The Stochastic Shortest Path Problem: 
A Polyhedral Perspective

Matthieu Guillot ¹  Gautier Stauffer ¹

¹G-SCOP, Univ. Grenoble Alpes, 38000 Grenoble, France

London School of Economics, January 2017
Outline of the talk

- Infinite horizon total cost MDP
Outline of the talk

- Infinite horizon total cost MDP
- The Stochastic Shortest Path Problem
Outline of the talk

- Infinite horizon total cost MDP
- The Stochastic Shortest Path Problem
- Contributions
Outline of the talk

- Infinite horizon total cost MDP
- The Stochastic Shortest Path Problem
- Contributions
- Main proof technique: Generalized flow decomposition theorem
Outline of the talk

- Infinite horizon total cost MDP
- The Stochastic Shortest Path Problem
- Contributions
- Main proof technique: Generalized flow decomposition theorem
- Open Questions
Infinite horizon Markov Decision Process

- Entries:
  - A finite set of states $S$
  - A finite set of actions $A$
  - A cost function on the actions $c : A \rightarrow R$
  - Conditional probabilities over the state space for each action $P(\cdot | a)$
  - An initial state $s_0$
Infinite horizon Markov Decision Process

- Entries:
  - $S$ a finite set of states
  - $A = \bigcup_{s \in S} A(s)$ a finite set of actions
  - $c : A \mapsto \mathbb{R}$, a cost function on the actions
  - $P(\cdot | a)$, conditional probabilities over the state space for each action
  - An initial state $s_0$. 

Guillot and Stauffer

The Stochastic Shortest Path Problem
Infinite horizon Markov Decision Process

- Entries:
  - \( S \) a finite set of states
  - \( A = \bigcup_{s \in S} A(s) \) a finite set of actions
  - Cost function on the actions \( c : A \rightarrow \mathbb{R} \)
  - Conditional probabilities over the state space for each action \( P(\cdot|a) \)
  - Initial state \( s_0 \)
Infinite horizon Markov Decision Process

Entries:
- \( S \) a finite set of states
- \( A = \bigcup_{s \in S} A(s) \) a finite set of actions

\[ c : A \to R \], a cost function on the actions

\[ P(\cdot | a) \], conditional probabilities over the state space for each action

An initial state \( s_0 \).
Infinite horizon Markov Decision Process

Entries:
- $\mathcal{S}$ a finite set of states
- $\mathcal{A} = \bigcup_{s \in \mathcal{S}} \mathcal{A}(s)$ a finite set of actions
- $c : \mathcal{A} \mapsto \mathbb{R}$, a cost function on the actions
Infinite horizon Markov Decision Process

Entries:

- $\mathcal{S}$ a finite set of states
- $\mathcal{A} = \bigcup_{s \in \mathcal{S}} \mathcal{A}(s)$ a finite set of actions
- $c : \mathcal{A} \mapsto \mathbb{R}$, a cost function on the actions
- $P(\cdot | a)$, conditional probabilities over the state space for each action $a$
Infinite horizon Markov Decision Process

- Entries:
  - $\mathcal{S}$ a finite set of states
  - $\mathcal{A} = \bigcup_{s \in \mathcal{S}} \mathcal{A}(s)$ a finite set of actions
  - $c : \mathcal{A} \mapsto \mathbb{R}$, a cost function on the actions
  - $P(\cdot | a)$, conditional probabilities over the state space for each action $a$
Infinite horizon Markov Decision Process

- Entries:
  - $\mathcal{S}$ a finite set of states
  - $\mathcal{A} = \bigcup_{s \in \mathcal{S}} \mathcal{A}(s)$ a finite set of actions
  - $c : \mathcal{A} \mapsto \mathbb{R}$, a cost function on the actions
  - $P(\cdot | a)$, conditional probabilities over the state space for each action $a$
  - An initial state $s_0$. 

---

The Stochastic Shortest Path Problem

Guillot and Stauffer

LSE 2017 3 / 18
Infinite horizon Markov Decision Process

- Dynamics:

![Graph Illustrating Infinite Horizon MDP]

- State transitions and action probabilities are visualized, with each state connected by arrows indicating possible actions and their respective transition probabilities.
Dynamics:

In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$. 
Infinite horizon Markov Decision Process

- **Dynamics:**
  - In each time period \( t \geq 0 \), the system is in state \( s_t \) and we need to decide upon an action \( a \) available in \( \mathcal{A}(s_t) \).
Dynamics:

In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$. 
Dynamics:

- In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$. 

