Modeling and Verification of Systems with Uncertainties

Benoît Delahaye

Habilitation defense

LABORATOIRE DES SCIENCES DU NUMÉRIQUE DE NANTES

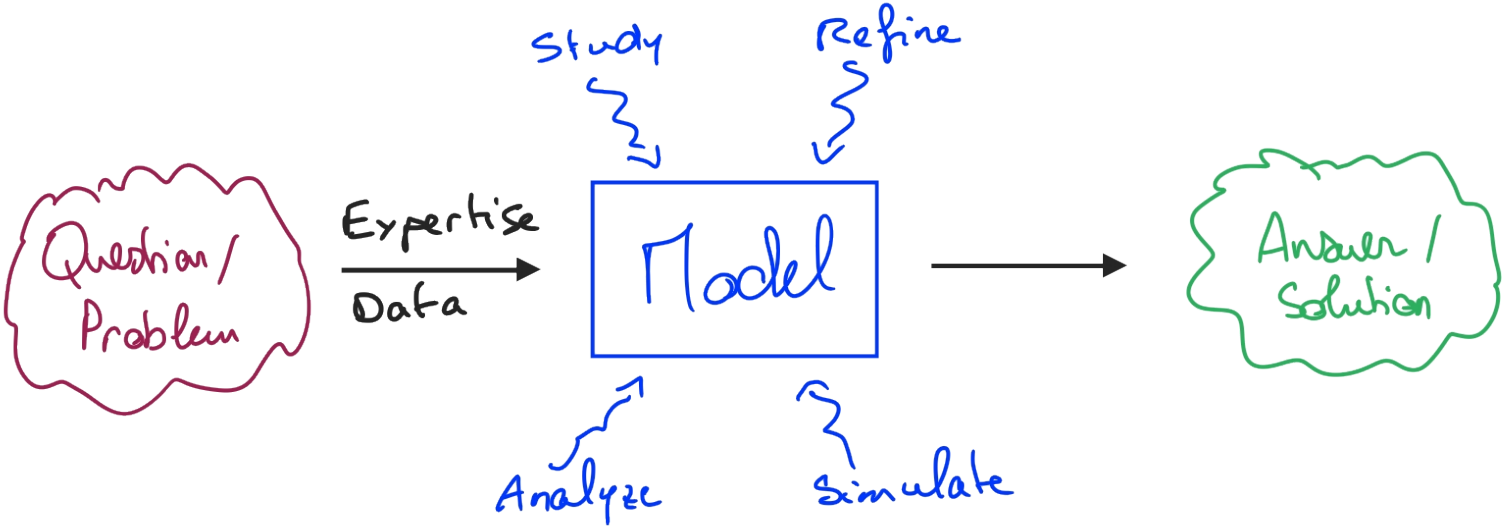
UNIVERSITÉ DE NANTES

Science

Science[1]

Science is a systematic enterprise that builds and organizes knowledge in the form of testable explanations and predictions about the universe. [...]

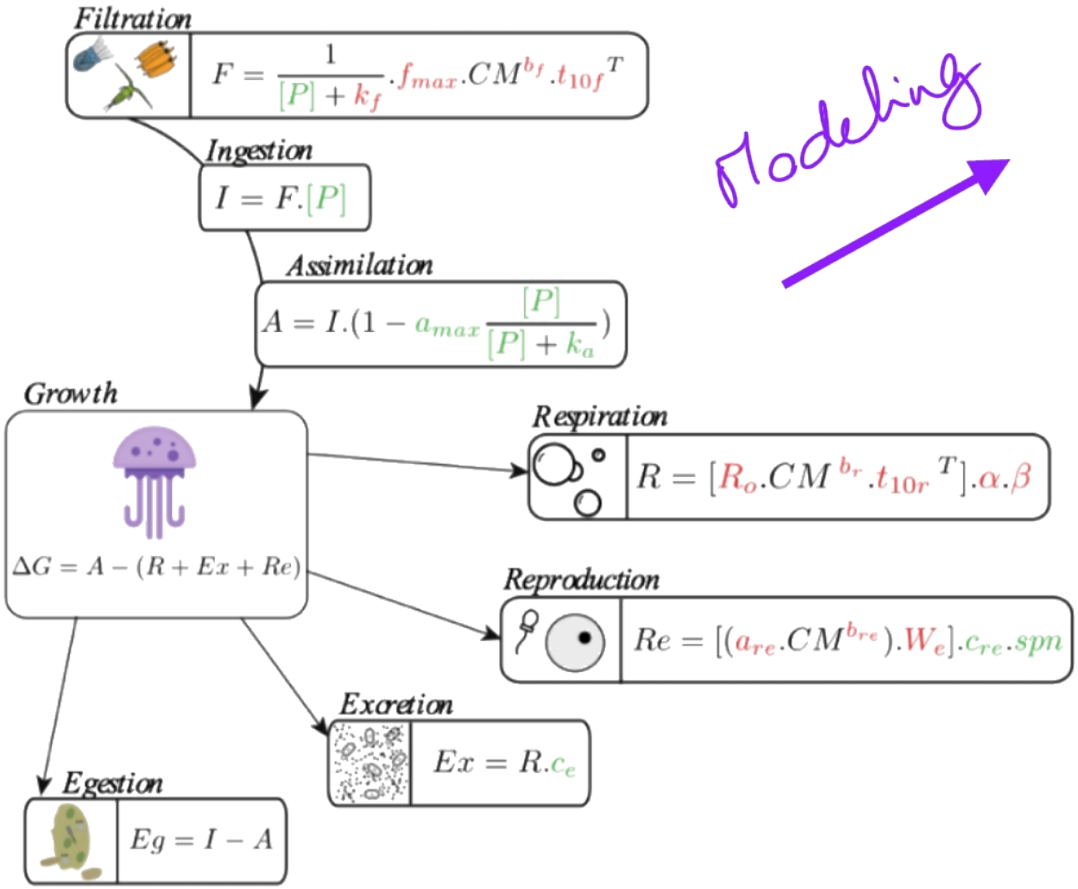
- Natural sciences
  Study nature
- Social sciences
  Study society
- Formal sciences
  Study abstract concepts



[1] https://en.wikipedia.org/wiki/Science

Example: Prediction of the growth of a jellyfish

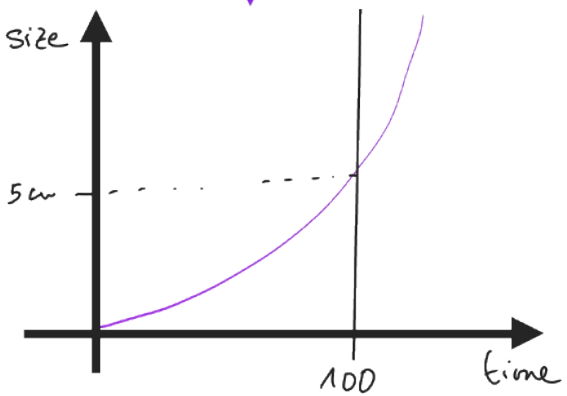
What is the size after 100 days?



Modeling

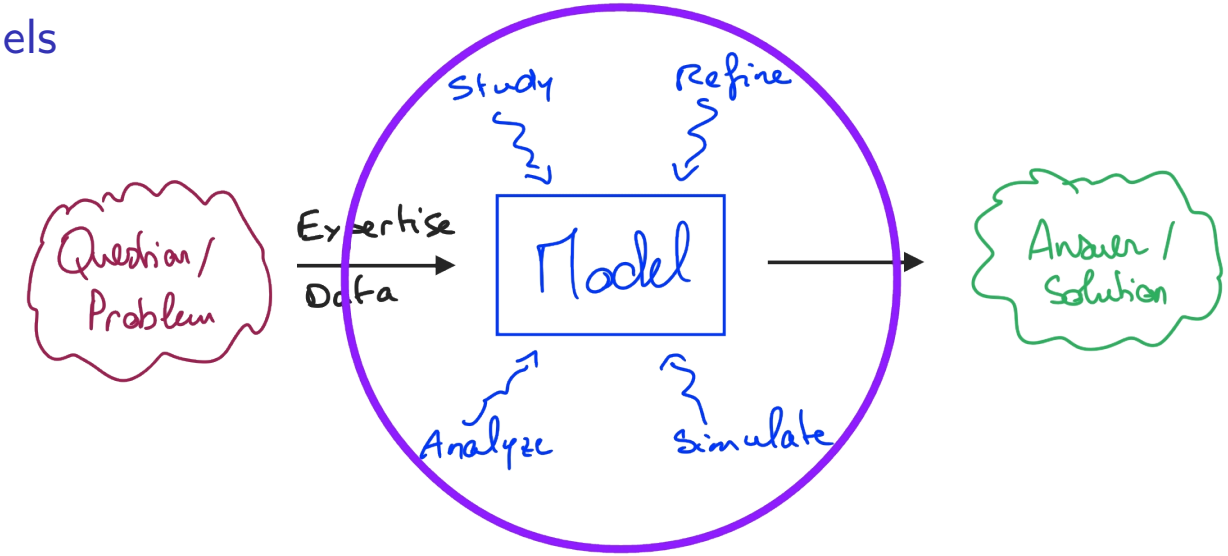
```
# Masse initiale en poids humide (en g)
            CW=0.0036*WM;
            cw=CW
            for t in Temps_D:
                if score_degrowth<lim_degrowth:
                # Application du modèle
                    X0=[CW, T, 0]
                    [Clear, Ingest, Assim, Respi, Excre, Prod, Egest, Scope_growth] = pelagia_feed(X0,param1,param2,0);
                    # Stockage des variables
                    CW=CW+ Scope_growth*dt;
                    WM=CW*(100/0.36);
                    pelsize=(wm/a)**(1/b);
                    if not pelsize>0:
                        print(pelsize)
                    cw=np.append(cw,CW) ;
                    WM=np.append(WM,wm);
                    Size1=np.append(Size1,pelsize);

                    val=vec_D[int(i*10)]
                    val=val[0]

                    if (val):
                        d=vec_D[int(i*10)]
                        d=d[1]
                        d=d[0]
                        if not (degrowth[d,1]-degrowth[d,2]<Size1[x_d[d]]<degrowth[d,1]+degrowth[d,2]):
                            score_degrowth=score_degrowth+1

                else:
                    bool_break1=True
                    break
U(DOS)---  modele2017-02-20-1.py   43% L241   (Python)
```

Verification



Theoretical Computer Science

Science of Models



Motivation

Providing **theories**, **tools** and **techniques** for building and analyzing models

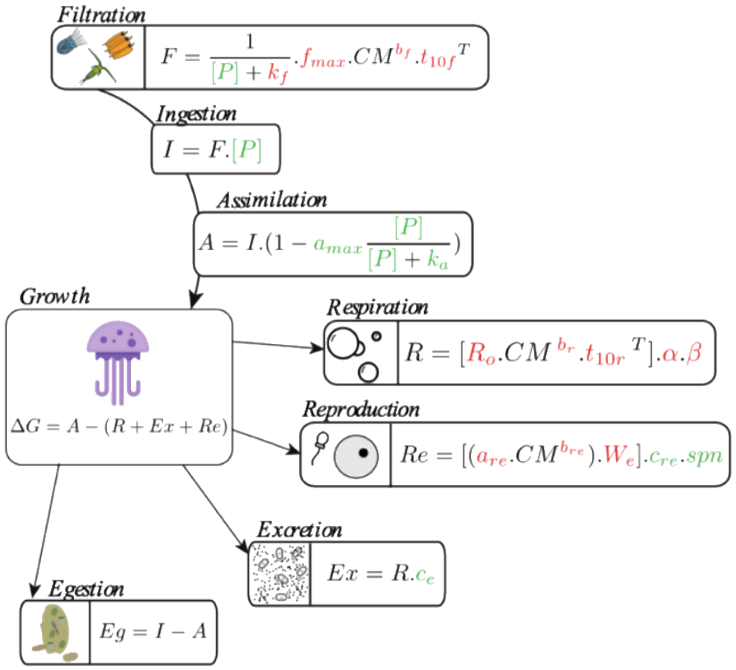
- **Modeling:** How to build models

  Language (formalism) – Expressivity – Abstraction – Manipulation

- **Verification:** How to analyze models

  (Semi-)Automation – Complexity/Efficiency – Diagnosis

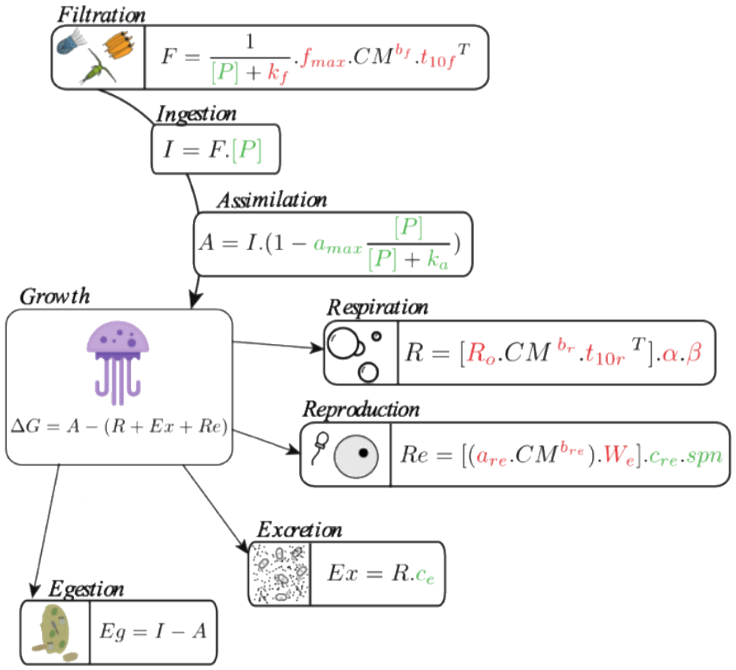
Inconvenience of deterministic modeling



Choosing the value of unknown variables

- Guess?
- Experimental data?
- Mean value?

Inconvenience of deterministic modeling



Choosing the value of unknown variables

- Guess?
- Experimental data?
- Mean value?

The **deterministic** model is **not** representative of a familly of variable systems

Modeling with uncertainties

Deterministic modeling requires perfect knowledge of the systems

Uncertainties can come from several sources

- Incomplete knowledge of the system (ex: unknown variables)
- Incomplete knowledge of the environment (ex: temperature, food)
- Abstraction: the model is too complex to be studied in its entirety
- ...

Modeling with uncertainties

Deterministic modeling requires perfect knowledge of the systems

Uncertainties can come from several sources

- Incomplete knowledge of the system (ex: unknown variables)
- Incomplete knowledge of the environment (ex: temperature, food)
- Abstraction: the model is too complex to be studied in its entirety
- ...

How are uncertainties accounted for?

- Non-determinism
- Probabilities
- Parameters

Modeling with uncertainties

Deterministic modeling requires perfect knowledge of the systems

Uncertainties can come from several sources

- Incomplete knowledge of the system (ex: unknown variables)
- Incomplete knowledge of the environment (ex: temperature, food)
- Abstraction: the model is too complex to be studied in its entirety
- ...

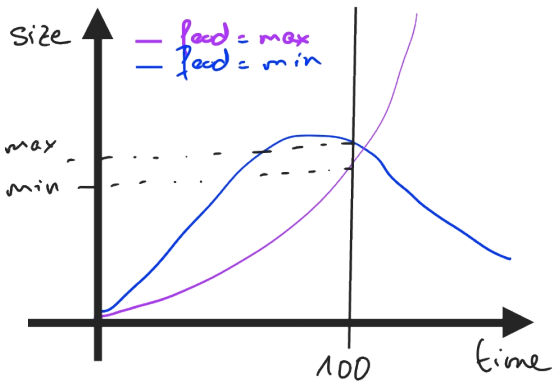
How are uncertainties accounted for?

