Parameter-Synthesis Problems for One-Counter Automata

Guillermo A. Pérez

(slides by Ritam Raha)

INFINITY 2020

Universiteit Antwerpen

Outline

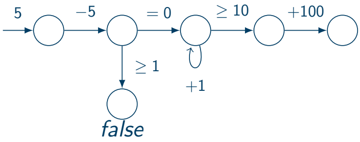
1. One-Counter Automata (Parametric) and Synthesis Problem
2. Previous Approach
3. Approach with Alternating Two-Way Automata for a subclass
4. (Failed) Approach with Partial Observation Games

One-Counter Automata

```
n = 5
n = max(0, n - 5)
if n = 0:
  while n < 10:
    n = n + 1
  n = n + 100
  # make progress
else:
  assert(False)
```

One-Counter Automata

```
n = 5
n = max(0, n - 5)
if n = 0:
  while n < 10:
    n = n + 1
  n = n + 100
  # make progress
else:
  assert(False)
```



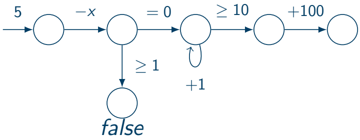
- Configurations: $(q, c), c \geq 0$;

One-Counter Automata

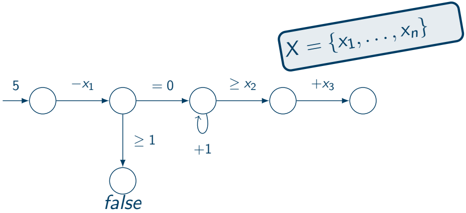
```python
def foobar(x):
  n = 5
  n = max(0, n - x)
  if n = 0:
    while n < 10:
      n = n + 1
    n = n + 100
    # make progress
  else:
    assert(False)
```

One-Counter Automata

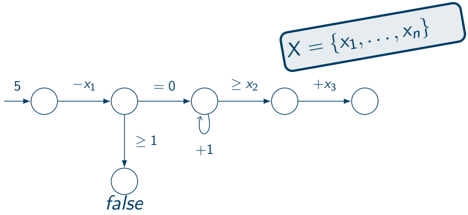
```python
def foobar(x):
  n = 5
  n = max(0, n - x)
  if n = 0:
    while n < 10:
      n = n + 1
    n = n + 100
    # make progress
  else:
    assert(False)
```



Parametric One-Counter Automata



Parametric One-Counter Automata

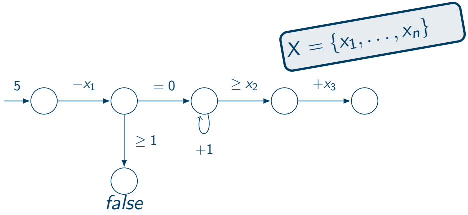


Definition (Succinct OCA with Parameters)

$\mathcal{A} = (Q, q_{in}, T, \delta, X)$

$\delta : T \to Op$ with Op the union of

Parametric One-Counter Automata



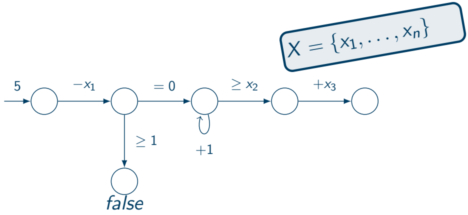
Definition (Succinct OCA with Parameters)

$\mathcal{A} = (Q, q_{in}, T, \delta, X)$

$\delta : T \to Op$ with Op the union of

- $CU := \{+a : a \in \mathbb{Z}\}$
- $CT := \{= 0, \geq a, = a : a \in \mathbb{Z}\}$

Parametric One-Counter Automata



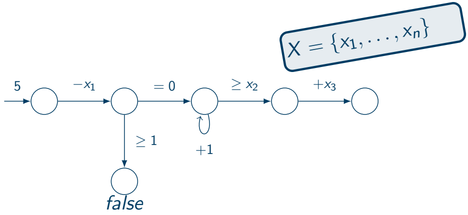
Definition (Succinct OCA with Parameters)

