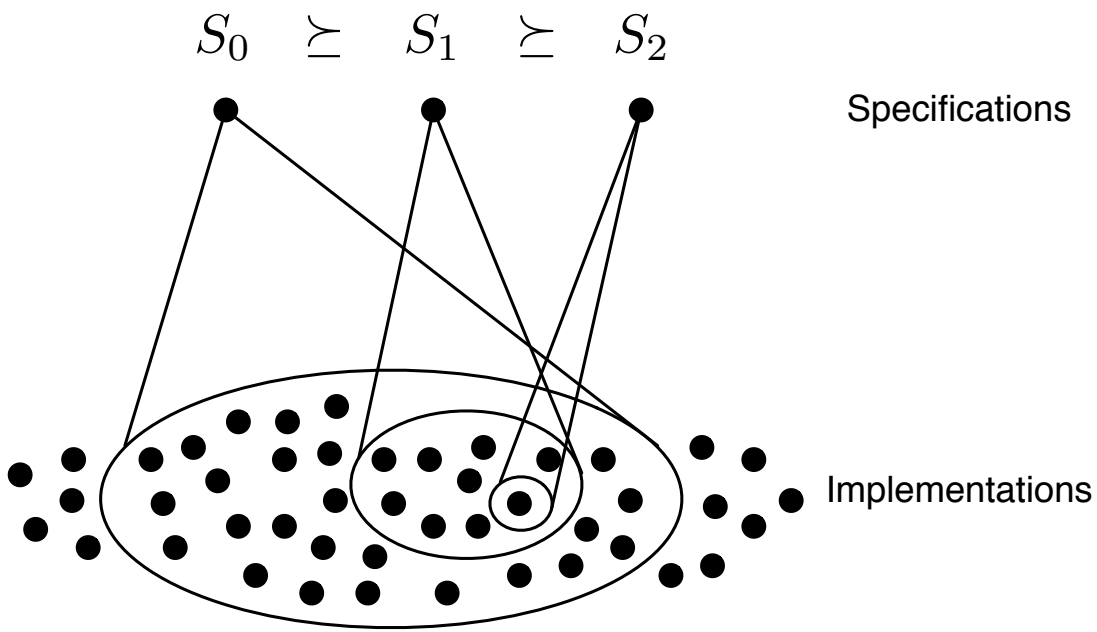
- Parallel composition



Motivation

Specification theories

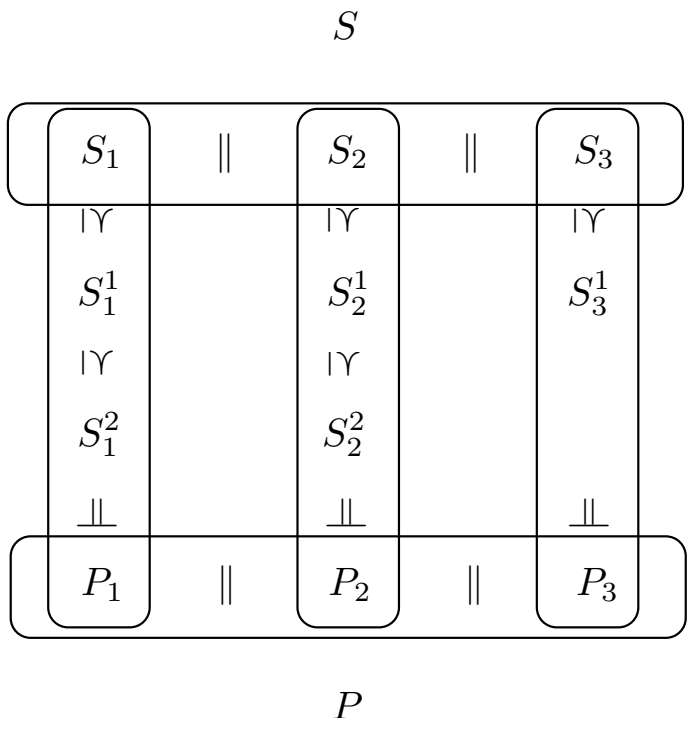
- Stepwise refinement



Motivation

Specification theories

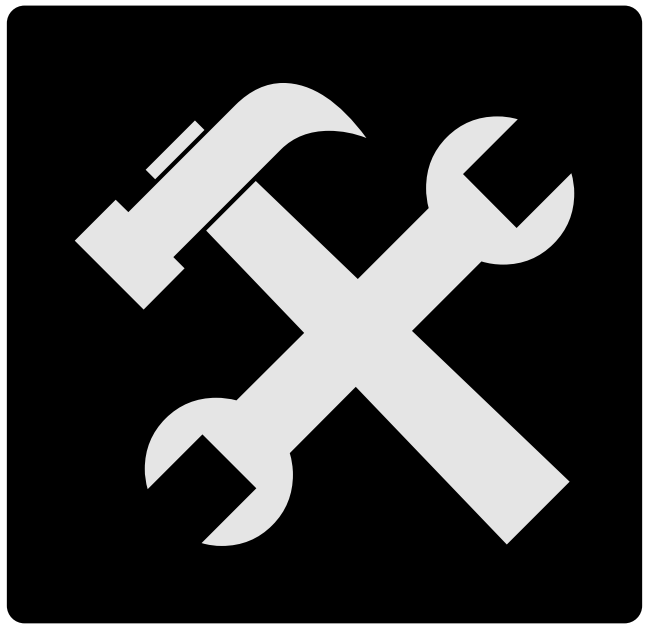
- Independent implementability



Motivation

Specification theories

- Tool support



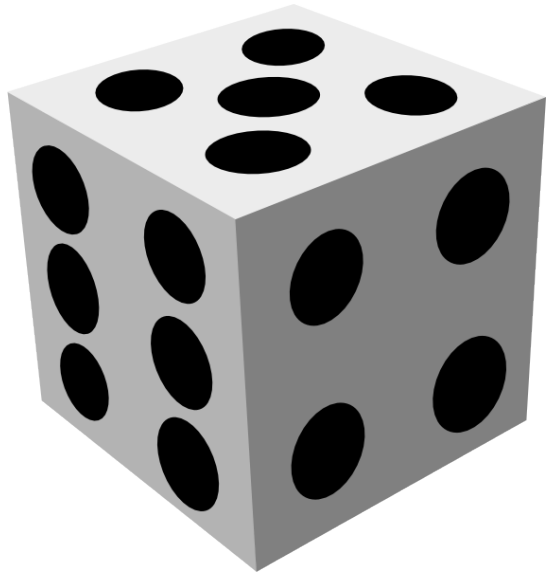
Motivation

Probabilities: in general

- Probabilities abstracts away hidden information
- Allows for quantitative reasoning
  - E.g. what is the probability to avoid a specific error?

Motivation

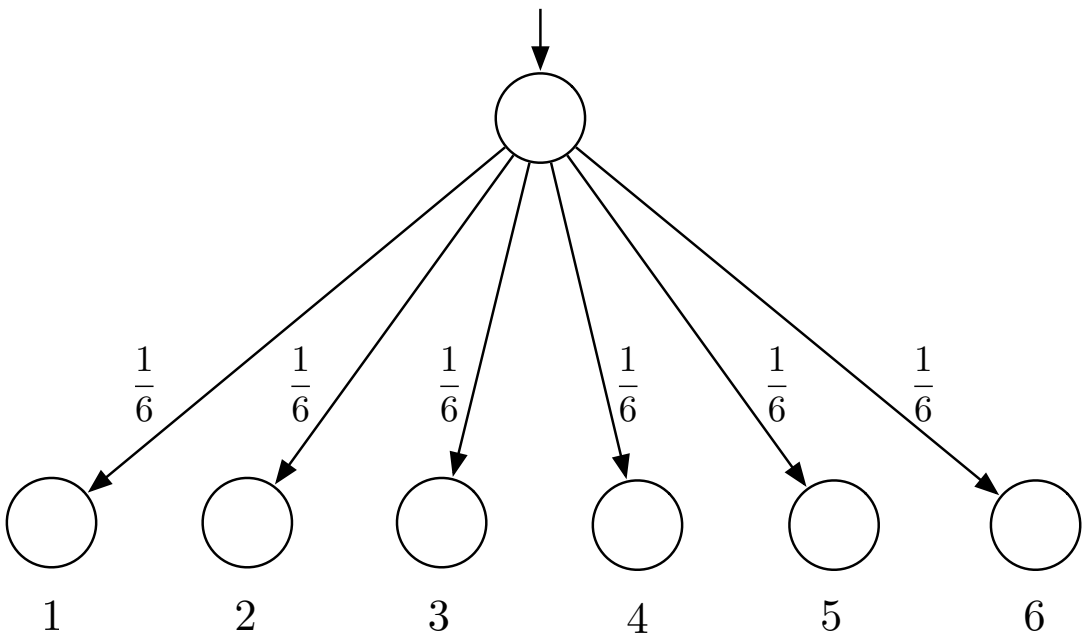
Probabilities: an informal example

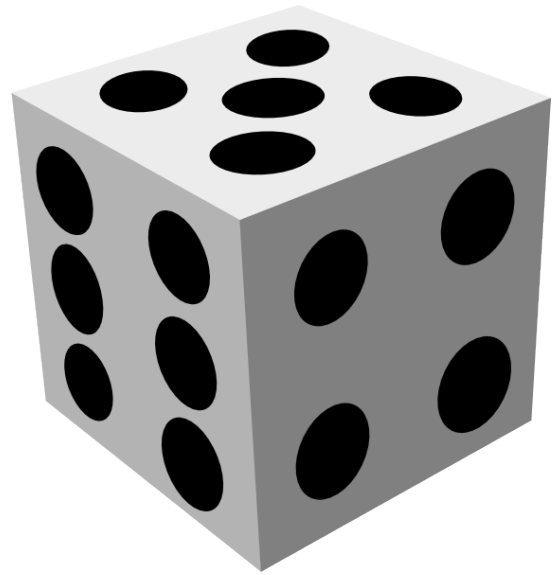


- How could this be modelled?

Motivation

Probabilities: an informal example

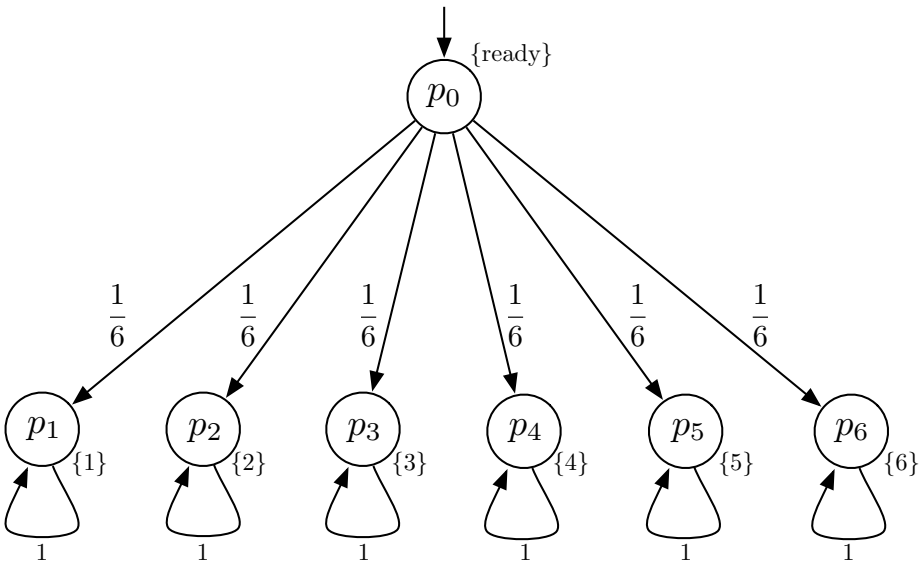




- Distribution is learned by e.g. sampling

Motivation

Probabilities: an informal example



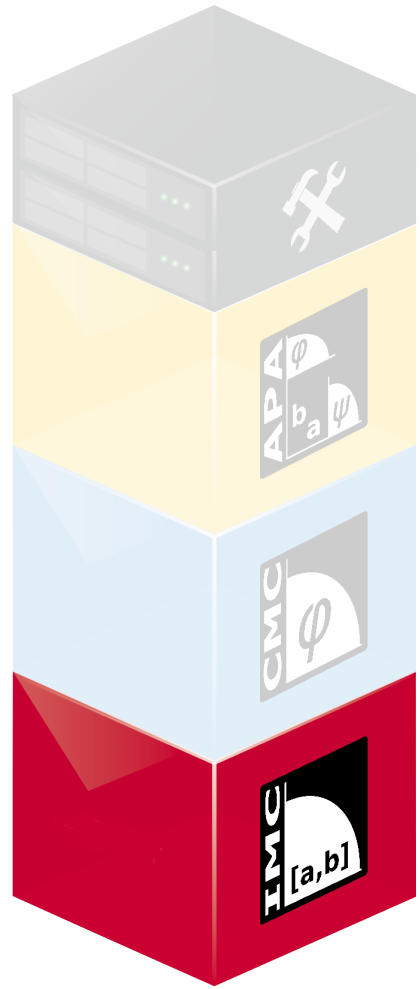
Definition

A *Markov Chain* consists of

- P: set of states, $p_0 \in P$
- A: atomic propositions,
- $V_C : P \to 2^A$: state valuation function
- $\pi : P \to \mathit{Dist}(Q)$: probability distribution assignment

Markov 1906

Interval Markov Chains



Paper A (**JLAP**, LATA'11)

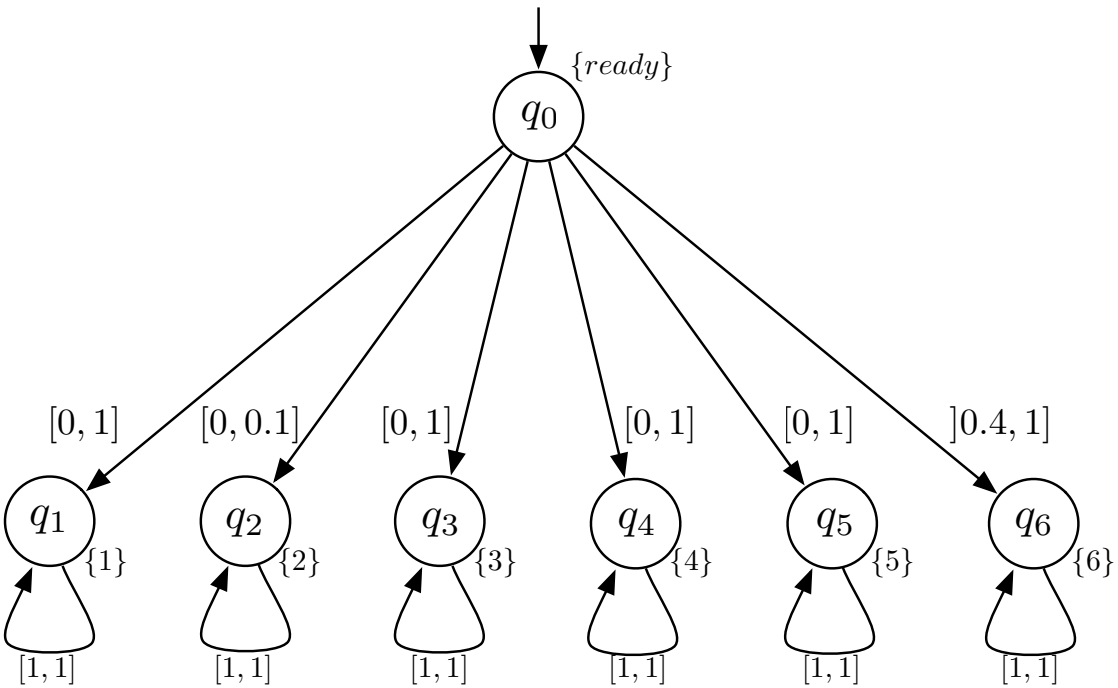
Interval Markov Chains

Motivation

- We modelled the behaviour of a dice
- How could we specify a dice?

