A Logical Characterization of Probabilistic Bisimulation

Prakash Panangaden

1School of Computer Science
McGill University
and
Simons Institute for Theoretical Computer Science
University of California, Berkeley

Berkeley Logic Colloquium 2nd December 2016
There is a simple modal logic which characterizes behavioural equivalence of states in a probabilistic transition system.
There is a simple modal logic which characterizes behavioural equivalence of states in a probabilistic transition system.

- Similar to the van Benthem-Hennessy-Milner result for (nondeterministic) transition systems, but
Main result

There is a simple modal logic which characterizes behavioural equivalence of states in a probabilistic transition system.

- Similar to the van Bentham-Hennessy-Milner result for (nondeterministic) transition systems, but
- for probabilistic systems.
Main result

There is a simple modal logic which characterizes behavioural equivalence of states in a probabilistic transition system.

- Similar to the van Benthem-Hennessy-Milner result for (nondeterministic) transition systems, but
- for probabilistic systems.
- In the last few weeks: Logical characterization for simulation in systems with countably many transitions; game characterization of bisimulation.
Related results

- Larsen and Skou proved a similar theorem but
Larsen and Skou proved a similar theorem but their transition systems were discrete,
Larsen and Skou proved a similar theorem but their transition systems were discrete, and there was a bound on the degree of branching of the transitions.
Larsen and Skou proved a similar theorem but their transition systems were discrete, there was a bound on the degree of branching of the transitions, all the probabilities had to be multiples of some fixed real number $\varepsilon$ and
Larsen and Skou proved a similar theorem but
their transition systems were discrete,
there was a bound on the degree of branching of the transitions,
all the probabilities had to be multiples of some fixed real number $\varepsilon$ and
their logic had some negative constructs.
Outline

1 Introduction
Outline

1. Introduction

2. Discrete probabilistic transition systems
Outline

1. Introduction
2. Discrete probabilistic transition systems
3. Labelled Markov processes
Outline

1. Introduction
2. Discrete probabilistic transition systems
3. Labelled Markov processes
4. Probabilistic bisimulation
Outline

1. Introduction
2. Discrete probabilistic transition systems
3. Labelled Markov processes
4. Probabilistic bisimulation
5. Game for bisimulation: Bartek Klin and Nathanaël Fijalkow
Outline

1. Introduction
2. Discrete probabilistic transition systems
3. Labelled Markov processes
4. Probabilistic bisimulation
5. Game for bisimulation: Bartek Klin and Nathanaël Fijalkow
6. Simulation
Outline

1. Introduction
2. Discrete probabilistic transition systems
3. Labelled Markov processes
4. Probabilistic bisimulation
5. Game for bisimulation: Bartek Klin and Nathanaël Fijalkow
6. Simulation
7. Concluding remarks
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]

Logical characterization. [LICS98, Info and Comp 2002]
Summary of Results

- Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
- Logical characterization. [LICS98, Info and Comp 2002]
- Metric analogue of bisimulation. [CONCUR99, TCS2004]
Summary of Results

- Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
- Logical characterization. [LICS98, Info and Comp 2002]
- Metric analogue of bisimulation. [CONCUR99, TCS2004]
- Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]

Logical characterization. [LICS98, Info and Comp 2002]

Metric analogue of bisimulation. [CONCUR99, TCS2004]

Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]

Weak bisimulation. [LICS02, CONCUR02]
**Summary of Results**

- Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
- Logical characterization. [LICS98, Info and Comp 2002]
- Metric analogue of bisimulation. [CONCUR99, TCS2004]
- Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]
- Weak bisimulation. [LICS02, CONCUR02]
- Real time. [QEST 2004, JLAP 2003, LMCS 2006]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]

Logical characterization. [LICS98, Info and Comp 2002]

Metric analogue of bisimulation. [CONCUR99, TCS2004]

Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]

Weak bisimulation. [LICS02, CONCUR02]

Real time. [QEST 2004, JLAP 2003, LMCS 2006]

Event bisimulation. [Info and Comp 2006]
Summary of Results

- Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
- Logical characterization. [LICS98, Info and Comp 2002]
- Metric analogue of bisimulation. [CONCUR99, TCS2004]
- Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]
- Weak bisimulation. [LICS02, CONCUR02]
- Real time. [QEST 2004, JLAP 2003, LMCS 2006]
- Event bisimulation. [Info and Comp 2006]
- Applications to machine learning [UAI 2004-06, AAAI 2015]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]

Logical characterization. [LICS98, Info and Comp 2002]

Metric analogue of bisimulation. [CONCUR99, TCS2004]

Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]

Weak bisimulation. [LICS02, CONCUR02]

Real time. [QEST 2004, JLAP 2003, LMCS 2006]