$\mathcal{A} = (Q, q_{in}, T, \delta, X)$

$\delta : T \to Op$ with Op the union of

- $CU := \{+a : a \in \mathbb{Z}\}$
- $CT := \{= 0, \geq a, = a : a \in \mathbb{Z}\}$
- $PU := \{+x, -x : x \in X\}$
- $PT := \{= x, \geq x : x \in X\}$

Parametric One-Counter Automata



Definition (Succinct OCA with Parameters)

$\mathcal{A} = (Q, q_{in}, T, \delta, X)$

$\delta : T \to Op$ with Op the union of

- $CU := \{+a : a \in \mathbb{Z}\}$
- $CT := \{= 0, \geq a, = a : a \in \mathbb{Z}\}$
- $PU := \{+x, -x : x \in X\}$
- $PT := \{= x, \geq x : x \in X\}$

Non-parametric: $X = \emptyset$

One-Counter Automata

Models		*CU*	*PU*	*ZT*	*PT*
Non-Parametric	SOCA	✓	✗	✓	✗
Parametric	OCAPT	$\{-1, 0, 1\}$	✗	✓	✓
	SOCAP	✓	✓	✓	✓

Decision Problems

Non-Parametric:

Parametric:

Decision Problems

Non-Parametric:

Reach

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

Parametric:

Decision Problems

Non-Parametric: 🙂

Reach

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

Parametric:

Decision Problems

Non-Parametric:

REACH

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

Parametric:

PAR-REACH

$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

Decision Problems

Non-Parametric: 🙂

REACH

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

Parametric: 🙂

PAR-REACH

$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

Decision Problems

Non-Parametric: 🙂

Reach

$\exists \rho$ such that
$(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

Parametric: 🙂

Par-Reach

$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$,
$(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

UnivReach

For all infinite ρ,
$(q_{in}, 0) \xrightarrow{\rho} q_f$

Decision Problems

Non-Parametric:

REACH
$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

UNIVREACH
For all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho} q_f$

Parametric:

PAR-REACH
$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

SYNTHREACH
$\exists V$ s.t. for all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

Decision Problems

Non-Parametric:

REACH
$\exists\rho$ such that
$(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

UNIVREACH
For all infinite ρ,
$(q_{in}, 0) \xrightarrow{\rho} q_f$

Parametric:

PAR-REACH
$\exists V : X \to \mathbb{N}$ s.t. $\exists\rho$,
$(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

SYNTHREACH
$\exists V$ s.t. for all infinite
ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

UnivReach for SOCA

Model and problem

- $Op = CU \cup ZT$
- UNIVREACH: all infinite paths reach q_f

UnivReach for SOCA

Model and problem

- $Op = CU \cup ZT$
- UNIVREACH: all infinite paths reach q_f

$\exists$ infinite path avoiding q_f = two REACH queries

UnivReach for SOCA

Model and problem

- $Op = CU \cup ZT$
- UNIVREACH: all infinite paths reach q_f

$\exists$ infinite path avoiding q_f = two REACH queries (coNP)

UnivReach for SOCA

Model and problem

- $Op = CU \cup ZT$
- UNIVREACH: all infinite paths reach q_f

$\exists$ infinite path avoiding q_f = two REACH queries (coNP)

Hardness from reduction from co-SUBSETSUM.

Proposition

The UNIVREACH *problem for SOCA is coNP-complete.*

Decision Problems

Non-Parametric:

REACH
$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

UNIVREACH
For all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho} q_f$

Parametric:

PAR-REACH
$\exists V : X\mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

SYNTHREACH
$\exists V$ s.t. for all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

Decision Problems

Non-Parametric: 🙂

Reach
$\exists\rho$ such that
$(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

🙂

UnivReach
For all infinite ρ,
$(q_{in}, 0) \xrightarrow{\rho} q_f$

[coNP-complete]

Parametric: 🙂

Par-Reach
$\exists V : X\mathbb{N}$ s.t. $\exists\rho$,
$(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

🙂?