![Diagram of an Infinite horizon Markov Decision Process](image-url)
Infinite horizon Markov Decision Process

Dynamics:

- In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$.
- The system evolves to state $s_{t+1}$ according to $P(\cdot|a)$. 

![Diagram of Markov Decision Process]
Dynamics:

- In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $\mathcal{A}(s_t)$.
- The system evolves to state $s_{t+1}$ according to $P(\cdot|a)$. 
Dynamics:

- In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$.
- The system evolves to state $s_{t+1}$ according to $P(\cdot|a)$. 

![Diagram of a Markov Decision Process]
Dynamics:

- In each time period $t \geq 0$, the system is in state $s_t$ and we need to decide upon an action $a$ available in $A(s_t)$.
- The system evolves to state $s_{t+1}$ according to $P(\cdot|a)$. 

![Diagram of the Infinite horizon Markov Decision Process](image-url)
Infinite horizon (total cost) Markov Decision Process

- Goal:

\[ \pi : S \rightarrow A \] (It defines a Markov Chain with transition matrix \( P^{\pi} \)).

\[ \min \sum_{t=0}^{\infty} c^{\pi}(s_0) \]
Goal:
- Find a policy \( \pi : S \mapsto A \)

Infinite horizon (total cost) Markov Decision Process

**Graph:**
- States: 1, 2, 3, 4, 5, 6, 7
- Actions: a, b, c, d, e, f
- Transitions and Costs
  - \( s_1 \) to \( s_2 \): Cost 7, Transition 0.2
  - \( s_2 \) to \( s_3 \): Cost 3, Transition 0.7
  - \( s_3 \) to \( s_4 \): Cost 4, Transition 0.1
  - \( s_4 \) to \( s_5 \): Cost 1, Transition 0.9
  - \( s_5 \) to \( s_6 \): Cost 10, Transition 0.8
  - \( s_6 \) to \( s_7 \): Cost 0, Transition 0.9

**Notes:**
- We might consider non-stationary and non-deterministic policies, but for most MDPs, ‘pure’ policies are optimal.
Goal:

- Find a policy \( \pi : S \mapsto A \)

(It defines a Markov Chain with transition matrix \( P_\pi \)).
Goal:

- Find a policy $\pi: S \mapsto A$
  (It defines a Markov Chain with transition matrix $P_\pi$).
- Minimizing $\sum_{k=0}^{+\infty} 1_{s_0} (P_\pi)^k c_\pi$

Infinite horizon (total cost) Markov Decision Process

Guillot and Stauffer

The Stochastic Shortest Path Problem

LSE 2017 5 / 18
Infinite horizon (total cost) Markov Decision Process

- **Goal:**
  - Find a policy \( \pi : S \mapsto A \)
    (It defines a Markov Chain with transition matrix \( P_\pi \)).
  - Minimizing \( \sum_{k=0}^{+\infty} \mathbb{1}_{s_0}(P_\pi)^k c_\pi \)

- **NB:** we might consider non stationary and non deterministic policies BUT for most MDPs ‘pure’ policies are optimal
Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} \mathbb{1}_{s_0}^t (P_\pi)^k c_\pi \) is not always defined

- **Discounted models**:
  \[
  V^*_{s_0} := \min \sum_{k=0}^{+\infty} \alpha^k (P_\pi)^k c_\pi
  \text{for some } 0 \leq \alpha < 1
  \]

- **Standards Methods from the 50's**:
  - Value Iteration: Bellman (1957)
  - Dynamic Programming
  - Policy Iteration: Howard (1960)
  - Block-Pivot Simplex algorithm
  - Linear Programming: Manne (1960)
**Discounted Markov Decision Process**

- **Issue**: \( \sum_{k=0}^{+\infty} 1_{s_0} (P_{\pi})^k c_{\pi} \) is not always defined

- **Discounted models**: \( V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k 1_{s_0} (P_{\pi})^k c_{\pi} \) for some \( 0 \leq \alpha < 1 \)
Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} \mathbb{1}_{s_0} (P_{\pi})^k c_\pi \) is not always defined