Event bisimulation. [Info and Comp 2006]

Applications to machine learning [UAI 2004-06, AAAI 2015]

Stone duality for Markov processes [LICS 2013]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]
Logical characterization. [LICS98, Info and Comp 2002]
Metric analogue of bisimulation. [CONCUR99, TCS2004]
Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]
Weak bisimulation. [LICS02, CONCUR02]
Real time. [QEST 2004, JLAP 2003, LMCS 2006]
Event bisimulation. [Info and Comp 2006]
Applications to machine learning [UAI 2004-06, AAAI 2015]
Stone duality for Markov processes [LICS 2013]
Dualized view of Markov processes [JACM 2014]
Probabilistic bisimulation can be defined for continuous state-space systems. [LICS97]

Logical characterization. [LICS98, Info and Comp 2002]

Metric analogue of bisimulation. [CONCUR99, TCS2004]

Approximation of LMPs. [LICS00, Info and Comp 2003, QEST 2006, CONCUR 2004]

Weak bisimulation. [LICS02, CONCUR02]

Real time. [QEST 2004, JLAP 2003, LMCS 2006]

Event bisimulation. [Info and Comp 2006]

Applications to machine learning [UAI 2004-06, AAAI 2015]

Stone duality for Markov processes [LICS 2013]

Dualized view of Markov processes [JACM 2014]

Simulation, games and continuous action spaces. [in preparation]
Collaborators

A set of states $S$, \[ \subseteq S \times A \times S, \]
Labelled Transition System - no probability

- A set of states $S$,
- a set of *labels* or *actions*, $L$ or $\mathcal{A}$ and
A set of states $S$,  

a set of *labels* or *actions*, $L$ or $A$ and  

a transition relation $\subseteq S \times A \times S$, usually written  

$$ \rightarrow_a \subseteq S \times S.$$  

The transitions could be indeterminate (nondeterministic).
A *discrete-time* Markov chain is a finite set $S$ (the state space) together with a transition probability function $T : S \times S \rightarrow [0, 1]$. 
A discrete-time Markov chain is a finite set $S$ (the state space) together with a transition probability function $T : S \times S \rightarrow [0, 1]$. The transition probability from $s$ to $s'$ only depends on $s$ and $s'$. 
Markov Chains

- A *discrete-time* Markov chain is a finite set $S$ (the state space) together with a transition probability function $T : S \times S \to [0, 1]$.
- The transition probability from $s$ to $s'$ only depends on $s$ and $s'$.
- This is what allows the probabilistic data to be given as a single matrix $T$. 
Discrete probabilistic transition systems

Just like a labelled transition system with probabilities associated with the transitions.
Discrete probabilistic transition systems

Just like a labelled transition system with probabilities associated with the transitions.

\[(S, L, \forall a \in L T_a : S \times S \rightarrow [0, 1])\]
Discrete probabilistic transition systems

- Just like a labelled transition system with probabilities associated with the transitions.

\[(S, L, \forall a \in L \ T_a : S \times S \rightarrow [0, 1])\]

- The model is reactive: All probabilistic data is internal - no probabilities associated with environment behaviour.
Examples of PTSs
Consider

\[
\begin{array}{c}
\begin{array}{c}
\text{Consider} \\
\text{Discrete probabilistic transition systems}
\end{array}
\end{array}
\]

\[
\begin{array}{c}
\begin{array}{c}
P_1
\end{array}
\end{array}
\]

\[
\begin{array}{c}
\begin{array}{c}
P_2
\end{array}
\end{array}
\]

\[
\begin{array}{c}
\begin{array}{c}
\text{Bisimulation for PTS: Larsen and Skou}
\end{array}
\end{array}
\]

Should \(s_0\) and \(t_0\) be bisimilar?

Yes, but we need to add the probabilities.
Consider

\[ P_1 \]

\[ P_2 \]

Should \( s_0 \) and \( t_0 \) be bisimilar?
Consider

Should $s_0$ and $t_0$ be bisimilar?

Yes, but we need to add the probabilities.
Let $S = (S, L, T_a)$ be a PTS. An equivalence relation $R$ on $S$ is a **bisimulation** if whenever $sRs'$, with $s, s' \in S$, we have that for all $a \in A$ and every $R$-equivalence class, $A$, $T_a(s, A) = T_a(s', A)$.
Let $S = (S, L, T_a)$ be a PTS. An equivalence relation $R$ on $S$ is a **bisimulation** if whenever $sRs'$, with $s, s' \in S$, we have that for all $a \in A$ and every $R$-equivalence class, $A$, $T_a(s, A) = T_a(s', A)$.