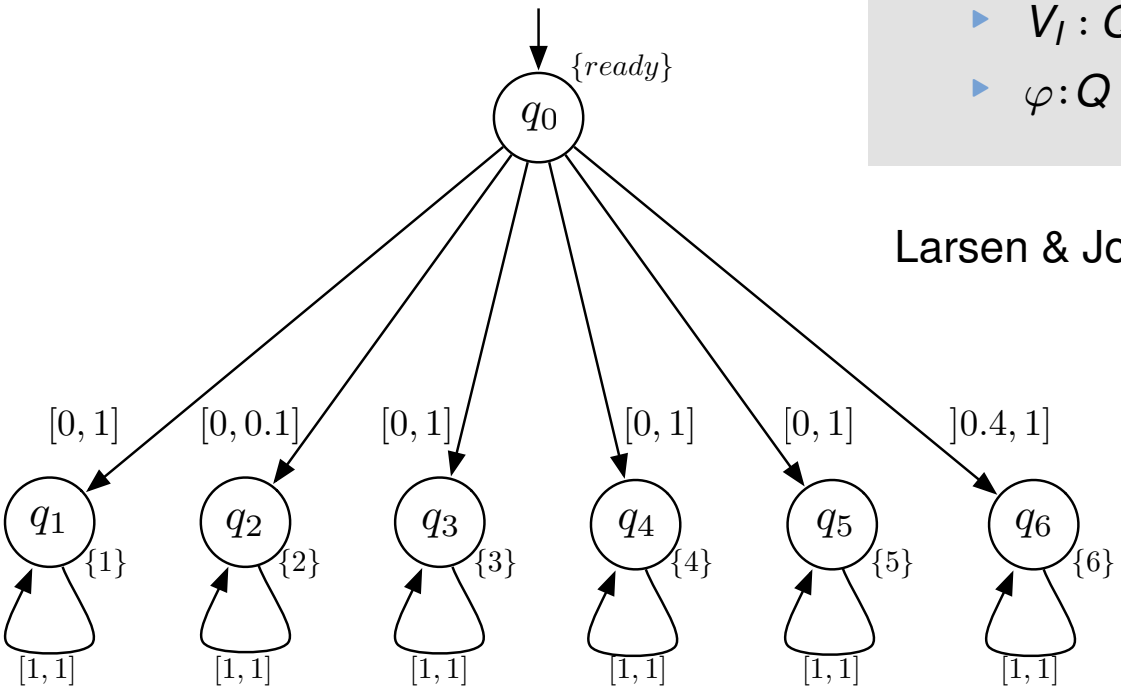
Interval Markov Chains

Motivation



Interval Markov Chains

Motivation



Definition

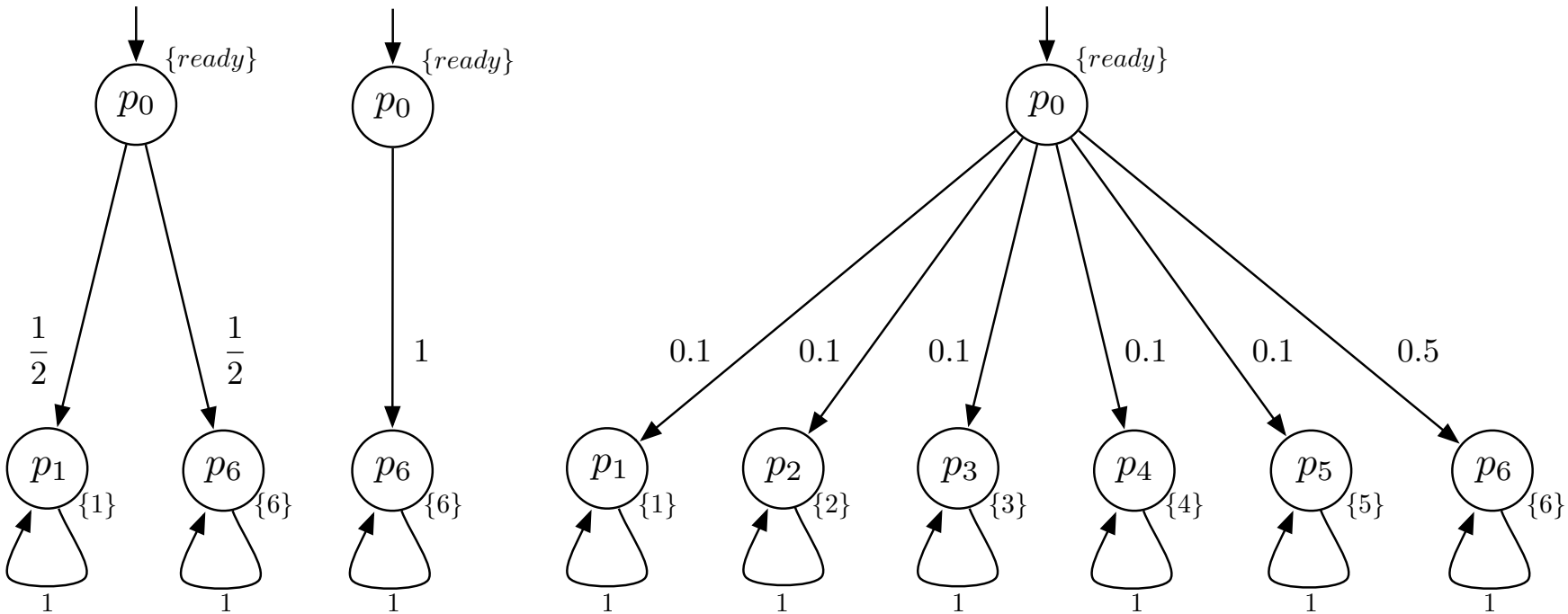
An *Interval Markov Chain* consists of

- Q: set of states, $q_0 \in Q$
- A: atomic propositions
- $V_l : Q \to 2^A$: state valuation function
- $\varphi : Q \to (Q \to \text{Intervals}_{[0,1]})$.

Larsen & Jonsson 1991

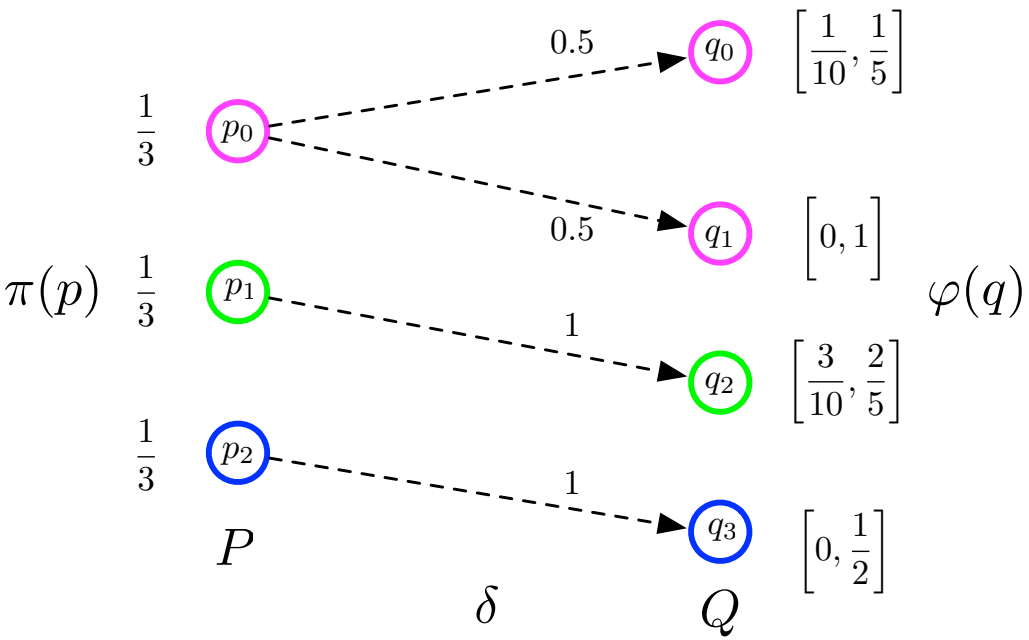
Interval Markov Chains

Motivation



Interval Markov Chains

Satisfaction - redistribution



Definition

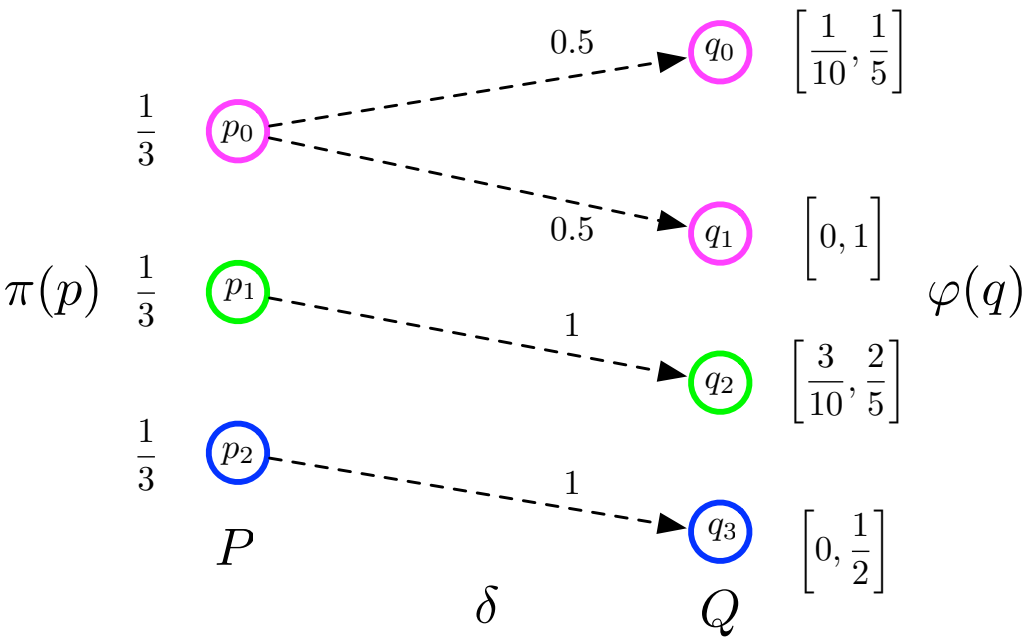
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

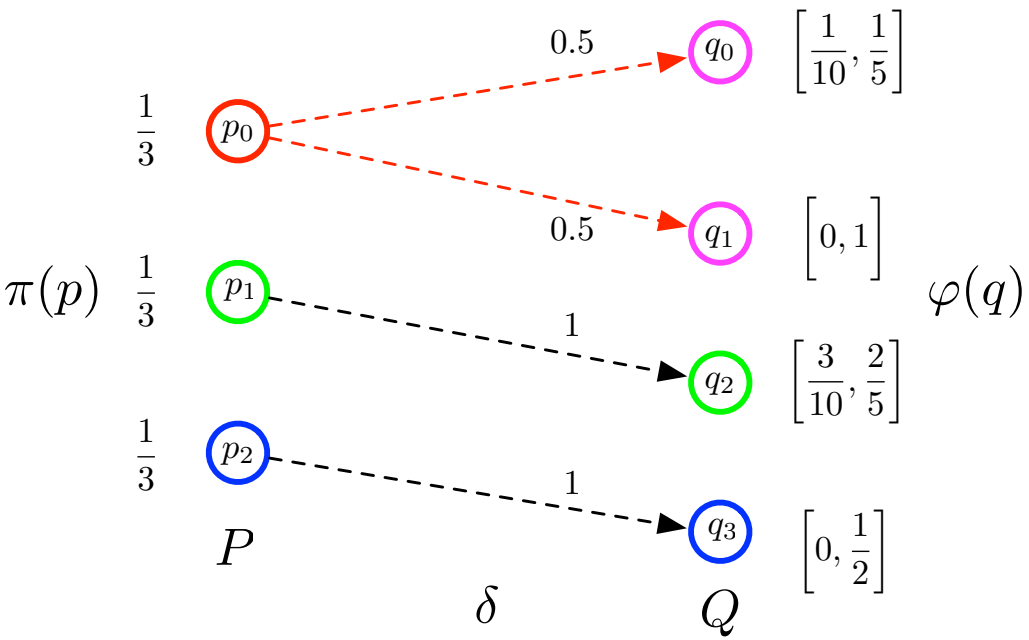
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

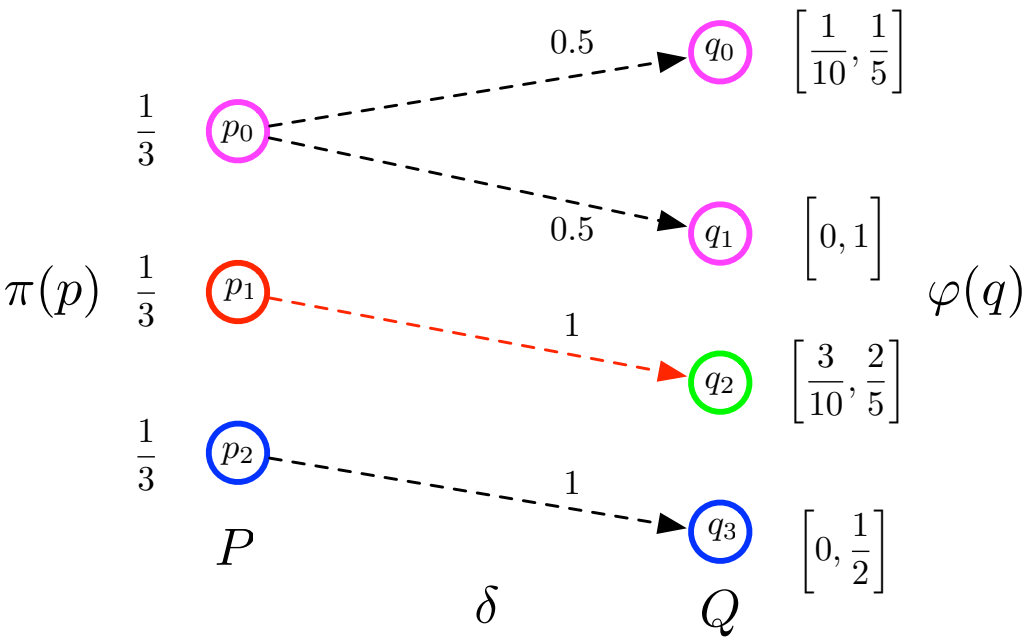
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

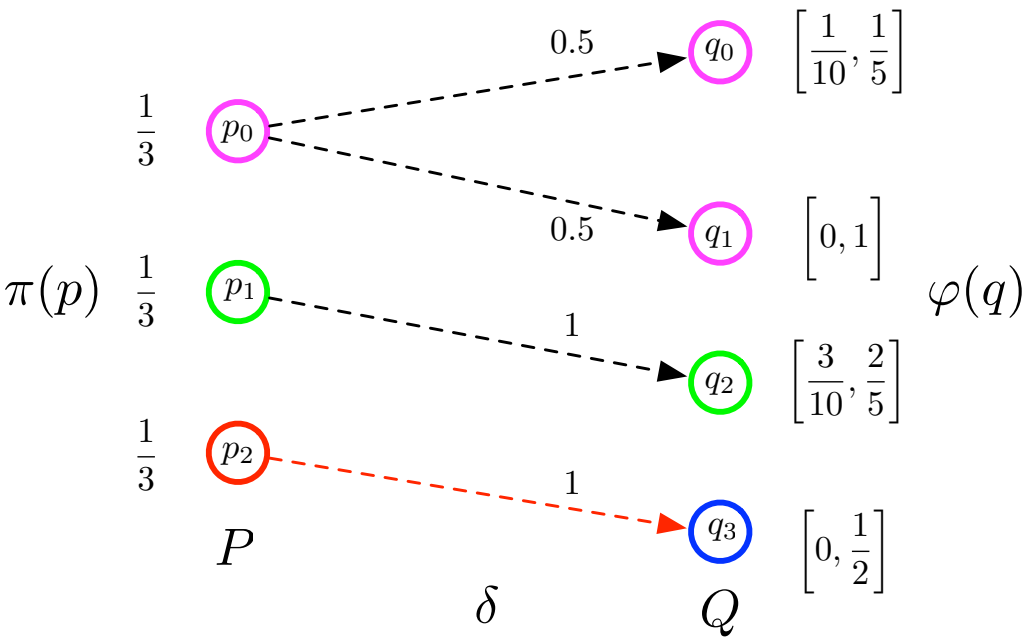
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