- **Discounted models**: \( V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k \mathbb{1}_{s_0} (P_{\pi})^k c_\pi \) for some \( 0 \leq \alpha < 1 \)

- **Standards Methods from the 50’s**:
  - Value Iteration: Bellman (1957)
  - Dynamic Programming
  - Policy Iteration: Howard (1960)
  - Block-Pivot Simplex algorithm
  - Linear Programming: Manne (1960)
Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} 1_{s_0}(P_{\pi})^k c_{\pi} \) is not always defined

- **Discounted models**: 
  \[ V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k 1_{s_0}(P_{\pi})^k c_{\pi} \text{ for some } 0 \leq \alpha < 1 \]

- **Standards Methods from the 50’s**: 
  \[ V^*(s) = \min_{a \in A(s)} \{ c(a) + \alpha \sum_{s'} P(s'|a) \cdot V^*(s') \} \]
### Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} 1_{s_0}^{t} (P_{\pi})^k c_{\pi} \) is not always defined

- **Discounted models**: \( V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k 1_{s_0}^{t} (P_{\pi})^k c_{\pi} \) for some \( 0 \leq \alpha < 1 \)

- **Standards Methods from the 50’s**:
  
  \[
  V^*(s) = \min_{a \in A(s)} \{ c(a) + \alpha \sum_{s'} P(s'|a) \cdot V^*(s') \}
  \]

- **Value Iteration**: Bellman (1957) Dynamic Programming
Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} \mathbb{1}_{s_0} (P^k \pi)_k c_{\pi} \) is not always defined

- **Discounted models**: \( V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k \mathbb{1}_{s_0} (P^k \pi)_k c_{\pi} \) for some \( 0 \leq \alpha < 1 \)

- **Standards Methods from the 50’s**: 
  \[
  V^*(s) = \min_{a \in A(s)} \{ c(a) + \alpha \sum_{s'} P(s'|a) \cdot V^*(s') \}
  \]

  - **Value Iteration**: Bellman (1957) Dynamic Programming
  - **Policy Iteration**: Howard (1960) Block-Pivot Simplex algorithm
Discounted Markov Decision Process

- **Issue**: \( \sum_{k=0}^{+\infty} \mathbb{1}_{s_0} (P_{\pi})^k c_{\pi} \) is not always defined

- **Discounted models**: \( V^*(s_0) := \min \sum_{k=0}^{+\infty} \alpha^k \mathbb{1}_{s_0} (P_{\pi})^k c_{\pi} \) for some \( 0 \leq \alpha < 1 \)

- **Standards Methods from the 50’s**:
  \[
  V^*(s) = \min_{a \in \mathcal{A}(s)} \{ c(a) + \alpha \sum_{s'} P(s'|a) \cdot V^*(s') \}
  \]

- **Value Iteration**: Bellman (1957) Dynamic Programming
- **Policy Iteration**: Howard (1960) Block-Pivot Simplex algorithm
- **Linear Programming**: Manne (1960)
Extension to **undiscounted** MDPs i.e. $\alpha = 1$ (discounted case is special case)
The Stochastic Shortest Path Problem

- Extension to undiscounted MDPs i.e. $\alpha = 1$ (discounted case is special case)
- Bertsekas and Tsitsiklis 1991: Value Iteration, Policy Iteration, LP all work

Hypothesis:
- There is an identified target state $T$ (from there no way to escape)
- There is a proper policy that leads to $T$ with probability 1
- 'Looping' in the system (outside $T$) is costly: $+\infty$ cost
The Stochastic Shortest Path Problem

- Extension to undiscounted MDPs i.e. $\alpha = 1$ (discounted case is special case)
- Bertsekas and Tsitsiklis 1991: Value Iteration, Policy Iteration, LP all work
- Hypothesis:
  - there is an identified target state $T$ (from there no way to escape)
  - there is a proper policy that leads to $T$ with probability 1
  - 'Looping' in the system (outside $T$) is costly: $+\infty$ cost

![Diagram of the stochastic shortest path problem](image)

- $s$: state
- $a$: action
- $a = p(s|a)$
- $s \rightarrow c(a) \rightarrow a$
The Stochastic Shortest Path Problem

- Extension to **undiscounted** MDPs i.e. $\alpha = 1$ (discounted case is special case)
- Bertsekas and Tsitsiklis 1991: Value Iteration, Policy Iteration, LP all work
- Hypothesis:
  - there is an identified target state $T$ (from there no way to escape)
The Stochastic Shortest Path Problem

