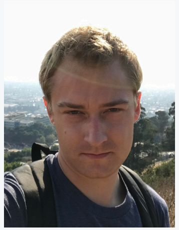
The Tractability Border of Reachability in Simple Vector Addition Systems with States

Henry Sinclair-Banks

Based on work with Dmitry Chistikov, Wojciech Czerwiński, Filip Mazowiecki, Łukasz Orlikowski, and Karol Węgrzycki to appear in FOCS'24.





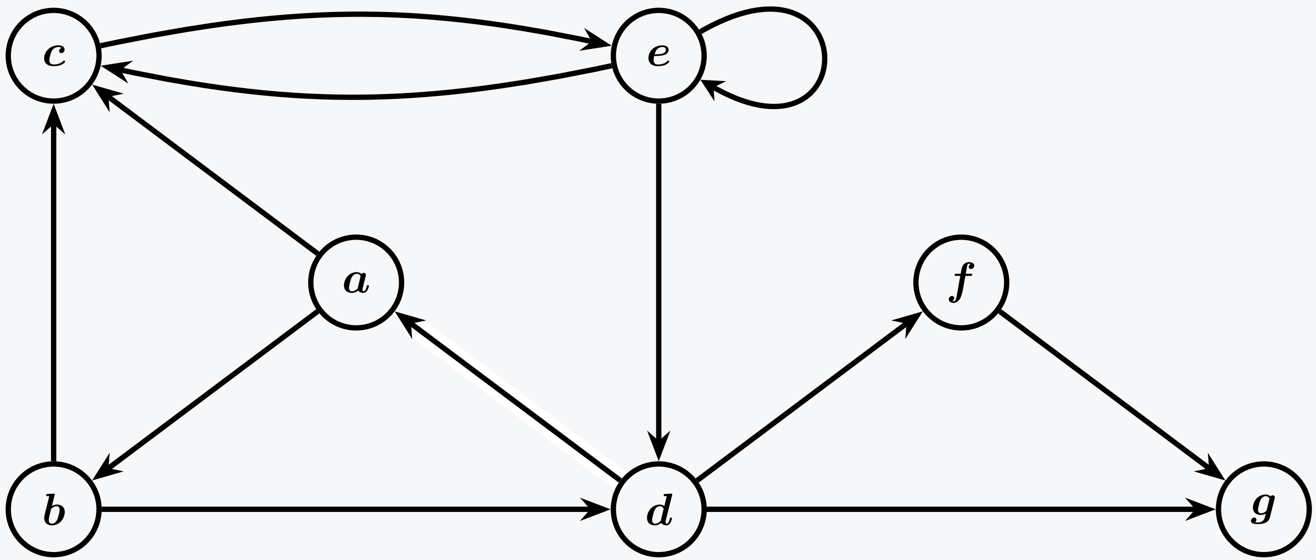



Algorithms & Complexity Seminar

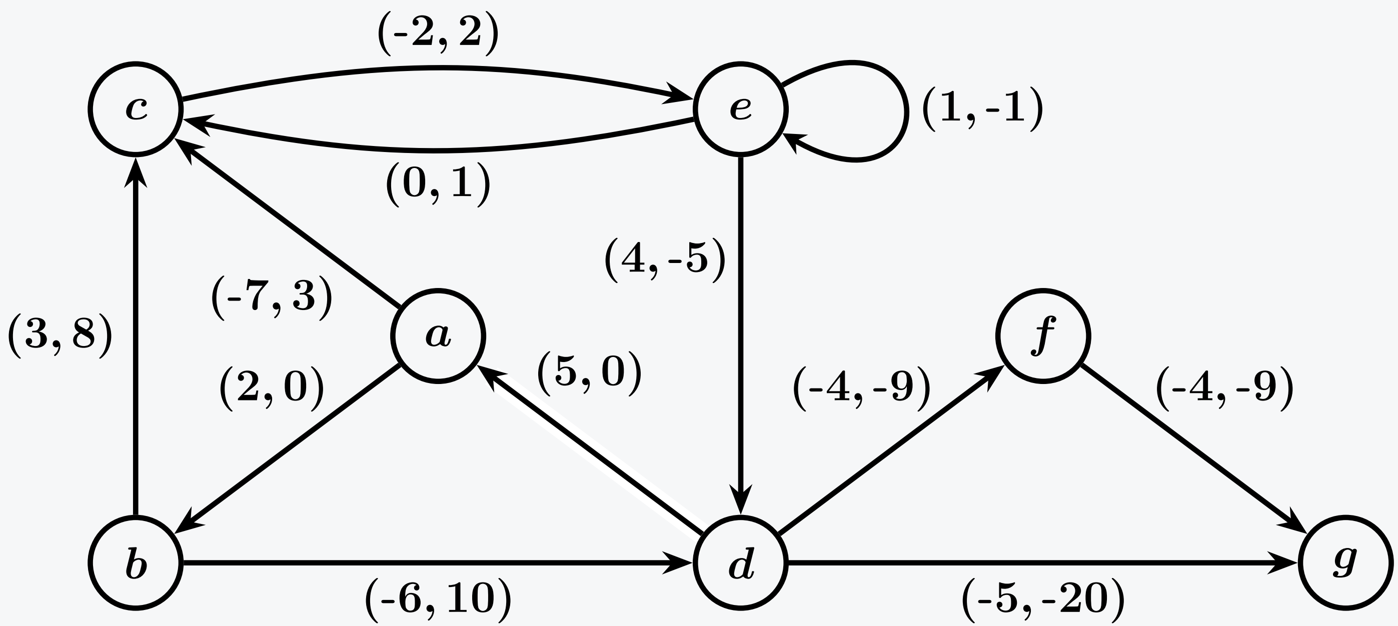
27th August 2024

KIT, Karlsruhe, Germany

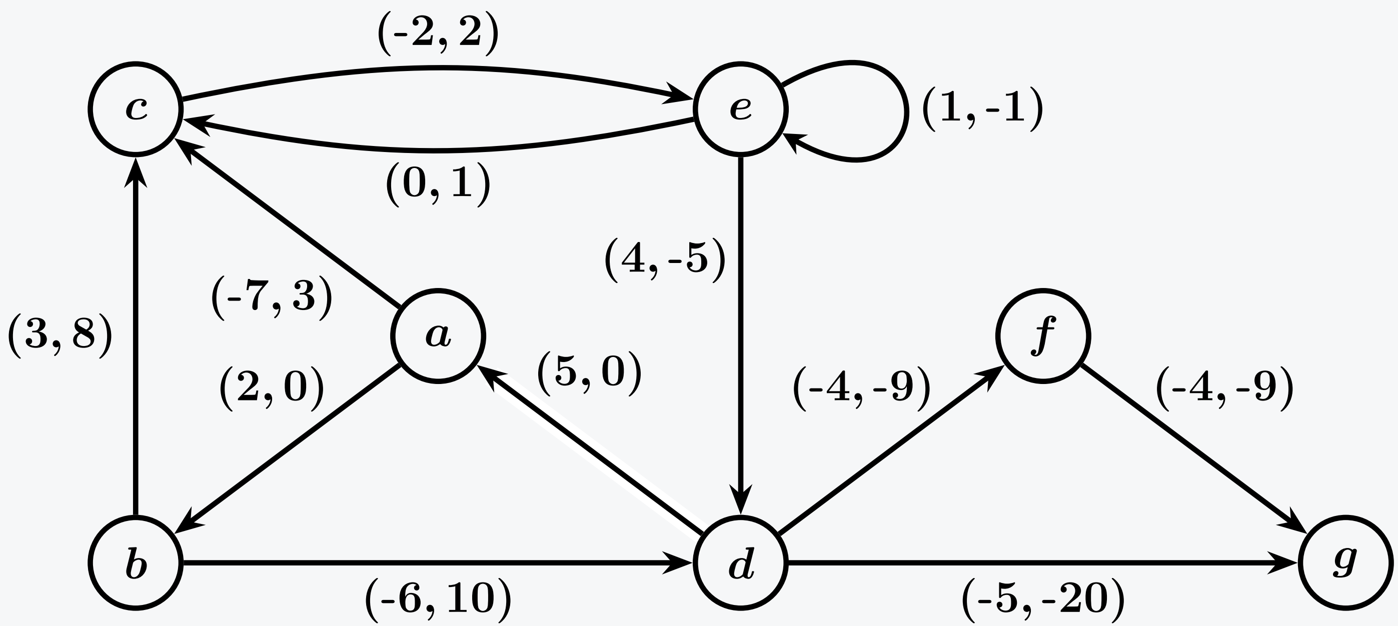
2-Dimensional Vector Addition System with States