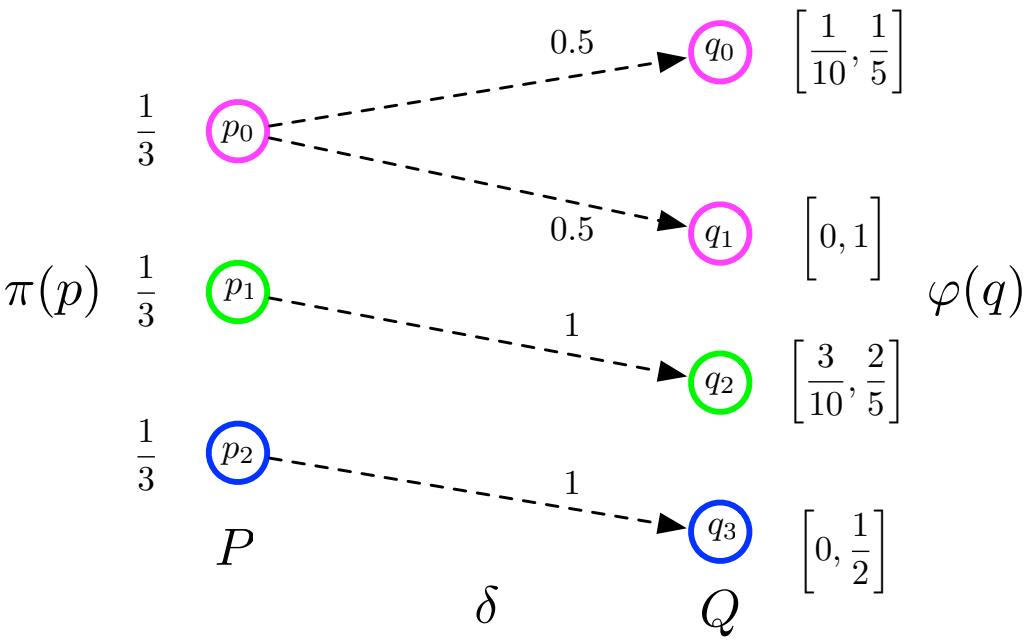
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

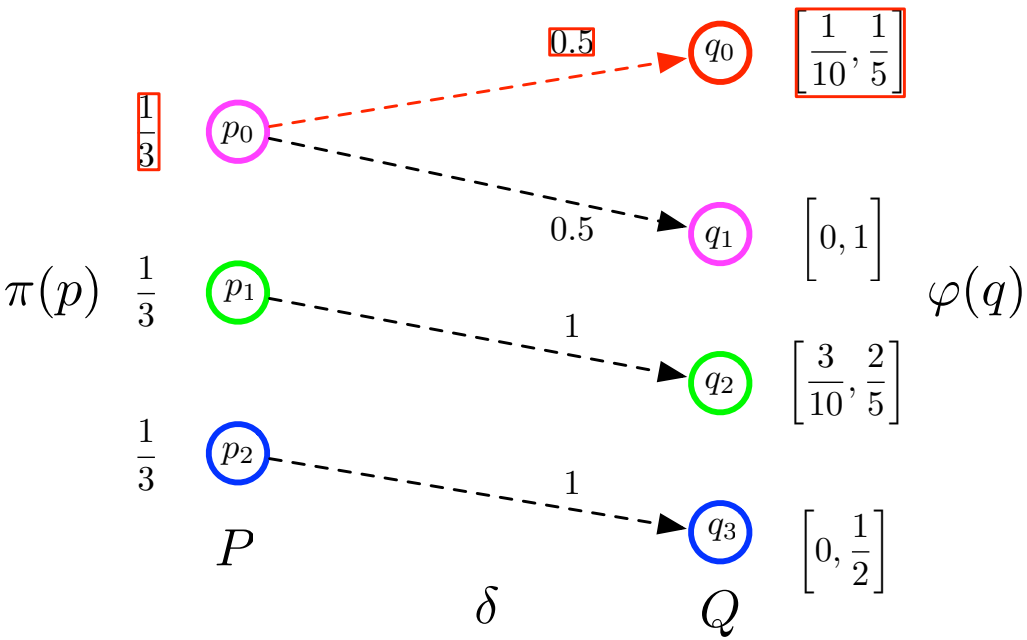
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

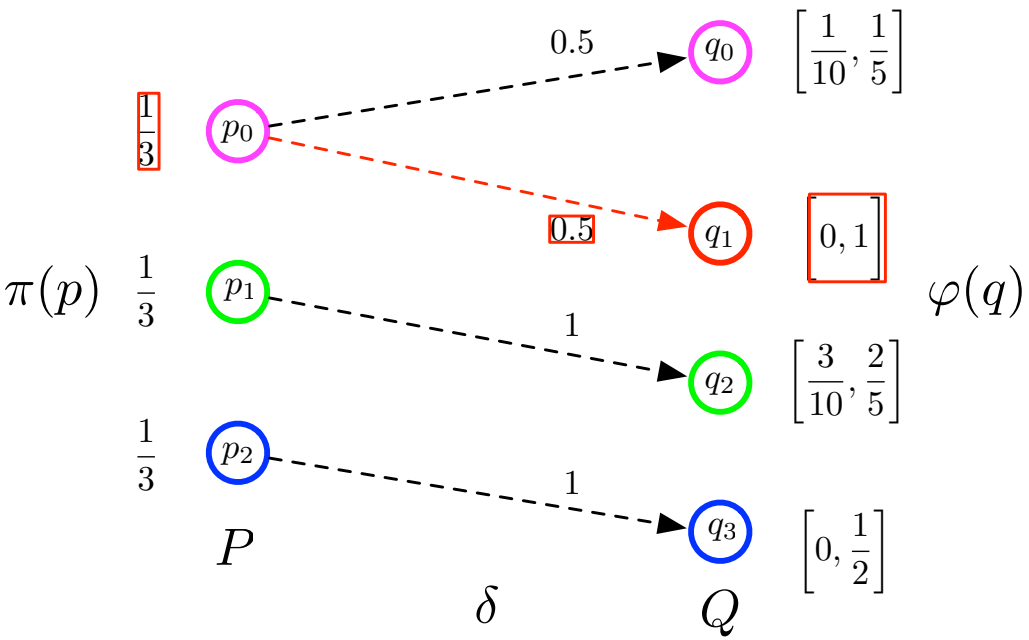
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

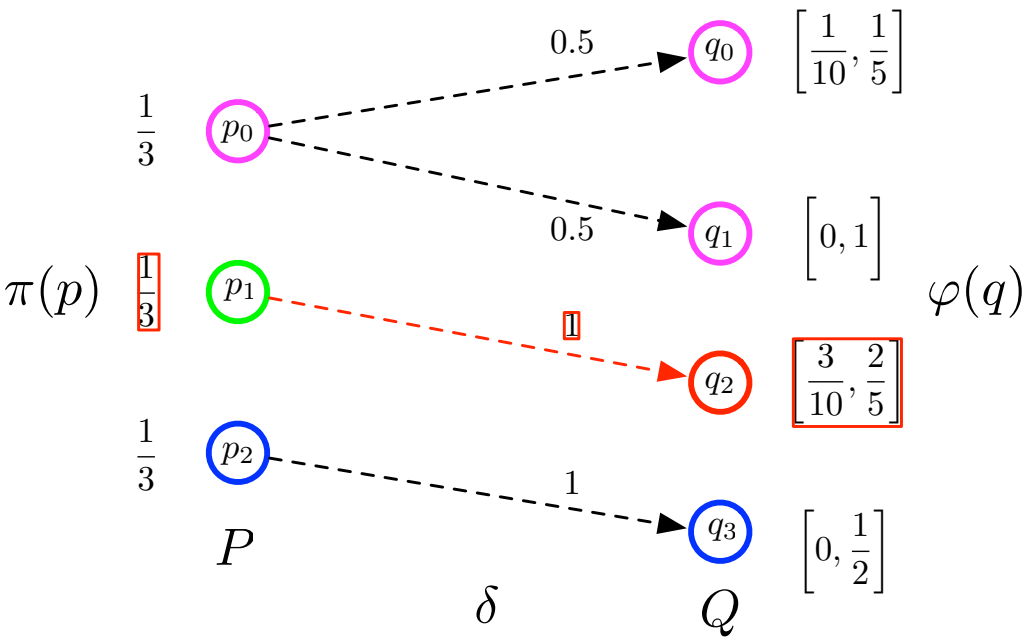
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

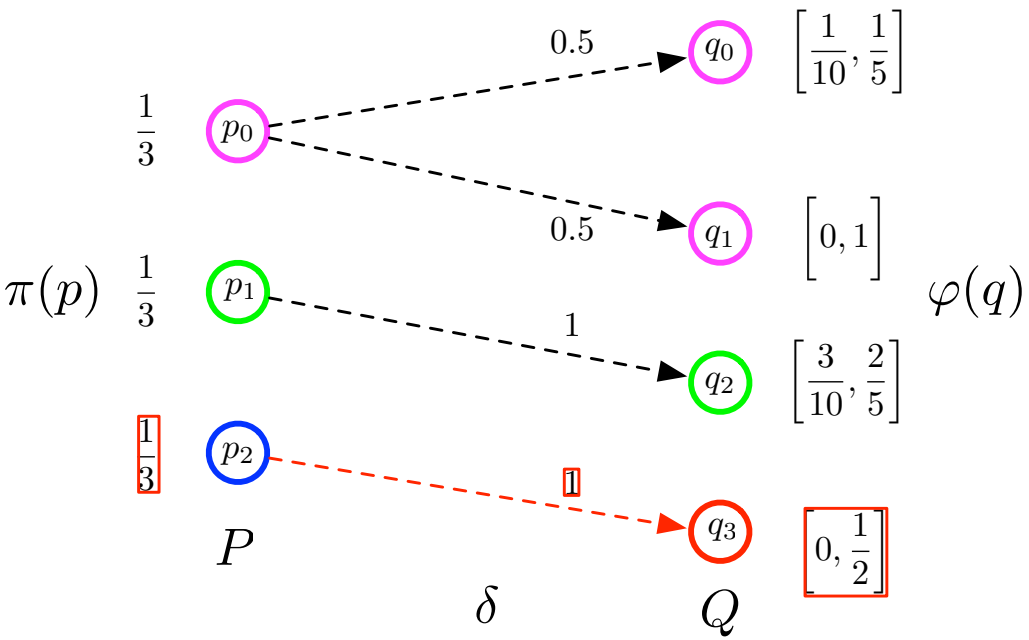
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

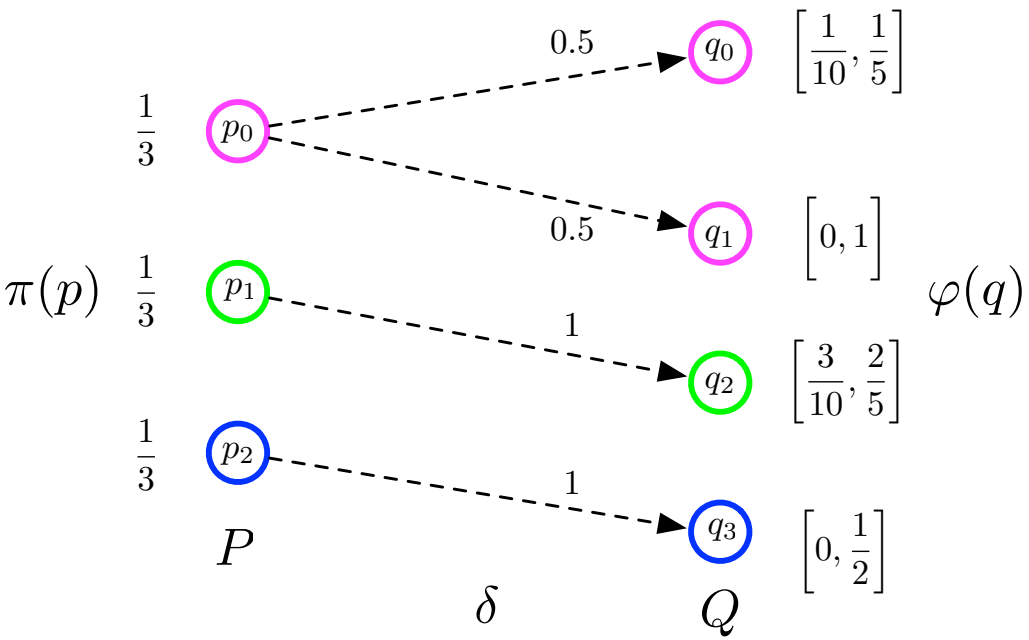
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction - redistribution



Definition

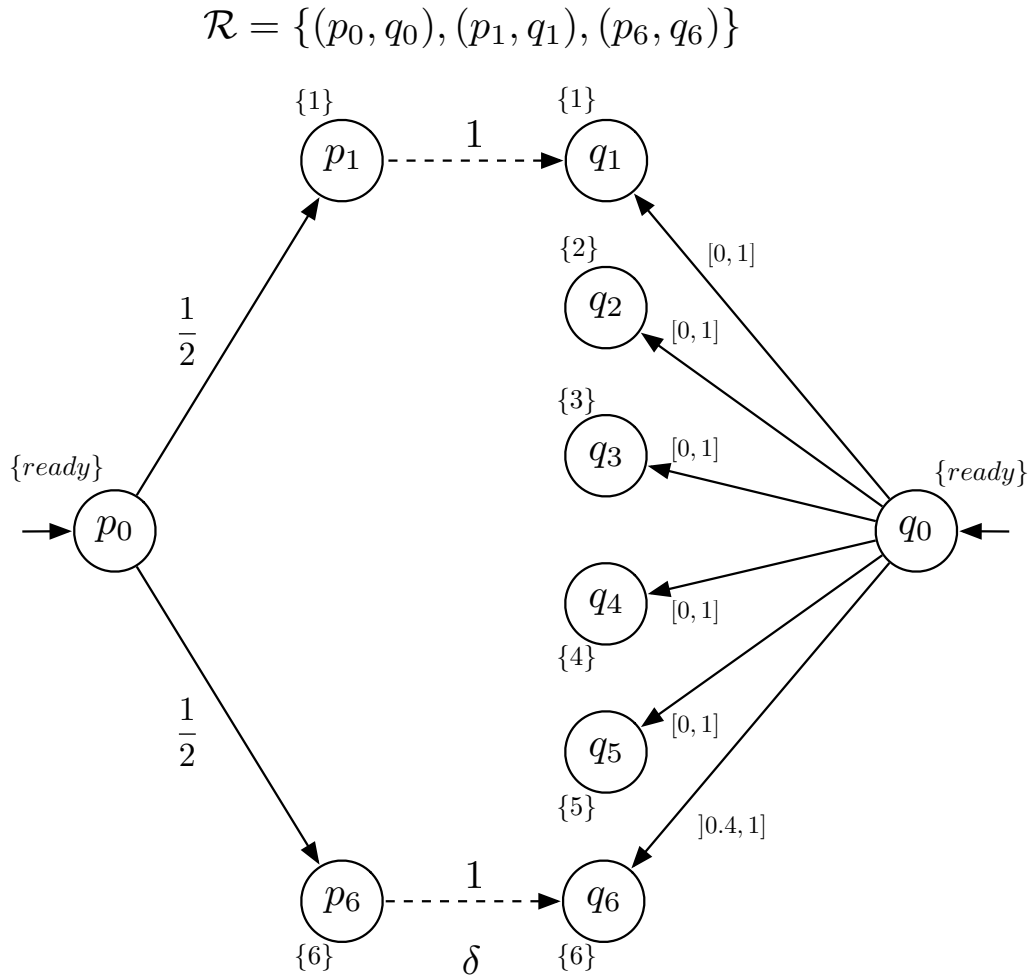
$p \in C$ MC, $q \in I$ IMC
$\delta : P \to (Q \to [0,1])$: correspondence function
$\mathcal{R} \subseteq C \times I$

$\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$, if and only if

- if $\pi(p)(p') > 0$, then $\delta(p')$ is a distribution on Q,
- $\sum_{p' \in P} \pi(p)(p')\delta(p')(q') \in \varphi(q)(q')$, and
- if $\delta(p')(q') > 0$, then $p' \mathcal{R} q'$.

