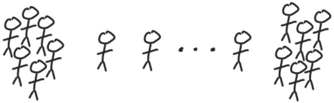
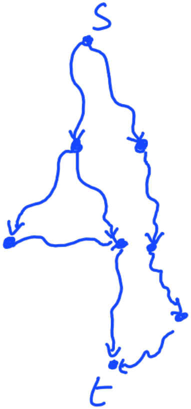
PaVeDys
Mai 2025

Parameterized Flows

Joint work with Hugo Gimbert
and Patrick Totzke



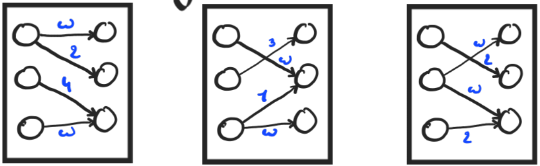


Model

Finite set of states S

q_1
q_2
q_3

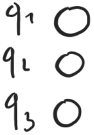
Finite set of tiles $\mathcal{T}$



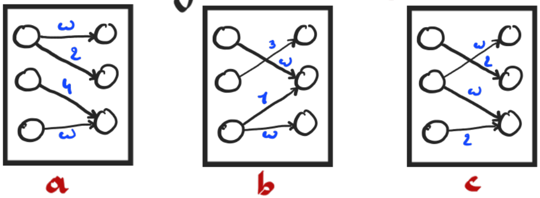
a b c

Model

Finite set of states S

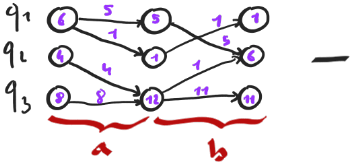


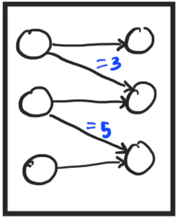
Finite set of tiles $\mathcal{T}$



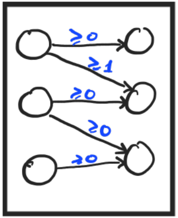
Configurations = $\mathbb{N}^S$

Step : transfer of tokens $\leq x \in \mathcal{T}$

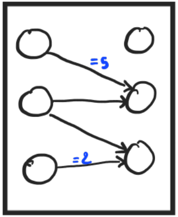




$\approx$ VASS

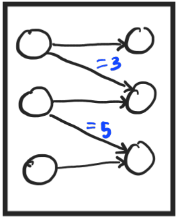


$\approx$ Lossy Broadcast.

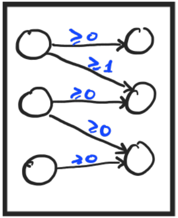


$\approx$ VASS with 0-test

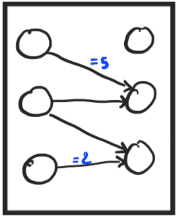
$\approx$ Reliable Broadcast



$\approx$ VASS

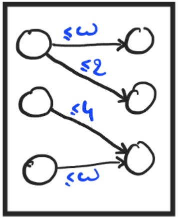


$\approx$ Lossy Broadcast.



$\approx$ VASS with 0-test

$\approx$ Reliable Broadcast



$\approx$ Locks ?

Sequential Flow Problem [Colcombet, Fijalkow, Ohlmann '20]

Input: Finite set of tiles $\tau \subseteq (\mathbb{N} \cup \{\omega\})^{S \times S}$

Source state $src \in S$

Target state $tgt \in S$

Question: $\forall N \in \mathbb{N}$,

there is a path from $(N \cdot src)$ to $(N \cdot tgt)$?

Sequential Flow Problem [Colcombet, Fijalkow, Ohlmann '20]

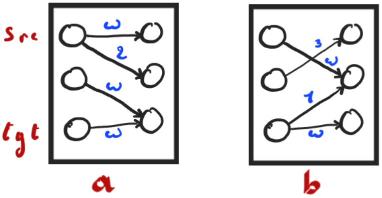
Input: Finite set of tiles $\tau \subseteq (\mathbb{N} \cup \{\omega\})^{S \times S}$

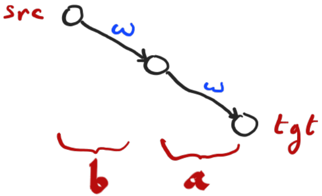
Source state $src \in S$

Target state $tgt \in S$

Question: $\forall N \in \mathbb{N}$,

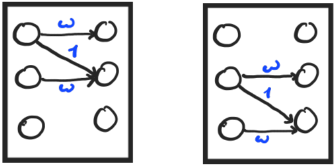
there is a path from $(N \cdot src)$ to $(N \cdot tgt)$?