SynthReach
$\exists V$ s.t. for all infinite
ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

Outline

1. One-Counter Automata (Parametric) and Synthesis Problem
2. Previous Approach
3. Approach with Alternating Two-Way Automata for a subclass
4. Approach with Partial Observation Games

PAD & Extensions

- Presburger arithmetic (PA) $:= \langle \mathbb{N}, 0, 1, +, < \rangle$

PAD & Extensions

- Presburger arithmetic (PA) $:= \langle \mathbb{N}, 0, 1, +, < \rangle$
- Presburger arithmetic with Divisibility (PAD) $:=$ PA $+ \mid$
  $(a \mid b \iff \exists c \in \mathbb{Z} : b = ac)$

PAD & Extensions

- Presburger arithmetic (PA) $:= \langle \mathbb{N}, 0, 1, +, < \rangle$
- Presburger arithmetic with Divisibility (PAD) $:=$ PA $+ \mid$ $(a \mid b \iff \exists c \in \mathbb{Z} : b = ac)$

Theorem (Robinson'49, Lipshitz'81)

Full PAD is undecidable; one alternation suffices for undecidability.

PAD & Extensions

- Presburger arithmetic (PA) $:= \langle \mathbb{N}, 0, 1, +, < \rangle$
- Presburger arithmetic with Divisibility (PAD) := PA $+ \mid$ $(a \mid b \iff \exists c \in \mathbb{Z} : b = ac)$

Theorem (Robinson'49, Lipshitz'81)

Full PAD is undecidable; one alternation suffices for undecidability.

Theorem (Lipshitz'78, Lechner-Ouaknine-Worrell'15)

The existential fragment of PAD (EPAD) is decidable in NEXP.

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\text{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m . \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\mathrm{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m . \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$
- $\forall\exists_R\mathrm{PAD}^+ := \forall\exists_R\mathrm{PAD}$ with $\neg$ not allowed before divisibility.

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\text{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m . \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$
- $\forall\exists_R\text{PAD}^+ := \forall\exists_R\text{PAD}$ with $\neg$ not allowed before divisibility.

Claim (Bozga-Iosif'05, Lechner'15)

SYNTHREACH for SOCAP is decidable by a reduction to $\forall\exists_R\text{PAD}^+$.

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\mathrm{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m . \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$
- $\forall\exists_R\mathrm{PAD}^+ := \forall\exists_R\mathrm{PAD}$ with $\neg$ not allowed before divisibility.

Claim (Bozga-Iosif'05, Lechner'15)

SynthReach for SOCAP is decidable by a reduction to $\forall\exists_R\mathrm{PAD}^+$.

$\forall\exists_R\mathrm{PAD} \equiv \forall\exists_R\mathrm{PAD}^+$

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\text{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m.\, \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$
- $\forall\exists_R\text{PAD}^+ := \forall\exists_R\text{PAD}$ with $\neg$ not allowed before divisibility.

Claim (Bozga-Iosif'05, Lechner'15)

SynthReach for SOCAP is decidable by a reduction to $\forall\exists_R\text{PAD}^+$.

$\forall\exists_R\text{PAD} \equiv \forall\exists_R\text{PAD}^+$

Idea: We have to rewrite $\neg(a \mid b)$

$\forall\exists_R$PAD & Undecidability

- $\forall\exists_R\text{PAD} := \forall z_1 \ldots \forall z_n \exists x_1 \ldots \exists x_m . \varphi(\mathbf{x}, \mathbf{z})$
  - divisibilities of the form $f(\mathbf{z}) \mid g(\mathbf{x}, \mathbf{z})$
- $\forall\exists_R\text{PAD}^+ := \forall\exists_R\text{PAD}$ with $\neg$ not allowed before divisibility.