Interval Markov Chains

Satisfaction



Definition

C be a MC, I IMC
$\mathcal{R} \subseteq C \times I$ is a *satisfaction relation* if and only if whenever $p \mathcal{R} q$ then

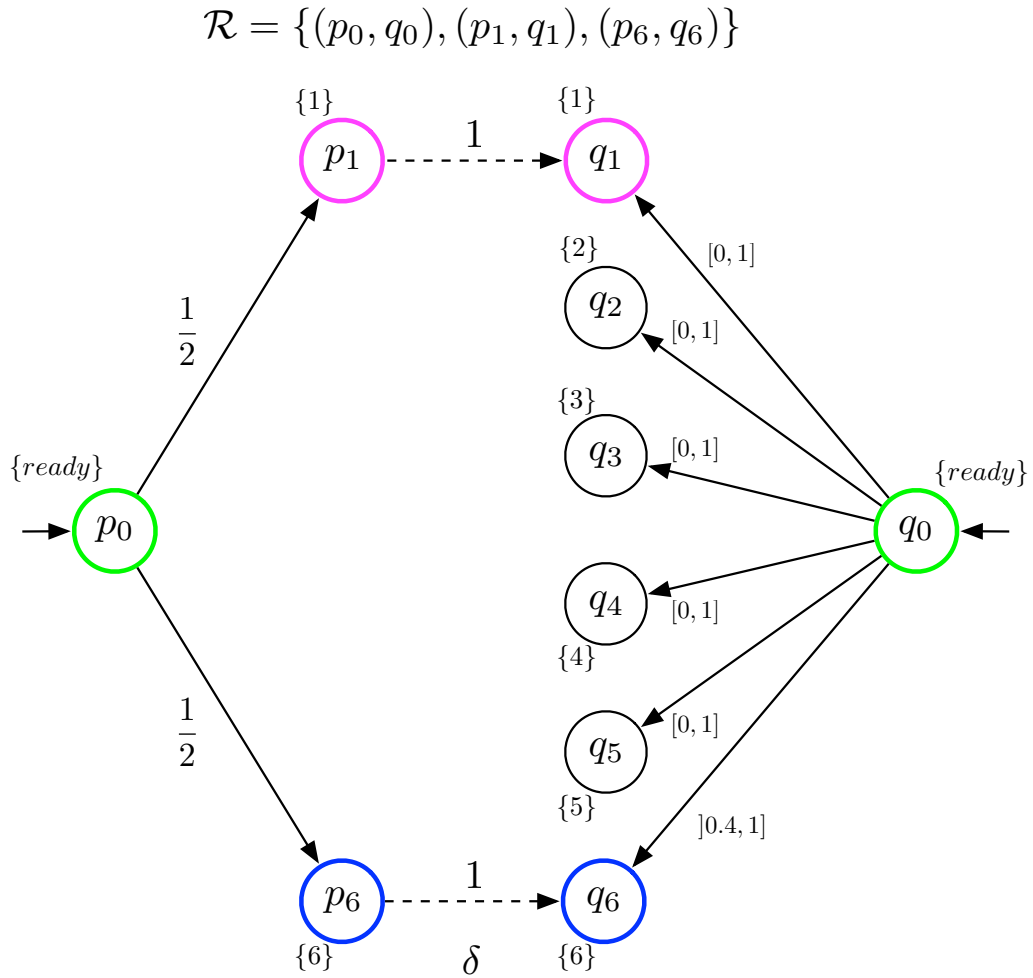
- $V_C(p) = V_I(q)$, and
- $\exists \delta : P \to (Q \to [0, 1])$ such that $\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$.

$C \models I \iff \exists\ \mathcal{R}$ such that $(p_0, q_0) \in \mathcal{R}$. We write that $C \in [\![I]\!]$.

Larsen & Jonsson 1991

Interval Markov Chains

Satisfaction



Definition

C be a MC, I IMC
$\mathcal{R} \subseteq C \times I$ is a *satisfaction relation* if and only if whenever $p \mathcal{R} q$ then

- $V_C(p) = V_I(q)$, and
- $\exists \delta : P \to (Q \to [0, 1])$ such that $\pi(p) \lhd^{\delta}_{\mathcal{R}} \varphi(q)$.

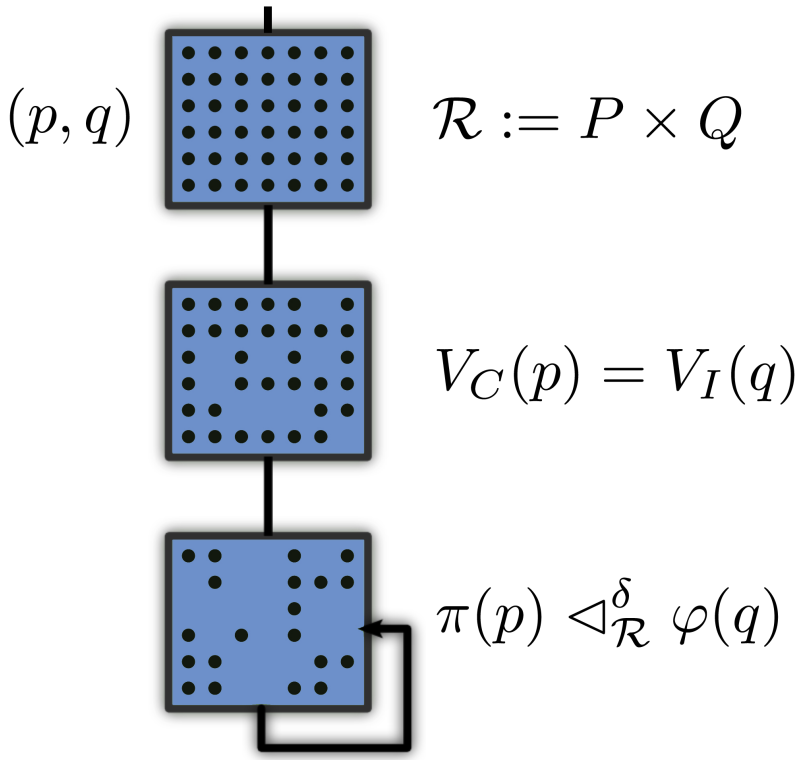
$C \models I \iff \exists\ \mathcal{R}$ such that $(p_0, q_0) \in \mathcal{R}$. We write that $C \in [\![I]\!]$.

Larsen & Jonsson 1991

Interval Markov Chains

Satisfaction: towards an algorithm

$\models$



Interval Markov Chains

Refinement relations

Assume I_1 and I_2 IMCs and $(q, s) \in Q \times S$.

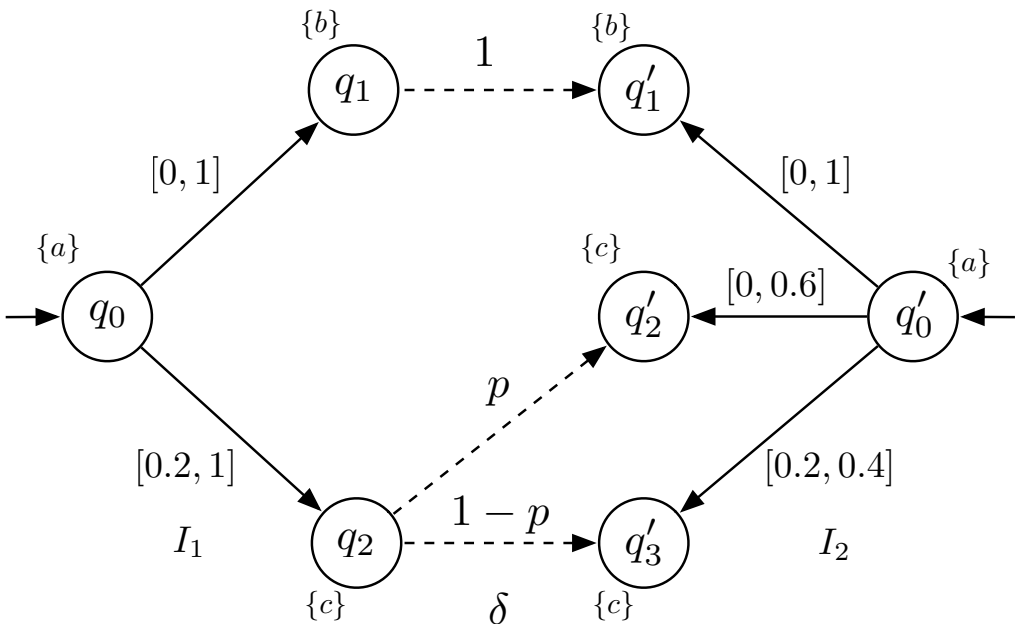
- Strong refinement $\preceq_S$: $\exists \delta \forall \sigma \in \varphi_1(q) : \sigma \lhd^{\delta}_{\mathcal{R}} \varphi_2(s)$
- Weak refinement $\preceq$: $\forall \sigma \in \varphi_1(q) \exists \delta : \sigma \lhd^{\delta}_{\mathcal{R}} \varphi_2(s)$
- Thorough refinement $\preceq_T$: $[\![I_1]\!] \subseteq [\![I_2]\!]$

Interval Markov Chains

Refinement relations

Assume I_1 and I_2 IMCs and $(q, s) \in Q \times S$.

- Strong refinement $\preceq_S$: $\exists \delta \forall \sigma \in \varphi_1(q) : \sigma \lhd^{\delta}_{\mathcal{R}} \varphi_2(s)$
- Weak refinement $\preceq$: $\forall \sigma \in \varphi_1(q) \exists \delta : \sigma \lhd^{\delta}_{\mathcal{R}} \varphi_2(s)$
- Thorough refinement $\preceq_T$: $[\![I_1]\!] \subseteq [\![I_2]\!]$



Interval Markov Chains

Results

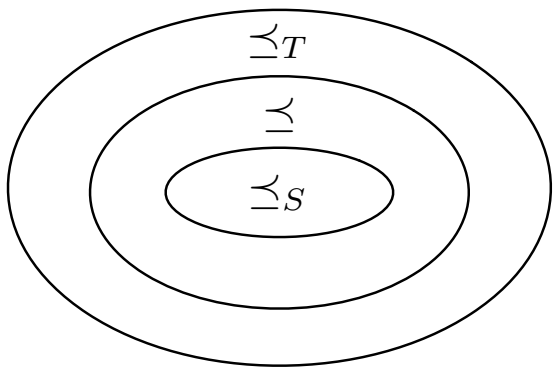
Theorem

Let I_1 and I_2 be IMCs. We have that

$$I_1 \preceq_S I_2 \Rightarrow I_1 \preceq I_2 \Rightarrow I_1 \preceq_T I_2,$$

and under determinism,

$$I_1 \preceq_S I_2 \Leftrightarrow I_1 \preceq I_2 \Leftrightarrow I_1 \preceq_T I_2.$$

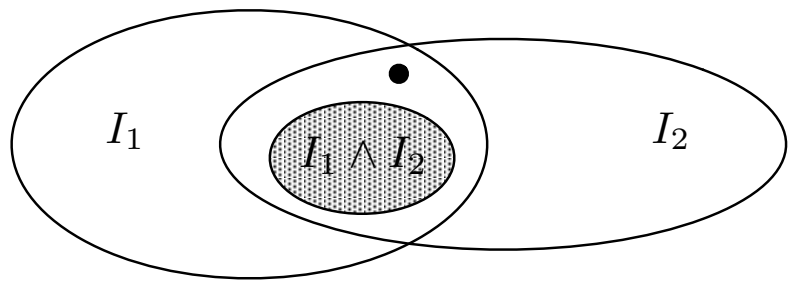


Interval Markov Chains

Closure problems: conjunction

Requirement for conjunction operator:

- $I_1 \wedge I_2 \preceq I_1$ and $I_1 \wedge I_2 \preceq I_2$, and
- for all I_3, if $I_3 \preceq I_1$ and $I_3 \preceq I_2$, then $I_3 \preceq I_1 \wedge I_2$.

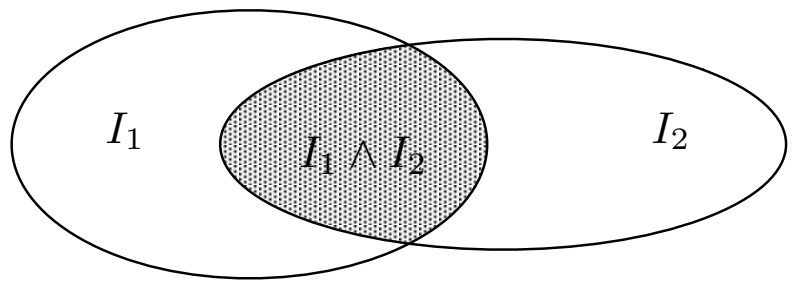


Interval Markov Chains

Closure problems: conjunction

Requirement for conjunction operator:

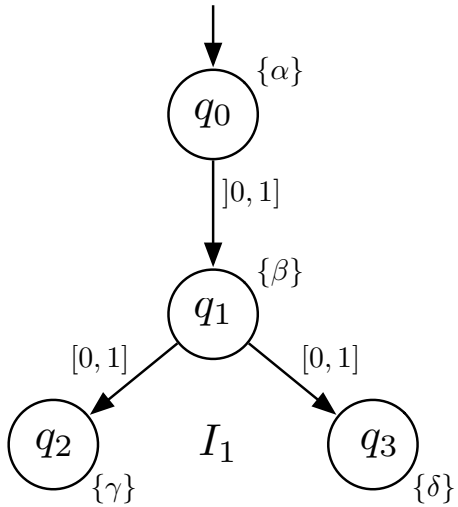
- $I_1 \wedge I_2 \preceq I_1$ and $I_1 \wedge I_2 \preceq I_2$, and
- for all I_3, if $I_3 \preceq I_1$ and $I_3 \preceq I_2$, then $I_3 \preceq I_1 \wedge I_2$.