- Non-determinism
- Probabilities
- Parameters
- **Left out**

Example: Jellyfish size ctd.

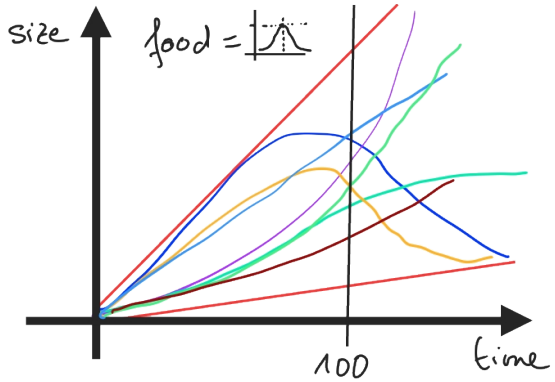
- **Non-determinism:** No further information

  ⇒ Study all potential outputs



- **Probabilities:** Imperfect information

  ⇒ Study the distribution of outputs



- **Parameters:** Building more information

  ⇒ The outcome is a function of p

size = f(food)
⇒ Optimization

Verification with uncertainties

Simulation is not sufficient when there are uncertainties.

- Statistical analysis
  - $\Rightarrow$ Using extensive simulations
- Automated techniques
  - Formal model analysis
  - Mix with statistical analysis
  - ...

Bottleneck of formal verification

- Complexity
- Undecidability in some cases
- Size of the models

Challenges

Modeling

- Capturing details / characteristics
  - Enhance expressivity
  - Develop new modeling formalisms
- Dealing with the size
  - Develop abstraction techniques
  - Study composition techniques

Verification

- Solving problems / Answering questions
  - Characterize decidability
  - Develop verification techniques
- Dealing with the size
  - Combine abstraction and verification
  - Enhance efficiency

Challenges

Modeling

- Capturing details / characteristics
  - Enhance expressivity
  - Develop new modeling formalisms
- Dealing with the size
  - Develop abstraction techniques
  - Study composition techniques

Verification

- Solving problems / Answering questions
  - Characterize decidability
  - Develop verification techniques
- Dealing with the size
  - Combine abstraction and verification
  - Enhance efficiency

Most of all: **Ease of use** by non-specialists

Contributions to the field 1/2

In the past 10 years, my contributions can be classified in 2 domains:

- Theoretical work on modeling formalisms and their verification
- Practical application of modeling and verification techniques to concrete case studies

Main "Theoretical" contributions

- Abstract probabilistic automata – [VMCAI'11], [Inf.Comp.'13], [LMCS'14] ...
- Timed algebra of services – [FORTE'13], [ICTAC'14], [MSCS'18]
- **Probabilistic Event-B** – [SAC'17], [SoSyM'19]
- Probabilistic time Petri nets – [PETRI NETS'16]
- **Parametric interval Markov chains** + timed extension
  [SynCoP'15], [VMCAI'16], [TIME'16], [QEST'17], [TCS'18], [JLAMP'20]

Manuscript **Manuscript + presentation**

Contributions to the field 2/2

Main "Practical" contributions

- Statistical model checking

  Several applications + Prototype tool – [ISoLA'12], [STTT'12], [STTT'15]

- **Parametric statistical model checking**

  Prototype tool + Application to UAV – [FORTE'19]

- Parameter synthesis using statistical model checking

  Application to oceanography – [MSystems'17], [Sci. Rep.'20]

- **Graphical event models learning and verification**

  Application to security assessment – [IEA/AIE'19]

Manuscript **Manuscript + presentation**

Parametric Interval Markov Chains

using constraints

Anicet Bart, Paulin Fournier, Didier Lime, Eric Monfroy, Laure Petrucci, Charlotte Truchet

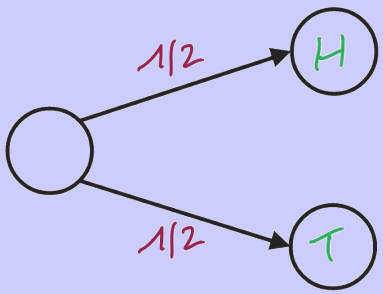
Outline

Context

Modeling and Analyzing famillies of Markov chains

System = **Markov chain**

- Probabilistic transition system
- Many existing verification techniques

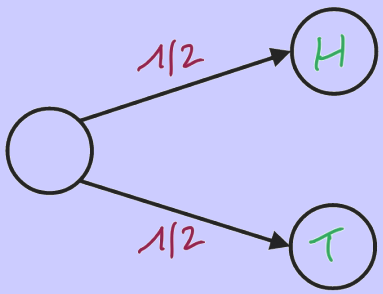


Context

Modeling and Analyzing famillies of Markov chains

System = **Markov chain**

- Probabilistic transition system
- Many existing verification techniques
- **What about variability?**

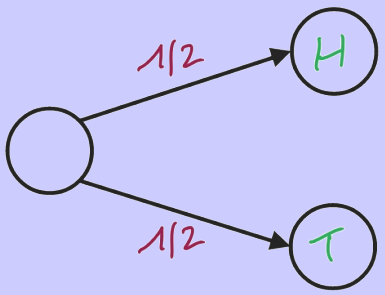


Context

Modeling and Analyzing famillies of Markov chains

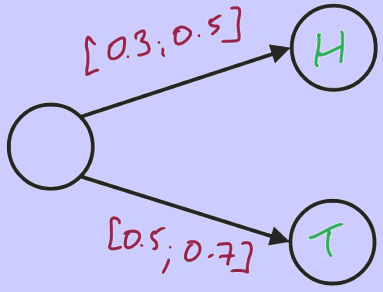
System = **Markov chain**

- Probabilistic transition system
- Many existing verification techniques
- **What about variability?**



Abstraction: **interval Markov chain** (Jonsson, Larsen - LICS'91)

- Represent famillies of Markov chains
- Several semantics
- Reason on **all** implementations at once



Motivation

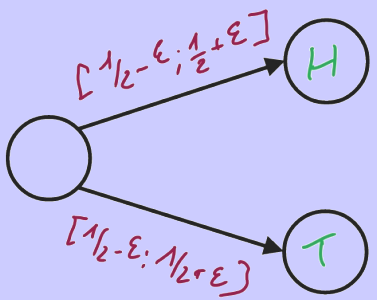
Controlling variability

Robustness / Dimensioning

- Choosing the size of the intervals in order to preserve properties

Example: Coin manufacture

How precise must the balance of a coin be to ensure that the probability of having 5 heads in 10 tosses is above 0.49?
Answer: $\varepsilon \leq 0.01$



Motivation

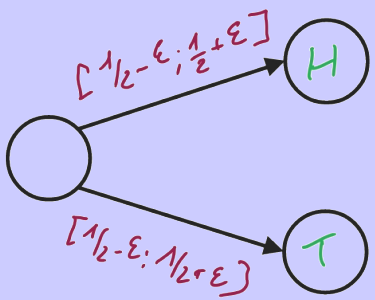
Controlling variability

Robustness / Dimensioning

- Choosing the size of the intervals in order to preserve properties

Example: Coin manufacture

How precise must the balance of a coin be to ensure that the probability of having 5 heads in 10 tosses is above 0.49?
Answer: $\varepsilon \leq 0.01$



$\Rightarrow$ All the coins that respect the above specification with $\varepsilon \leq 0.01$ will satisfy the property

Challenges

Compatibility

Preserving the characteristics of IMCs

- Considering all semantics

Decidability

Being able to solve the original problem

- Compute the set of parameter values ensuring a given property

Usefullness

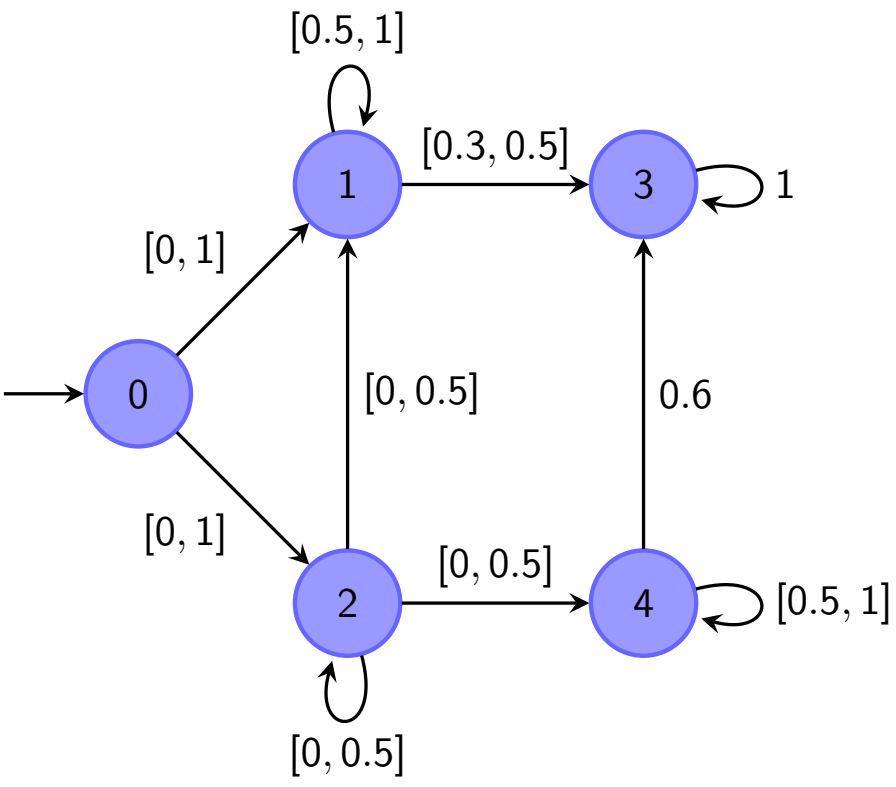
Being able to deal with "large enough" models

- Number of states/transitions
- Number of parameters

Outline

Interval Markov Chains (IMCs)

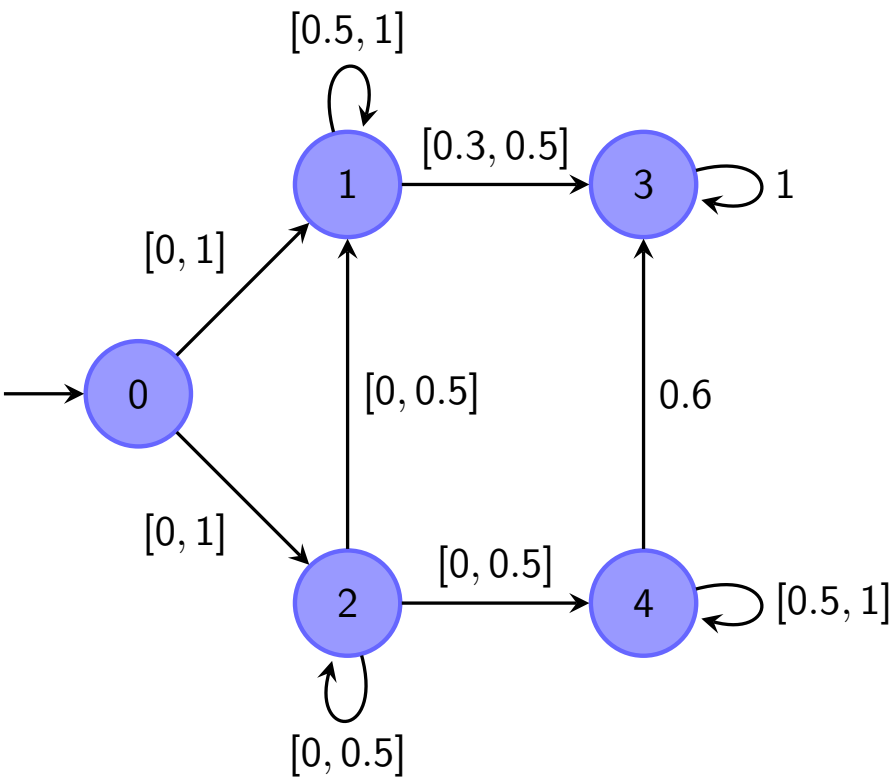
Semantics



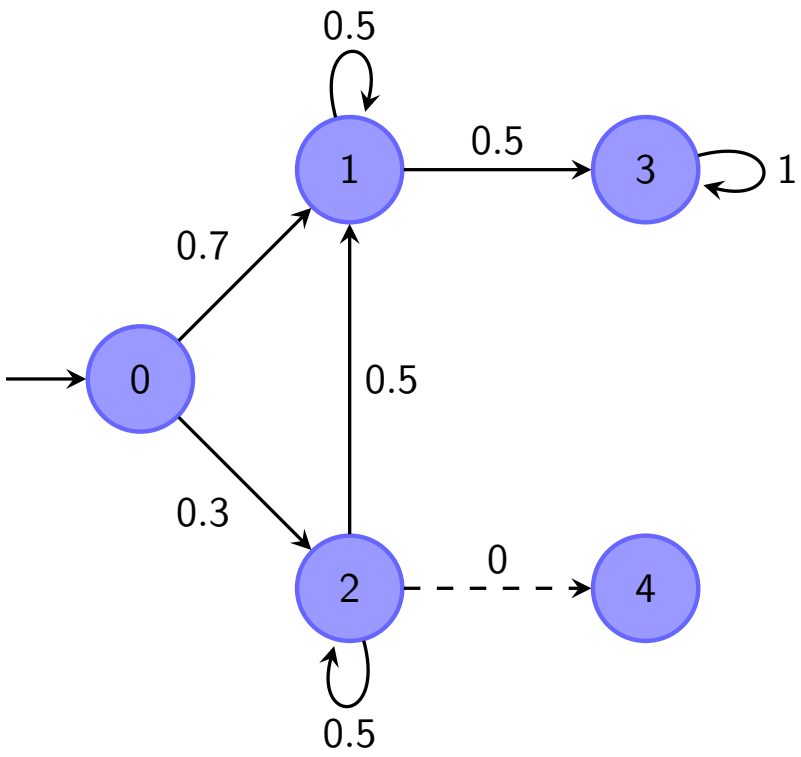
Specification (IMC)

Interval Markov Chains (IMCs)

Semantics



Specification (IMC)

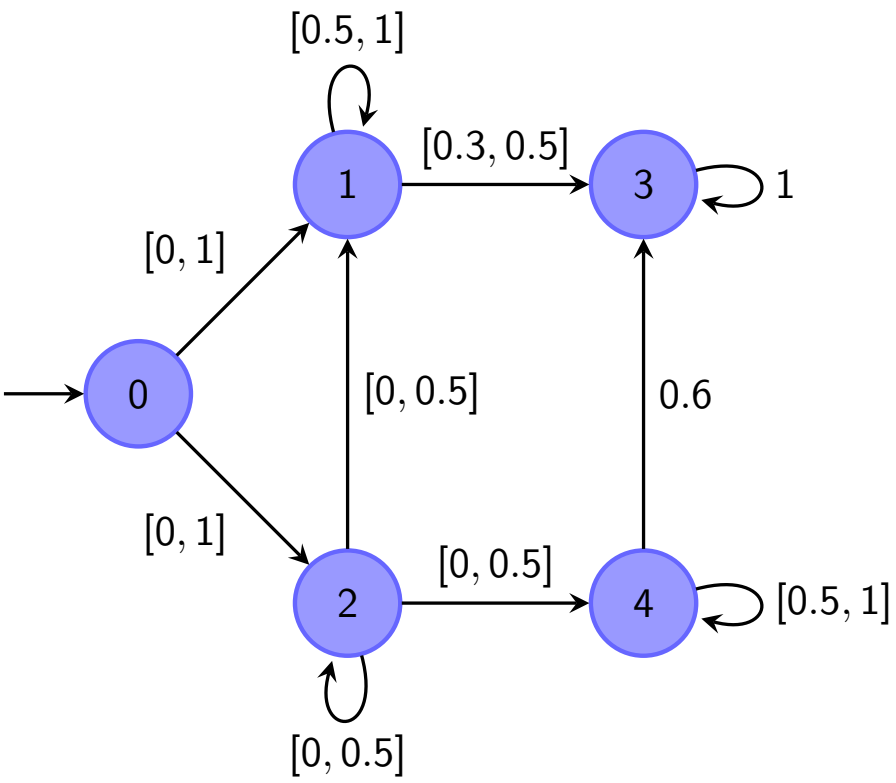


Implementation (MC)