The notation $T_a(s, A)$ means “the probability of starting from $s$ and jumping to a state in the set $A$.”
Let $S = (S, L, T_a)$ be a PTS. An equivalence relation $R$ on $S$ is a **bisimulation** if whenever $s Rs'$, with $s, s' \in S$, we have that for all $a \in A$ and every $R$-equivalence class, $A$, $T_a(s, A) = T_a(s', A)$.

The notation $T_a(s, A)$ means “the probability of starting from $s$ and jumping to a state in the set $A$.”

Two states are bisimilar if there is some bisimulation relation $R$ relating them.
Labelled Markov processes are probabilistic versions of labelled transition systems. Labelled transition systems where the final state is governed by a probability distribution - no other indeterminacy.
What are labelled Markov processes?

- Labelled Markov processes are probabilistic versions of labelled transition systems. Labelled transition systems where the final state is governed by a probability distribution - no other indeterminacy.

- All probabilistic data is *internal* - no probabilities associated with environment behaviour.
What are labelled Markov processes?

- Labelled Markov processes are probabilistic versions of labelled transition systems. Labelled transition systems where the final state is governed by a probability distribution - no other indeterminacy.
- All probabilistic data is *internal* - no probabilities associated with environment behaviour.
- We observe the interactions - not the internal states.
What are labelled Markov processes?

- Labelled Markov processes are probabilistic versions of labelled transition systems. Labelled transition systems where the final state is governed by a probability distribution - no other indeterminacy.
- All probabilistic data is *internal* - no probabilities associated with environment behaviour.
- We observe the interactions - not the internal states.
- In general, the state space of a labelled Markov process may be a *continuum*.
Motivation

Model and reason about systems with *continuous* state spaces or continuous time evolution or both.

- Hybrid control systems; e.g. flight management systems.
Motivation

Model and reason about systems with *continuous* state spaces or continuous time evolution or both.

- Hybrid control systems; e.g. flight management systems.
- Telecommunication systems with spatial variation; e.g. cell phones.
Motivation

Model and reason about systems with *continuous* state spaces or continuous time evolution or both.

- Hybrid control systems; e.g. flight management systems.
- Telecommunication systems with spatial variation; e.g. cell phones.
- Performance modelling.
Motivation

Model and reason about systems with *continuous* state spaces or continuous time evolution or both.

- Hybrid control systems; e.g. flight management systems.
- Telecommunication systems with spatial variation; e.g. cell phones.
- Performance modelling.
- Continuous time systems.
Motivation

Model and reason about systems with continuous state spaces or continuous time evolution or both.

- Hybrid control systems; e.g. flight management systems.
- Telecommunication systems with spatial variation; e.g. cell phones.
- Performance modelling.
- Continuous time systems.
- Probabilistic programming languages with recursion or iteration.
Markov Kernels

A Markov kernel is a function $h : S \times \Sigma \rightarrow [0, 1]$ with (a) $h(s, \cdot) : \Sigma \rightarrow [0, 1]$ a (sub)probability measure and (b) $h(\cdot, A) : S \rightarrow [0, 1]$ a measurable function.
A **Markov kernel** is a function $h : S \times \Sigma \to [0, 1]$ with (a) $h(s, \cdot) : \Sigma \to [0, 1]$ a (sub)probability measure and (b) $h(\cdot, A) : S \to [0, 1]$ a measurable function.

Though apparently asymmetric, these are the stochastic analogues of binary relations.
Markov Kernels

A Markov kernel is a function $h : S \times \Sigma \rightarrow [0, 1]$ with (a) $h(s, \cdot) : \Sigma \rightarrow [0, 1]$ a (sub)probability measure and (b) $h(\cdot, A) : S \rightarrow [0, 1]$ a measurable function.

Though apparently asymmetric, these are the stochastic analogues of binary relations

and the uncountable generalization of a matrix.
Formal Definition of LMPs

- An LMP is a tuple \((S, \Sigma, L, \forall \alpha \in L. \tau_\alpha)\) where \(\tau_\alpha : S \times \Sigma \to [0, 1]\) is a transition probability function such that
Formal Definition of LMPs

An LMP is a tuple \((S, \Sigma, L, \forall \alpha \in L. \tau_\alpha)\) where \(\tau_\alpha : S \times \Sigma \rightarrow [0, 1]\) is a transition probability function such that

\[ \forall s : S. \lambda A : \Sigma. \tau_\alpha(s, A) \] is a subprobability measure and

\[ \forall A : \Sigma. \lambda s : S. \tau_\alpha(s, A) \] is a measurable function.
Larsen-Skou Bisimulation