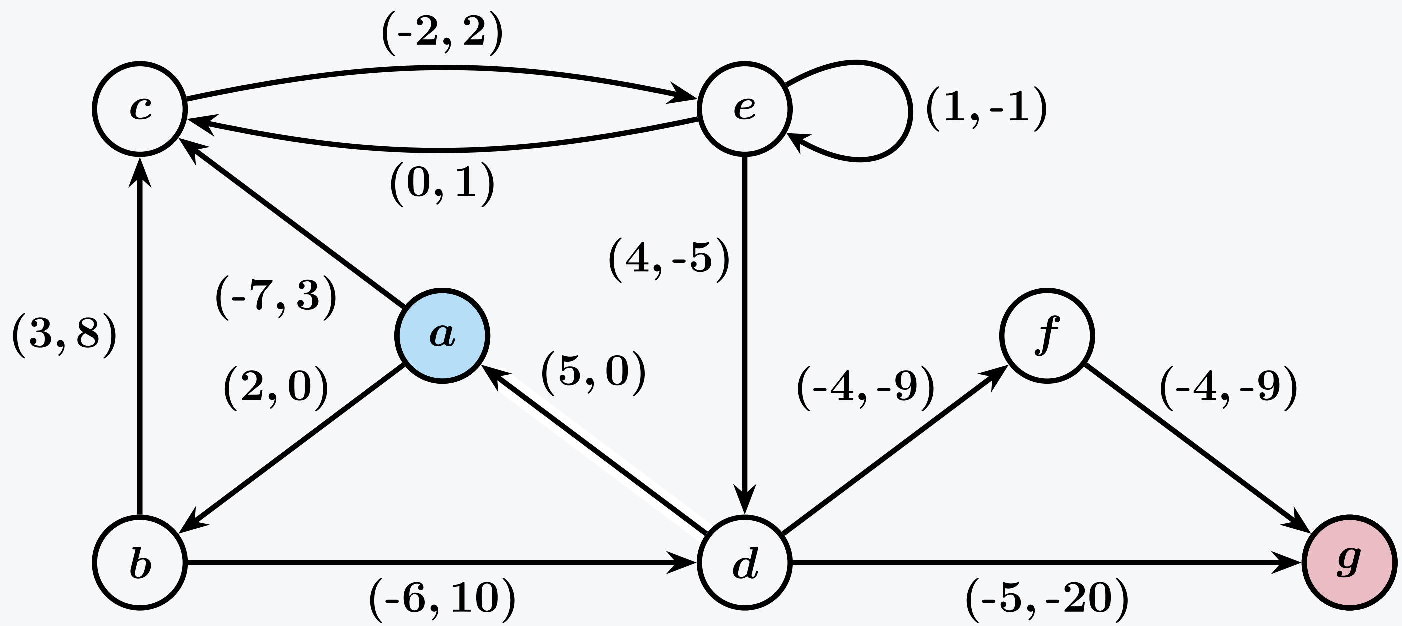
2-Dimensional Vector Addition System with States



2-Dimensional VASS



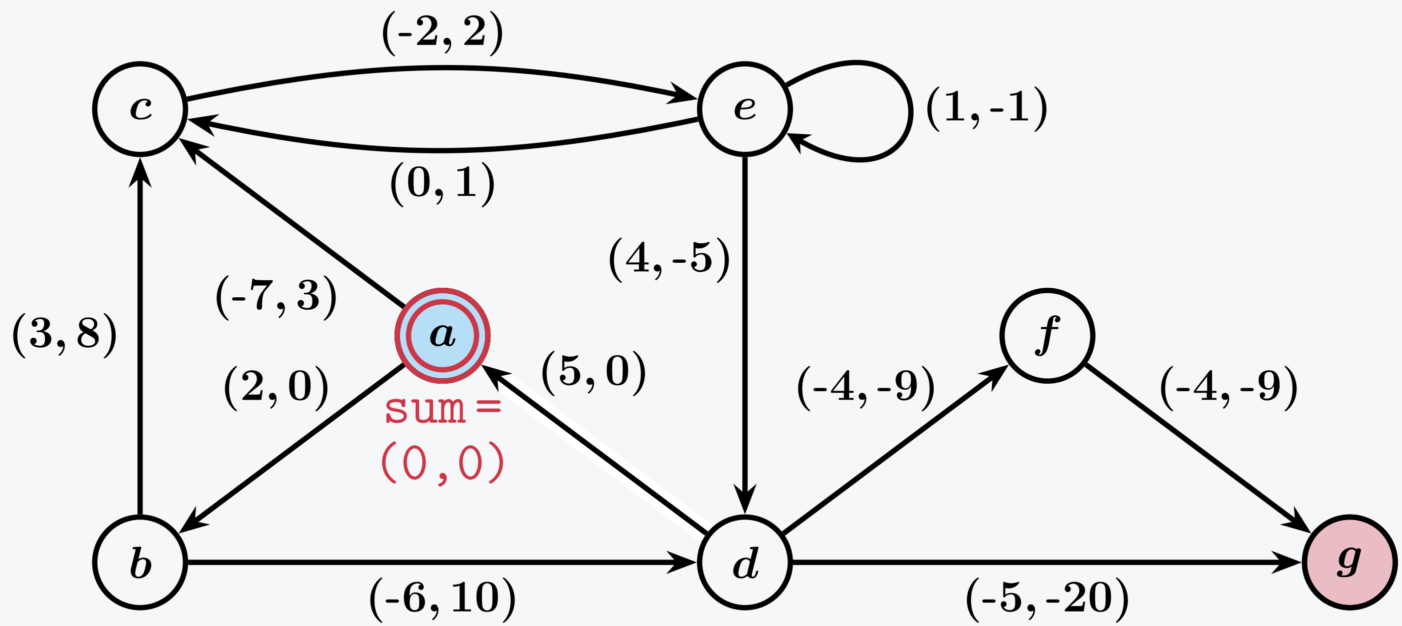
Reachability in 2-Dimensional VASS



Does there exist a run from *a* with counter values (0,0) to *g* with counter values (4,5) ?

(the counters must remain nonnegative at all times)