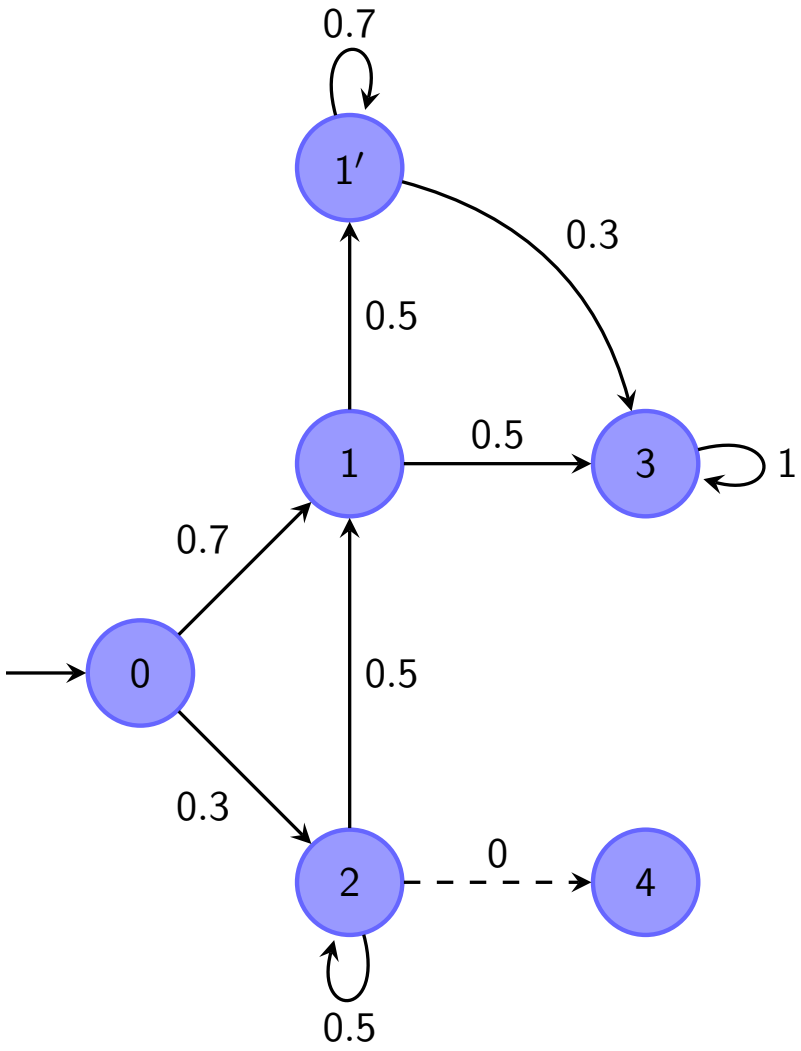
"Once and for all" semantics $\models_{\mathtt{I}}^{o}$

Interval Markov Chains (IMCs)

Semantics



Specification (IMC)

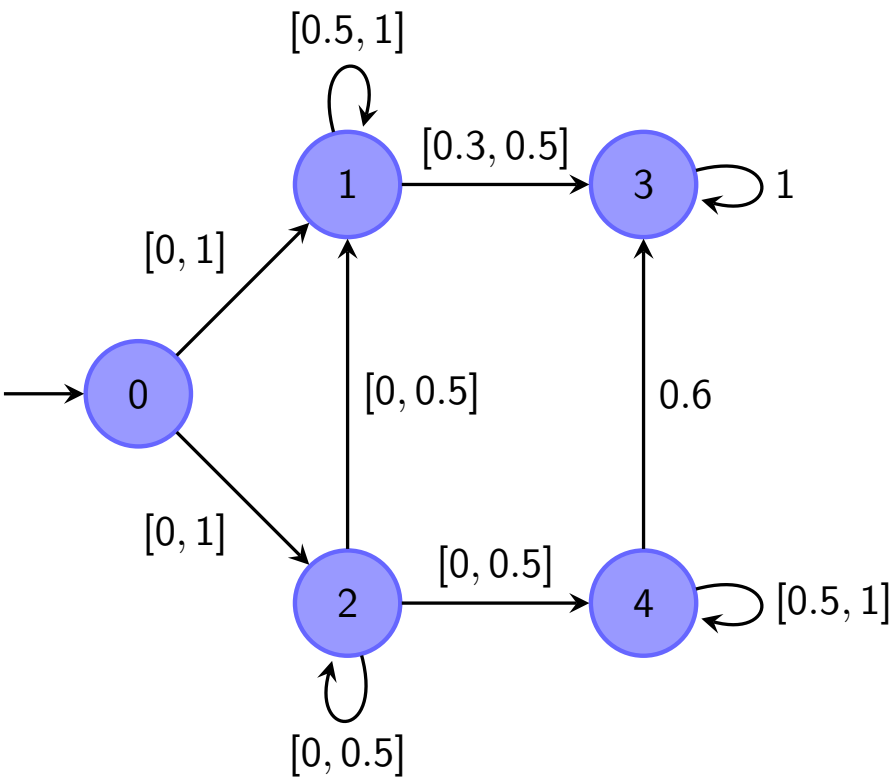


Implementation (MC)

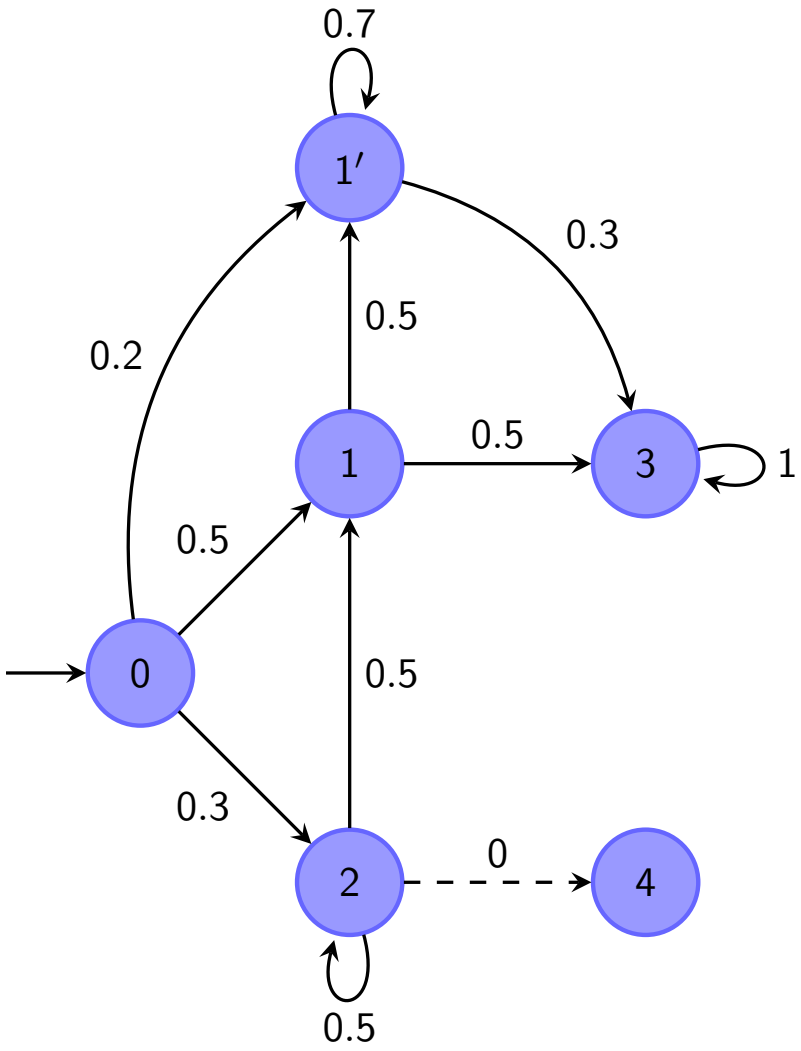
"IMDP" semantics $\models_{\mathrm{I}}^{d}$

Interval Markov Chains (IMCs)

Semantics



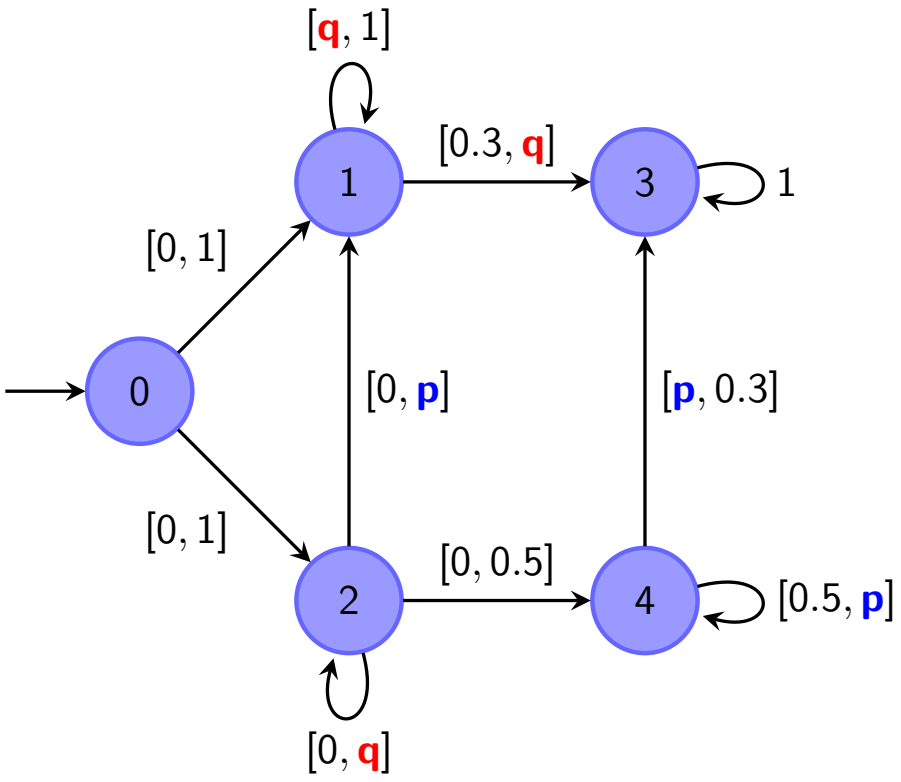
Specification (IMC)



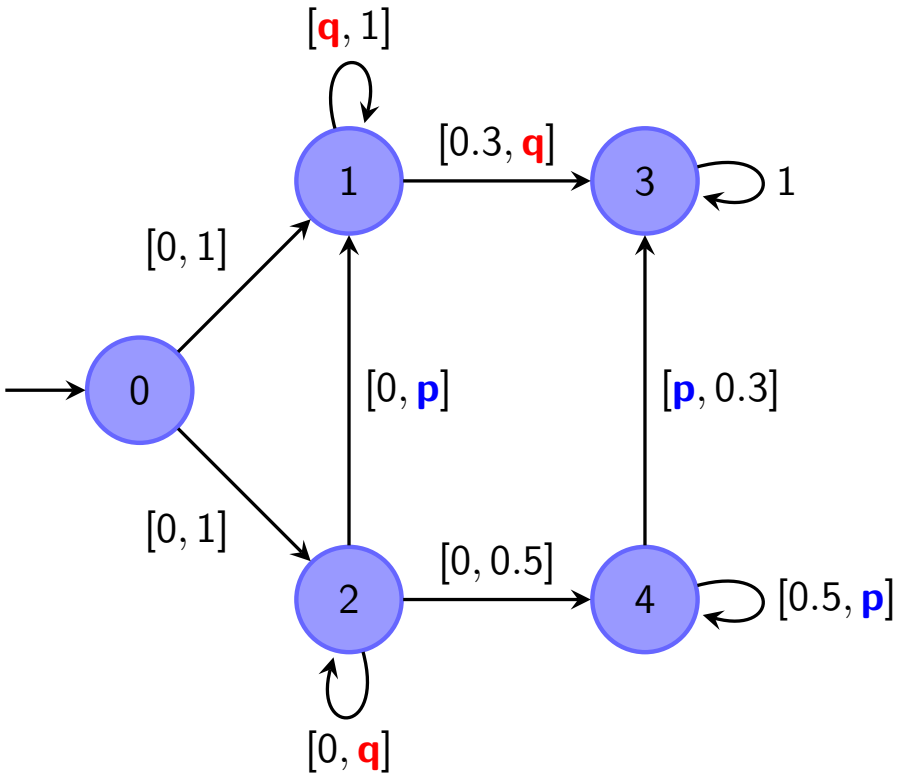
Implementation (MC)

"At every step" semantics $\models_{\mathrm{I}}^{a}$

Parametric Interval Markov Chains (pIMCs)

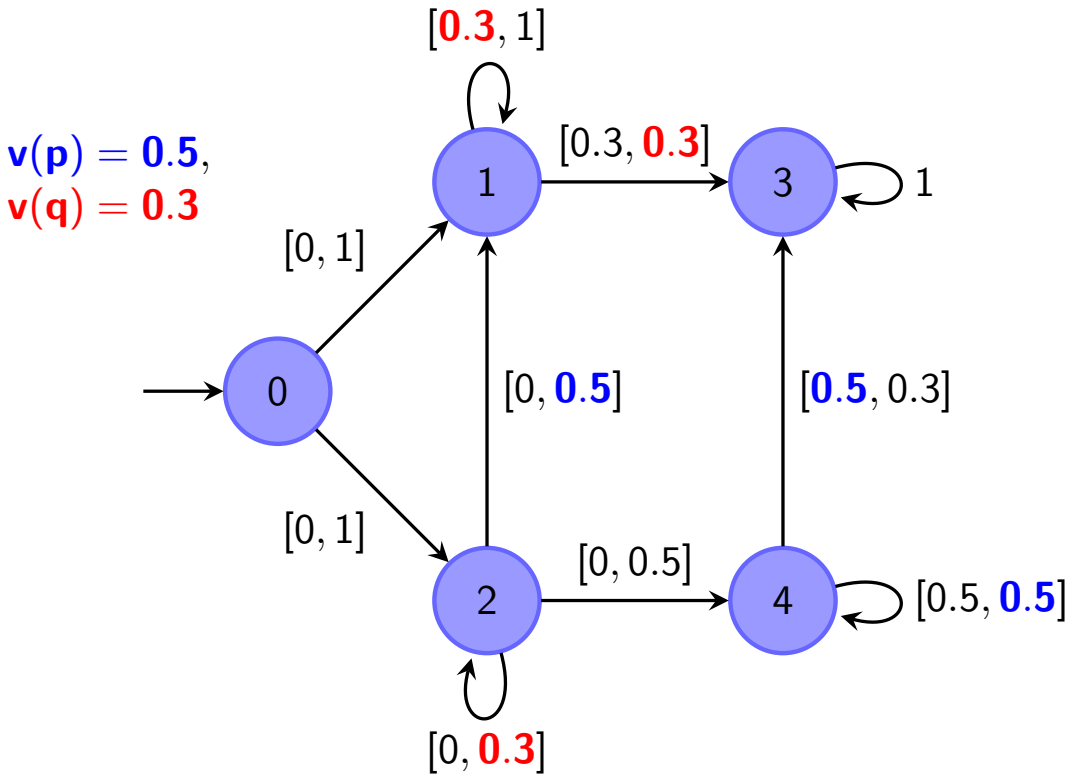


Parametric Interval Markov Chains (pIMCs)