Definition

Let \( S = (S, i, \Sigma, \tau) \) be a labelled Markov process. An equivalence relation \( R \) on \( S \) is a **bisimulation** if whenever \( sRs' \), with \( s, s' \in S \), we have that for all \( a \in A \) and every \( R \)-closed **measurable** set \( A \in \Sigma \),
\[
\tau_a(s, A) = \tau_a(s', A).
\]

Two states are bisimilar if they are related by a bisimulation relation.
The logic

\[ \mathcal{L} ::= T | \phi_1 \land \phi_2 | a_q \phi \]

We say \( s \models a_q \phi \) iff

\[ \exists A \in \Sigma. (\forall s' \in A. s' \models \phi) \land (\tau_a(s, A) > q). \]
The logic

\[ \mathcal{L} ::= T \mid \phi_1 \land \phi_2 \mid \langle a \rangle_q \phi \]

We say \( s \models \langle a \rangle_q \phi \) iff

\[ \exists A \in \Sigma. (\forall s' \in A. s' \models \phi) \land (\tau_a(s, A) > q). \]

The main theorem

Two systems are bisimilar iff they obey the same formulas of \( \mathcal{L} \). [DEP 1998 LICS, I and C 2002]
That cannot be right?

Two processes that cannot be distinguished without negation. The formula that distinguishes them is $\langle a \rangle (\neg \langle b \rangle \top)$. 
But it is!

We add probabilities to the transitions.
But it is!

We add probabilities to the transitions.

- If $p + q < r$ or $p + q > r$ we can easily distinguish them.
But it is!

We add probabilities to the transitions.

- If $p + q < r$ or $p + q > r$ we can easily distinguish them.
- If $p + q = r$ and $p > 0$ then $q < r$ so $\langle a \rangle_r \langle b \rangle_1 \top$ distinguishes them.
Proof idea

- Show that the relation “$s$ and $s'$ satisfy exactly the same formulas” is a bisimulation.
Proof idea

- Show that the relation “s and s’ satisfy exactly the same formulas” is a bisimulation.
- Can easily show that $\tau_a(s, A) = \tau_a(s', A)$ for $A$ of the form $[\phi]$. 
Proof idea

- Show that the relation “$s$ and $s'$ satisfy exactly the same formulas” is a bisimulation.
- Can easily show that $\tau_a(s, A) = \tau_a(s', A)$ for $A$ of the form $[\phi]$.
- Use Dynkin’s $\lambda - \pi$ theorem to show that we get a well defined measure on the $\sigma$-algebra generated by such sets and the above equality holds.
Proof idea

- Show that the relation “s and s’ satisfy exactly the same formulas” is a bisimulation.
- Can easily show that $\tau_a(s, A) = \tau_a(s', A)$ for $A$ of the form $[\phi]$.
- Use Dynkin’s $\lambda - \pi$ theorem to show that we get a well defined measure on the $\sigma$-algebra generated by such sets and the above equality holds.
- Use special properties of analytic spaces to show that this $\sigma$-algebra is the same as the original $\sigma$-algebra.
Let $R$ be a bisimulation relation on an LMP $(S, \Sigma, \tau_a)$. We prove by induction on $\phi$ that $\forall \phi \in \mathcal{L}$

$$\forall s, s' \in S. sR s' \Rightarrow s \models \phi \iff s' \models \phi.$$
The Easy Direction

Let $R$ be a bisimulation relation on an LMP $(S, \Sigma, \tau_a)$. We prove by induction on $\phi$ that $\forall \phi \in \mathcal{L}$

$$\forall s, s' \in S. sR s' \Rightarrow s \models \phi \iff s' \models \phi.$$
The Easy Direction

Let $R$ be a bisimulation relation on an LMP $(S, \Sigma, \tau_a)$. We prove by induction on $\phi$ that $\forall \phi \in \mathcal{L}$

$$\forall s, s' \in S. sRs' \Rightarrow s \models \phi \iff s' \models \phi.$$ 

Base case trivial.

∧ is obvious from Inductive Hypothesis.
The Easy Direction

- Let $R$ be a bisimulation relation on an LMP $(S, \Sigma, \tau_a)$. We prove by induction on $\phi$ that $\forall \phi \in \mathcal{L}$

$$\forall s, s' \in S. sR s' \Rightarrow s \models \phi \iff s' \models \phi.$$ 

- Base case trivial.
- $\land$ is obvious from Inductive Hypothesis.
- For $\phi = \langle a \rangle q \psi$ we have that $[\psi]$ is $R$-closed from inductive hypothesis. Thus

$$\tau_a(s, [\psi]) = \tau_a(s', [\psi])$$

and thus $sR s' \Rightarrow s \models \phi \iff s' \models \phi$. 
Digression on Analytic Spaces

- An analytic set $A$ is the image of a Polish space $X$ (or a Borel subset of $X$) under a continuous (or measurable) function $f : X \to Y$, where $Y$ is Polish. If $(S, \Sigma)$ is a measurable space where $S$ is an analytic set in some ambient topological space and $\Sigma$ is the Borel $\sigma$-algebra on $S$. 