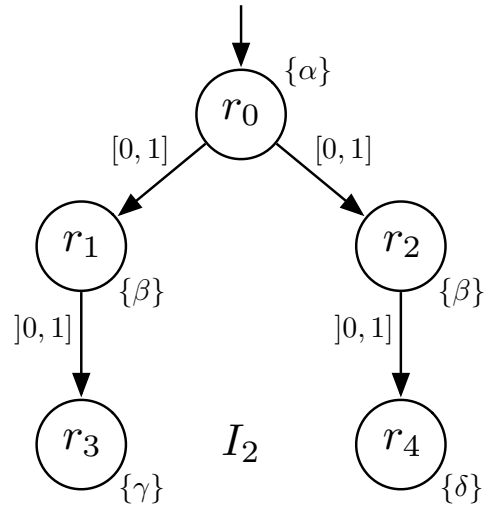


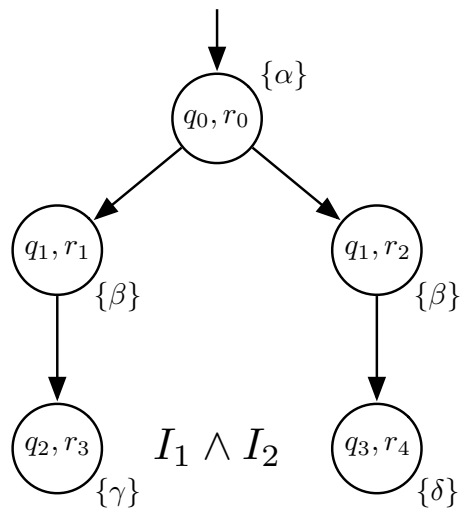
$$[\![I_1 \wedge I_2]\!] = [\![I_1]\!] \cap [\![I_2]\!]$$

Interval Markov Chains

Closure problems: conjunction

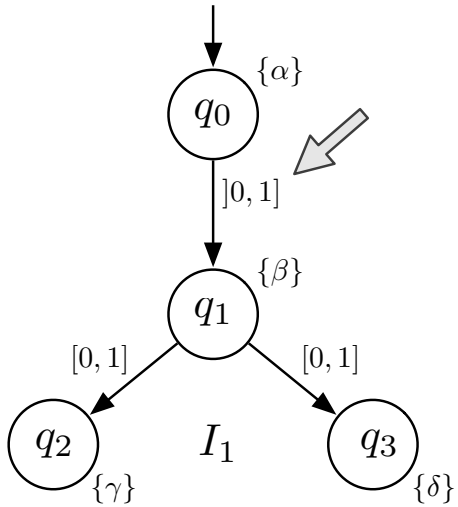


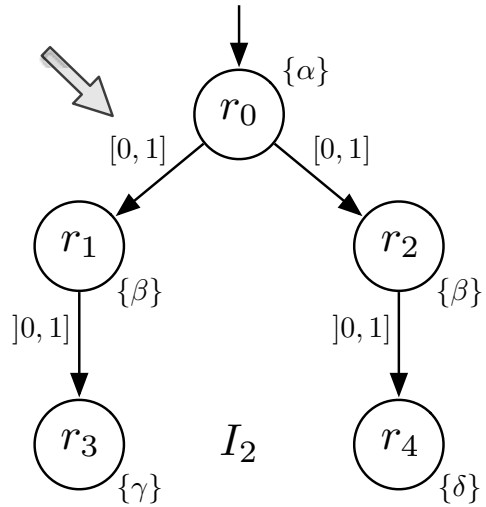


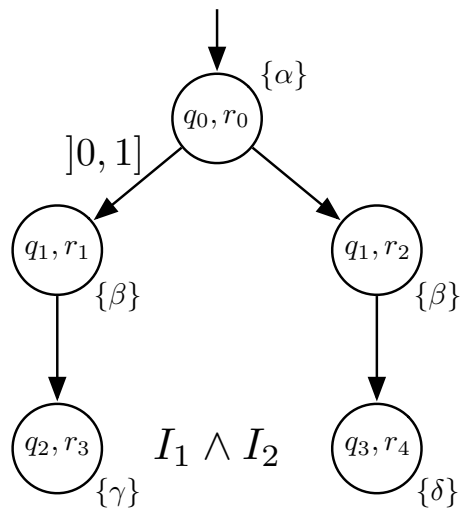


Interval Markov Chains

Closure problems: conjunction

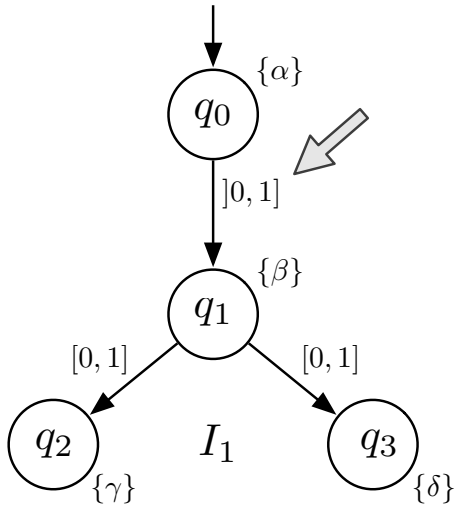


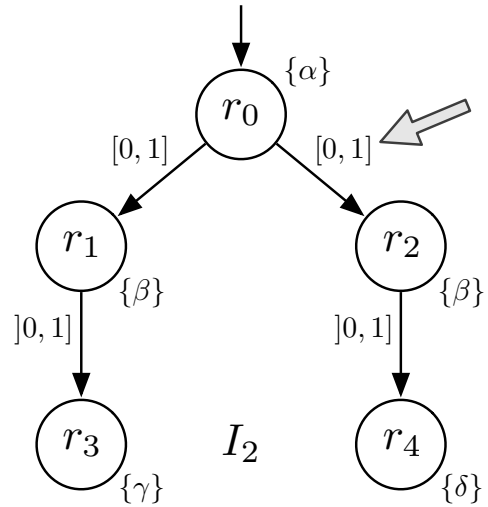


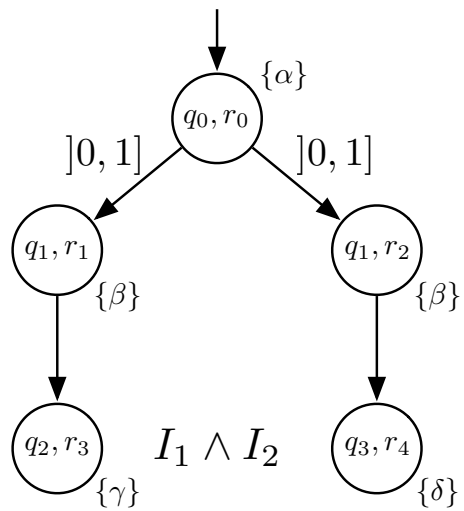


Interval Markov Chains

Closure problems: conjunction

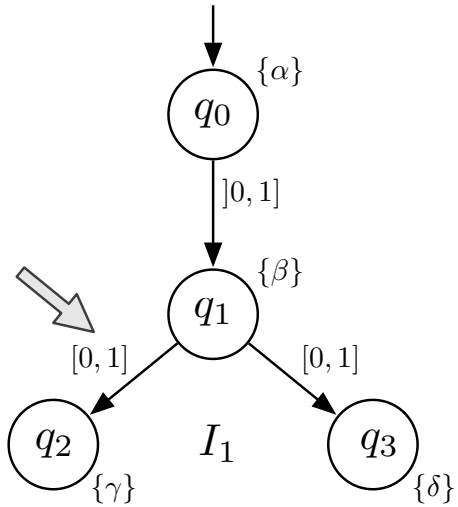


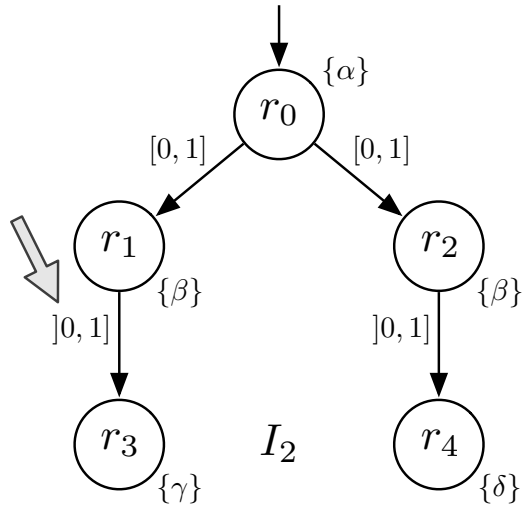


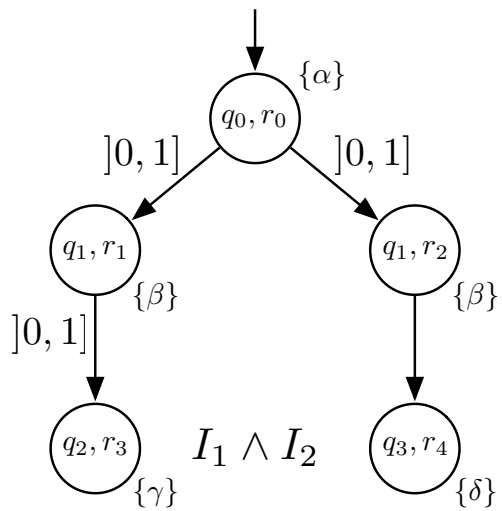


Interval Markov Chains

Closure problems: conjunction

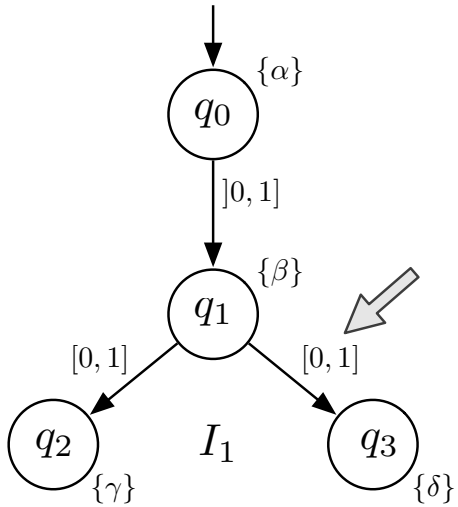


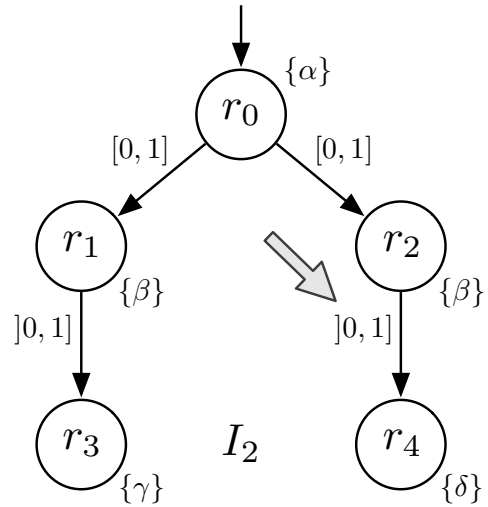


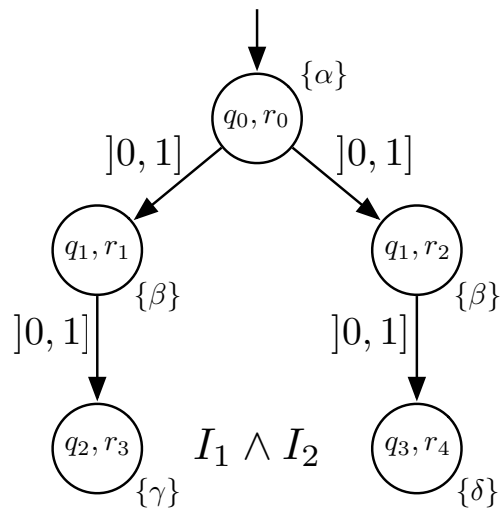


Interval Markov Chains

Closure problems: conjunction

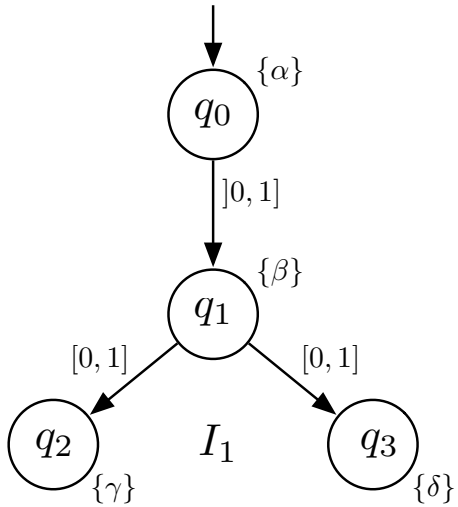


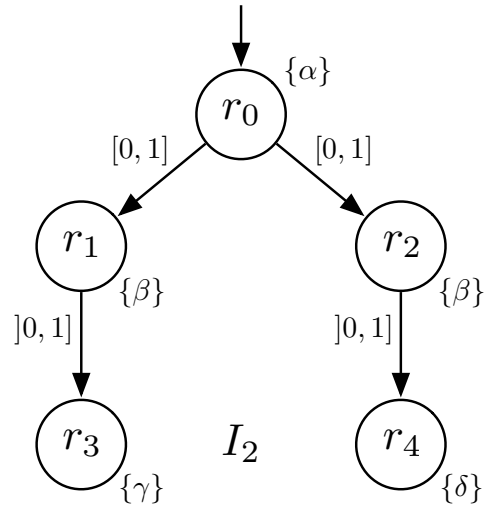


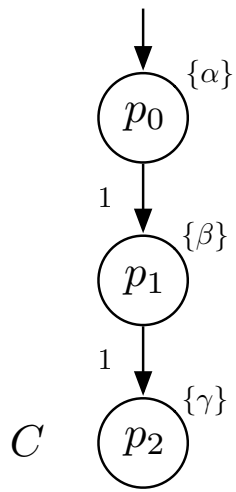


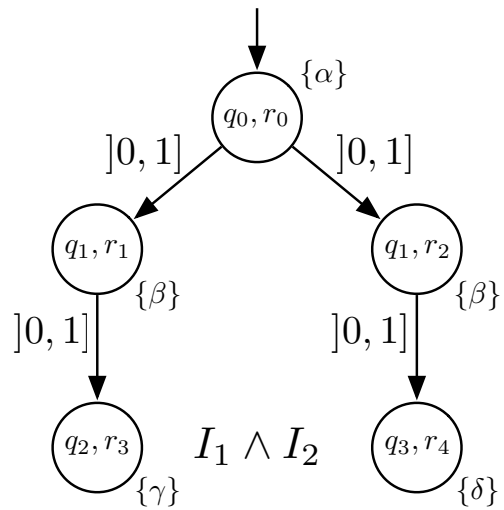
Interval Markov Chains

Closure problems: conjunction





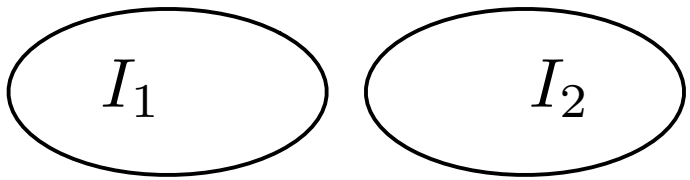




Interval Markov Chains

Common implementation

- Common implementation

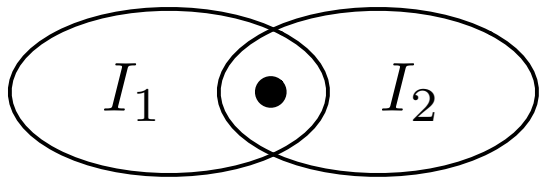


- Consistency
- Pruning

Interval Markov Chains

Common implementation

- Common implementation



- Consistency
- Pruning

Interval Markov Chains

Complexity results

Theorem

Checking for the existence of a common implementation is EXPTIME-complete

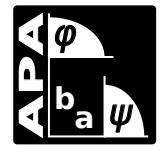
Theorem

Checking consistency is in PTIME

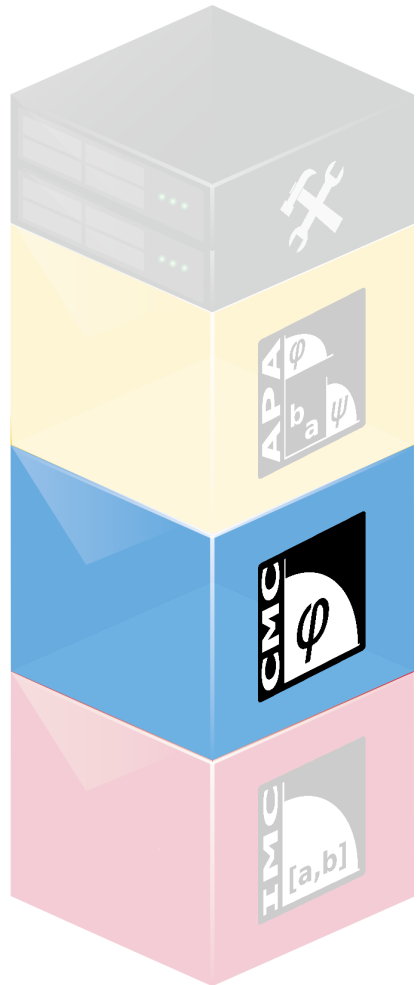
Theorem

Checking thorough refinement is EXPTIME-complete

Conclusion – IMC

			
Satisfaction	✓		
Refinement	✓		
Consistency	✓		
Pruning	✓		
Conjunction			
Parallel Composition			