- Extension to undiscounted MDPs i.e. $\alpha = 1$ (discounted case is special case)
- Bertsekas and Tsitsiklis 1991: Value Iteration, Policy Iteration, LP all work

Hypothesis:
- there is an identified target state $T$ (from there no way to escape)
- there is a proper policy that lead to $T$ with proba 1
The Stochastic Shortest Path Problem

- Extension to undiscounated MDPs i.e. $\alpha = 1$ (discounted case is special case)
- Bertsekas and Tsitsiklis 1991: Value Iteration, Policy Iteration, LP all work
- Hypothesis:
  - there is an identified target state $T$ (from there no way to escape)
  - there is a proper policy that lead to $T$ with proba 1
  - ‘looping’ in the system (outside $T$) is costly: $+\infty$ cost
The Stochastic Shortest Path Problem

Almost an extension of the standard deterministic shortest path:

- An identified target state \( T \) (from there no way to escape)
- A proper policy that leads to \( T \) with probability 1
- "Looping" in the system (outside \( T \)) is costly: \(+\infty\) cost

NB: Bertsekas and Yu (2016) proved that perturbed versions of PI and VI converge in the presence of zero cost cycles.
Almost an extension of the standard deterministic shortest path:

- There is an identified target state $T$ (from there no way to escape).
- There is a proper policy that leads to $T$ with probability 1.
- Looping in the system (outside $T$) is costly: $+\infty$ cost.

This forbids zero cost cycles.

NB: Bertsekas and Yu (2016) proved that perturbed versions of PI and VI converge in the presence of zero cost cycles.
Almost an extension of the standard deterministic shortest path:
- there is an identified target state $T$ (from there no way to escape)

NB: Bertsekas and Yu (2016) proved that perturbed versions of PI and VI converge in the presence of zero cost cycles.
Almost an extension of the standard deterministic shortest path:
- there is an identified target state $T$ (from there no way to escape)
- there is a proper policy that lead to $T$ with proba 1

NB: Bertsekas and Yu (2016) proved that perturbated version of PI and VI converge in the presence of zero cost cycles.
Almost an extension of the standard deterministic shortest path:

- there is an identified target state $T$ (from there no way to escape)
- there is a proper policy that lead to $T$ with proba 1
- ‘looping’ in the system (outside $T$) is costly: $+\infty$ cost
  $\rightarrow$ this forbids zero cost cycles

NB: Bertsekas and Yu (2016) proved that perturbated version of PI and VI converge in the presence of zero cost cycles.
The Stochastic Shortest Path Problem

Almost an extension of the standard deterministic shortest path:
- there is an identified target state $T$ (from there no way to escape)
- there is a proper policy that lead to $T$ with proba 1
- ‘looping’ in the system (outside $T$) is costly: $+\infty$ cost
  $\rightarrow$ this forbids zero cost cycles

\[ \begin{array}{ccc}
1 & 7 & 3 \\
5 & 3 & 2 \\
2 & 2 & 10 \\
\end{array} \]

$T$

NB: Bertsekas and Yu (2016) proved that perturbated version of PI and VI converge in the presence of zero cost cycles.
This is not only a technical problem!

- Many applications with zero cost cycles!
- Maximizing the probability of reaching a target
- Ex: Robot motion planning in turbulent water
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
- A proof that we can actually restrict to ‘pure’ policies
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
- A proof that we can actually restrict to ‘pure’ policies
- Proof of convergence of Value Iteration by a simple analysis: a natural extension of Bellman-Ford
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
- A proof that we can actually restrict to ‘pure’ policies
- Proof of convergence of Value Iteration by a simple analysis: a natural extension of Bellman-Ford
- Proof that Policy Iteration converges
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
- A proof that we can actually restrict to ‘pure’ policies
- Proof of convergence of Value Iteration by a simple analysis: a natural extension of Bellman-Ford
- Proof that Policy Iteration converges
- A generalization of Dijkstra’s algorithm through primal-dual
Our Contribution

- A Generalization of the framework by Bertsekas and Tsitsiklis that encapsulates the deterministic version (i.e. zero cost cycles)
- A proof that we can actually restrict to ‘pure’ policies
- Proof of convergence of Value Iteration by a simple analysis: a natural extension of Bellman-Ford
- Proof that Policy Iteration converges
- A generalization of Dijkstra’s algorithm through primal-dual