Claim (Bozga-Iosif'05, Lechner'15)

SynthReach for SOCAP is decidable by a reduction to $\forall\exists_R\text{PAD}^+$.

$\forall\exists_R\text{PAD} \equiv \forall\exists_R\text{PAD}^+$

Idea: We have to rewrite $\neg(a \mid b)$

$\neg(a \mid b) \equiv b = aq + r$ where, $0 < r < b$.

$\forall\exists_R$PAD & Undecidability

Theorem (Bozga-Iosif'05)

$\forall\exists_R PAD$ is undecidable.

Idea: Using the single restricted alternation we define

1. LCM

$\forall\exists_R$PAD & Undecidability

Theorem (Bozga-Iosif'05)

$\forall\exists_R PAD$ is undecidable.

Idea: Using the single restricted alternation we define

1. LCM
2. Square (x^2)

$\forall\exists_R$PAD & Undecidability

Theorem (Bozga-Iosif'05)

$\forall\exists_R PAD$ is undecidable.

Idea: Using the single restricted alternation we define

1. LCM
2. Square (x^2)
3. Multiplication

Undecidable!!

Decision Problems

Non-Parametric:

REACH

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete (HKOW'09)]

UNIVREACH

For all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho} q_f$

[coNP-complete]

Parametric:

PAR-REACH

$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP (HKOW'09,LOW'15)]

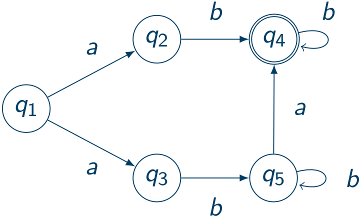
SYNTHREACH

$\exists V$ s.t. for all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

Outline

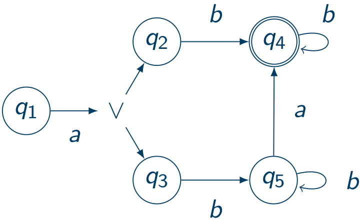
Alternating Two-Way Automaton

$w = ab^\omega$



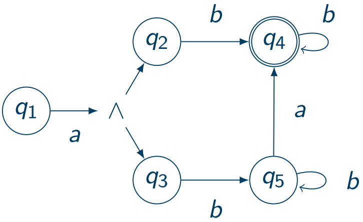
Alternating Two-Way Automaton

$w = ab^\omega$



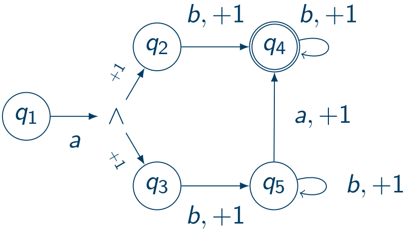
Alternating Two-Way Automaton

$w = ab^\omega$



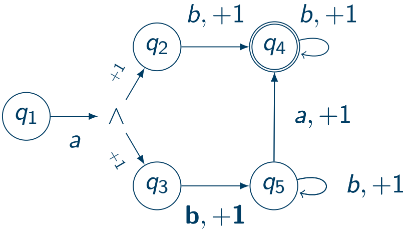
Alternating Two-Way Automaton

$w = ab^\omega$



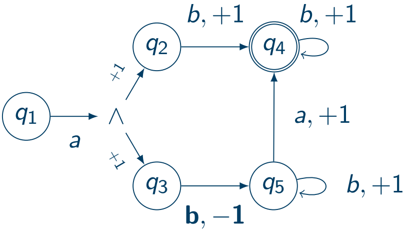
Alternating Two-Way Automaton

$w = ab^\omega$



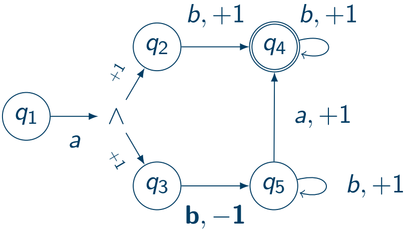
Alternating Two-Way Automaton

$w = ab^{\omega}$



Alternating Two-Way Automaton

$w = ab^\omega$



Theorem (Serre'06)

The non-emptiness problem for A2As is in PSPACE.