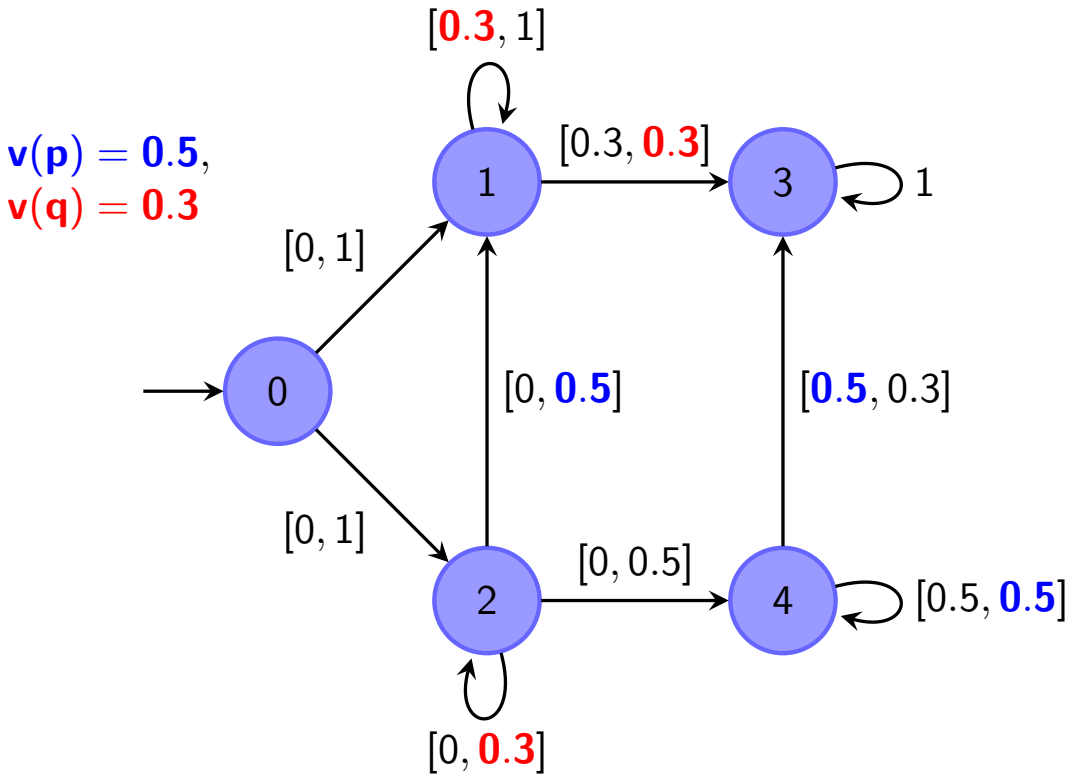
Valuating the parameters of $\mathcal{I}$ with valuation v gives an IMC $v(\mathcal{I})$

Parametric Interval Markov Chains (pIMCs)



Valuating the parameters of $\mathcal{I}$ with valuation v gives an IMC $v(\mathcal{I})$

Parametric Interval Markov Chains (pIMCs)



Valuating the parameters of $\mathcal{I}$ with valuation v gives an IMC $v(\mathcal{I})$

Implementations are implementations of the resulting IMCs w.r.t chosen semantics

Properties

Given pIMC $\mathcal{P}$ and state label α, synthesize **all parameter valuations** ensuring that

Consistency

$\mathcal{P}$ admits at least one implementation.

Existential/Universal qualitative reachability

- $\exists \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) > 0$
- $\forall \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) > 0$

Existential/Universal quantitative reachability

- $\exists \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) \sim \lambda$
- $\forall \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) \sim \lambda$

Properties

Given pIMC $\mathcal{P}$ and state label α, synthesize **all parameter valuations** ensuring that

Consistency

$\mathcal{P}$ admits at least one implementation.

Existential/Universal qualitative reachability

- $\exists \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) > 0$
- $\forall \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) > 0$

Existential/Universal quantitative reachability

- $\exists \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) \sim \lambda$
- $\forall \mathcal{M} \models \mathcal{P}$ s.t. $\mathbb{P}^{\mathcal{M}}(\Diamond \alpha) \sim \lambda$

Depends on the chosen IMC semantics

Outline

Reducing to "once-and-for-all"

Observation

The "once-and-for-all" semantics can be dealt with

- The implementation structure is known
- MC techniques can be adapted

Reducing to "once-and-for-all"

Observation

The "once-and-for-all" semantics can be dealt with

- The implementation structure is known
- MC techniques can be adapted

- Consistency:

Theorem

An IMC is consistent iff it admits an implementation **with the same structure**.

So we focus on implementations respecting the structure of the IMC.

Reducing to "once-and-for-all"

Observation

The "once-and-for-all" semantics can be dealt with

- The implementation structure is known
- MC techniques can be adapted

- Consistency:

Theorem

An IMC is consistent iff it admits an implementation **with the same structure**.

So we focus on implementations respecting the structure of the IMC.

- Extends to qualitative reachability

Quantitative reachability

Equivalence of $\models_{\mathtt{I}}^{a}$, $\models_{\mathtt{I}}^{d}$ and $\models_{\mathtt{I}}^{o}$ for IMCs

Theorem

Let $\mathcal{I}$ be an IMC*. For all MC $\mathcal{M}$ such that $\mathcal{M} \models_{\mathtt{I}}^{a} \mathcal{I}$ and all state label α, there exist MCs $\mathcal{M}_{\leq}$ and $\mathcal{M}_{\geq}$ such that $\mathcal{M}_{\leq} \models_{\mathtt{I}}^{o} \mathcal{I}$, $\mathcal{M}_{\geq} \models_{\mathtt{I}}^{o} \mathcal{I}$ and*

$$\mathbb{P}^{\mathcal{M}_{\leq}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}_{\geq}}(\Diamond\alpha)$$

Constructive proof:

Quantitative reachability

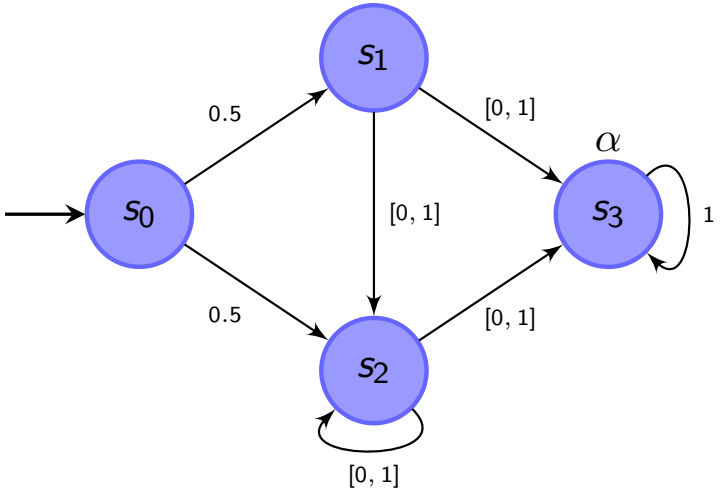
Equivalence of $\models_{\mathtt{I}}^{a}$, $\models_{\mathtt{I}}^{d}$ and $\models_{\mathtt{I}}^{o}$ for IMCs

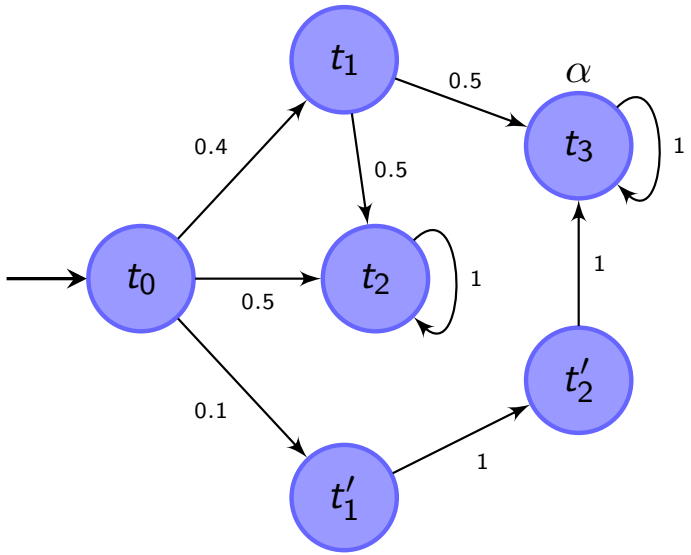
Theorem

Let $\mathcal{I}$ be an IMC. *For all MC $\mathcal{M}$ such that $\mathcal{M} \models_{\mathtt{I}}^{a} \mathcal{I}$ and all state label α, there exist MCs $\mathcal{M}_{\leq}$ and $\mathcal{M}_{\geq}$ such that $\mathcal{M}_{\leq} \models_{\mathtt{I}}^{o} \mathcal{I}$, $\mathcal{M}_{\geq} \models_{\mathtt{I}}^{o} \mathcal{I}$ and*

$$\mathbb{P}^{\mathcal{M}_{\leq}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}_{\geq}}(\Diamond\alpha)$$

Constructive proof:





Quantitative reachability

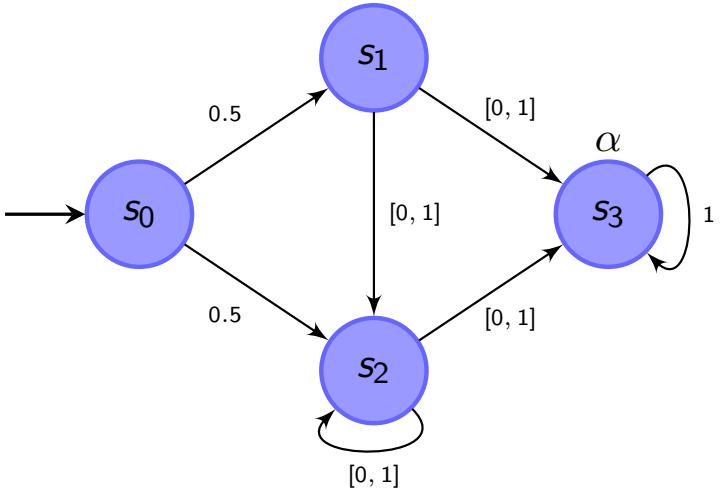
Equivalence of $\models_{\mathtt{I}}^{a}$, $\models_{\mathtt{I}}^{d}$ and $\models_{\mathtt{I}}^{o}$ for IMCs

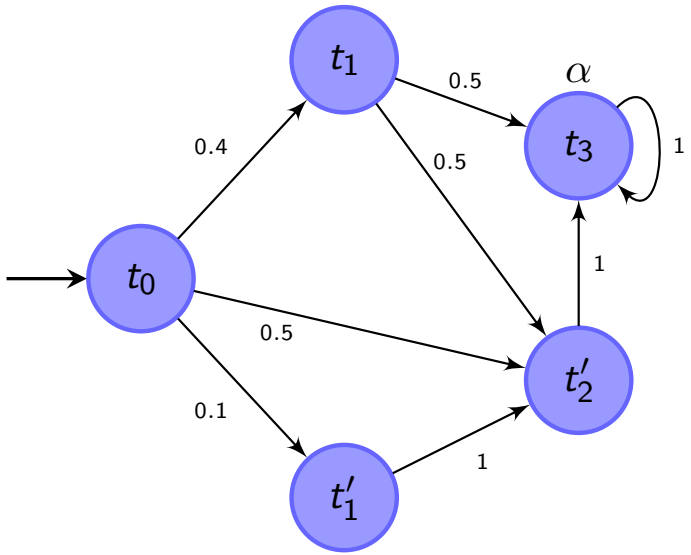
Theorem

Let $\mathcal{I}$ be an IMC. *For all MC $\mathcal{M}$ such that $\mathcal{M} \models_{\mathtt{I}}^{a} \mathcal{I}$ and all state label α, there exist MCs $\mathcal{M}_{\leq}$ and $\mathcal{M}_{\geq}$ such that $\mathcal{M}_{\leq} \models_{\mathtt{I}}^{o} \mathcal{I}$, $\mathcal{M}_{\geq} \models_{\mathtt{I}}^{o} \mathcal{I}$ and*

$$\mathbb{P}^{\mathcal{M}_{\leq}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}_{\geq}}(\Diamond\alpha)$$

Constructive proof:





Quantitative reachability

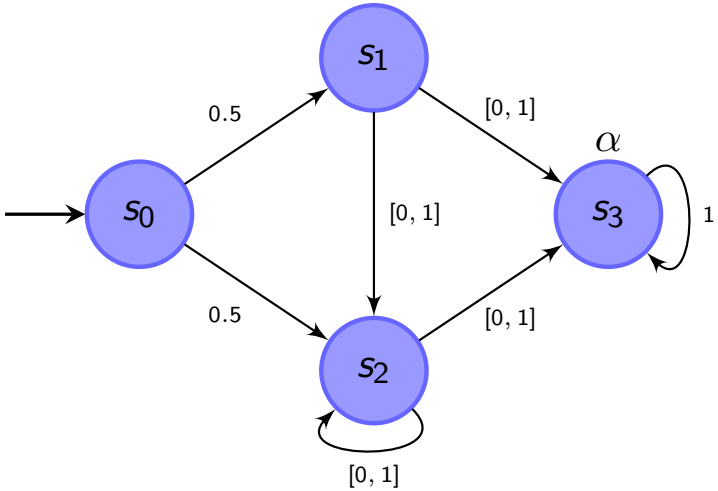
Equivalence of $\models_{\mathtt{I}}^{a}$, $\models_{\mathtt{I}}^{d}$ and $\models_{\mathtt{I}}^{o}$ for IMCs

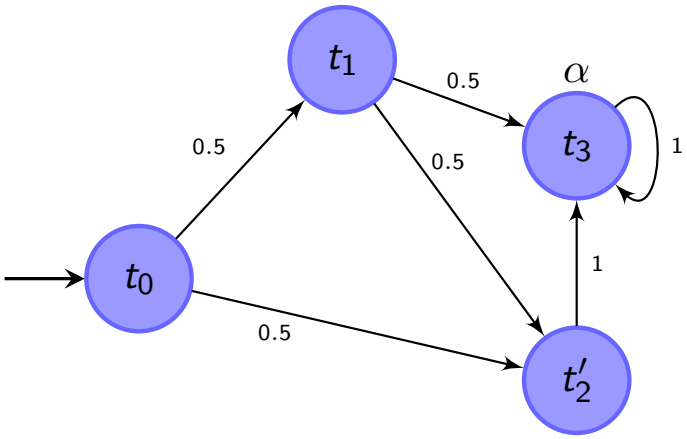
Theorem

Let $\mathcal{I}$ be an IMC. *For all MC $\mathcal{M}$ such that $\mathcal{M} \models_{\mathtt{I}}^{a} \mathcal{I}$ and all state label α, there exist MCs $\mathcal{M}_{\leq}$ and $\mathcal{M}_{\geq}$ such that $\mathcal{M}_{\leq} \models_{\mathtt{I}}^{o} \mathcal{I}$, $\mathcal{M}_{\geq} \models_{\mathtt{I}}^{o} \mathcal{I}$ and*

$$\mathbb{P}^{\mathcal{M}_{\leq}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \leq \mathbb{P}^{\mathcal{M}_{\geq}}(\Diamond\alpha)$$

Constructive proof:

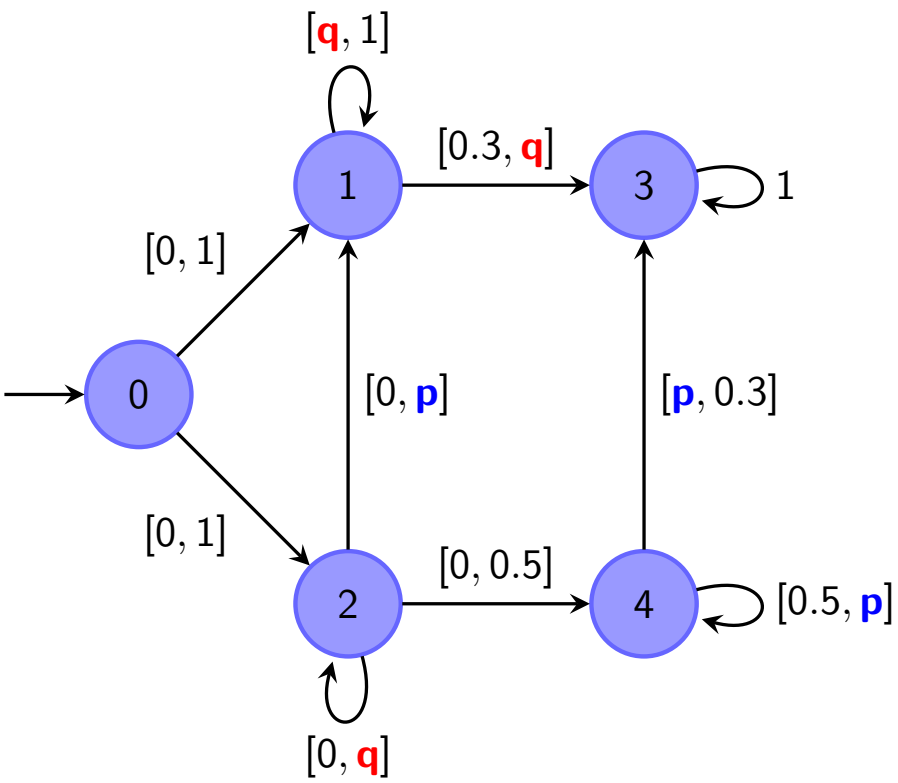




Outline

Constraint encoding $\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$

For consistency



$\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$: For each state $s \in S$, Constraints are as follows:

(1) ρ_s, if $s = s_0$

(2) $\rho_s \Leftrightarrow \sum_{s' \in \mathtt{Succ}(s)} \theta_s^{s'} = 1$

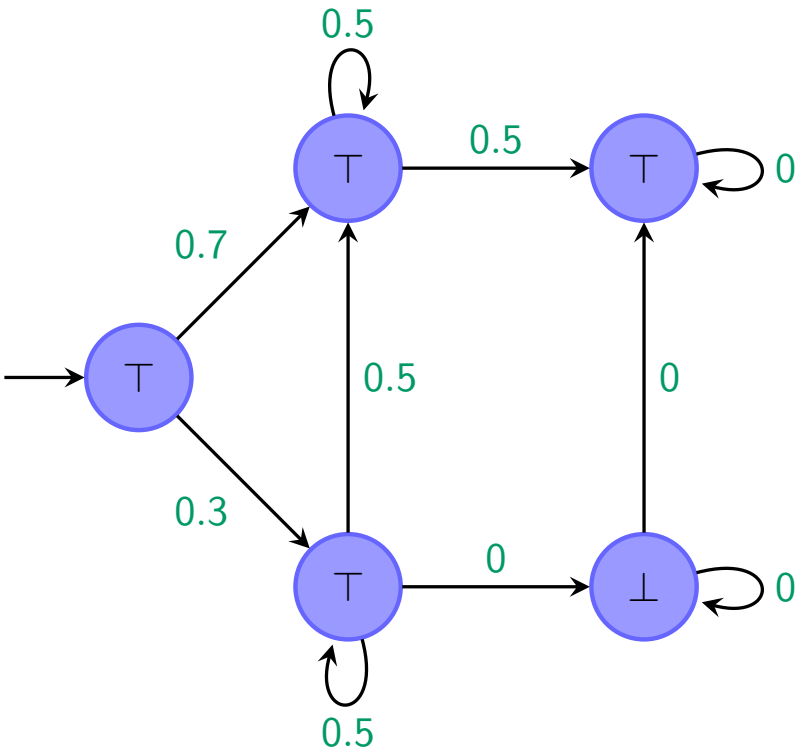
(3) $\neg\rho_s \Leftrightarrow \sum_{s' \in \mathtt{Pred}(s) \setminus \{s\}} \theta_{s'}^{s} = 0$, if $s \neq s_0$

(4) $\neg\rho_s \Leftrightarrow \sum_{s' \in \mathtt{Succ}(s)} \theta_s^{s'} = 0$

(5) $\rho_s \Rightarrow \theta_s^{s'} \in P(s, s')$, for all $s' \in \mathtt{Succ}(s)$

Constraint encoding $\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$

For consistency



$\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$: For each state $s \in S$,
Ex. solution:

$$p = 0.5$$

$$q = 0.5$$

Consistency and Qualitative Reachability

Theorem

A pIMC $\mathcal{P}$ *is existential consistent iff* $\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$ *is satisfiable.*

Linear in the size of $\mathcal{P}$

Consistency and Qualitative Reachability

Theorem

A pIMC $\mathcal{P}$ *is existential consistent iff* $\mathbf{C}_{\exists\mathbf{c}}(\mathcal{P})$ *is satisfiable.*

Linear in the size of $\mathcal{P}$

- Extends to qualitative reachability (Existential + Universal): $\mathbf{C}_{\exists\mathbf{r}}(\mathcal{P})$

Quantitative Reachability Encoding $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha)$

Based on the classical construction

($S_\top$ as usual, $S_\bot$ using $\mathbf{C}_{\exists\mathbf{r}}(\mathcal{P})$, and additionnal real-valued variables and constraints for reachability properties).

Theorem

- $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha) \cup (\pi_{s_0} \sim p))$ *is satisfiable iff* $\exists \mathcal{M} \models \mathcal{P}$: $\mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \sim p$
- $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha) \cup (\pi_{s_0} \not\sim p))$ *is unsatisfiable iff* $\forall \mathcal{M} \models \mathcal{P}$: $\mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \sim p$

Quantitative Reachability Encoding $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha)$

Based on the classical construction

($S_\top$ as usual, $S_\bot$ using $\mathbf{C}_{\exists\mathbf{r}}(\mathcal{P})$, and additionnal real-valued variables and constraints for reachability properties).

Theorem

- $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha) \cup (\pi_{s_0} \sim p))$ *is satisfiable iff* $\exists \mathcal{M} \models \mathcal{P}$: $\mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \sim p$
- $\mathbf{C}_{\exists\overline{\mathbf{r}}}(\mathcal{P}, \alpha) \cup (\pi_{s_0} \not\sim p))$ *is unsatisfiable iff* $\forall \mathcal{M} \models \mathcal{P}$: $\mathbb{P}^{\mathcal{M}}(\Diamond\alpha) \sim p$

Regardless of the chosen semantics!

Implementation

Prototype tool

Generates the CSP encodings

- QF_LRA (qualitative) / QF_NRA (quantitative) logic in SMT_LIB format
- No integer variables $\Rightarrow$ Real
- Plugged in to Z3

Implementation

Prototype tool

Generates the CSP encodings

- QF_LRA (qualitative) / QF_NRA (quantitative) logic in SMT_LIB format
- No integer variables ⇒ Real
- Plugged in to Z3

Benchmark NAND	pIMC			$C_{\exists c}$			$C_{\exists r}$			$C_{\exists \bar{r}}$		
	#states	#trans.	#par.	#var.	#cstr.	time	#var.	#cstr.	time	#var.	#cstr.	time
K=1; N=2	104	147	4	255	1,526	0.17s	170	1,497	0.19s	296	2,457	69.57s
K=1; N=3	252	364	5	621	3,727	0.24s	406	3,557	0.30s	703	5,828	31.69s
K=1; N=5	930	1,371	7	2,308	13,859	0.57s	1,378	12,305	0.51s	2,404	20,165	T.O.
K=1; N=10	**7,392**	**11,207**	**12**	18,611	111,366	9.46s	9,978	89,705	13.44s	17,454	147,015	T.O.

Outline

Discussion

- **Summary**:
  - New modeling formalism and verification technique
  - Constraint encodings
  - Linear size
  - Prototype implementation
  - Equivalence of semantics for quantitative reachability

Discussion

- **Summary**:
  - New modeling formalism and verification technique
  - Constraint encodings
  - Linear size
  - Prototype implementation
  - Equivalence of semantics for quantitative reachability
- **Perspectives**:
  - Extension of equivalence/encoding to other properties (LTL?)
  - Generation and representation of *all solutions*
  - Extension to Constaint Markov Chains
  - ...

Parametric Statistical Model Checking

and application to UAV case study

Christian Attiogbe, Ran Bao, Paulin Fournier, Abhignya Kamma, Didier Lime

Outline

Context

Verification of (parametric) probabilistic systems

Markov chain verification: well established

- Model-checking algorithms
- Goes through the whole state space
- Optimizations

Context

Verification of (parametric) probabilistic systems

Markov chain verification: well established

- Model-checking algorithms
- Goes through the whole state space
- Optimizations

- **State-space explosion**
- **Infinite systems?**

Context

Verification of (parametric) probabilistic systems

Markov chain verification: well established

- Model-checking algorithms
- Goes through the whole state space
- Optimizations

- **State-space explosion**
- **Infinite systems?**

Parametric Markov chains verification

- Similar to Markov chains verification
- Yields rational functions of the parameters

Context

Verification of (parametric) probabilistic systems

Markov chain verification: well established

- Model-checking algorithms
- Goes through the whole state space
- Optimizations

- **State-space explosion**
- **Infinite systems?**

Parametric Markov chains verification

- Similar to Markov chains verification
- Yields rational functions of the parameters

- **State-space explosion**
- **Infinite systems?**
- **Number of parameters?**

Motivation

Dealing with large (infinite) probabilistic systems

Statistical model checking (SMC)

Multiple approximation techniques

- Simulation-based
- Formal guarantees

Motivation

Dealing with large (infinite) probabilistic systems

Statistical model checking (SMC)

Multiple approximation techniques

- Simulation-based
- Formal guarantees

- Model-size independent
- Any executable model
- Infinite systems

Motivation

Dealing with large (infinite) probabilistic systems

Statistical model checking (SMC)

Multiple approximation techniques

- Simulation-based
- Formal guarantees

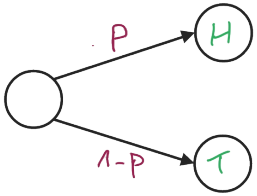
- Model-size independent
- Any executable model
- Infinite systems

Limitations:

- **Bounded linear properties only**
- **Purely probabilistic systems only**

What about parametric models?

1. How to simulate?

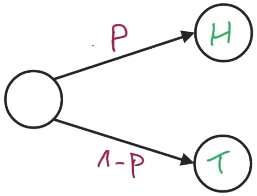


   - Guess and optimize?

2. How to give meaningfull guarantees?

What about parametric models?

1. How to simulate?



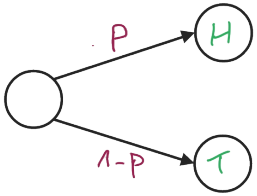
- Guess and optimize?

$\Rightarrow$ Importance sampling

2. How to give meaningfull guarantees?
   $\Rightarrow$ Importance sampling

What about parametric models?

1. How to simulate?



- Guess and optimize?

⇒ Importance sampling

2. How to give meaningfull guarantees?
   ⇒ Importance sampling

⇒ We extend Statistical Model Checking (Monte Carlo) to **parametric Markov chains**

- Based on **importance sampling** with symbolic outcome
- Output is a polynomial function of the parameters
- Comes with parametric confidence intervals

\+ Prototype tool

\+ Application to UAV case study

Outline

Properties

We only consider properties on bounded runs.

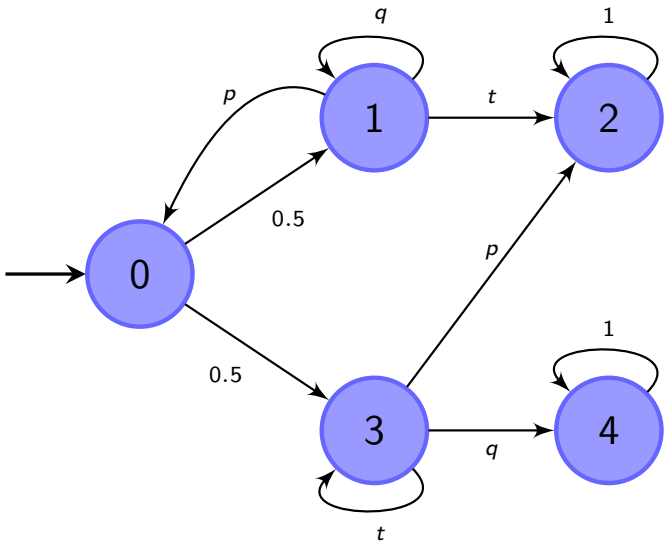
- **Bounded reachability**: Studying the probability of reaching a given state in less than ℓ steps. Written $\mathbb{P}_{\mathcal{M}}(\Diamond^{\leq \ell} s)$.
- **Bounded safety**: Studying the probability of staying in a set of **safe** states during ℓ steps. Written $\mathbb{P}_{\mathcal{M}}(\Box^{=\ell} E)$.
- **Expected reward**: Studying the expected value of a given **reward** function r on runs of length ℓ. Written $\mathbb{E}^{\ell}_{\mathcal{M}}(r)$.

Remark

Expected reward properties capture bounded safety and reachability.

Intuition for pMCs

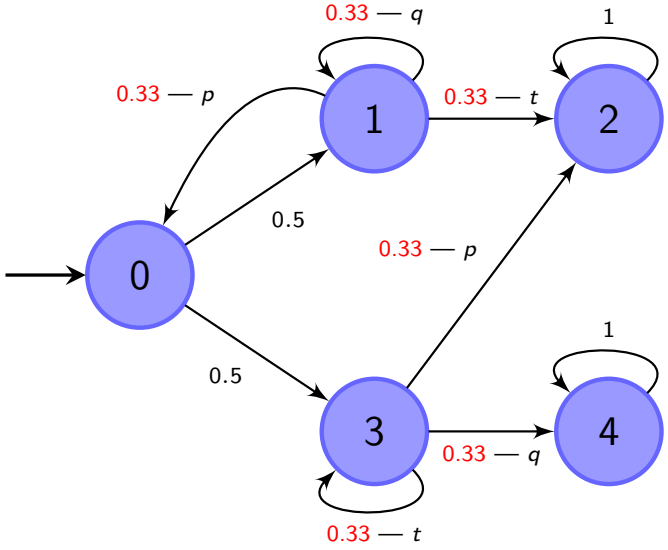
Expected reward r



- How to run simulations?