Analytic sets do not form a $\sigma$-algebra but they are in the completion of the Borel algebra under any measure. [Universally measurable.] Regular conditional probability densities (disintegrations) can be defined on analytic spaces.
An analytic set $A$ is the image of a Polish space $X$ (or a Borel subset of $X$) under a continuous (or measurable) function $f : X \rightarrow Y$, where $Y$ is Polish. If $(S, \Sigma)$ is a measurable space where $S$ is an analytic set in some ambient topological space and $\Sigma$ is the Borel $\sigma$-algebra on $S$.

Analytic sets do not form a $\sigma$-algebra but they are in the completion of the Borel algebra under any measure. [Universally measurable.]
Digression on Analytic Spaces

- An analytic set $A$ is the image of a Polish space $X$ (or a Borel subset of $X$) under a continuous (or measurable) function $f : X \to Y$, where $Y$ is Polish. If $(S, \Sigma)$ is a measurable space where $S$ is an analytic set in some ambient topological space and $\Sigma$ is the Borel $\sigma$-algebra on $S$.

- Analytic sets do not form a $\sigma$-algebra but they are in the completion of the Borel algebra under any measure. [Universally measurable.]

- Regular conditional probability densities (disintegrations) can be defined on analytic spaces.
Amazing Facts about Analytic Spaces

Given $A$ an analytic space and $\sim$ an equivalence relation such that there is a *countable* family of real-valued measurable functions $f_i : S \to \mathbb{R}$ such that

$$\forall s, s' \in S. s \sim s' \iff \forall i. f_i(s) = f_i(s')$$

is called *smooth*.
Given $A$ an analytic space and $\sim$ an equivalence relation such that there is a *countable* family of real-valued measurable functions $f_i : S \to \mathbb{R}$ such that

$$\forall s, s' \in S. s \sim s' \iff \forall f_i.f_i(s) = f_i(s')$$

is called *smooth*.

If $\sim$ is smooth then the quotient space $(Q, \Omega)$ - where $Q = S/\sim$ and $\Omega$ is the finest $\sigma$-algebra making the canonical surjection $q : S \to Q$ measurable - is also analytic.
Given $A$ an analytic space and $\sim$ an equivalence relation such that there is a \textit{countable} family of real-valued measurable functions $f_i : S \to \mathbb{R}$ such that

$$\forall s, s' \in S. s \sim s' \iff \forall i. f_i(s) = f_i(s')$$

is called \textit{smooth}.

If $\sim$ is smooth then the quotient space $(Q, \Omega)$ - where $Q = S/ \sim$ and $\Omega$ is the finest $\sigma$-algebra making the canonical surjection $q : S \to Q$ measurable - is also analytic.

If an analytic space $(S, \Sigma)$ has a sub-$\sigma$-algebra $\Sigma_0$ of $\Sigma$ which separates points and is countably generated then $\Sigma_0$ is $\Sigma$! The Unique Structure Theorem (UST).
We have LMP \((S, \Sigma, L, \tau_a)\) and we want to quotient by \(\simeq\) where \(s \simeq s'\) if they agree on all formulas of the logic.

\[
\begin{array}{c}
(S, \Sigma, L, \tau_a) \\
\downarrow q \\
(S/ \simeq, \Sigma/ \simeq, L, \rho_a)
\end{array}
\]
The big picture

1. We have LMP \((S, \Sigma, L, \tau_a)\) and we want to quotient by \(\simeq\) where \(s \simeq s'\) if they agree on all formulas of the logic.

\[
\begin{array}{c}
(S, \Sigma, L, \tau_a) \\
\downarrow q \\
(S/ \simeq, \Sigma/ \simeq, L, \rho_a)
\end{array}
\]

2. We want to define \(\rho_a\) in such a way that

\[
\rho_a(q(s), B) = \tau_a(s, q^{-1}(B)).
\]
We have LMP \((S, \Sigma, L, \tau_a)\) and we want to quotient by \(\simeq\) where \(s \simeq s'\) if they agree on all formulas of the logic.

\[
(S, \Sigma, L, \tau_a) \xrightarrow{q} (S/\simeq, \Sigma/\simeq, L, \rho_a)
\]

We want to define \(\rho_a\) in such a way that

\[
\rho_a(q(s), B) = \tau_a(s, q^{-1}(B)).
\]

Why?
We have LMP \((S, \Sigma, L, \tau_a)\) and we want to quotient by \(\simeq\) where \(s \simeq s'\) if they agree on all formulas of the logic.

\[
\begin{array}{c}
(S, \Sigma, L, \tau_a) \\
\downarrow q \\
(S/ \simeq, \Sigma/ \simeq, L, \rho_a)
\end{array}
\]

We want to define \(\rho_a\) in such a way that

\[
\rho_a(q(s), B) = \tau_a(s, q^{-1}(B)).
\]

Why?