Constraint Markov Chains

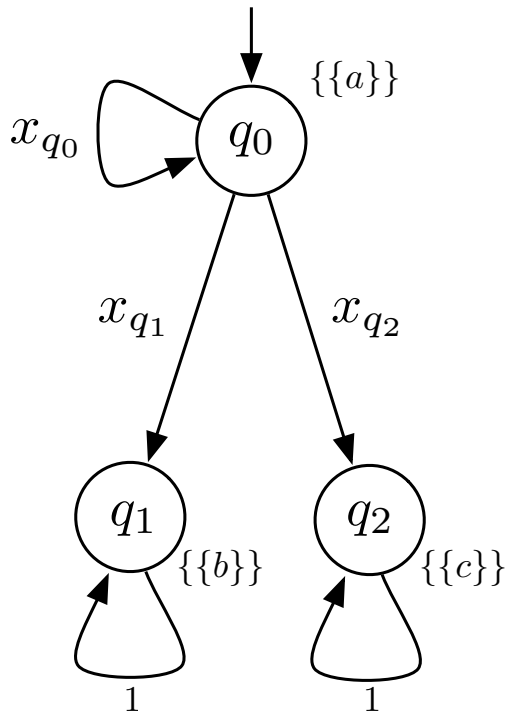


Paper B (**TCS**, QEST'10)

Paper C (**PEVA**, –"–)

Constraint Markov Chains

Motivation



$$\varphi(q_0)(x) \equiv 0.1 \leq x_{q_0} \leq 0.2 \vee 0.7 \leq x_{q_0} \leq 0.9$$

Definition

A *Constraint Markov Chain* consists of

- Q: set of states, $q_0 \in Q$
- A: atomic propositions
- $V : Q \to 2^{2^A}$: admissible state valuations
- $\varphi : Q \to (Dist(Q) \to \{0, 1\})$: constraint function

Constraint Markov Chains

Results

Theorem

Let S_1 and S_2 be CMCs. We have that

$$S_1 \preceq_S S_2 \Rightarrow S_1 \preceq S_2 \Rightarrow S_1 \preceq_T S_2,$$

and under determinism and single valuations in initial states,

$$S_1 \preceq_S S_2 \Leftrightarrow S_1 \preceq S_2 \Leftrightarrow S_1 \preceq_T S_2.$$

Constraint Markov Chains

Closure properties: results

- Conjunction

Theorem

$$[\![S_1 \wedge S_2]\!] = [\![S_1]\!] \cap [\![S_2]\!]$$

- Parallel composition

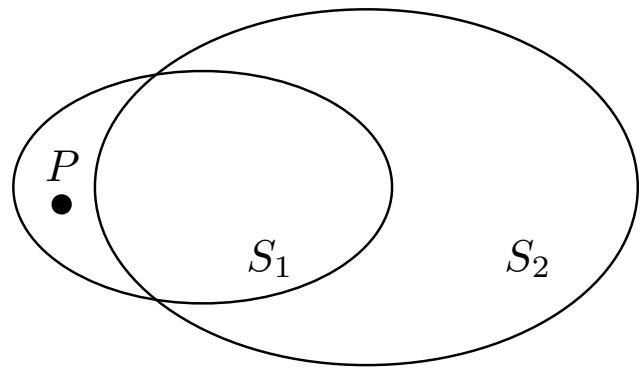
Theorem

If $S_1 \preceq S_3$ *and* $S_2 \preceq S_4$*, then* $S_1 \| S_2 \preceq S_3 \| S_4$

Constraint Markov Chains

Counter-example generation and weak refinement

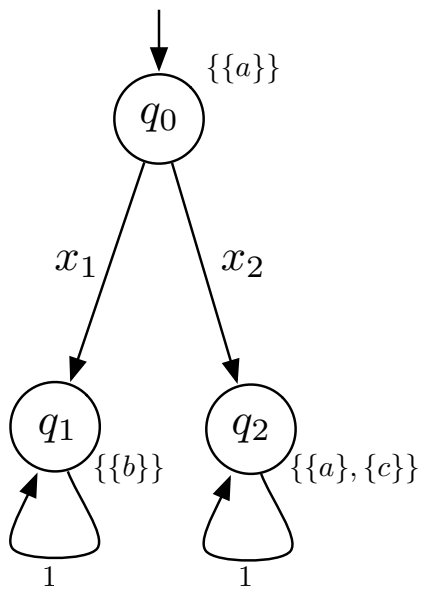
- $S_1 \not\leq S_2$ (deterministic and in single valuation normal form)
- Counter-example P: $P \in [\![S_1]\!]$ and $P \notin [\![S_2]\!]$
- Uses equivalences of refinement relations



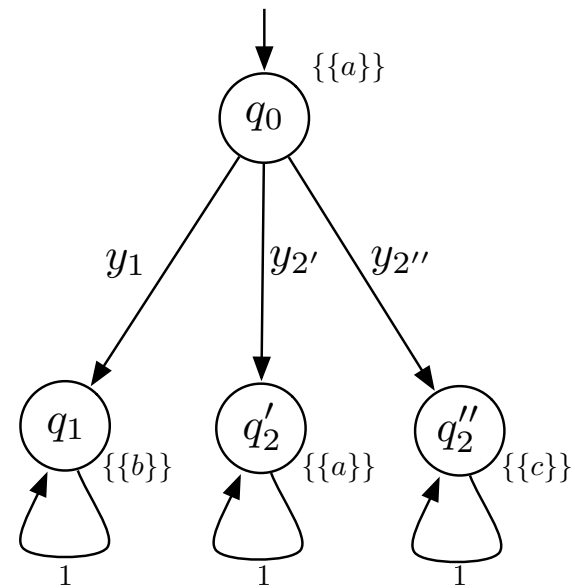
Constraint Markov Chains

Further results

- Normalization: $[\![S]\!] = [\![\mathcal{N}(S)]\!]$



$$\varphi(q_0)(x) \equiv x_1 \geq 0.7 \wedge x_2 \leq 0.4 \wedge x_1 + x_2 = 1$$

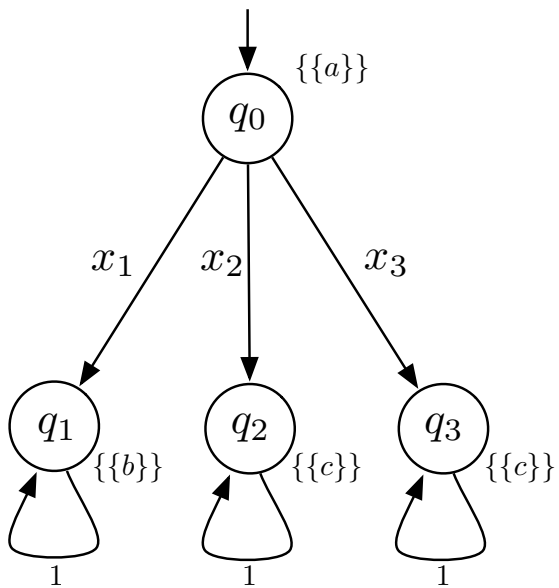


$$\varphi(q_0)(y) \equiv y_1 \geq 0.7 \wedge y_{2'} + y_{2''} \leq 0.4 \wedge y_1 + y_{2'} + y_{2''} = 1$$

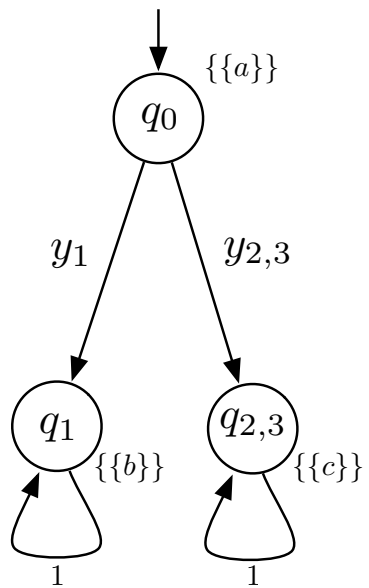
Constraint Markov Chains

Further results

- Determinization: $S \preceq \rho(S)$



$\varphi(q_0)(x) \equiv x_1 + x_2 + x_3 = 1$

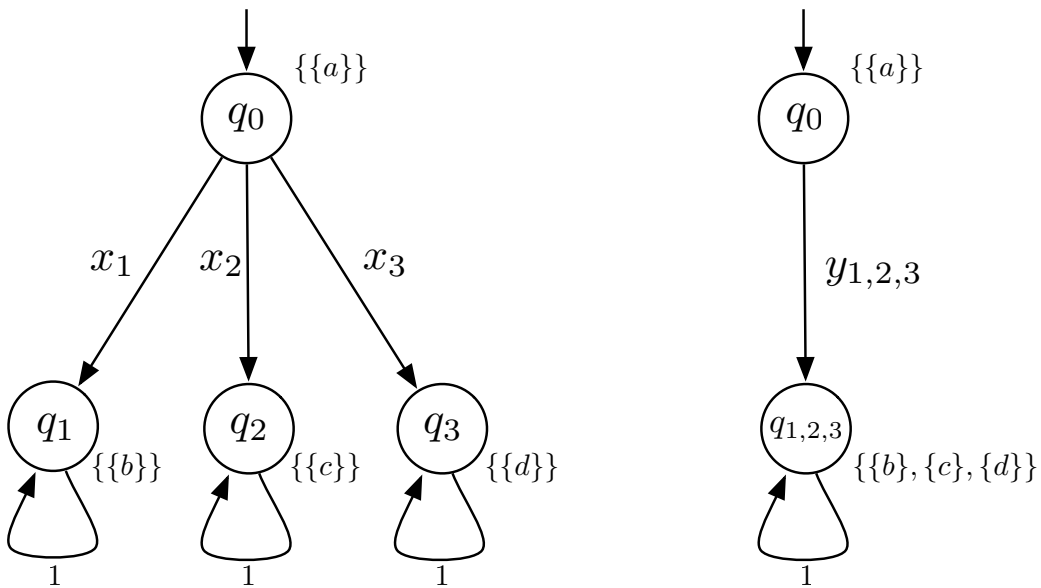


$\varphi(q_0)(y) \equiv x_1 + y_{2,3} = 1$

Constraint Markov Chains

Further results

- State abstraction: $S \preceq \alpha(S)$

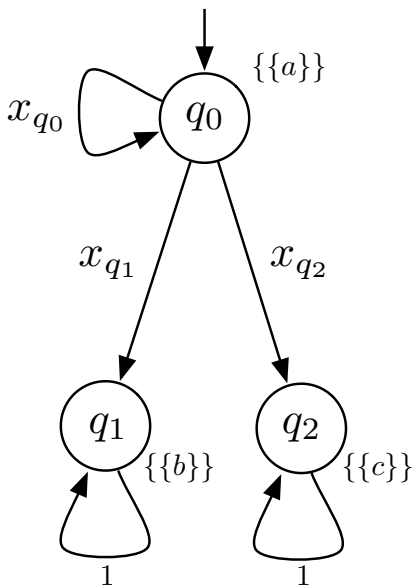


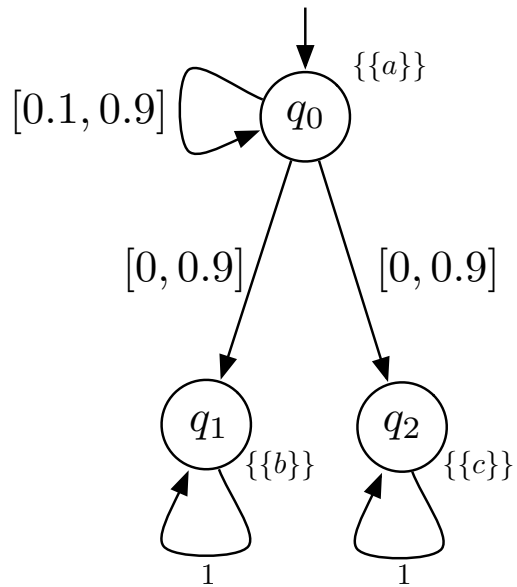
$\varphi(q_0)(x) \equiv x_1 + x_2 + x_3 = 1$ $\varphi(q_0)(y) \equiv y_{1,2,3} = 1$

Constraint Markov Chains

Further results

- Constraint abstraction: $S \leq \chi(S)$
- Minimality: $S \leq I$, then $\chi(S) \leq I$



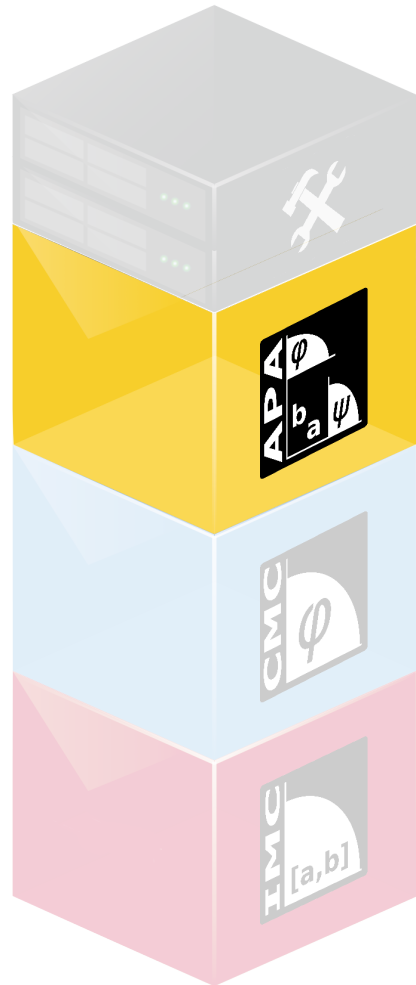


$$\varphi(q_0)(x) \equiv 0.1 \leq x_{q_0} \leq 0.2 \vee 0.7 \leq x_{q_0} \leq 0.9$$

Conclusion – CMC

	IMC [a,b]	CMC φ	APA φ b a ψ
Satisfaction	✓	✓	
Refinement	✓	✓	
Consistency	✓	✓	
Pruning	✓	✓	
Normalization		✓	
Constraint-abstraction		✓	
State-abstraction		✓	
Conjunction		✓	
Parallel Composition		✓	

Abstract Probabilistic Automata



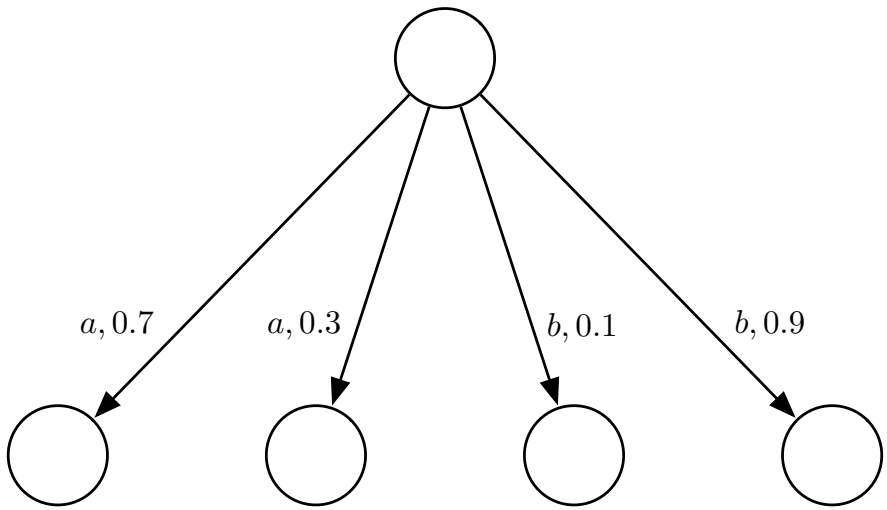
Paper D (**VMCAI'11**)