Intuition for pMCs

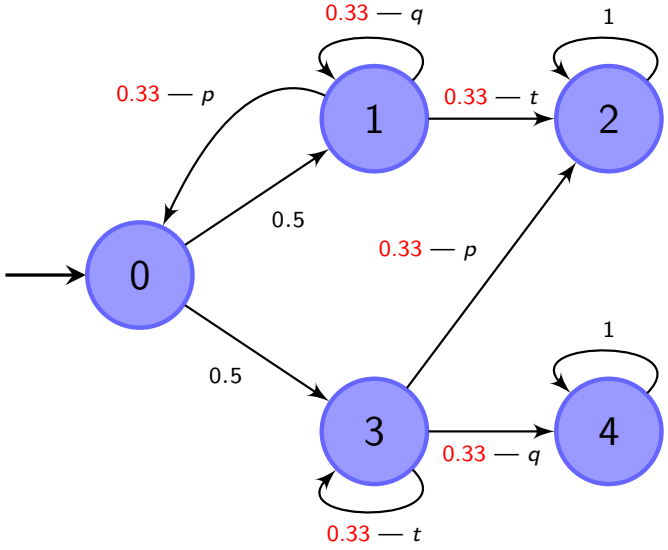
Expected reward r



- How to run simulations? Use a *normalization function* $f \to \mathcal{M}^f$

Intuition for pMCs

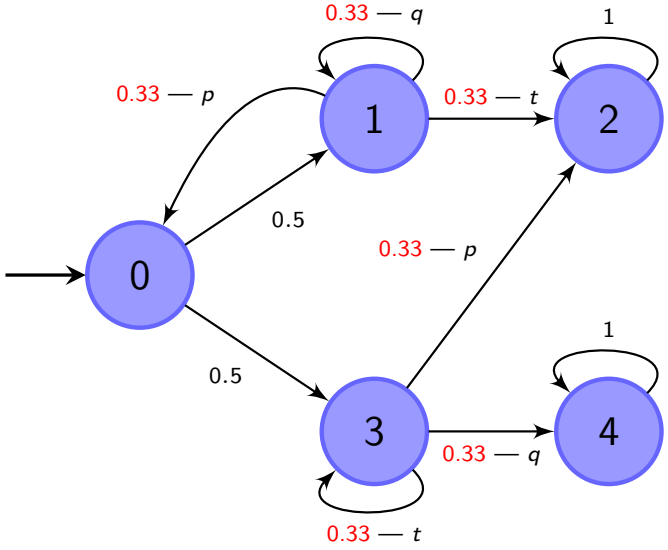
Expected reward r



- How to run simulations? Use a *normalization function* $f \to \mathcal{M}^f$
- Use $r'(\rho) = \frac{\mathbb{P}_{\mathcal{M}}(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$ instead of r

Intuition for pMCs

Expected reward r



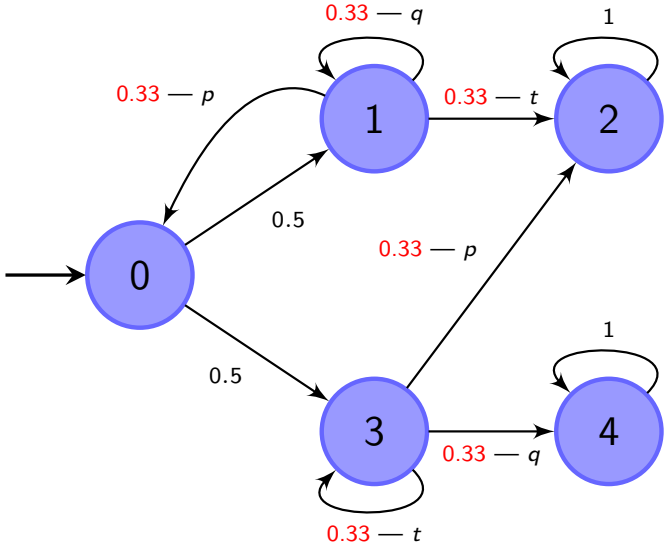
- How to run simulations? Use a *normalization function* $f \to \mathcal{M}^f$
- Use $r'(\rho) = \frac{\mathbb{P}_{\mathcal{M}}(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$ instead of r

$$\boxed{\mathbb{E}^{\ell}_{\mathcal{M}}(r')(v) = \mathbb{E}^{\ell}_{\mathcal{M}^v}(r) \text{ for all valid parameter valuation } v}$$

If $\mathcal{M}^f$ and $\mathcal{M}^v$ have the same structure

Intuition for pMCs

Expected reward r



- How to run simulations? Use a *normalization function* $f \to \mathcal{M}^f$
- Use $r'(\rho) = \frac{\mathbb{P}_{\mathcal{M}}(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$ instead of r

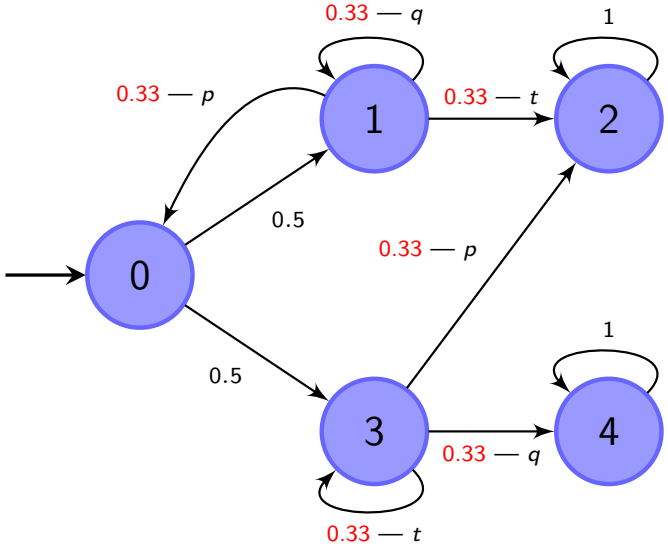
$$\boxed{\mathbb{E}^{\ell}_{\mathcal{M}}(r')(v) = \mathbb{E}^{\ell}_{\mathcal{M}^v}(r) \text{ for all valid parameter valuation } v}$$

If $\mathcal{M}^f$ and $\mathcal{M}^v$ have the same structure

- **Bonus:** confidence interval = polynomial function of the parameters

Example

Measure the probability of reaching 4 in less than 5 steps

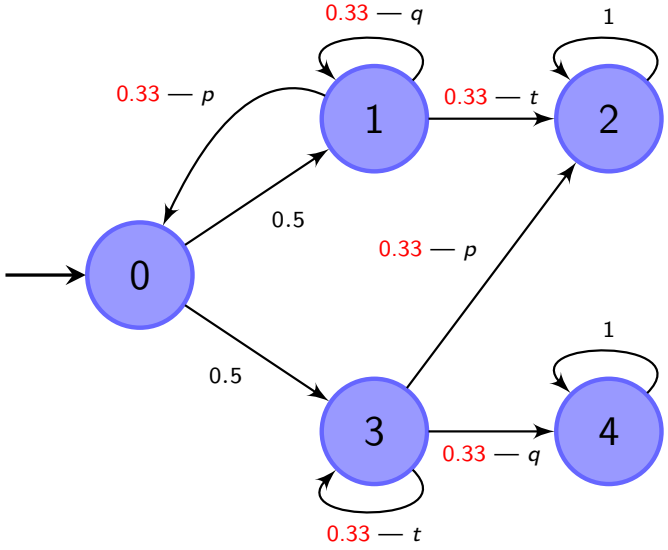


$$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2$$
$$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4$$
$$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2$$
$$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0$$
$$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4$$
$$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4$$
$$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2$$
$$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2$$

- $r(\rho_i) = 1$ if ρ_i reaches 4

Example

Measure the probability of reaching 4 in less than 5 steps

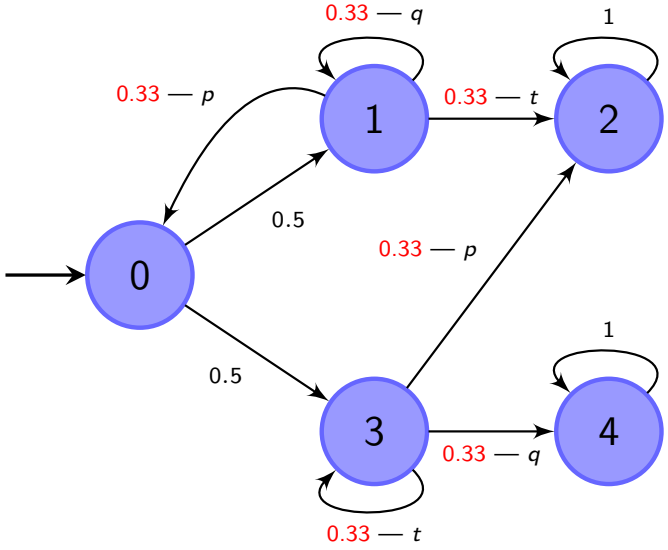


$$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_1) = 0$$
$$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4 \qquad r'(\rho_2) = 27pqt$$
$$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_3) = 0$$
$$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0 \qquad r'(\rho_4) = 0$$
$$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4 \qquad r'(\rho_5) = 3q$$
$$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4 \qquad r'(\rho_6) = 27qt^2$$
$$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 \qquad r'(\rho_7) = 0$$
$$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_8) = 0$$

- $r(\rho_i) = 1$ if ρ_i reaches 4

Example

Measure the probability of reaching 4 in less than 5 steps



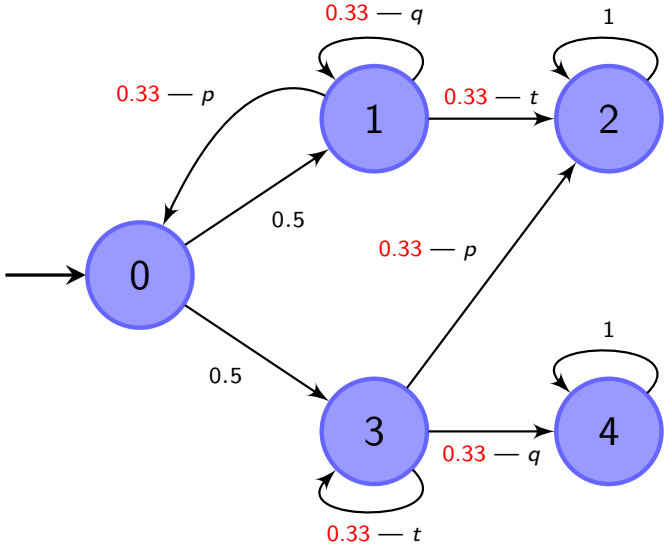
$$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_1) = 0$$
$$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4 \qquad r'(\rho_2) = 27pqt$$
$$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_3) = 0$$
$$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0 \qquad r'(\rho_4) = 0$$
$$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4 \qquad r'(\rho_5) = 3q$$
$$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4 \qquad r'(\rho_6) = 27qt^2$$
$$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 \qquad r'(\rho_7) = 0$$
$$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_8) = 0$$

- $r(\rho_i) = 1$ if ρ_i reaches 4
- Here, $\mathbb{E}^5_{\mathcal{M}}(r') \sim 3.375pqt + 0.375q + 3.375qt^2$

  For $v(p) = v(q) = 0.25, v(t) = 0.5$: $\mathbb{E}^5_{\mathcal{M}}(r')(v) \sim 0.41$ (exact: 0.261..)

Example

Measure the probability of reaching 4 in less than 5 steps



$$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_1) = 0$$
$$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4 \qquad r'(\rho_2) = 27pqt$$
$$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_3) = 0$$
$$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0 \qquad r'(\rho_4) = 0$$
$$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4 \qquad r'(\rho_5) = 3q$$
$$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4 \qquad r'(\rho_6) = 27qt^2$$
$$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 \qquad r'(\rho_7) = 0$$
$$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \qquad r'(\rho_8) = 0$$

- $r(\rho_i) = 1$ if ρ_i reaches 4
- Here, $\mathbb{E}^5_{\mathcal{M}}(r') \sim 3.375pqt + 0.375q + 3.375qt^2$

  For $v(p) = v(q) = 0.25, v(t) = 0.5$: $\mathbb{E}^5_{\mathcal{M}}(r')(v) \sim 0.41$ (exact: 0.261..)
- With 1 000 runs:

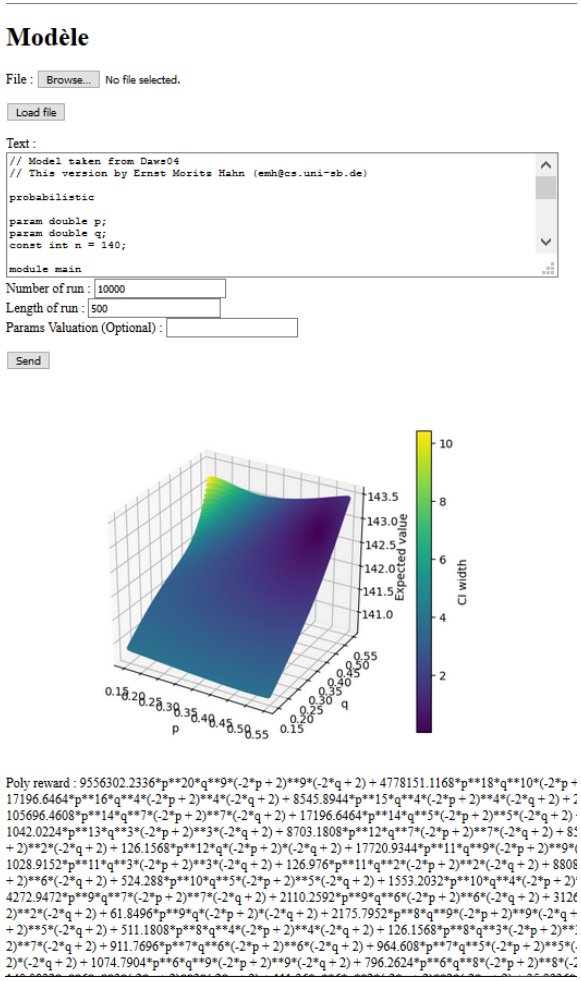
  $\mathbb{E}^5_{\mathcal{M}}(r') \sim 0.2801$
  Size of CI $= 0.0296$

Prototype Implementation

MCpMC[a]

parametric Statistical Model Checking

- Written in Python
- input: prism model or python class
- output: parametric probability function / confidence interval
- graphical web interface (ongoing work)



[a] Available at `https://github.com/Astlo/IMCpMC`

Outline

Context: UAVs flying above a crowd (Entertainment)

PIXIEL GROUP

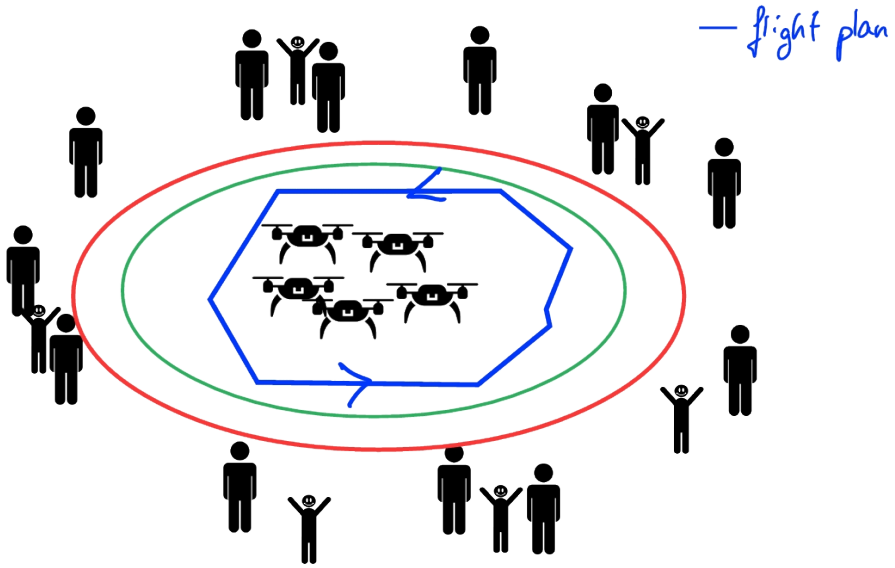
- UAV public shows
- Concerned with safety

Context: UAVs flying above a crowd (Entertainment)