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

Proposition (Based on Bollig-Quaas-Sangnier'19)

For every OCAPT $\mathcal{A}$ we construct an A2A $\mathcal{T}$ of poly-size which accepts words corresponding to valuations that witness SynthReach *is true.*

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

Proposition (Based on Bollig-Quaas-Sangnier'19)

For every OCAPT $\mathcal{A}$ we construct an A2A $\mathcal{T}$ of poly-size which accepts words corresponding to valuations that witness $\textsc{SynthReach}$ *is true.*

Idea:

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

Proposition (Based on Bollig-Quaas-Sangnier'19)

For every OCAPT $\mathcal{A}$ we construct an A2A $\mathcal{T}$ of poly-size which accepts words corresponding to valuations that witness SynthReach *is true.*

Idea:

- Encode valuations as parameter words

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

Proposition (Based on Bollig-Quaas-Sangnier'19)

For every OCAPT $\mathcal{A}$ we construct an A2A $\mathcal{T}$ of poly-size which accepts words corresponding to valuations that witness SynthReach *is true.*

Idea:

- Encode valuations as parameter words
- For every transition we build an A2A

From OCAPT to A2A

$Op = \{-1, 0, +1\} \cup ZT \cup PT$

Proposition (Based on Bollig-Quaas-Sangnier'19)

For every OCAPT $\mathcal{A}$ we construct an A2A $\mathcal{T}$ of poly-size which accepts words corresponding to valuations that witness SynthReach *is true.*

Idea:

- Encode valuations as parameter words
- For every transition we build an A2A
- Accept the reaching runs
- (...and runs that "die off")

OCAPT to A2A

Valuation to words:

OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N};\qquad \Sigma = X \cup \{\square\};$

OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N};\qquad \Sigma = X \cup \{\square\};$

$\square$

OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N}$; $\Sigma = X \cup \{\square\}$;

$\square\square\square x_1 \square x_2 \square^\omega$

Valuation to words:

$V : X \to \mathbb{N}$; $\Sigma = X \cup \{\Box\}$;

$\Box\Box\Box x_1 \Box x_2 \Box^\omega \Rightarrow x_1 \mapsto 2, x_2 \mapsto 3$;

OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N};\qquad \Sigma = X \cup \{\Box\};$

$\Box\Box\Box x_1\Box x_2\Box^\omega \Rightarrow x_1 \mapsto 2, x_2 \mapsto 3;$

Op to A2A:

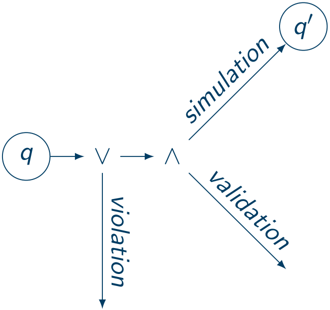
OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N}; \qquad \Sigma = X \cup \{\Box\};$

$\Box\Box\Box x_1 \Box x_2 \Box^\omega \Rightarrow x_1 \mapsto 2, x_2 \mapsto 3;$

Op to A2A:



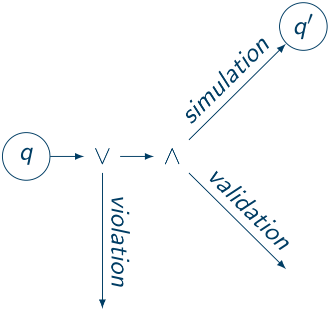
OCAPT to A2A

Valuation to words:

$V : X \to \mathbb{N};\qquad \Sigma = X \cup \{\Box\};$

$\Box\Box\Box x_1 \Box x_2 \Box^\omega \Rightarrow x_1 \mapsto 2, x_2 \mapsto 3;$

Op to A2A:



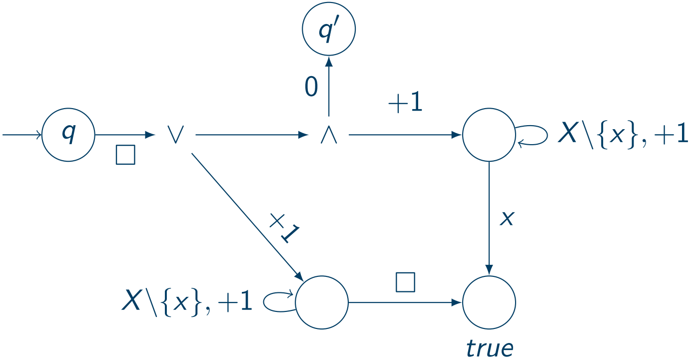
Accept reaching runs:

OCAPT to A2A

- Encoding Equality Test:

OCAPT to A2A

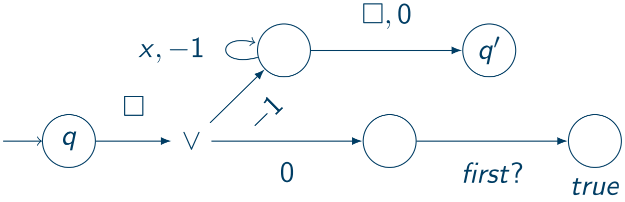
- Encoding Equality Test:



OCAPT to A2A

- Encoding Decrement:

- Encoding Decrement:



OCAPT to A2A

We want to check all runs:

OCAPT to A2A

We want to check all runs:

$q \xrightarrow{op_1} q_1 \Rightarrow (q, \square, \mathcal{T}_1)$

OCAPT to A2A

We want to check all runs:

$$q \xrightarrow{op_1} q_1 \Rightarrow (q, \Box, \mathcal{T}_1)$$

$$q \xrightarrow{op_2} q_2 \Rightarrow (q, \Box, \mathcal{T}_2)$$

OCAPT to A2A

We want to check all runs:

$$q \xrightarrow{op_1} q_1 \Rightarrow (q, \Box, \mathcal{T}_1)$$

$$q \xrightarrow{op_2} q_2 \Rightarrow (q, \Box, \mathcal{T}_2)$$

$$(q, \Box, \mathcal{T}_1 \wedge \mathcal{T}_2) \in \mathcal{T}$$

OCAPT to A2A

Proposition

SYNTHREACH *for OCAPT is in PSPACE.*

OCAPT to A2A

Proposition

SynthReach *for OCAPT is in PSPACE.*

- For every A2A $\mathcal{T}$ there is an NBA $\mathcal{B}$ of exponential size accepting same language
- Non-emptiness witnesses for NBAs are simple "lassos"
- $\implies$ SynthReach admits exponential (w.r.t. the OCAPT) witnesses and thus polynomial in binary encoding

OCAPT to A2A

Proposition

SYNTHREACH *for OCAPT is in PSPACE.*

- For every A2A $\mathcal{T}$ there is an NBA $\mathcal{B}$ of exponential size accepting same language
- Non-emptiness witnesses for NBAs are simple "lassos"
- $\implies$ SYNTHREACH admits exponential (w.r.t. the OCAPT) witnesses and thus polynomial in binary encoding
- Guess a valuation and check UNIVREACH for resulting SOCA

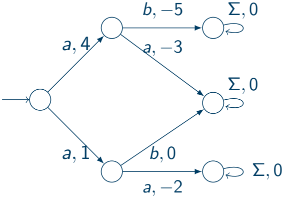
Theorem

SYNTHREACH *for OCAPT is in* NP^{coNP}.