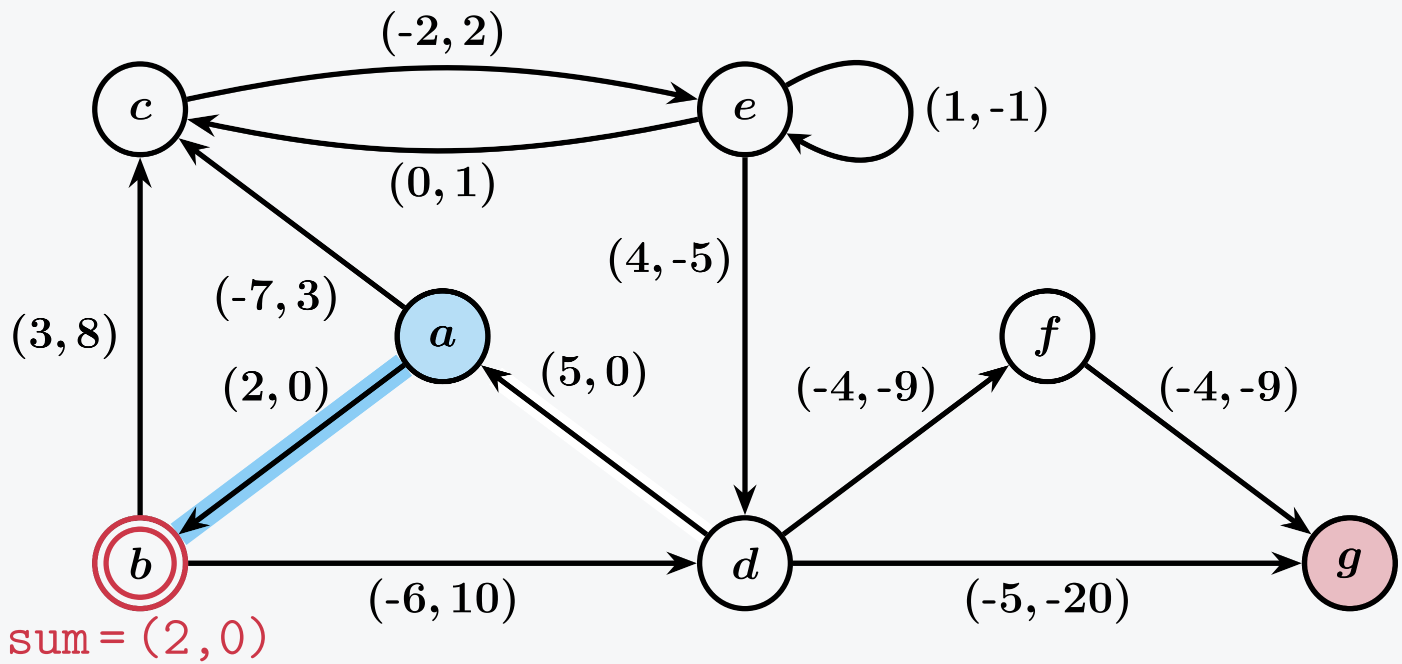
Reachability in 2-Dimensional VASS



Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

(the counters must remain nonnegative at all times)

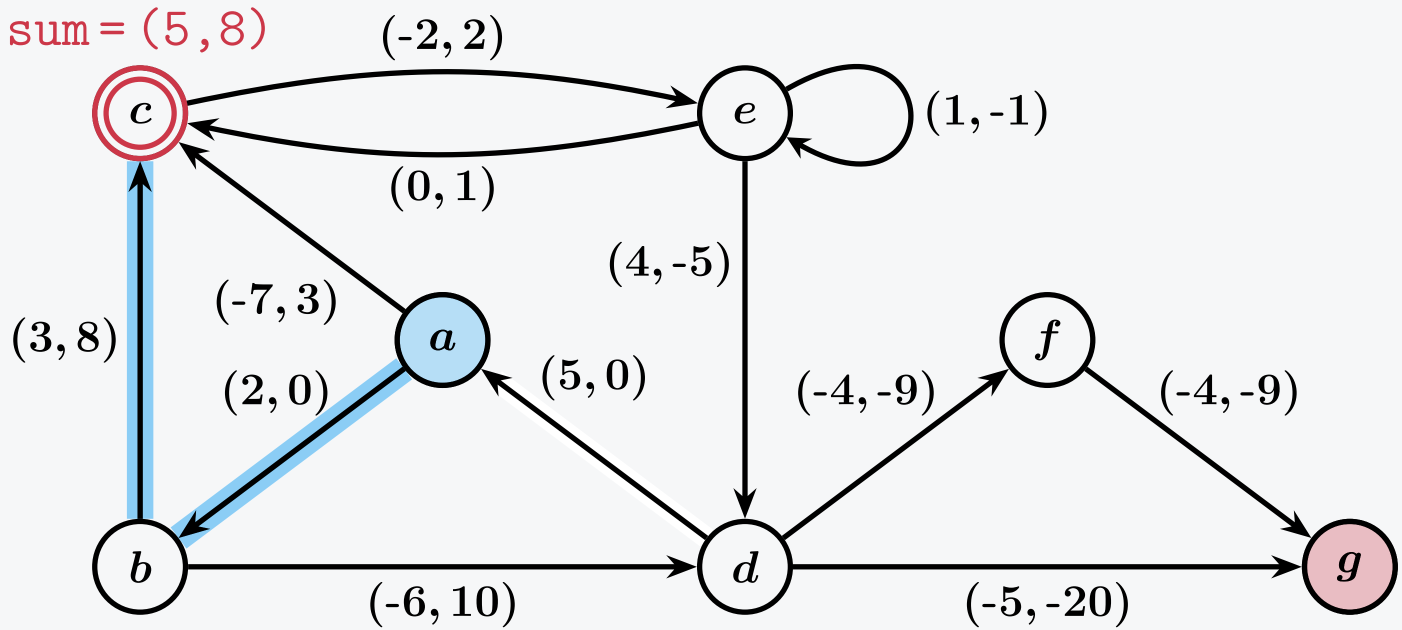
Reachability in 2-Dimensional VASS



Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

(the counters must remain nonnegative at all times)

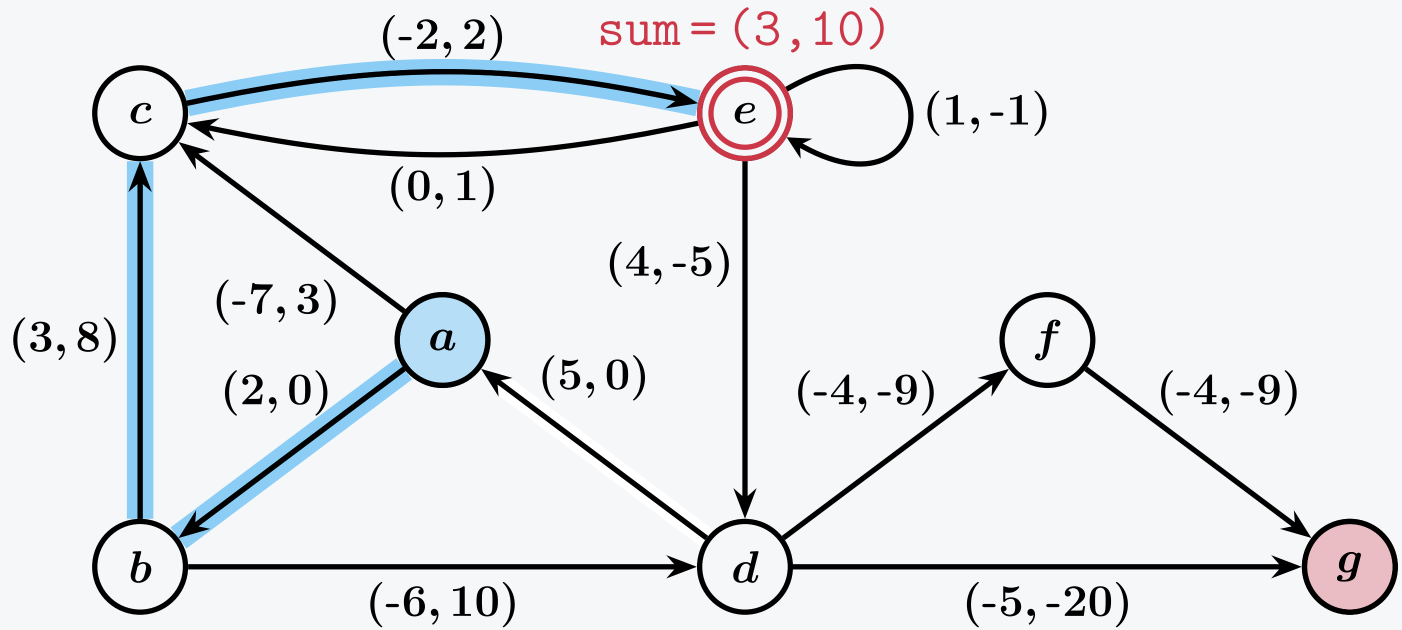
Reachability in 2-Dimensional VASS



Does there exist a run from *a* with counter values (0,0) to *g* with counter values (4,5) ?