- UAV public shows
- Concerned with safety

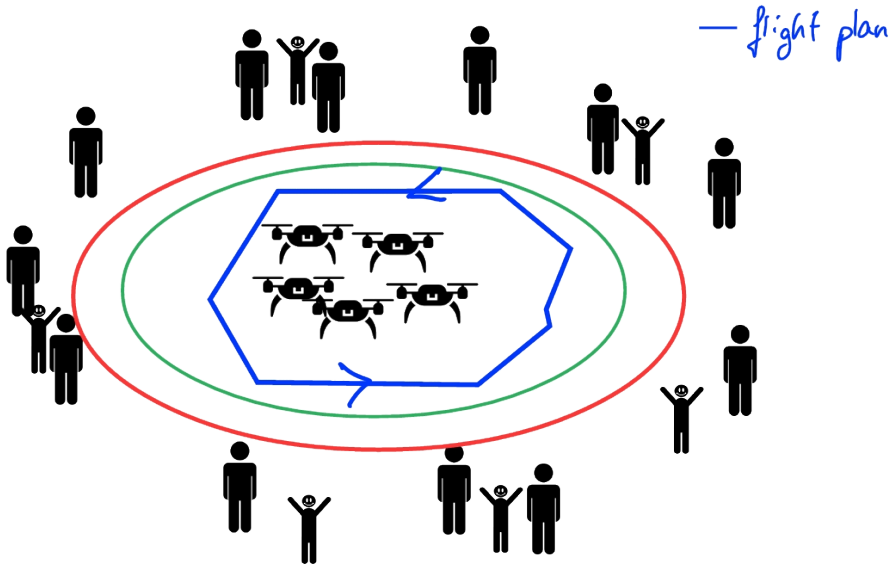
⇒ **How to ensure that the flight is safe?**

Measuring safety of the crowd



Guaranteeing human safety

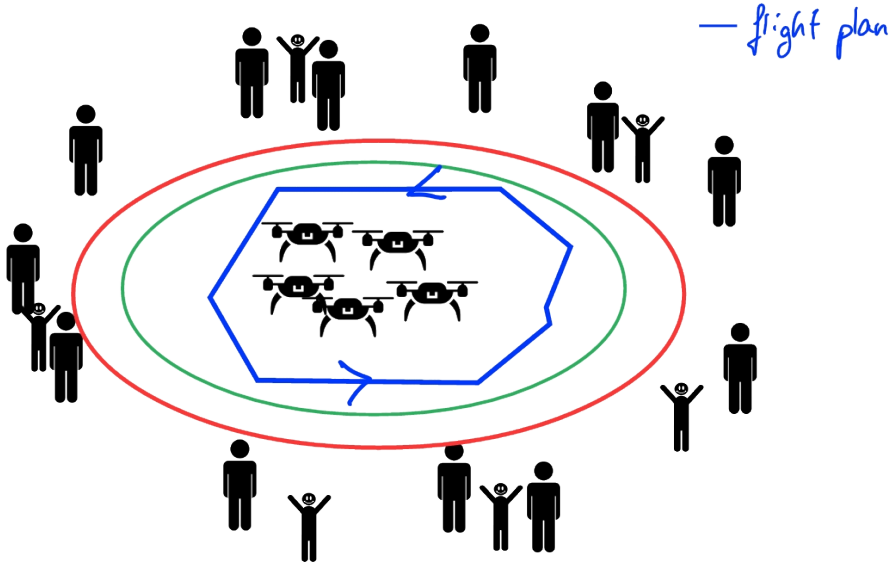
Measuring safety of the crowd



Guaranteeing human safety

- **Humans are safe if UAVs are far**

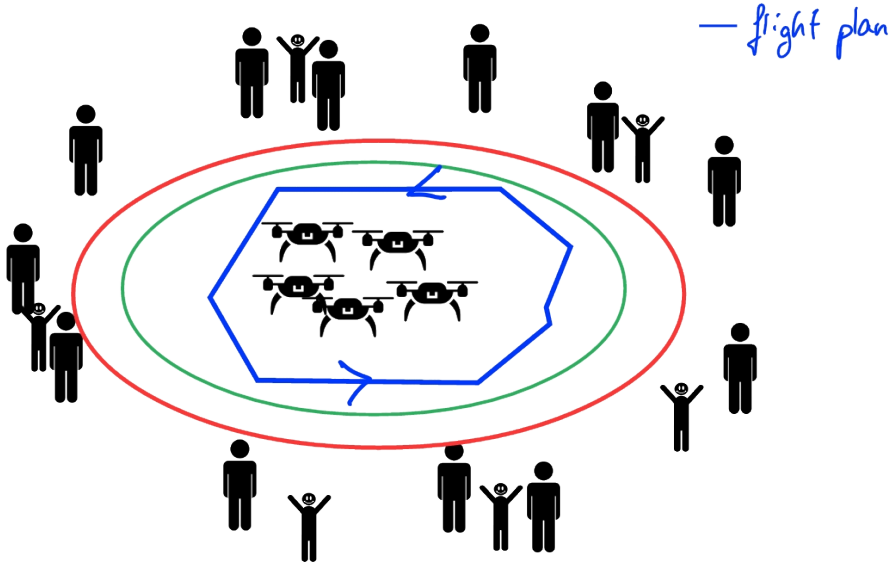
Measuring safety of the crowd



Guaranteeing human safety

- **Humans are safe if UAVs are far**
- Trajectory computation depends on position estimation

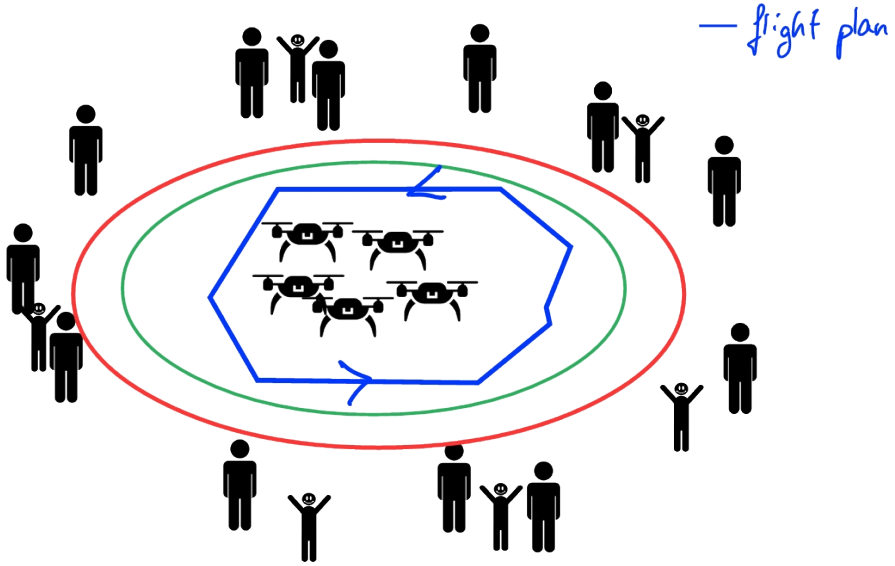
Measuring safety of the crowd



Guaranteeing human safety

- **Humans are safe if UAVs are far**
- Trajectory computation depends on position estimation
- Precise position estimation ensures human safety

Measuring safety of the crowd



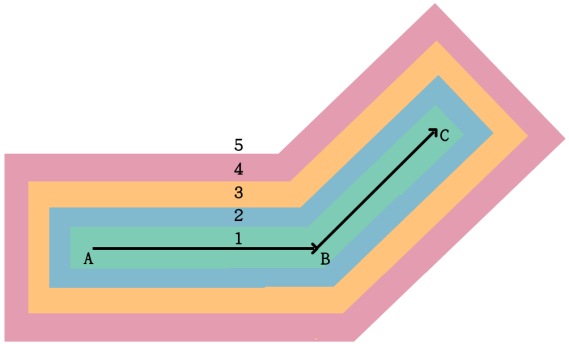
Guaranteeing human safety

- **Humans are safe if UAVs are far**
- Trajectory computation depends on position estimation
- Precise position estimation ensures human safety

⇒ Focus on the probability that a UAV stays in a "safe zone"

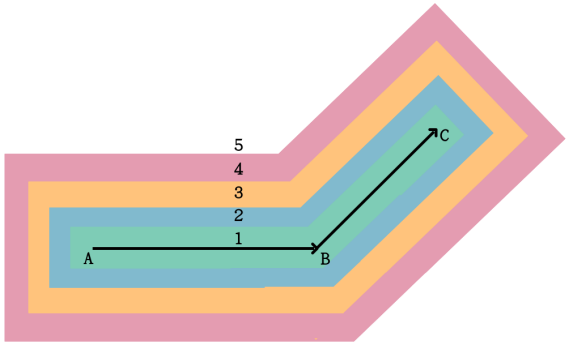
Modeling UAV trajectory

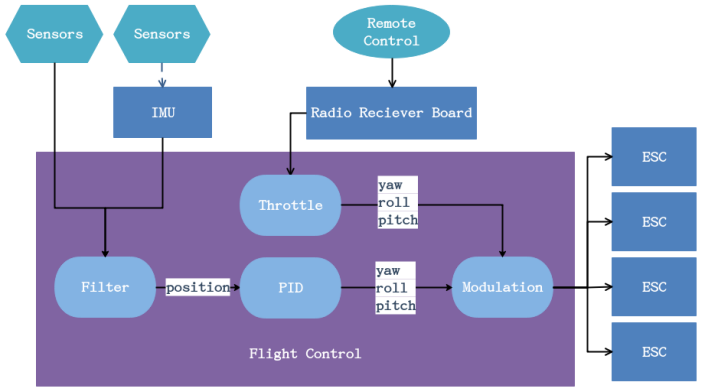
- Flight plan is cut in segments
- 5 **safety zones** of fixed size



Modeling UAV trajectory

- Flight plan is cut in segments
- 5 **safety zones** of fixed size



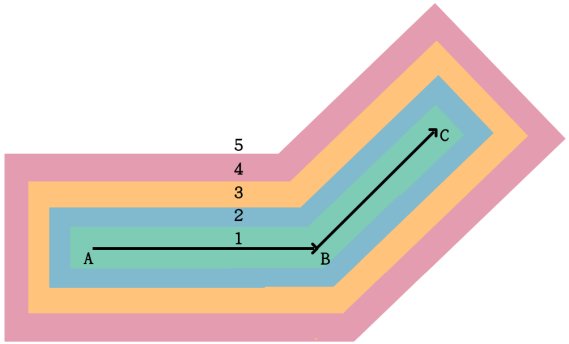


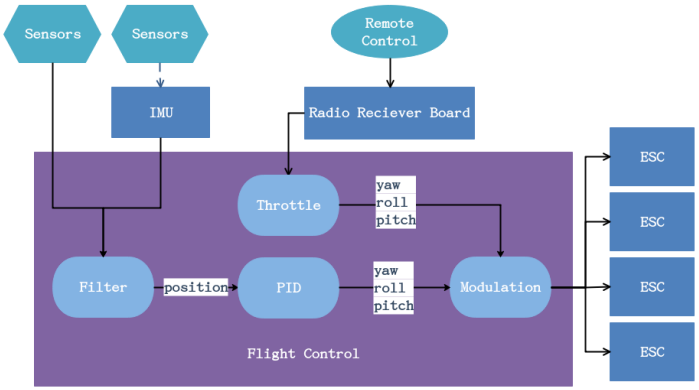
Position estimation:

Sensor precision
Filter correction
} Precision parameter

Modeling UAV trajectory

- Flight plan is cut in segments
- 5 **safety zones** of fixed size



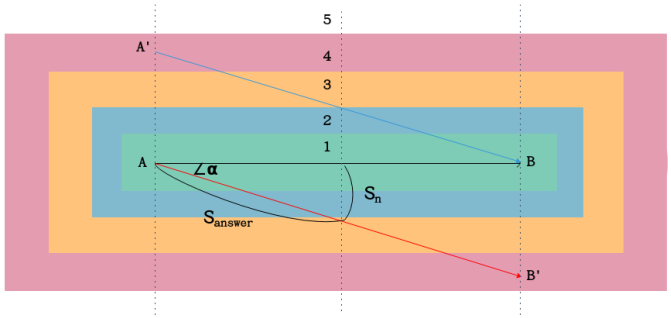


Position estimation:

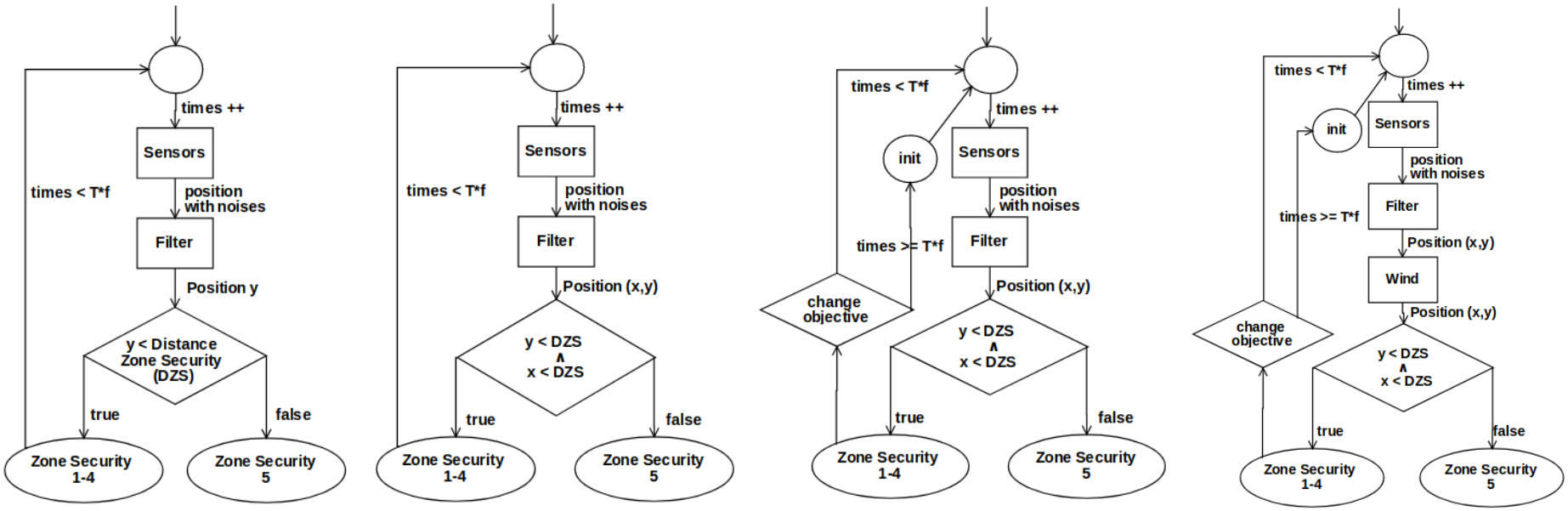
Sensor precision
Filter correction
} Precision parameter

For each segment

- Estimated position according to precision parameters
- Next position = function of estimated position



Resulting Models



(a) Precision on y

(b) Precision on x, y

(c) Complex flight plan

(d) Wind disturbance

- 4 models of increasing complexity
- Position = real-valued variables
- Prism (a) and Python (b,c,d) models
- Parameters = precision of Position(x,y) + wind force

Results: Probability of entering zones 4-5

#sim			10k		20k		50k	
	Model	#p	V1	V2	V1	V2	V1	V2
Running time	**(d)(np)**	**5**	**28s**		**53-54s**		**149-155s**	
Scenario 1			5.44%	5.31%	5.61%	5.21%	5.59%	5.47%
Conf. interv.			±0.98%	±0.86%	±0.69%	±0.64%	±0.42%	±0.43%
Scenario 2			10.8%	10.9%	10.8%	10.8%	10.9%	10.7%
Conf. interv.			±1.35%	±1.32%	±0.91%	±0.92%	±0.57%	±0.57%
Running time	**(d)(p)**	**9**	**185-190s**		**311-314s**		**612-621s**	
Scenario 1			4.95%	5.97%	5.28%	6.62%	4.16%	5.61%
Conf. interv.			±5.22%	±5.71%	±4.71%	±6.25%	±1.86%	±4.38%
Scenario 2			9.55%	9.87%	10.3%	11.3%	9.57%	10.7%
Conf. interv.			±8.40%	±7.86%	±7.04%	±7.89%	±5.29%	±3.99%

Precision parameters PF0/1/2/3/4: 0-2m / 2-4m / 4-6m / 6-8m / 8-10m

Zone4: 8m from trajectory. Zone5: 50m from trajectory

Scenario 1

PF0/1/2/3/4 = 0.15/0.3/0.4/0.1/0.05

Scenario 2

PF0/1/2/3/4 = 0.1/0.25/0.35/0.2/0.1

Outline

Discussion

- **Summary**:
  - Parametric Monte Carlo procedure for pMC
  - Polynomial parametric confidence interval
  - Prototype implementation
  - Application to UAV flight plan analysis

Discussion

- **Summary**:
  - Parametric Monte Carlo procedure for pMC
  - Polynomial parametric confidence interval
  - Prototype implementation
  - Application to UAV flight plan analysis
- **Perspectives**:
  - Impact of the normalization function
  - Improving MCpMC
  - Comparison/Integration(?) to existing tools

Outline

Gap between theory and practice

Remark

The modeling formalism used in the UAV case study **is not** (parametric/interval) Markov chains.