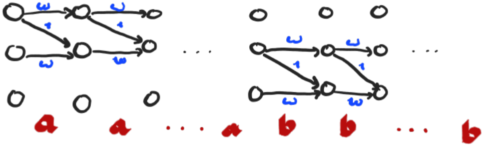


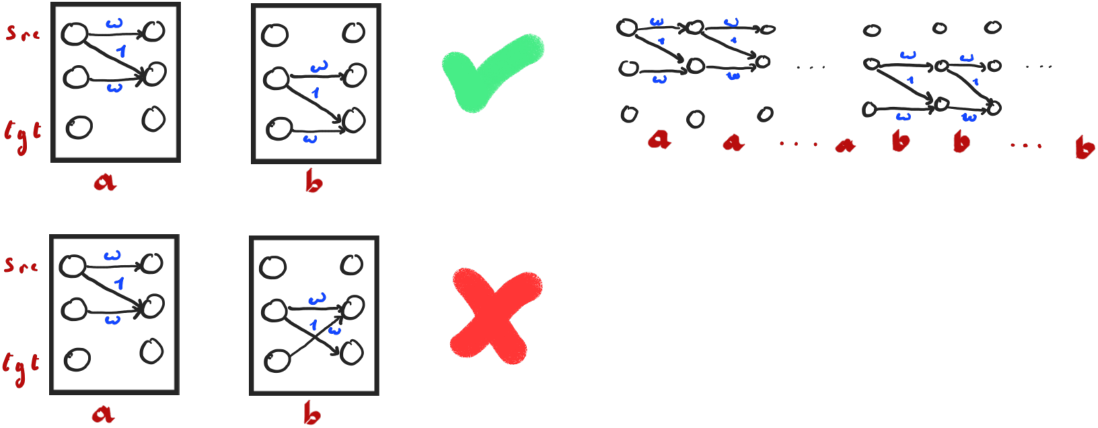
src

tgt

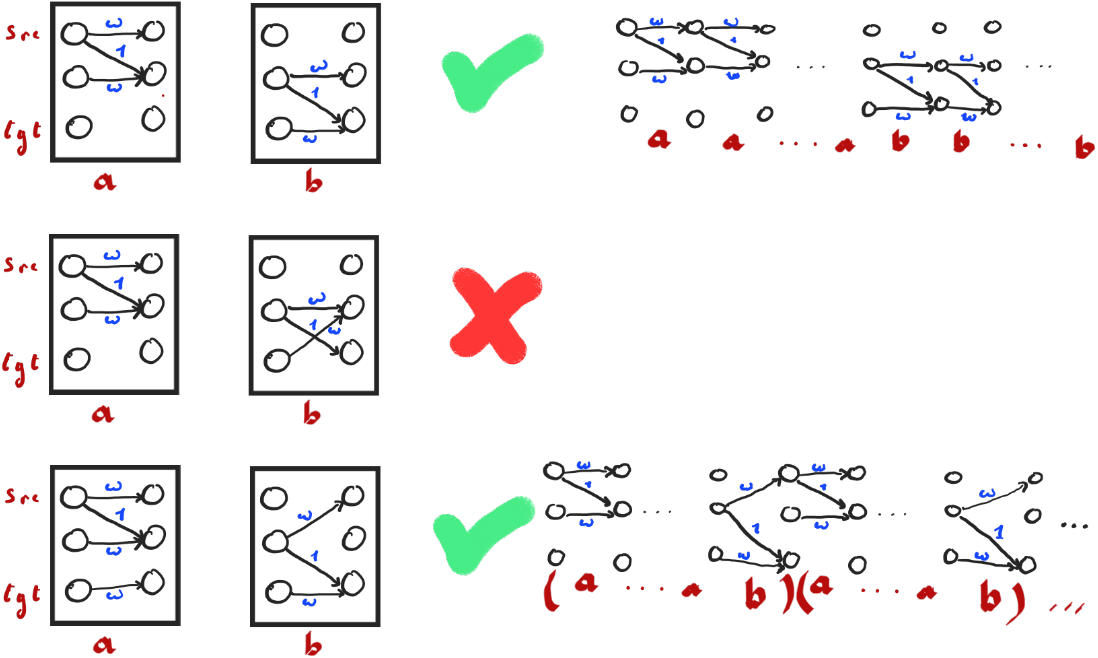


a b





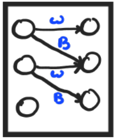
src
tgt
ω
1
a
b


src
tgt
ω
1
a
b
a a ... a b b ... b
(a ... a b)(a ... a b) ...