In lieu of an answer: maps between LMP’s satisfying the above condition are called “zigzags” and bisimulation can be defined as the existence of a span of zigzags.
Easy to check that $q^{-1}(q([\phi])) = [\phi]$:

$s \in q^{-1}(q([\phi]))$ implies that $q(s) \in q([\phi])$, i.e. $\exists s' \in [\phi]. s \simeq s'$, so $s \models \phi$ so $s \in [\phi]$. Thus $q([\phi])$ is measurable. Thus the $\sigma$-algebra generated—say, $\Lambda$—by $q([\phi])$ is a $\sigma$-algebra of $\Omega$. $\Lambda$ is countably generated and separates points so by UST it is $\Omega$. Thus $q([\phi])$ generates $\Omega$. The $\rho$ is well defined - I
\( \rho \) is well defined - 1

- Easy to check that \( q^{-1}(q(\llbracket \phi \rrbracket)) = \llbracket \phi \rrbracket \):
  \( s \in q^{-1}(q(\llbracket \phi \rrbracket)) \) implies that \( q(s) \in q(\llbracket \phi \rrbracket) \), i.e. \( \exists s' \in \llbracket \phi \rrbracket . s \simeq s' \), so \( s \models \phi \) so \( s \in \llbracket \phi \rrbracket \).

- Thus \( q(\llbracket \phi \rrbracket) \) is measurable.
\( \rho \) is well defined - I

- Easy to check that \( q^{-1}(q([\phi])) = [\phi] \):
  
  \[ s \in q^{-1}(q([\phi])) \text{ implies that } q(s) \in q([\phi]), \text{ i.e. } \exists s' \in [\phi]. s \simeq s', \text{ so } s \models \phi \text{ so } s \in [\phi]. \]

- Thus \( q([\phi]) \) is measurable.

- Thus the \( \sigma \)-algebra generated -say, \( \Lambda \) - by \( q([\phi]) \) is a sub-\( \sigma \)-algebra of \( \Omega \).
\( \rho \) is well defined - I

- Easy to check that \( q^{-1}(q([\phi])) = [\phi] \):
  
  \[ s \in q^{-1}(q([\phi])) \text{ implies that } q(s) \in q([\phi]), \text{ i.e. } \exists s' \in [\phi]. s \sim s', \text{ so } s \models \phi \text{ so } s \in [\phi]. \]

- Thus \( q([\phi]) \) is measurable.

- Thus the \( \sigma \)-algebra generated -say, \( \Lambda \) - by \( q([\phi]) \) is a sub-\( \sigma \)-algebra of \( \Omega \).

- \( \Lambda \) is countably generated and separates points so by UST it is \( \Omega \).

  Thus \( q([\phi]) \) generates \( \Omega \).
\( \rho \) is well defined - II

- The collection \( q([\phi]) \) is a \( \pi \)-system (because \( \mathcal{L}_0 \) has conjunction) and it generates \( \Omega \); thus if we can show that two measures agree on these sets they agree on all of \( \Omega \).
The collection $q([\phi])$ is a $\pi$-system (because $L_0$ has conjunction) and it generates $\Omega$; thus if we can show that two measures agree on these sets they agree on all of $\Omega$.

If $q(s) = q(s') = t$ then $\tau_a(s, [\phi]) = \tau_a(s', [\phi])$ (simple interpolation).
\( \rho \) is well defined - II

- The collection \( q([\phi]) \) is a \( \pi \)-system (because \( \mathcal{L}_0 \) has conjunction) and it generates \( \Omega \); thus if we can show that two measures agree on these sets they agree on all of \( \Omega \).
- If \( q(s) = q(s') = t \) then \( \tau_a(s, [\phi]) = \tau_a(s', [\phi]) \) (simple interpolation).
- Thus \( \tau_a(s, q^{-1}(q([\phi]))) = \tau_a(s', q^{-1}(q([\phi]))) \) and hence \( \rho \) is well defined. We have \( \rho_a(q(s), B) = \tau_a(s, q^{-1}(B)) \).
Let $X$ be any $\simeq$-closed subset of $\mathcal{S}$.
Let $X$ be any $\simeq$-closed subset of $S$.

Then $q^{-1}(q(X)) = X$ and $q(X) \in \Omega$. 