Gap between theory and practice

Remark

The modeling formalism used in the UAV case study **is not** (parametric/interval) Markov chains.

- Not suited to practical applications
  - Real valued variables
  - Complex behaviour
  - Huge model
- Model practitionners are not aware / expert of those formalisms

Gap between theory and practice

Remark

The modeling formalism used in the UAV case study **is not** (parametric/interval) Markov chains.

- Not suited to practical applications
  - Real valued variables
  - Complex behaviour
  - Huge model
- Model practitionners are not aware / expert of those formalisms

We need to make theory usable in practice

Gap between theory and practice

Remark

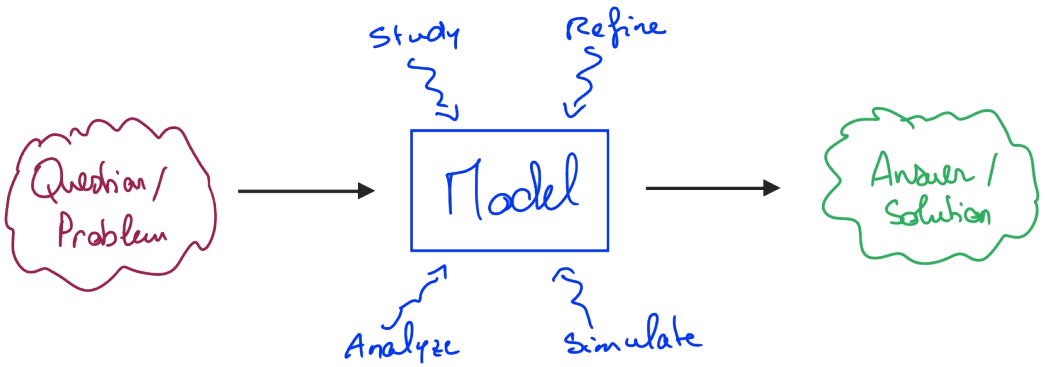
The modeling formalism used in the UAV case study **is not** (parametric/interval) Markov chains.

- Not suited to practical applications
  - Real valued variables
  - Complex behaviour
  - Huge model
- Model practitionners are not aware / expert of those formalisms

We need to make theory usable in practice

- pSMC is promising (few modeling requirements)

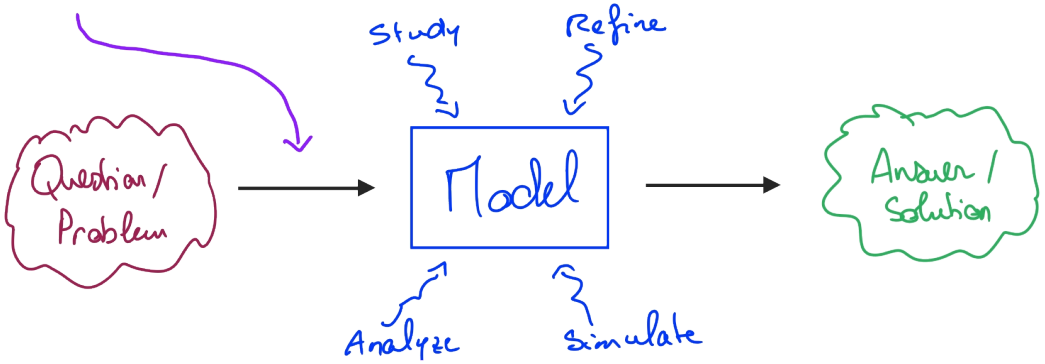
Research axes



Research axes

Reducing model size

- Abstraction / Composition theories
- Automated model learning using experimental data



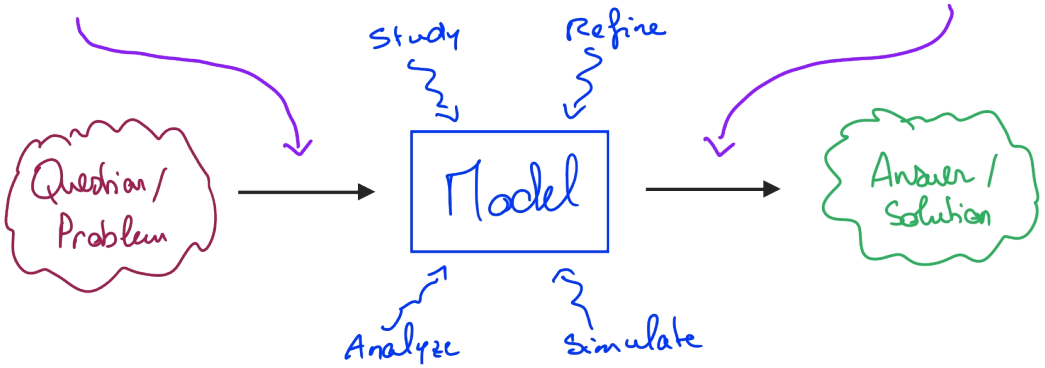
Research axes

Reducing model size

- Abstraction / Composition theories
- Automated model learning using experimental data

Enhancing pSMC

- More insightfull types of parameters
- Choice of normalization functions



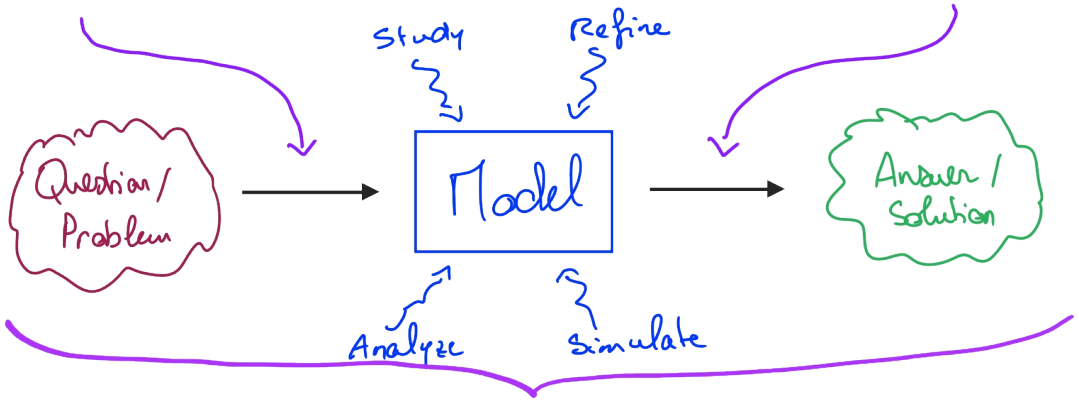
Research axes

Reducing model size

- Abstraction / Composition theories
- Automated model learning using experimental data

Enhancing pSMC

- More insightfull types of parameters
- Choice of normalization functions



Developing a verification platform

- Usable by computer scientists **and** modelers

  Ex: Parameterisation of the jellyfish model

Thanks

- The jury
- Ph.D. students and postdocs

  Amine, Dimitri, Eva, Hadrien, Paulin
- Co-authors

  Anicet, Arnaud, Charlotte, Christian, Damien E., Didier, Etienne, Eric, Kim, Laure, Lionel, Olivier H., Philippe, Simon, ...
- LS2N Colleagues

  Anne-Françoise, Annie, Assia, Audrey, Charlery, Christine, Claude, Damien V., Elodie, Géraldine, Gilles, Guillaume, Laura, Marion, Pascal, Samuel, Virginie, ...
- All my friends and familly

Thanks

- The jury
- Ph.D. students and postdocs

  Amine, Dimitri, Eva, Hadrien, Paulin
- Co-authors

  Anicet, Arnaud, Charlotte, Christian, Damien E., Didier, Etienne, Eric, Kim, Laure, Lionel, Olivier H., Philippe, Simon, ...
- LS2N Colleagues

  Anne-Françoise, Annie, Assia, Audrey, Charlery, Christine, Claude, Damien V., Elodie, Géraldine, Gilles, Guillaume, Laura, Marion, Pascal, Samuel, Virginie, ...
- All my friends and familly

Thank you for listening!

References I

[TIME'16] Étienne André and Benoît Delahaye. Consistency in parametric interval probabilistic timed automata. In *International Symposium on Temporal Representation and Reasoning, TIME 2016, Proceedings*, pages 110–119. IEEE Computer Society, 2016.

[JLAMP'20] Étienne André, Benoît Delahaye, and Paulin Fournier. Consistency in parametric interval probabilistic timed automata. *J. Log. Algebr. Meth. Program.*, 110, 2020.

[SAC'17] Mohamed Amine Aouadhi, Benoît Delahaye, and Arnaud Lanoix. Moving from Event-B to probabilistic Event-B. In *Annual ACM Symposium on Applied Computing, 2017, Proceedings*. ACM, 2017.

[IEA/AIE'19] Dimitri Antakly, Benoît Delahaye, and Philippe Leray. Graphical event model learning and verification for security assessment. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2019, Proceedings*, volume 11606 of *LNCS*, pages 245–252. Springer, 2019.

[SoSyM'19] Mohamed Amine Aouadhi, Benoît Delahaye, and Arnaud Lanoix. Introducing probabilistic reasoning within Event-B. *Software and Systems Modeling*, 18(3):1953–1984, 2019.

[FORTE'19] Ran Bao, J. Christian Attiogbé, Benoît Delahaye, Paulin Fournier, and Didier Lime. Parametric statistical model checking of UAV flight plan. In *International Conference on Formal Techniques for Distributed Objects, Components, and Systems, FORTE 2019*, volume 11535 of *LNCS*, pages 57–74. Springer, 2019.

[STTT'12] Ananda Basu, Saddek Bensalem, Marius Bozga, Benoît Delahaye, and Axel Legay. Statistical abstraction and model-checking of large heterogeneous systems. *International Journal on Software Tools for Technology Transfer*, 14(1):53–72, 2012.

[ISoLA'12] Saddek Bensalem, Marius Bozga, Benoît Delahaye, Cyrille Jégourel, Axel Legay, and Ayoub Nouri. Statistical model checking qos properties of systems with sbip. In *International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, ISoLA, Proceedings*, volume 7609 of *LNCS*, pages 327–341. Springer, 2012.

[TCS'18] Anicet Bart, Benoît Delahaye, Paulin Fournier, Didier Lime, Eric Monfroy, and Charlotte Truchet. Reachability in parametric interval markov chains using constraints. *Theor. Comput. Sci.*, 747:48–74, 2018.

[QEST'17] Anicet Bart, Benoît Delahaye, Didier Lime, Eric Monfroy, and Charlotte Truchet. Reachability in parametric interval Markov chains using constraints. In *International Conference on Quantitative Evaluation of Systems, 2017, Proceedings*, volume 10503 of *LNCS*, pages 173–189. Springer, 2017.

References II

[MSystems'17] Benoît Delahaye, Damien Eveillard, and Nicholas Bouskill. On the power of uncertainties in microbial system modeling: No need to hide them anymore. *MSystems*, 2(6):e00169–17, 2017.

[SynCoP'15] Benoît Delahaye. Consistency for parametric interval Markov chains. In *International Workshop on Synthesis of Complex Parameters, 2015, Proceedings*, volume 44 of *OASICS*, pages 17–32. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[FORTE'13] Benoît Delahaye, José Luiz Fiadeiro, Axel Legay, and Antónia Lopes. A timed component algebra for services. In *International Conference on Formal Techniques for Distributed Systems, FORTE, 2013, Proceedings*, volume 7892 of *LNCS*, pages 242–257. Springer, 2013.

[LMCS'14] Benoît Delahaye, Uli Fahrenberg, Kim G. Larsen, and Axel Legay. Refinement and difference for probabilistic automata. *Logical Methods in Computer Science*, 10(3), 2014.

[ICTAC'14] Benoît Delahaye, José Luiz Fiadeiro, Axel Legay, and Antónia Lopes. Heterogeneous timed machines. In *International Colloquium on Theoretical Aspects of Computing, ICTAC, 2014, Proceedings*, volume 8687 of *LNCS*, pages 115–132. Springer, 2014.

[VMCAI'11] Benoît Delahaye, Joost-Pieter Katoen, Kim G. Larsen, Axel Legay, Mikkel L. Pedersen, Falak Sher, and Andrzej Wasowski. Abstract probabilistic automata. In *International Conference on Verification, Model Checking, and Abstract Interpretation, VMCAI, 2011, Proceedings*, volume 6538 of *LNCS*, pages 324–339. Springer, 2011.

[Inf. Comp.'13] Benoît Delahaye, Joost-Pieter Katoen, Kim G. Larsen, Axel Legay, Mikkel L. Pedersen, Falak Sher, and Andrzej Wasowski. Abstract probabilistic automata. *Inf. Comput.*, 232:66–116, 2013.

[VMCAI'16] Benoît Delahaye, Didier Lime, and Laure Petrucci. Parameter synthesis for parametric interval Markov chains. In *International Conference on Verification, Model Checking, and Abstract Interpretation, 2016, Proceedings*, volume 9583 of *LNCS*, pages 372–390. Springer, 2016.

[PETRI NETS'16] Yrvann Emzivat, Benoît Delahaye, Didier Lime, and Olivier H. Roux. Probabilistic time petri nets. In *International Conference on Application and Theory of Petri Nets and Concurrency, PETRI NETS, 2016, Proceedings*, volume 9698 of *LNCS*, pages 261–280. Springer, 2016.

[MSCS'18] José Luiz Fiadeiro, Antónia Lopes, Benoît Delahaye, and Axel Legay. Dynamic networks of heterogeneous timed machines. *Math. Struct. Comput. Sci.*, 28(6):800–855, 2018.

References III

[STTT'15] Ayoub Nouri, Saddek Bensalem, Marius Bozga, Benoît Delahaye, Cyrille Jégourel, and Axel Legay. Statistical model checking qos properties of systems with SBIP. *International Journal on Software Tools for Technology Transfer*, 17(2):171–185, 2015.

[Sci. Rep.'20] Simon Ramondenc, Damien Eveillard, Lionel Guidi, Fabien Lombard, and Benoît Delahaye. Probabilistic modeling to estimate jellyfish ecophysiological properties and size distributions. *Scientific Reports*, 10, 2020.