Paper E (**ACSD'11**)

Extending the notion of MCs

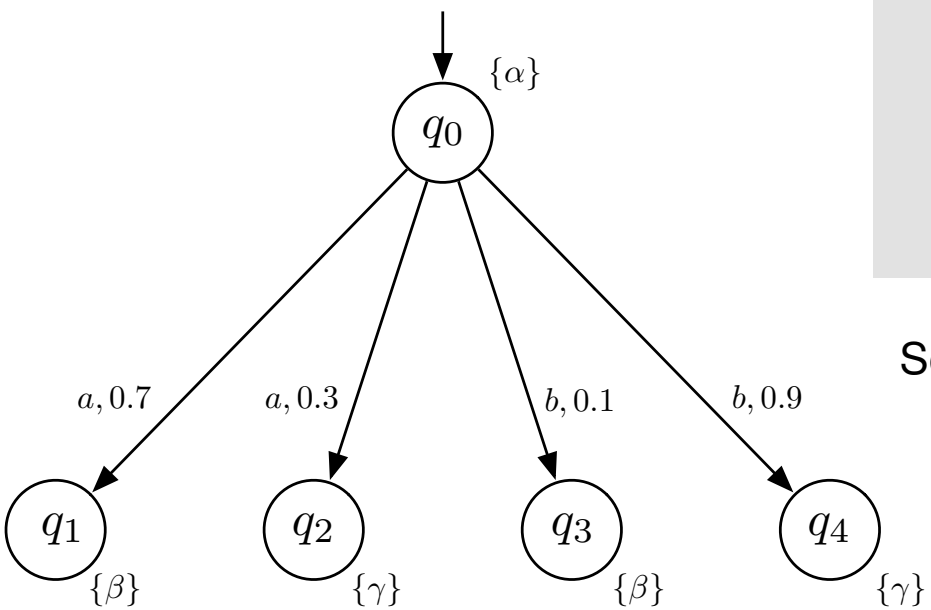
Probabilistic automata

- MCs: precisely one outgoing transitions from each state
- Non-determinism



Extending the notion of MCs

Motivation



Definition

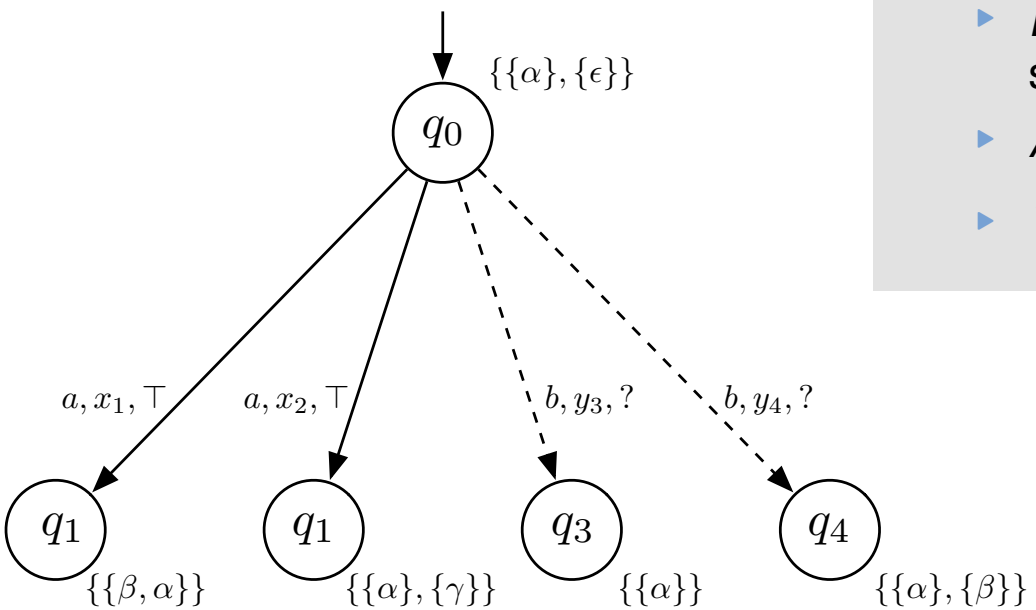
A *Probabilistic Automaton* consists of

- S: set of states, $s_0 \in S$
- A: actions
- L: $S \times A \times Dist(S) \to \{\bot, \top\}$: transition function
- AP: atomic propositions
- V: $S \to 2^{AP}$: state-labeling function

Segala & Lynch 1994

Abstract Probabilistic Automata

Motivation



$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Definition

An *Abstract Probabilistic Automaton* consists of

- S: set of states, $s_0 \in S$
- A: actions
- $L : S \times A \times C(S) \longrightarrow \{\bot, ?, \top\}$: state-constraint function
- AP: atomic propositions,
- V: $S \to 2^{2^{AP}}$: state-labeling function.

Abstract Probabilistic Automata

Weak refinement

Definition

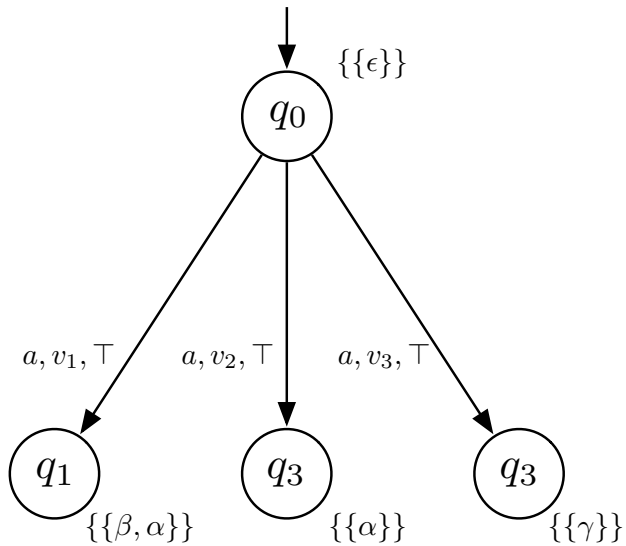
N, N' APAs
$\mathcal{R} \subseteq N \times N'$ is a weak refinement relation if and only if, for all $(s, s') \in \mathcal{R}$,

- $L'(s', a, \varphi') = \top \Rightarrow \exists \varphi \in C(S), L(s, a, \varphi) = \top$ and $\forall \mu \in Sat(\varphi) \exists \delta : S \to (S' \to [0,1]), \mu \lhd^{\delta}_{\mathcal{R}} \varphi'$,
- $L(s, a, \varphi) \neq \bot \Rightarrow \exists \varphi' \in C(S'), L'(s', a, \varphi') \neq \bot$ and $\forall \mu \in Sat(\varphi) \exists \delta : S \to (S' \to [0,1]), \mu \lhd^{\delta}_{\mathcal{R}} \varphi'$, and
- $V(s) \subseteq V'(s')$.

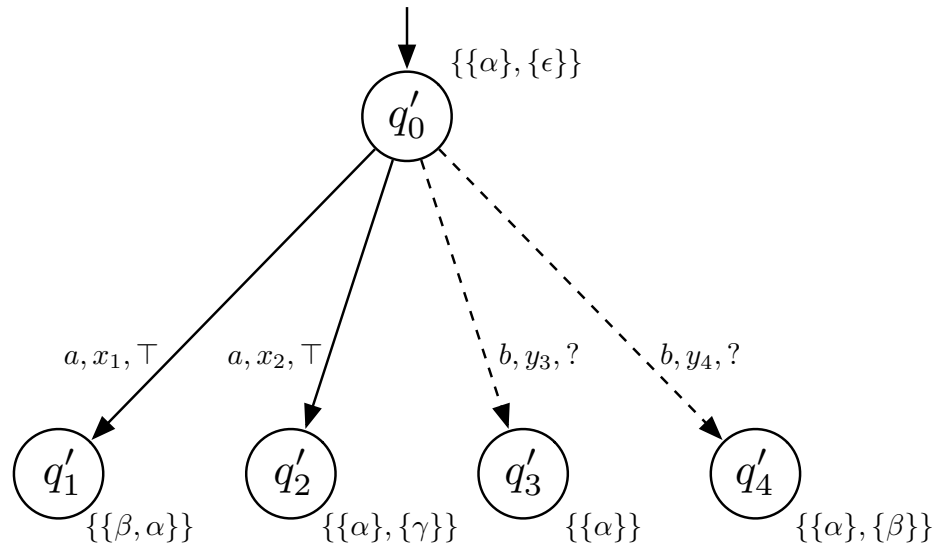
$N \preceq N' \iff \exists \mathcal{R},\ (s_0, s'_0) \in \mathcal{R}$.

Abstract Probabilistic Automata

Refinement: example



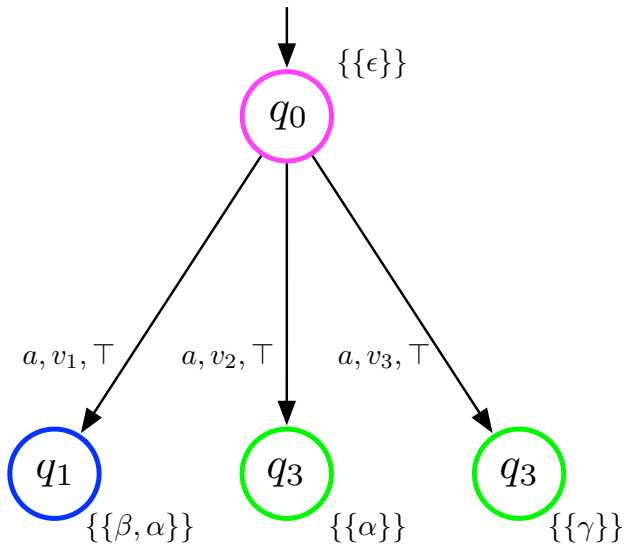
$$\varphi_v \equiv (v_1 = 1 \vee v_2 + v_3 = 1) \wedge v_1 + v_2 + v_3 = 1$$



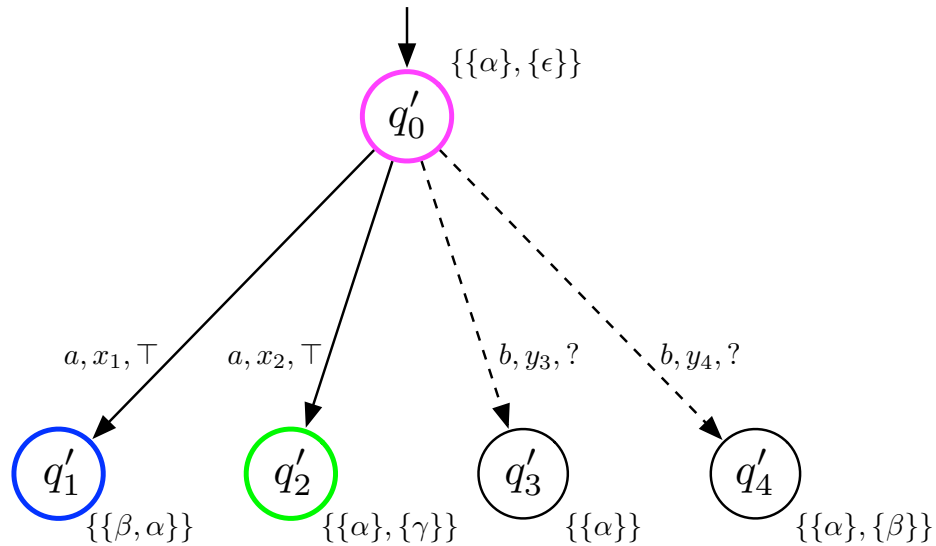
$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Abstract Probabilistic Automata

Refinement: example



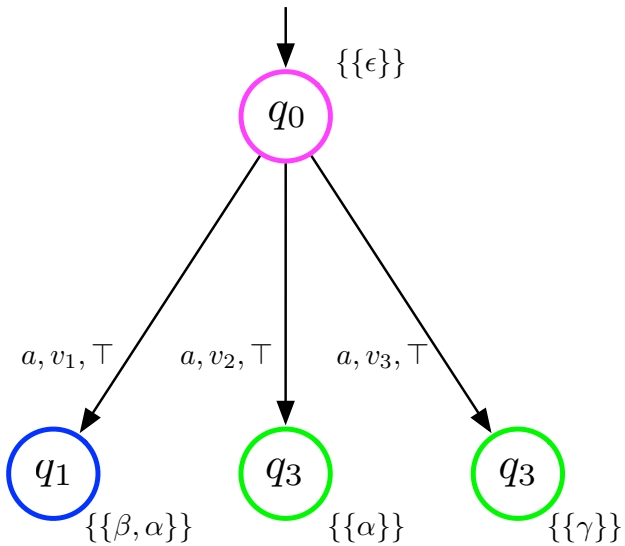
$$\varphi_v \equiv (v_1 = 1 \vee v_2 + v_3 = 1) \wedge v_1 + v_2 + v_3 = 1$$



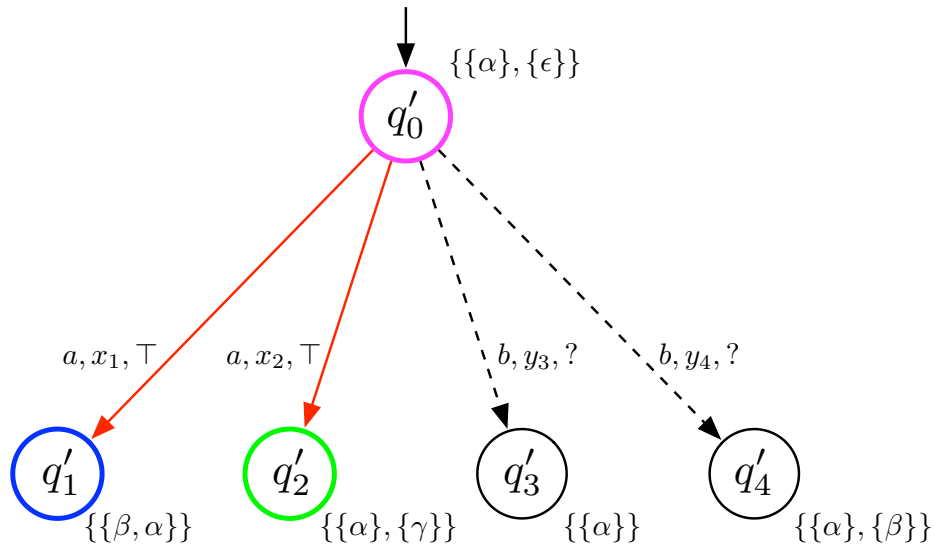
$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Abstract Probabilistic Automata

Refinement: example



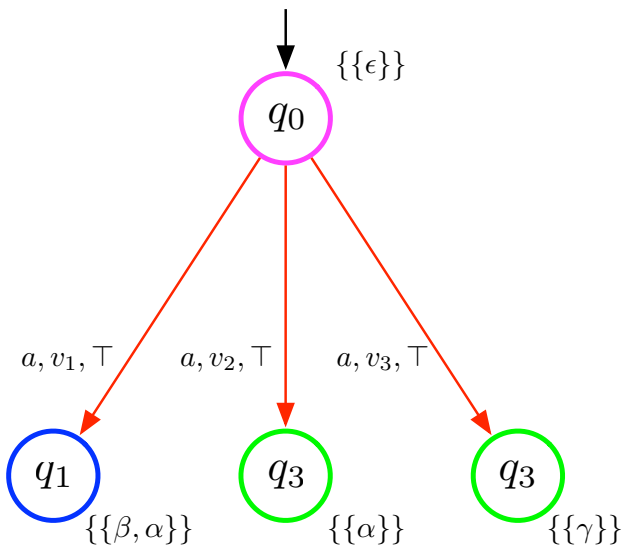
$$\varphi_v \equiv (v_1 = 1 \vee v_2 + v_3 = 1) \wedge v_1 + v_2 + v_3 = 1$$