Let $X$ be any $\sim$-closed subset of $S$.

Then $q^{-1}(q(X)) = X$ and $q(X) \in \Omega$.

If $s \sim s'$ then $q(s) = q(s')$ and

$$
\tau_a(s, X) = \tau_a(s, q^{-1}(q(X))) = \rho_a(q(s), q(X)) = \\
\rho_a(q(s'), q(X)) = \tau_a(s', q^{-1}(q(X))) = \tau_a(s', X).
$$
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states \((s, t)\) are **not** bisimilar.
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states $(s, t)$ are not bisimilar.
- Spoiler move: Choose a measurable set $C$ and an action $a$ such that $\tau_a(x, C) \neq \tau_a(y, C)$ and $C$ is a bisimulation equivalence class.
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states \((s, t)\) are **not** bisimilar.

- Spoiler move: Choose a measurable set \(C\) and an action \(a\) such that \(\tau_a(x, C) \not= \tau_a(y, C)\) **and** \(C\) is a bisimulation equivalence class.

- Duplicator will deny that \(C\) is an equivalence class by choosing \(s' \in C\) and \(y' \not\in C\) and claiming that \((x', y')\) are bisimilar.
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states \((s, t)\) are **not** bisimilar.

- Spoiler move: Choose a measurable set \(C\) and an action \(a\) such that \(\tau_a(x, C) \neq \tau_a(y, C)\) and \(C\) is a bisimulation equivalence class.

- Duplicator will deny that \(C\) is an equivalence class by choosing \(s' \in C\) and \(y' \notin C\) and claiming that \((x', y')\) are bisimilar.

- Duplicator wins if she can go on forever or if Spoiler is stuck.
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states $(s, t)$ are not bisimilar.
- Spoiler move: Choose a measurable set $C$ and an action $a$ such that $\tau_a(x, C) \neq \tau_a(y, C)$ and $C$ is a bisimulation equivalence class.
- Duplicator will deny that $C$ is an equivalence class by choosing $s' \in C$ and $y' \notin C$ and claiming that $(x', y')$ are bisimilar.
- Duplicator wins if she can go on forever or if Spoiler is stuck.
- Spoiler can only win if Duplicator is stuck. For example if $C$ is all of $S$. 
Rules of the game

- Spoiler/duplicator game. Spoiler tries to show that a pair of states \((s, t)\) are **not** bisimilar.
- Spoiler move: Choose a measurable set \(C\) and an action \(a\) such that \(\tau_a(x, C) \neq \tau_a(y, C)\) and \(C\) is a bisimulation equivalence class.
- Duplicator will deny that \(C\) is an equivalence class by choosing \(s' \in C\) and \(y' \notin C\) and claiming that \((x', y')\) are bisimilar.
- Duplicator wins if she can go on forever or if Spoiler is stuck.
- Spoiler can only win if Duplicator is stuck. For example if \(C\) is all of \(S\).
- \(s\) and \(t\) are bisimilar if and only if Duplicator has a winning strategy.
Let $S = (S, \Sigma, \tau)$ be a labelled Markov process. A preorder $R$ on $S$ is a simulation if whenever $sRs'$, we have that for all $a \in A$ and every $R$-closed measurable set $A \in \Sigma$, $\tau_a(s, A) \leq \tau_a(s', A)$. We say $s$ is simulated by $s'$ if $sRs'$ for some simulation relation $R$. 
Logic for simulation?

- The logic used in the characterization has no negation, not even a limited negative construct.
Logic for simulation?

- The logic used in the characterization has no negation, not even a limited negative construct.
- One can show that if \( s \) simulates \( s' \) then \( s \) satisfies all the formulas of \( \mathcal{L} \) that \( s' \) satisfies.
Logic for simulation?

- The logic used in the characterization has no negation, not even a limited negative construct.
- One can show that if $s$ simulates $s'$ then $s$ satisfies all the formulas of $\mathcal{L}$ that $s'$ satisfies.
- What about the converse?
Counter example!

In the following picture, \( t \) satisfies all formulas of \( \mathcal{L} \) that \( s \) satisfies but \( t \) does not simulate \( s \).

All transitions from \( s \) and \( t \) are labelled by \( a \).
A formula of $\mathcal{L}$ that is satisfied by $t$ but not by $s$.

$$\langle a \rangle_0 (\langle a \rangle_0 T \land \langle b \rangle_0 T).$$
Counter example (contd.)