→ Simplifies, Improves and Extends all previous results and analysis for infinite horizon total cost MDPs!
Our technique: polyhedral analysis

- Observation that the (dual of the) linear programming formulation for SSP is a natural relaxation of a more general problem

→ The corresponding polyhedra generalizes the network flow polyhedra

\[
\begin{align*}
\min & \quad cx \\
\sum_{a \in \delta^+(v)} x(a) & - \sum_{a \in \delta^-(v)} x(a) = \begin{cases} 
1, & \text{if } v = s \\
-1, & \text{if } v = t \\
0, & \text{otherwise}
\end{cases}, \forall v \in V \\
x & \geq 0
\end{align*}
\]
Our technique: polyhedral analysis

- Observation that the (dual of the) linear programming formulation for SSP is a natural relaxation of a more general problem

→ The corresponding polyhedra generalizes the network flow polyhedra
Our technique: polyhedral analysis

- Observation that the (dual of the) linear programming formulation for SSP is a natural relaxation of a more general problem.
  - The corresponding polyhedra generalizes the network flow polyhedra.

\[
\begin{align*}
\min & \quad cx \\
\sum_{a \in \mathcal{A}(s)} x(a) - \sum_{a \in \mathcal{A}} p(s|a)x(a) &= \begin{cases} 
1, & \text{if } s = s_0 \\
-1, & \text{if } s = T \\
0, & \text{otherwise}
\end{cases} \\
x &\geq 0
\end{align*}
\]
Our technique: polyhedral analysis

- Observation that the (dual of the) linear programming formulation for SSP is a natural relaxation of a more general problem
  → The corresponding polyhedra generalizes the network flow polyhedra
Linear Programming relaxation: proof sketch

- A policy $\pi$ induces a probability distribution over all possible $(s_0, T)$-walks
- $y^\pi_k(s)$: probability of being in state $s$ in period $k$ following policy $\pi$
- $x^\pi_k(a)$: probability of taking action $a$ in period $k$ following policy $\pi$
- We have for all $\pi$ and for all $k \geq 0$:
  \[
  \sum_{a \in \mathcal{A}(s)} x^\pi_k(a) = y^\pi_k(s) \text{ and } y^\pi_{k+1}(s) = \sum_{j \in \mathcal{A}} p(s|a)x^\pi_k(a)
  \]
- It implies $\sum_k \sum_{a \in \mathcal{A}(s)} x^\pi_{k+1}(a) = \sum_k \sum_{a \in \mathcal{A}} p(s|a)x^\pi_k(a)$
- Together with $y^\pi_0 = 1_{s_0} = \sum_{a \in \mathcal{A}(s)} x^\pi_0(a)$ this yields
  \[
  \sum_{a \in \mathcal{A}(s)} x^\pi(a) - \sum_{a \in \mathcal{A}} p(s|a)x^\pi(a) = 1_{s_0}
  \]
as long as $x^\pi(a) := \sum_k x^\pi_k(a)$ is well-defined for all $a$ (this is our new def. of proper)
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)

→ The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)

→ The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
- The proof relies on a generalization of the ‘flow’ decomposition theorem

![Graph diagram](image-url)
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)

→ The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem

Diagram:

1 → 2 → 3 → 4 → T
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)

→ The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are 'associated' with 'pure' policies (NB: the extreme points are NOT integral)
- The proof relies on a generalization of the 'flow' decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are 'associated' with 'pure' policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the 'flow' decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)

→ The proof relies on a generalization of the ‘flow’ decomposition theorem
Our technique: polyhedral analysis

- Proof that the extreme points of this relaxation are ‘associated’ with ‘pure’ policies (NB: the extreme points are NOT integral)
  → The proof relies on a generalization of the ‘flow’ decomposition theorem.
Idea of contributions: framework

- Decomposition theorem implies that extreme points are ‘pure’ strategies and extreme rays of the relaxation are ‘transition cycles’.
Idea of contributions: framework

- Decomposition theorem implies that extreme points are ‘pure’ strategies and extreme rays of the relaxation are ‘transition cycles’
- A transition cycle is a solution $x \geq 0$ to $\sum_{a \in A(s)} x(a) - \sum_{a \in A} p(s|a)x(a) = 0$