Monoid abstraction M

$1, 2, 3, \ldots \rightsquigarrow B$

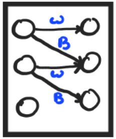
Elements :



$\{0, B, \omega\}^{S \times S}$

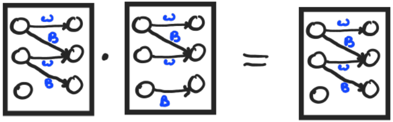
Monoid abstraction M

1, 2, 3, ... $\rightsquigarrow$ B

Elements : 

$\{0, B, \omega\}^{S \times S}$

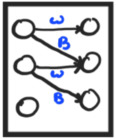
Product : max-min $\rightsquigarrow$ $(\tau_1 \cdot \tau_2)[s, s'] = \max_{s'' \in S} \min(\tau_1[s, s''], \tau_2[s'', s'])$



Morphism $\tau^* \longrightarrow M$

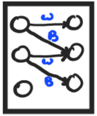
Monoid abstraction M

$1, 2, 3, \ldots \rightsquigarrow B$

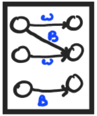
Elements : 

$\{0, B, \omega\}^{S \times S}$

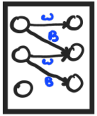
Product : max-min $\rightsquigarrow$ $(\tau_1 \cdot \tau_2)[s, s'] = \max_{s'' \in S} \min(\tau_1[s, s''], \tau_2[s'', s'])$



$\cdot$

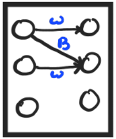


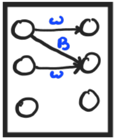
$=$



Morphism $\mathcal{T}^* \longrightarrow M$

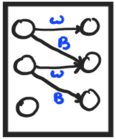
⚠ Problem :

$\forall k,$ 

$^k =$ 

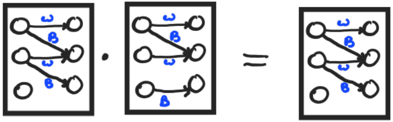
Monoid abstraction M

$1, 2, 3, \ldots \rightsquigarrow B$

Elements : 

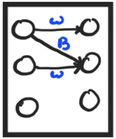
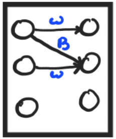
$\{0, B, \omega\}^{S \times S}$

Product : max-min $\rightsquigarrow$ $(\tau_1 \cdot \tau_2)[s, s'] = \max_{s'' \in S} \min(\tau_1[s, s''], \tau_2[s'', s'])$

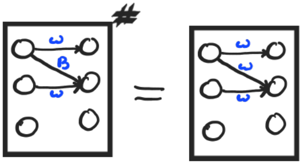


Morphism $\tau^* \longrightarrow M$

⚠ Problem :

$\forall k,$  ^{k} = 

Define an operator φ (on idempotents)



Morphism φ from #-expressions to M

$(a^\# b)^\#$

If there is a #expression E such that $\varphi(E)[src, tgt] = \omega$ then return Yes

↳ For all N, we can transfer N tokens by replacing # with N.

$(a^N b)^N$

Simon's theorem

- → Finite monoid M
- → Alphabet Σ
- → Morphism $\varphi : \Sigma^* \to M$

Factorisation tree for $w \in \Sigma^*$

M-labelled tree

Leaves = w

Nodes: Product nodes

Iteration nodes

$e \cdot e = e$

Simon's theorem

- → Finite monoid M
- → Alphabet Σ
- → Morphism $\varphi : \Sigma^* \to M$

Factorisation tree for $w \in \Sigma^*$

M-labelled tree

Leaves $= w$

Nodes: Product nodes

Iteration nodes

$e \cdot e = e$

Théorème

$\boxed{\forall w \in \Sigma^*, \text{ there is a factorisation tree for } w \text{ of height} \leq 3|M|}$

Application

If there is no #expression E such that $\varphi(E)[src, tgt] = \omega$ then return No

↳ Every word can be evaluated by a tree of height $\leq 3|M|$

↳ Show that a word evaluated by a tree of height h has flow $\leq g(h, |M|)$

Application

If there is no #expression E such that $\varphi(E)[src, tgt] = \omega$ then return No