- A formula of $\mathcal{L}$ that is satisfied by $t$ but not by $s$:
  $$\langle a \rangle_0 (\langle a \rangle_0 T \land \langle b \rangle_0 T).$$

- A formula with disjunction that is satisfied by $s$ but not by $t$:
  $$\langle a \rangle_{\frac{3}{4}} (\langle a \rangle_0 T \lor \langle b \rangle_0 T).$$
A logical characterization for simulation

- The logic $\mathcal{L}$ does **not** characterize simulation. One needs disjunction.

$$\mathcal{L}_\lor := \mathcal{L} \vert \phi_1 \lor \phi_2.$$
A logical characterization for simulation

- The logic $\mathcal{L}$ does not characterize simulation. One needs disjunction.

\[
\mathcal{L}_\vee := \mathcal{L} \mid \phi_1 \vee \phi_2.
\]

- With this logic we have:
  An \textbf{LMP} $s_1$ simulates $s_2$ if and only if for every formula $\phi$ of $\mathcal{L}_\vee$ we have
  \[
s_1 \models \phi \Rightarrow s_2 \models \phi.
  \]
A logical characterization for simulation

- The logic $\mathcal{L}$ does not characterize simulation. One needs disjunction.

$$\mathcal{L}_\lor := \mathcal{L} | \phi_1 \lor \phi_2.$$ 

- With this logic we have:
  An LMP $s_1$ simulates $s_2$ if and only if for every formula $\phi$ of $\mathcal{L}_\lor$ we have

  $$s_1 \models \phi \Rightarrow s_2 \models \phi.$$ 

- The original proof uses domain theory and only works for finitely many labels.
A logical characterization for simulation

- The logic $\mathcal{L}$ does **not** characterize simulation. One needs disjunction.

$$\mathcal{L}_\lor := \mathcal{L} | \phi_1 \lor \phi_2.$$  

- With this logic we have:
  - An **LMP** $s_1$ simulates $s_2$ if and only if for every formula $\phi$ of $\mathcal{L}_\lor$ we have
    $$s_1 \models \phi \Rightarrow s_2 \models \phi.$$  

- The original proof uses domain theory and only works for finitely many labels.

- New proof, with Nathanaël Fijalkow and Bartek Klin, works with countably many labels and uses topology.
Other Logics

\[
\begin{align*}
\mathcal{L}_{\text{Can}} & := \mathcal{L}_0 \mid \text{Can}(a) \\
\mathcal{L}_\Delta & := \mathcal{L}_0 \mid \Delta_a \\
\mathcal{L}_\neg & := \mathcal{L}_0 \mid \neg \phi \\
\mathcal{L}_\lor & := \mathcal{L}_0 \mid \phi_1 \lor \phi_2 \\
\mathcal{L}_\land & := \mathcal{L}_\neg \mid \bigwedge_{i \in \mathbb{N}} \phi_i
\end{align*}
\]

where

\[
\begin{align*}
s \models \text{Can}(a) & \quad \text{to mean that } \tau_a(s, S) > 0; \\
s \models \Delta_a & \quad \text{to mean that } \tau_a(s, S) = 0.
\end{align*}
\]

We need \(\mathcal{L}_\lor\) to characterise simulation.
Concluding remarks

Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
- There is a logical characterisation of simulation.
Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
- There is a logical characterisation of simulation.
- There is a “metric” on LMPs which is based on this logic.
Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
- There is a logical characterisation of simulation.
- There is a “metric” on LMPs which is based on this logic.
- Why did the proof require so many subtle properties of analytic spaces? There is a more general definition of bisimulation for which the logical characterisation proof is “easy” but to prove that that definition coincides with this one in analytic spaces requires roughly the same proof as that given here.

Recently, Fijalkow showed that if there are uncountably many labels then the logical characterization of bisimulation fails. However, if we introduce a topology on the space of labels and a continuity assumption, we can regain the logical characterization result.
Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
- There is a logical characterisation of simulation.
- There is a “metric” on LMPs which is based on this logic.
- Why did the proof require so many subtle properties of analytic spaces? There is a more general definition of bisimulation for which the logical characterisation proof is “easy” but to prove that that definition coincides with this one in analytic spaces requires roughly the same proof as that given here.
- Recently, Fijalkow showed that if there are uncountably many labels then the logical characterization of bisimulation fails.
Concluding remarks

Conclusions

- Strong probabilistic bisimulation is characterised by a very simple modal logic with no negative constructs.
- There is a logical characterisation of simulation.
- There is a “metric” on LMPs which is based on this logic.
- Why did the proof require so many subtle properties of analytic spaces? There is a more general definition of bisimulation for which the logical characterisation proof is “easy” but to prove that that definition coincides with this one in analytic spaces requires roughly the same proof as that given here.
- Recently, Fijalkow showed that if there are uncountably many labels then the logical characterization of bisimulation fails.
- However, if we introduce a topology on the space of labels and a continuity assumption, we can regain the logical characterization result.