Assumptions

- There exists a path between all node $i$ and 0 in the support graph
- There is no negative cost transition cycle
Idea of contributions: framework

- Decomposition theorem implies that extreme points are ‘pure’ strategies and extreme rays of the relaxation are ‘transition cycles’
- A transition cycle is a solution $x \geq 0$ to $\sum_{a \in A(s)} x(a) - \sum_{a \in A} p(s|a)x(a) = 0$
- The optimum of the relaxation and of the original problem coincide when no transition cycles of negative costs: this is our new framework
Idea of contributions: framework

- Decomposition theorem implies that extreme points are ‘pure’ strategies and extreme rays of the relaxation are ‘transition cycles’
- A transition cycle is a solution $x \geq 0$ to $\sum_{a \in A(s)} x(a) - \sum_{a \in A} p(s|a)x(a) = 0$
- The optimum of the relaxation and of the original problem coincide when no transition cycles of negative costs: this is our new framework

Assumptions

- There exists a path between all node $i$ and 0 in the support graph
- There is no negative cost transition cycle
Idea of contributions: algorithms

- Value iteration is very similar to Bellman-Ford: we essentially prove that
  \[ \min \lim_{\mathcal{P}} \lim_{K \to \infty} \sum_{k=0}^{K} c^T x_k^\mathcal{P} = \lim_{K \to \infty} \min_{\mathcal{P}_K} \sum_{k=0}^{K} c^T x_k^\mathcal{P} \]

  \((\mathcal{P} \sim \text{all proper policies}, \mathcal{P}_K \sim \text{all proper policies that terminate in } K \text{ steps})\)
Idea of contributions: algorithms

- **Value iteration** is very similar to Bellman-Ford: we essentially prove that
  \[
  \min \lim_{\Pi \in \mathcal{P}} \sum_{k=0}^{K} c^T x^\Pi_k = \lim_{K \to \infty} \min_{\Pi \in \mathcal{P}_K} \sum_{k=0}^{K} c^T x^\Pi_k
  \]
  (\(\mathcal{P}\) \(\sim\) all proper policies, \(\mathcal{P}_K\) \(\sim\) all proper policies that terminate in \(K\) steps)

- **Policy iteration** is a block-pivot simplex: we prove strict improvement to guarantee finiteness.
Idea of contributions : algorithms

- **Value iteration is very similar to Bellman-Ford**: we essentially prove that
  \[ \min \lim_{K \to \infty} \sum_{k=0}^{K} c^T x_k^\pi = \lim_{K \to \infty} \min_{\pi \in \mathcal{P}_K} \sum_{k=0}^{K} c^T x_k^\pi \]

  (\( \mathcal{P} \sim \) all proper policies, \( \mathcal{P}_K \sim \) all proper policies that terminate in \( K \) steps)

- **Policy iteration is a block-pivot simplex**: we prove strict improvement to guarantee finiteness.

- **We can apply a primal-dual algorithm**, the subproblem is a reachability question: Dijkstra-like algorithm (we fall into the same class, not the case before because of zero cost cycles !!)
Main Open questions

- The stochastic shortest path problem is polynomial through LP

Ye (2011): true for discounted MDPs if \( \alpha \) is fixed

Is our generalization of Dijkstra's algorithm strongly polynomial?

Is the reachability subproblem strongly polynomial?

Guillot and Stauffer

The Stochastic Shortest Path Problem
Main Open questions

- The stochastic shortest path problem is polynomial through LP
- Is it strongly polynomial?
Main Open questions

- The stochastic shortest path problem is polynomial through LP
- Is it strongly polynomial?
- Ye (2011) : true for discounted MDPs if $\alpha$ is fixed
Main Open questions

- The stochastic shortest path problem is polynomial through LP
- Is it strongly polynomial?
- Ye (2011): true for discounted MDPs if $\alpha$ is fixed
- Is our generalization of Dijkstra’s algorithm strongly polynomial?
Main Open questions

- The stochastic shortest path problem is polynomial through LP
- Is it strongly polynomial?
- Ye (2011): true for discounted MDPs if $\alpha$ is fixed
- Is our generalization of Disjkstra’s algorithm strongly polynomial?
- Is the reachability subproblem strongly polynomial?