(the counters must remain nonnegative at all times)

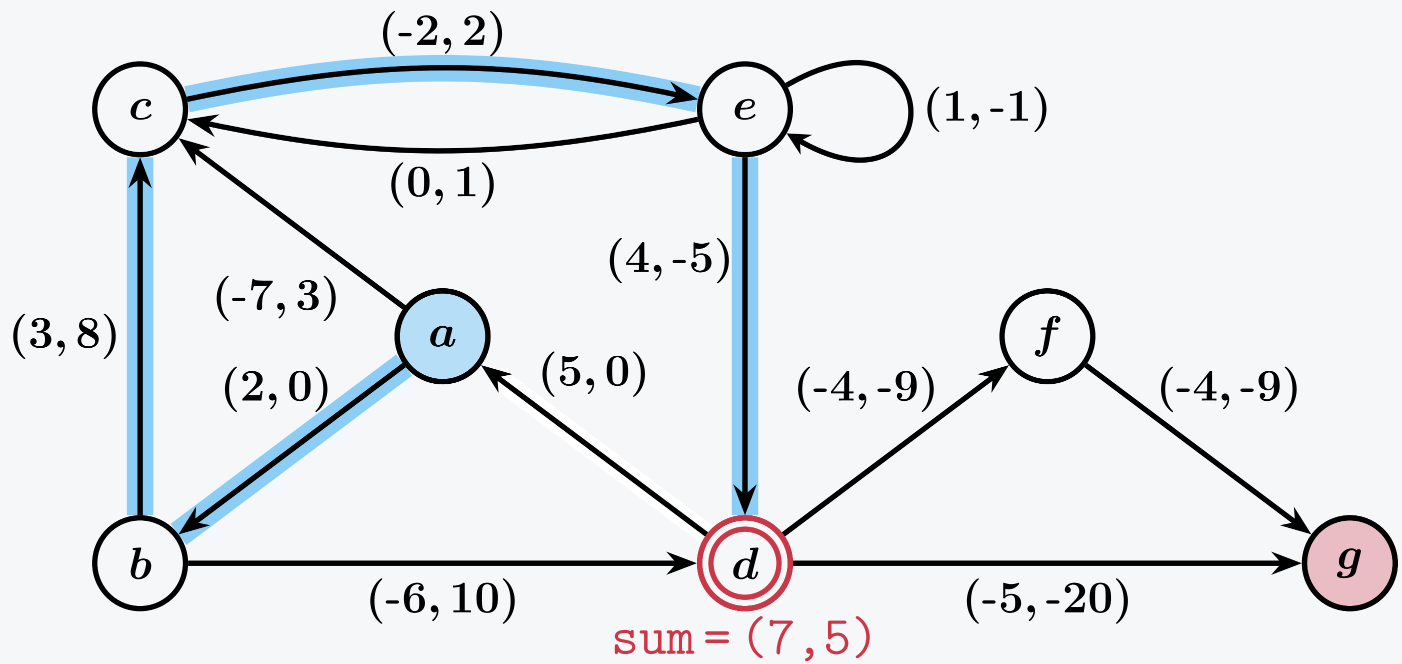
Reachability in 2-Dimensional VASS



Does there exist a run from *a* with counter values (0,0) to *g* with counter values (4,5) ?

(the counters must remain nonnegative at all times)

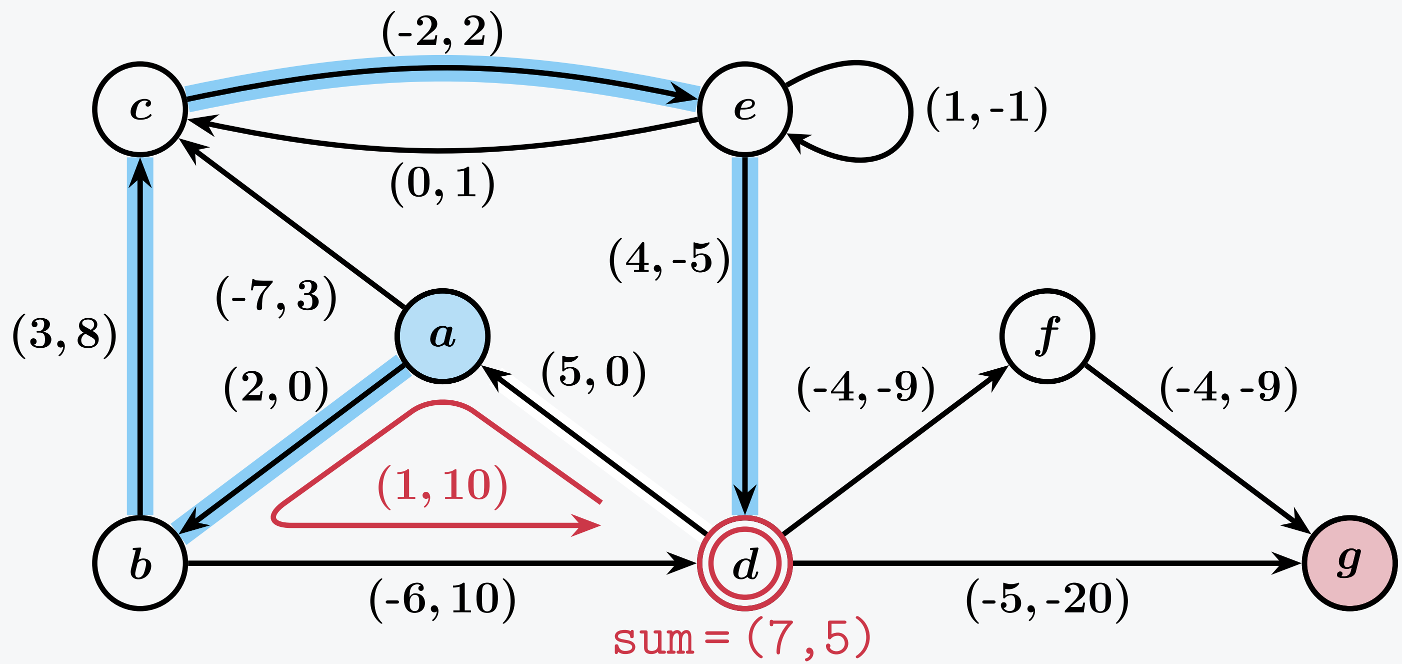
Reachability in 2-Dimensional VASS



Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

(the counters must remain nonnegative at all times)

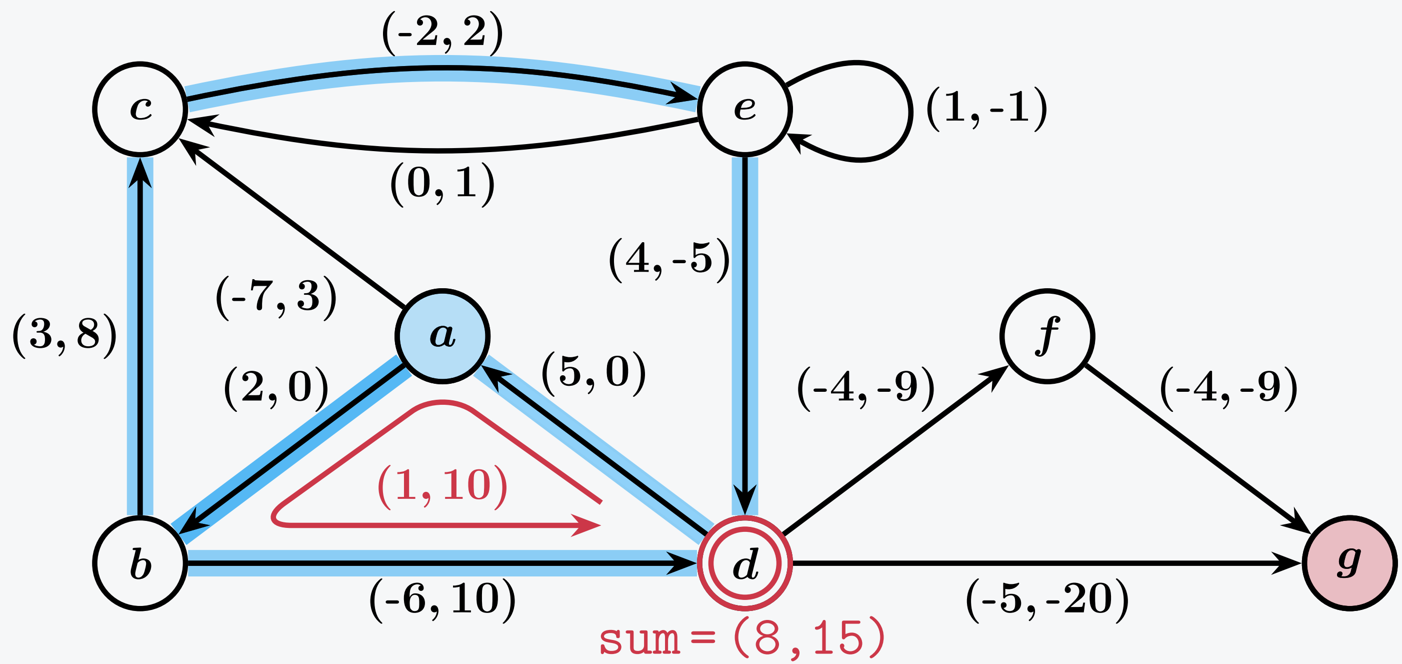
Reachability in 2-Dimensional VASS



Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

(the counters must remain nonnegative at all times)