↳ Every word can be evaluated by a tree of height $\leq 3|M|$

↳ Show that a word evaluated by a tree of height h has flow $\leq g(h, |M|)$

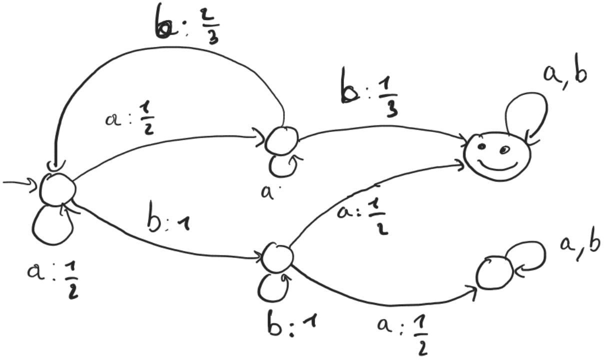
Théorème (by Saturation)

$$\boxed{\text{The parameterized flow problem is in EXPTIME}}$$

Application

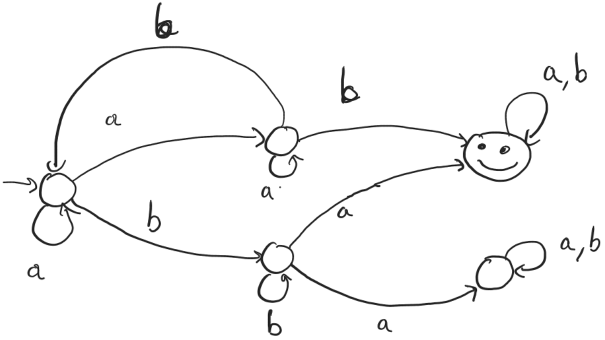
Parameterized MDPs

Almost-sure reachability.



Goal: reach ☺ with probability 1.

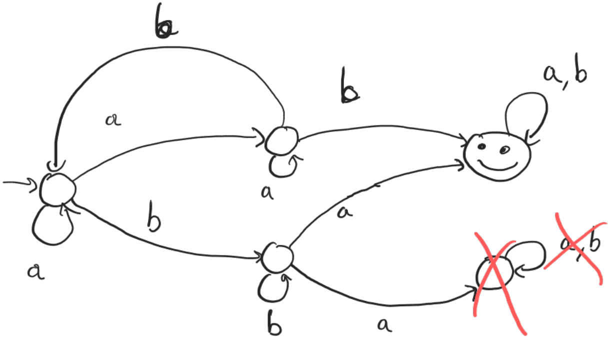
Almost-sure reachability.



→ Erase probabilities

Goal: reach ☺ with probability 1.

Almost-sure reachability.



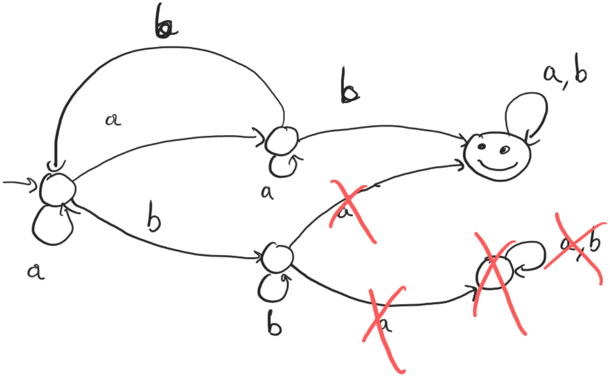
→ Erase probabilities

→ Compute winning region as greatest fix-point.

↳ Repeatedly remove states from which ☺ is not reachable.

Goal: reach ☺ with probability 1.

Almost-sure reachability.



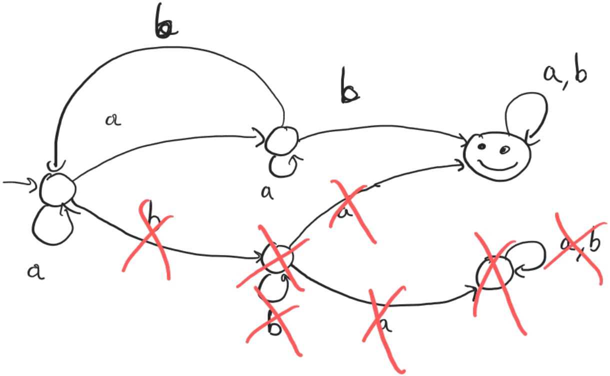
→ Erase probabilities

→ Compute winning region as greatest fix-point.

↳ Repeatedly remove states from which ☺ is not reachable.

Goal: reach ☺ with probability 1.

Almost-sure reachability.