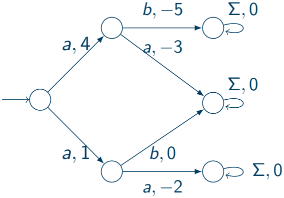
Outline

1. One-Counter Automata (Parametric) and Synthesis Problem
2. Previous Approach
3. Approach with Alternating Two-Way Automata for a subclass
4. Approach with Partial Observation Games

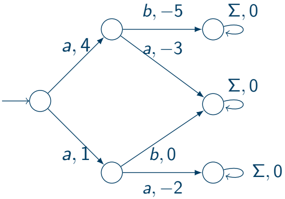
Partial Observation Energy Games



Partial Observation Energy Games



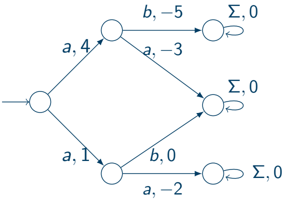
Partial Observation Energy Games



- Chooses an action

- Resolves non-determinism

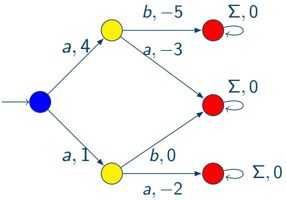
Partial Observation Energy Games



- Chooses an action
- Keeps the energy level positive

- Resolves non-determinism
- Wants it eventually negative

Partial Observation Energy Games



- Chooses an action
- Keeps the energy level positive

- Resolves non-determinism
- Wants it eventually negative

SOCAP to POEG

Claim

For every SOCAP $\mathcal{A}$ we construct a POEG which Eve wins iff for all valuations V there exists a reaching run of $\mathcal{A}$.

SOCAP to POEG

Claim

For every SOCAP $\mathcal{A}$ we construct a POEG which Eve wins iff for all valuations V there exists a reaching run of $\mathcal{A}$.

Idea:

- Adam chooses valuations, Eve simulates a reaching run

SOCAP to POEG

Claim

For every SOCAP $\mathcal{A}$ we construct a POEG which Eve wins iff for all valuations V there exists a reaching run of $\mathcal{A}$.

Idea:

- Adam chooses valuations, Eve simulates a reaching run
- We create gadgets in the game simulating Op

SOCAP to POEG

Claim

For every SOCAP $\mathcal{A}$ we construct a POEG which Eve wins iff for all valuations V there exists a reaching run of $\mathcal{A}$.

Idea:

- Adam chooses valuations, Eve simulates a reaching run
- We create gadgets in the game simulating *Op*
- The PO helps Adam force Eve to simulate runs faithfully

SOCAP to POEG

Claim

For every SOCAP $\mathcal{A}$ we construct a POEG which Eve wins iff for all valuations V there exists a reaching run of $\mathcal{A}$.

Idea:

- Adam chooses valuations, Eve simulates a reaching run
- We create gadgets in the game simulating *Op*
- The PO helps Adam force Eve to simulate runs faithfully

Conjecture

SynthReach *problem for SOCAP is decidable.*

Decision Problems

Non-Parametric:

REACH

$\exists \rho$ such that $(q_{in}, 0) \xrightarrow{\rho} q_f$

[NP-complete]

UNIVREACH

For all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho} q_f$

[coNP-complete]

Parametric:

PAR-REACH

$\exists V : X \to \mathbb{N}$ s.t. $\exists \rho$, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[in NEXP]

SYNTHREACH

$\exists V$ s.t. for all infinite ρ, $(q_{in}, 0) \xrightarrow{\rho}_V q_f$

[Decidable?]

Conclusion

- If parameters are only allowed on tests, the problem is in NP^{NP}
- In full generality, for SOCAP it is still open