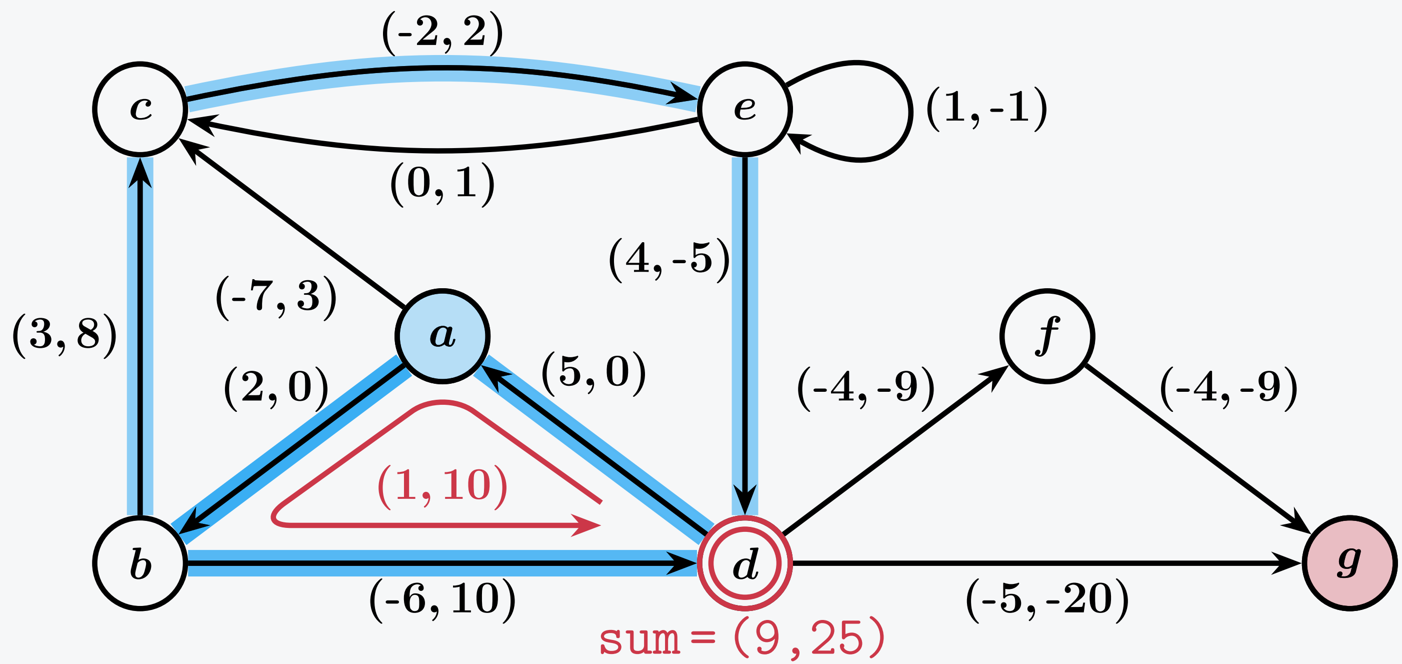
Reachability in 2-Dimensional VASS



Does there exist a run from *a* with counter values (0,0) to *g* with counter values (4,5) ?

(the counters must remain nonnegative at all times)

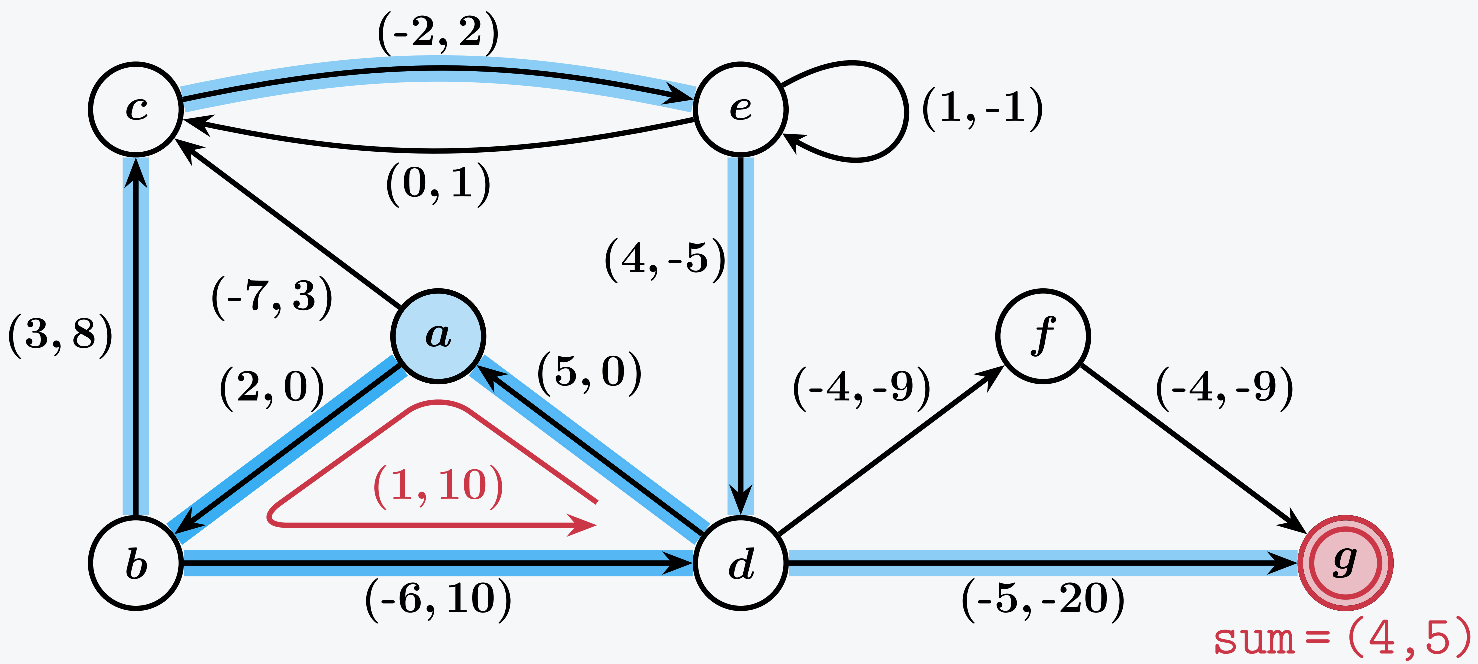
Reachability in 2-Dimensional VASS



Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

(the counters must remain nonnegative at all times)