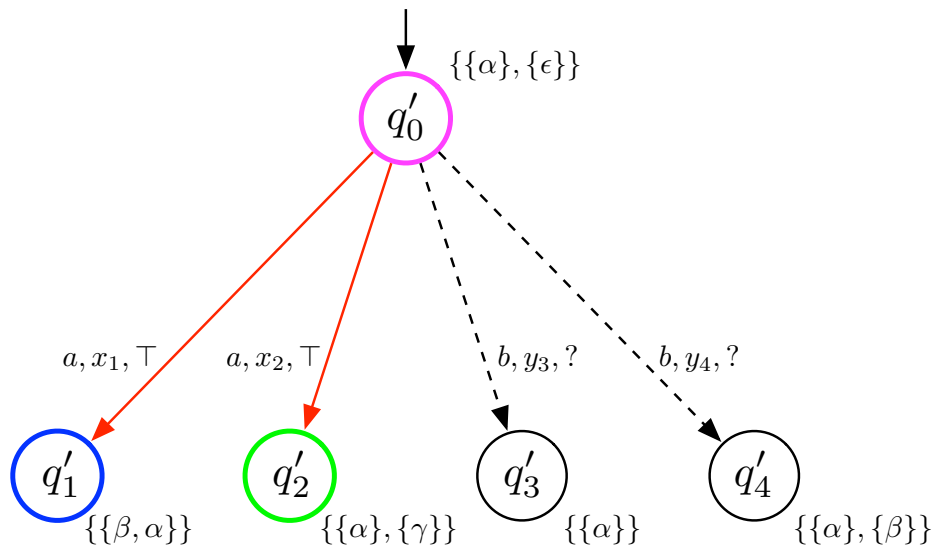
$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Abstract Probabilistic Automata

Refinement: example



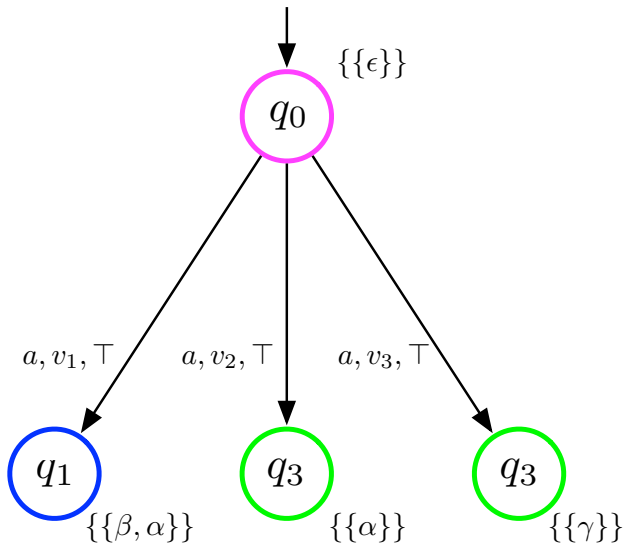
$$\varphi_v \equiv (v_1 = 1 \vee v_2 + v_3 = 1) \wedge v_1 + v_2 + v_3 = 1$$



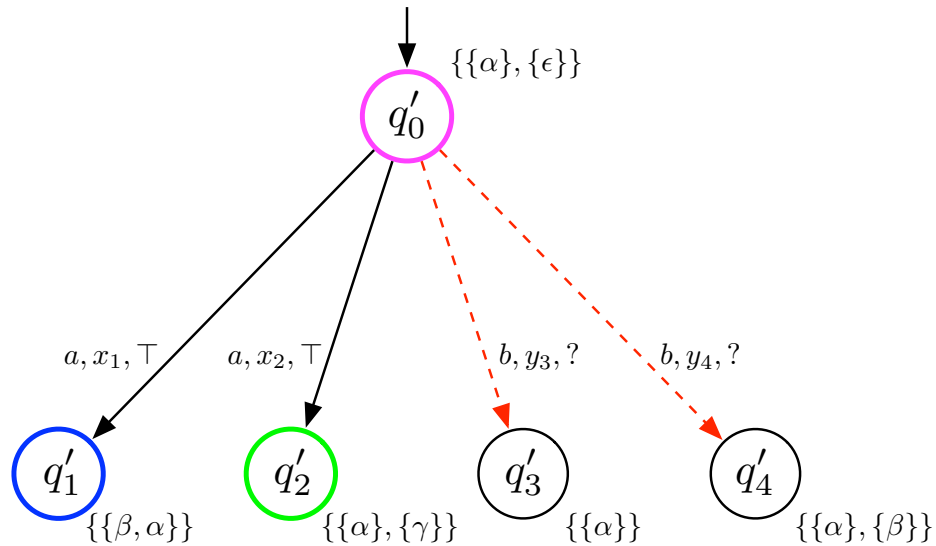
$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Abstract Probabilistic Automata

Refinement: example



$$\varphi_v \equiv (v_1 = 1 \vee v_2 + v_3 = 1) \wedge v_1 + v_2 + v_3 = 1$$



$$\varphi_x \equiv (x_1 = 1 \vee x_2 = 1) \wedge x_1 + x_2 = 1$$
$$\varphi_y \equiv y_3 \geq 0.1 \wedge y_3 + y_4 = 1$$

Abstract Probabilistic Automata

Refinement relations

- Weak refinement $\preceq$
- Strong refinement $\preceq_S$
- Thorough refinement $\preceq_T$
- Weak weak refinement $\preceq_W$

Abstract Probabilistic Automata

Refinement relations

- Weak refinement $\preceq$
- Strong refinement $\preceq_S$
- Thorough refinement $\preceq_T$
- Weak weak refinement $\preceq_W$
  - $L'(s', a, \varphi') = \top \Rightarrow \exists \varphi \in C(S), L(s, a, \varphi) = \top$ and $\forall \mu \in Sat(\varphi) \exists \delta : S \to (S' \to [0,1]), \mu \lhd^{\delta}_{\mathcal{R}} \varphi'$
  - $L(s, a, \varphi) \neq \bot \Rightarrow \forall \mu \in Sat(\varphi) \exists \varphi' \in C(S'), L'(s', a, \varphi') \neq \bot$ and $\exists \delta : S \to (S' \to [0,1]), \mu \lhd^{\delta}_{\mathcal{R}} \varphi'$

Abstract Probabilistic Automata

Results

Theorem

Let N_1 and N_2 be APAs. We have that

$$N_1 \preceq_S N_2 \Rightarrow N_1 \preceq N_2 \Rightarrow N_1 \preceq_W N_2 \Rightarrow N_1 \preceq_T N_2,$$

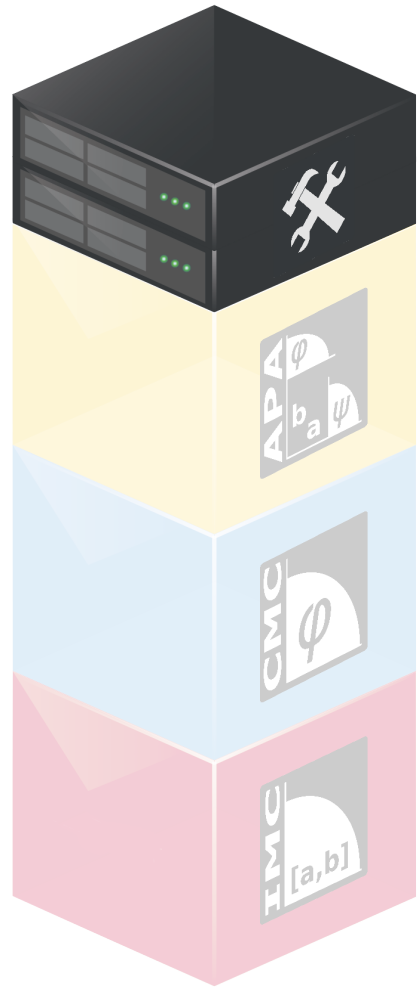
and under determinism and single valuations in initial states,

$$N_1 \preceq_S N_2 \Leftrightarrow N_1 \preceq N_2 \Leftrightarrow N_1 \preceq_W N_2 \Leftrightarrow N_1 \preceq_T N_2.$$

Conclusion – APA

	IMC [a,b]	CMC φ	APA φ b a ψ
Satisfaction	✓	✓	✓
Refinement	✓	✓	✓
Consistency	✓	✓	✓
Pruning	✓	✓	✓
Normalization		✓	✓
Constraint-abstraction		✓	✓
State-abstraction		✓	✓
Conjunction		✓	✓
Parallel Composition		✓	✓

APAC



Paper C

Paper E

Paper F (**QEST'11**)

APAC

Features

- Language for specifying CMCs and APAs
- Written in C# and uses the SMT solver Z3

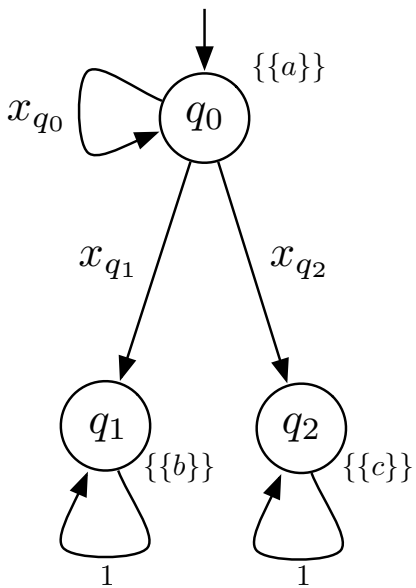
APAC

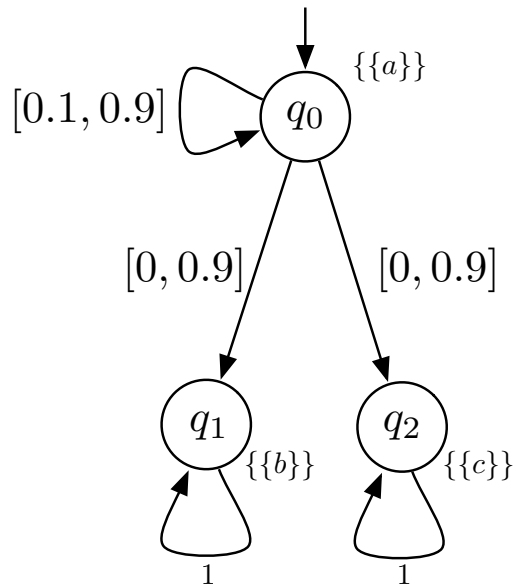
Z3

- Simplifies expressions by e.g. quantifier elimination
- Example: $\exists x : ax^2 + bx + c = 0 \rightsquigarrow b^2 - 4ac \geq 0$
  - Model instantiation
- Solving equations
- Z3 supports linear arithmetic for reals

APAC

- Constraint abstraction





$$\varphi(q_0)(x) \equiv 0.1 \leq x_{q_0} \leq 0.2 \vee 0.7 \leq x_{q_0} \leq 0.9$$

APAC

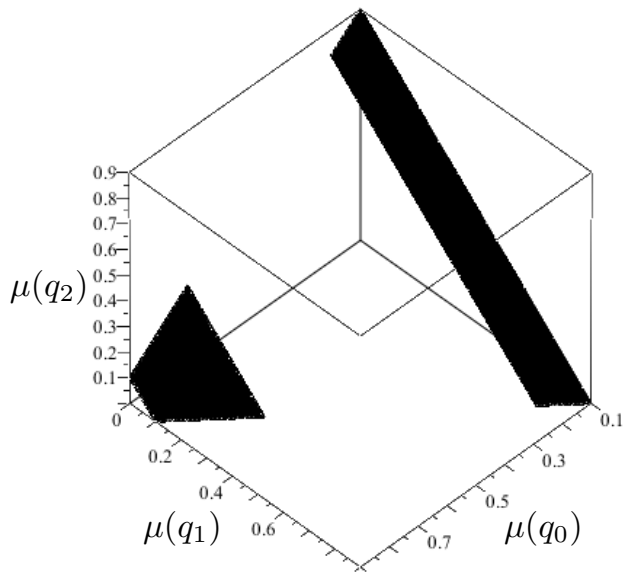
Constraint abstraction

Let $u \in Q_1$.

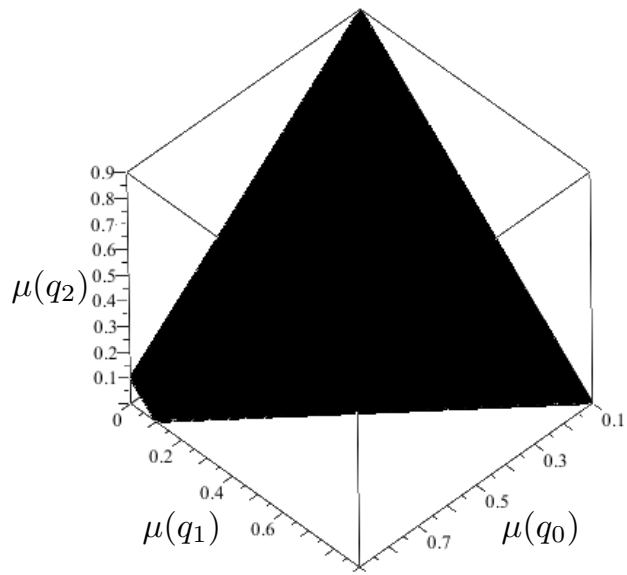
$$
\begin{aligned}
&\forall v \in Q_1 : (0 \leq L_v \leq U_v \leq 1) \wedge \\
&(\forall x \in \mathit{Dist}(Q_1) : \varphi(u)(x) \Rightarrow \forall v \in Q_1 : L_v \leq x(v) \leq U_v) \wedge \\
&\forall (L'_1, U'_1, \ldots, L'_k, U'_k) \in [0,1]^{|Q_1|} : \\
&\quad \Big[((\forall x \in \mathit{Dist}(Q_1) : \varphi(u)(x) \Rightarrow \forall v \in Q_1 : L'_v \leq x(v) \leq U'_v) \wedge \\
&\quad (\forall v \in Q_1 : 0 \leq L'_v \leq U'_v \leq 1)) \Rightarrow (\forall v \in Q_1 : L'_v \leq L_v \wedge U_v \leq U'_v) \Big].
\end{aligned}
$$

APAC

Constraint abstraction



$$\varphi(q_0)(x) \equiv 0.1 \leq x_{q_0} \leq 0.2 \vee 0.7 \leq x_{q_0} \leq 0.9$$



$$\varphi(q_0)(q_0) = [0.1, 0.9]$$
$$\varphi(q_0)(q_1) = [0, 0.9]$$
$$\varphi(q_0)(q_2) = [0, 0.9]$$

Conclusion

	IMC [a,b]	CMC φ	APA φ b a ψ
Satisfaction	✓	✓	✓
Refinement	✓	✓	✓
Consistency	✓	✓	✓
Pruning	✓	✓	✓
Normalization		✓	✓
Constraint-abstraction		✓	✓
State-abstraction		✓	✓
Conjunction		✓	✓
Parallel Composition		✓	✓

Conclusion

	IMC [a,b]	CMC φ	APA φ b a ψ
Satisfaction	✓	🛠	🛠
Refinement	✓	🛠	🛠
Consistency	✓	🛠	🛠
Pruning	✓	🛠	🛠
Normalization		🛠	🛠
Constraint-abstraction		🛠	🛠
State-abstraction		🛠	🛠
Conjunction		🛠	🛠
Parallel Composition		✓	✓

Conclusion

Recent, current and future work

- New notions of satisfaction and refinement:
  - Stuttering satisfaction
- A modal logic for APAs
- A generalisation of counter-example generation
- Improvements to APAC (+GUI, +internet-based execution)

Conclusion

Recent, current and future work

- New notions of satisfaction and refinement:
  - Stuttering satisfaction
- A modal logic for APAs
- A generalisation of counter-example generation
- Improvements to APAC (+GUI, +internet-based execution)