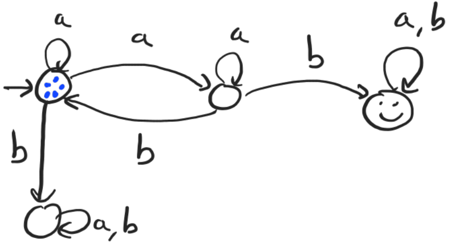
→ Erase probabilities

→ Compute winning region as greatest fix-point.

↳ Repeatedly remove states from which ☺ is not reachable.

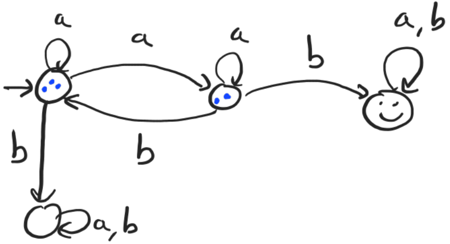
Goal: reach ☺ with probability 1.

Parameterized MDPs



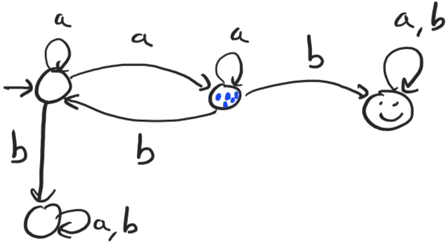
- Put N tokens in the initial state
- Each token reacts independently to the selected action.
- Try to make them all reach ☺

Parameterized MDPs



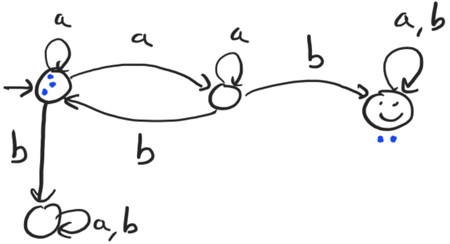
- Put N tokens in the initial state
- Each token reacts independently to the selected action.
- Try to make them all reach ☺

Parameterized MDPs



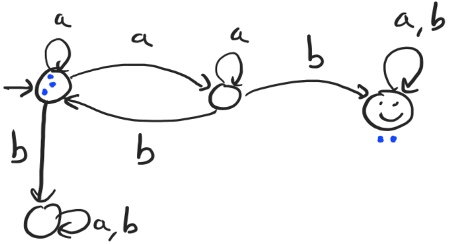
- Put N tokens in the initial state
- Each token reacts independently to the selected action.
- Try to make them all reach ☺

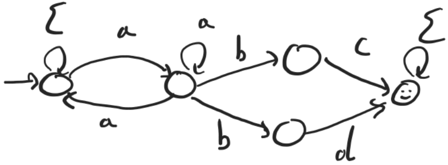
Parameterized MDPs



- Put N tokens in the initial state
- Each token reacts independently to the selected action.
- Try to make them all reach ☺

Parameterized MDPs





- Put N tokens in the initial state
- Each token reacts independently to the selected action.
- Try to make them all reach ☺

Characterisation of winning MDPs

Winning region
- → ☺ is always reachable with proba >0.
- → Never leave the region

+ Winning region is $\downarrow$-closed

∃ winning strategy

$\iff$

$\exists W \subseteq N^S \times \Sigma$

- W is $\downarrow$-closed
- $\forall (v,a) \in W$, if $v \xrightarrow{a} v'$ then $v' \in W$
- $\forall v \in W$, $\exists$ a path $v \rightsquigarrow F$ in W
- $N \cdot init \subseteq W$

$W \leftarrow \mathbb{N}^S \times \Sigma$

Do

| If $\exists (v, a) \in W,\ v \xrightarrow{a} v'$ with $v' \notin W$ (1)

| | $W \leftarrow W \setminus v\uparrow$

| If $\exists v \in W,\ v \not\rightsquigarrow F$ in W (2)

| | $W \leftarrow W \setminus v\uparrow$

Until fixpoint

If $I \subseteq W \rightarrow$ Yes

Else $\rightarrow$ No

(1) $\rightarrow$ Easy

(2) $\rightarrow$ Sequential flow problem

Theorem

The Random population control problem is Exptime-complete.

Proof: Show that

winning strat $\Rightarrow$ $\exists$ a region within the winning region in which at all times we have

↳ large crowds of tokens ω

↳ small sets of isolated tokens $\leq |S|$

Future work

Applications to - Verification
- Flows in infinite graphs

Quantitative objectives for parameterized MDPs

Speed of victory " " "