Reachability in 2-Dimensional VASS

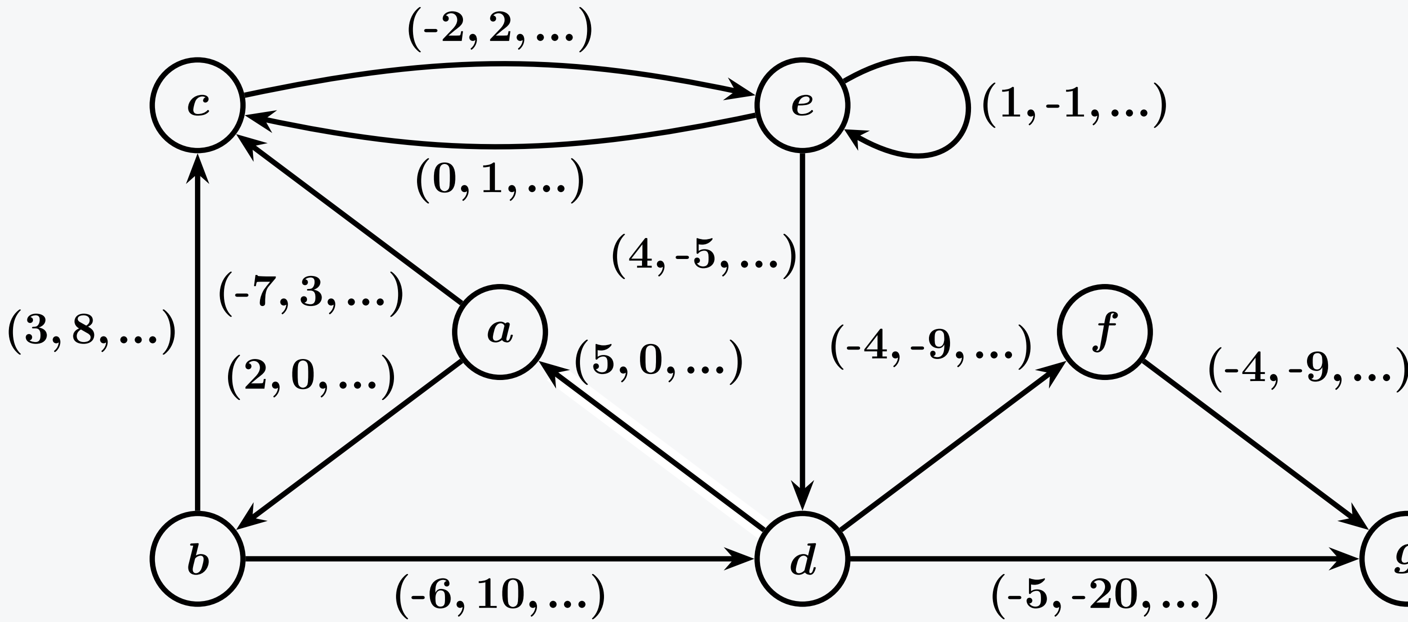


Does there exist a run from a with counter values (0,0) to g with counter values (4,5) ?

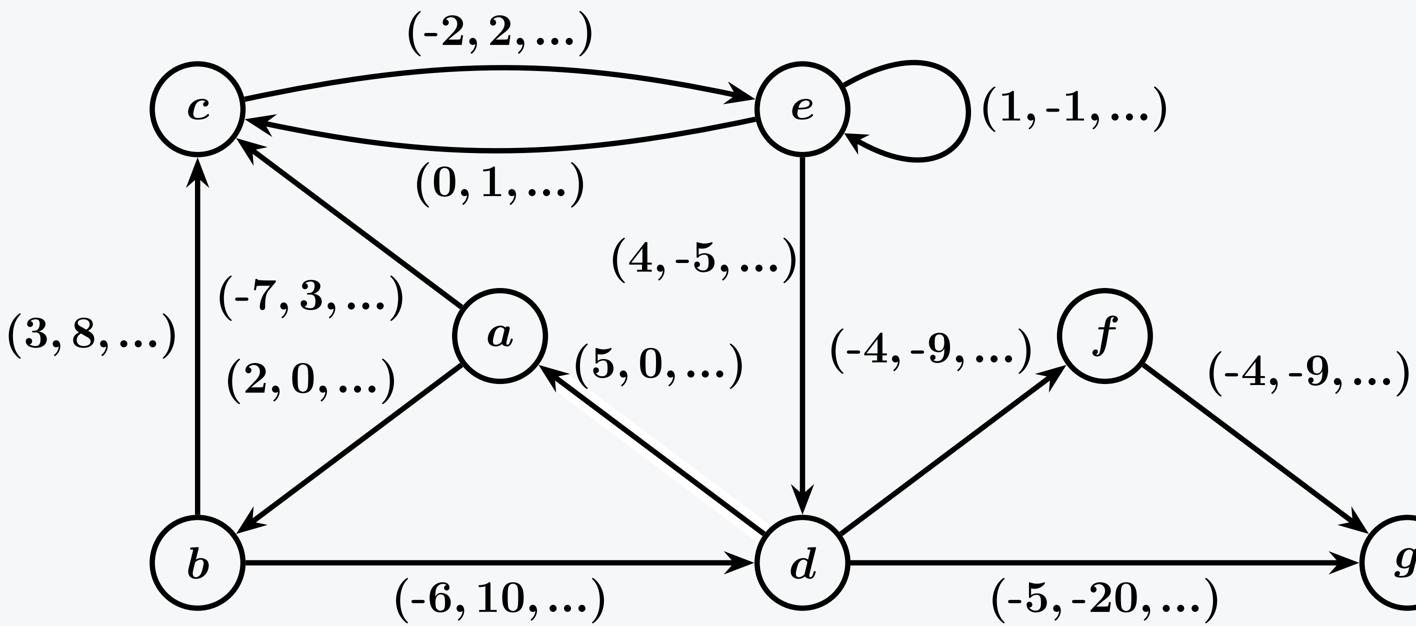
YES!

(the counters must remain nonnegative at all times)

Reachability in VASS



Reachability in VASS



Reachability problem: does there exist a run from $p(\mathbf{u})$ to $q(\mathbf{v})$?

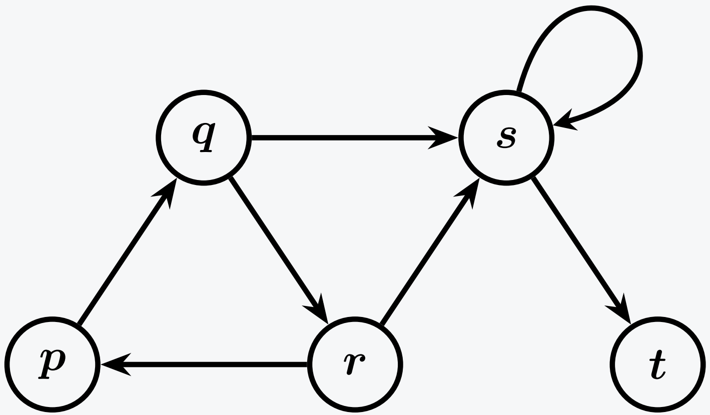
Flat VASS

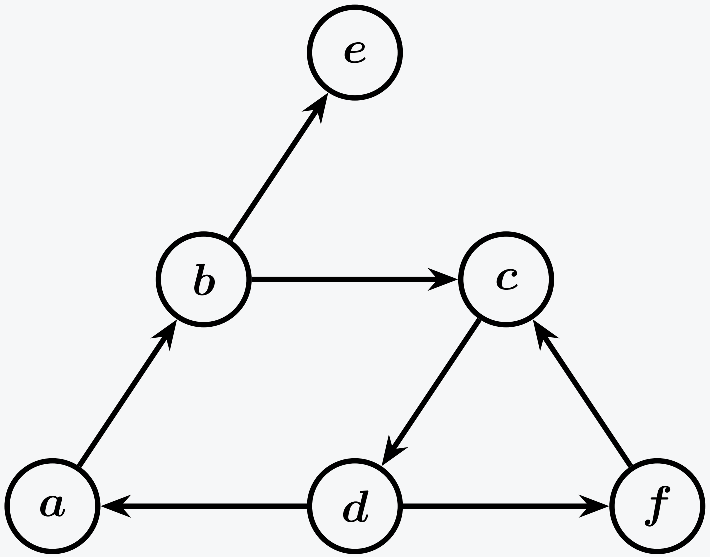
Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Flat VASS

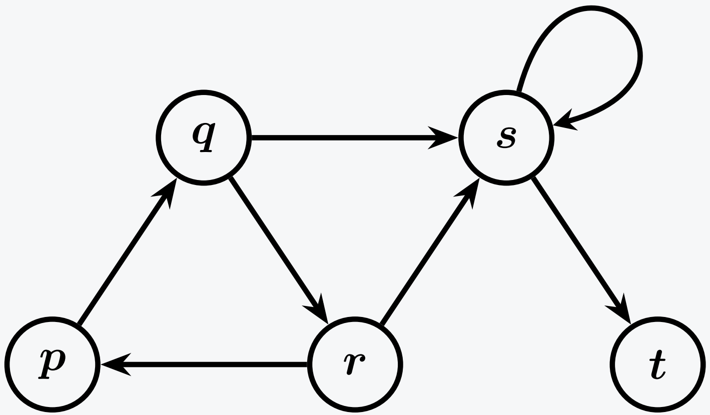
Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.



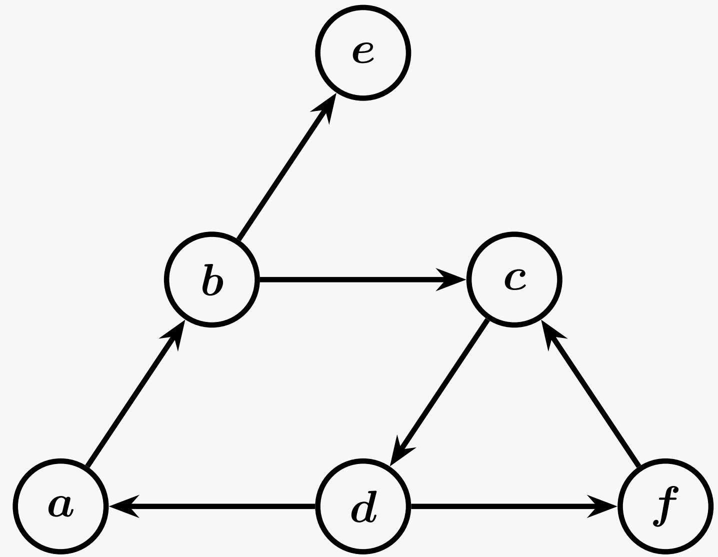


Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

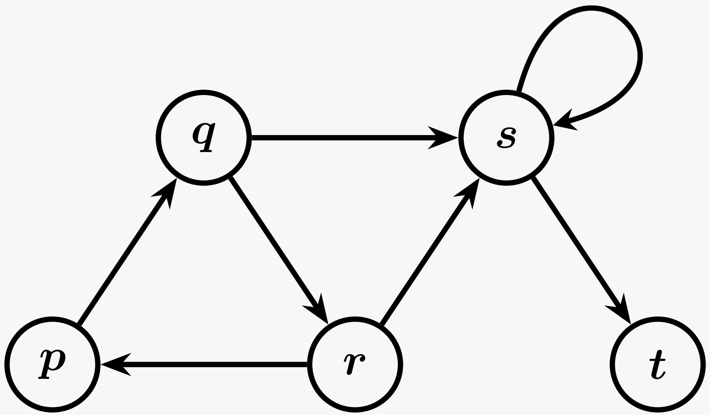


Flat :)

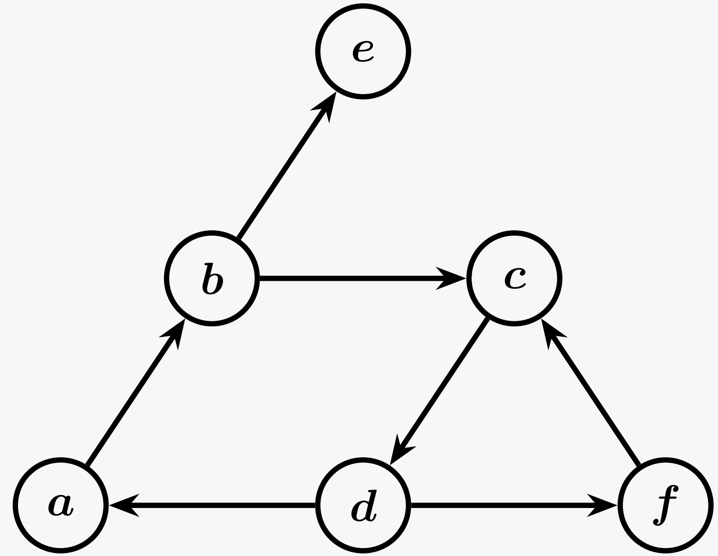


Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.



Flat :)



Not flat :(

Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

Flat VASS

Definition (Flat)**.** For every state $\boldsymbol{q} \in \boldsymbol{Q}$, there is at most one simple cycle that contains $\boldsymbol{q}$.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

Proof sketch. Let $(\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}, \boldsymbol{t})$ be an instance of subset sum (with multiplicities).

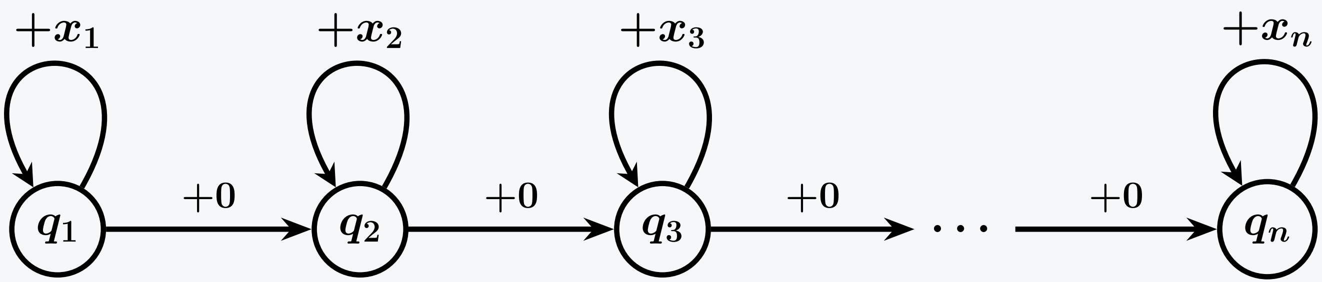
Flat VASS

Definition (Flat). For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

Proof sketch. Let $(\{x_1, \ldots, x_n\}, t)$ be an instance of subset sum (with multiplicities).



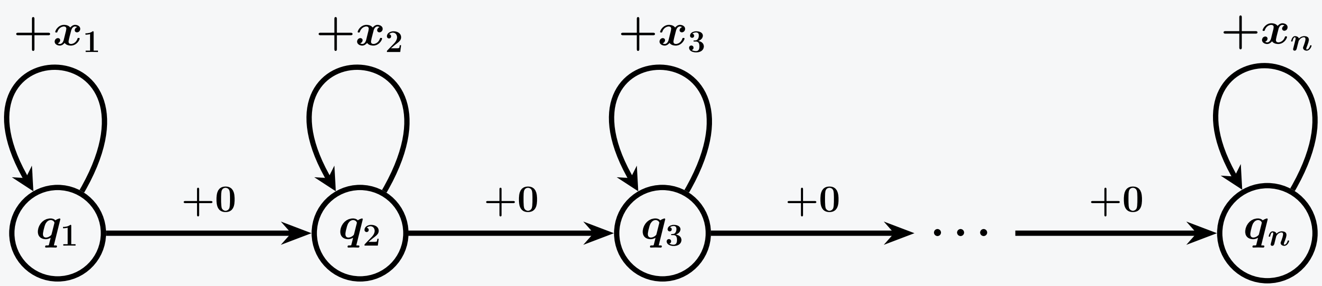
Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

Proof sketch. Let $(\{x_1, \ldots, x_n\}, t)$ be an instance of subset sum (with multiplicities).



There exist $k_1, \ldots, k_n$ such that $t = \Sigma k_i \cdot x_i$ if and only if there is a run from $q_1(0)$ to $q_n(t)$.

Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

Flat VASS

Definition (Flat). For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]
[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

What is the complexity of reachability in unary flat VASS?

Flat VASS

Definition (Flat)**.** For every state $q \in Q$, there is at most one simple cycle that contains q.

Theorem. Reachability in flat VASS is in NP (even with binary encoding). [Fribourg and Olsén '97]

[Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Theorem. Reachability in binary flat 1-VASS is NP-hard. [Rosier and Yen '85]

What is the complexity of reachability in unary flat VASS?

[Blondin, Finkel, Göller, Haase, and McKenzie '15]

[Englert, Lazić, and Totzke '16]

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
$\mathbf{1}$	NP-complete [Rosier and Yen '85]	

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	
2	NP-complete	

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
$\mathbf{1}$	NP-complete [Rosier and Yen '85]	
$\mathbf{2}$	NP-complete	
	$\vdots$	

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
$\mathbf{1}$	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
$\mathbf{2}$	NP-complete	
	$\vdots$	

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
$\mathbf{1}$	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
$\mathbf{2}$	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
	$\vdots$	

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
2	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
	⋮	
7		NP-complete [Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
2	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
	⋮	
5		NP-complete [Dubiak '20]
7		NP-complete [Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
2	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
	⋮	
5		NP-complete [Dubiak '20]
6		NP-complete
7		NP-complete [Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
2	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
	⋮	
4		NP-complete [Czerwiński and Orlikowski '22]
5		NP-complete [Dubiak '20]
6		NP-complete
7		NP-complete [Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

Complexity of Reachability in Flat VASS

Dimension	Binary encoding	Unary encoding
1	NP-complete [Rosier and Yen '85]	NL-complete [Valiant and Paterson '73]
2	NP-complete	NL-complete [Englert, Lazić, and Totzke '16]
3		NP-complete This presentation!
4	⋮	NP-complete [Czerwiński and Orlikowski '22]
5		NP-complete [Dubiak '20]
6		NP-complete
7		NP-complete [Czerwiński, Lasota, Lazić, Leroux, and Mazowiecki '20]

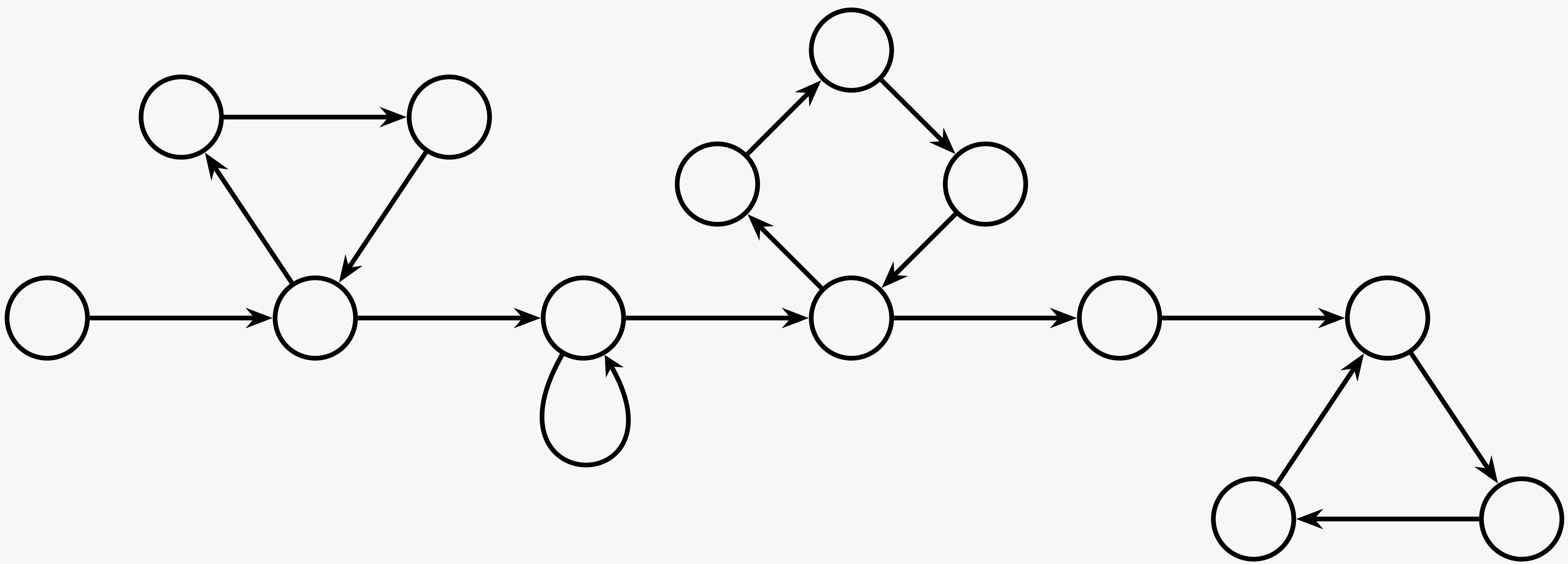
~~Flat VASS~~ Linear Path Schemes

~~Flat VASS~~ Linear Path Schemes

Definition (LPS)**.** A VASS where the states and transitions form a simple path between disjoint cycles.

~~Flat VASS~~ Linear Path Schemes

Definition (LPS)**.** A VASS where the states and transitions form a simple path between disjoint cycles.



~~Flat VASS~~ Linear Path Schemes

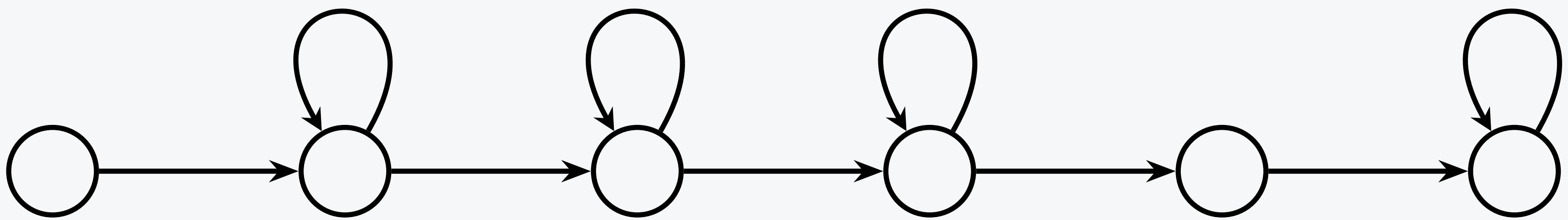
Definition (LPS)**.** A VASS where the states and transitions form a simple path between disjoint cycles.

Definition (SLPS)**.** A *Simple* LPS has cycles of length one ("self-loops").

~~Flat VASS~~ Linear Path Schemes

Definition (LPS)**.** A VASS where the states and transitions form a simple path between disjoint cycles.

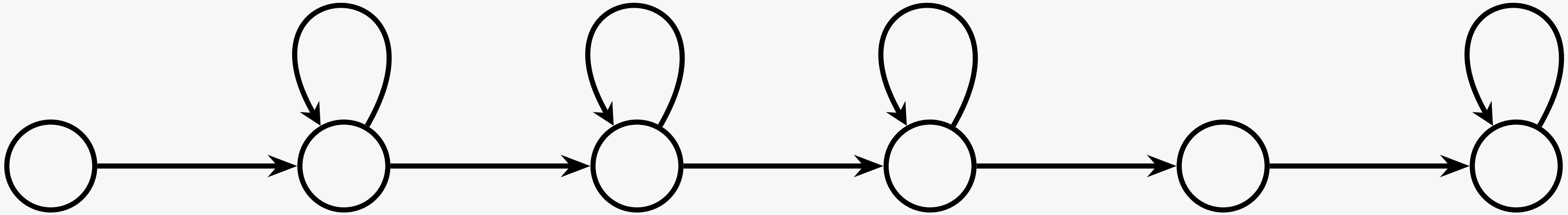
Definition (SLPS)**.** A *Simple* LPS has cycles of length one ("self-loops").



~~Flat VASS~~ Linear Path Schemes

Definition (LPS)**.** A VASS where the states and transitions form a simple path between disjoint cycles.

Definition (SLPS)**.** A *Simple* LPS has cycles of length one ("self-loops").



For $d \geq 3$, is reachability in unary d-dimensional linear path schemes in P?

[Englert, Lazić, and Totzke '16]

[Leroux '21]

Main Contribution

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

   - Let $p_1, \ldots, p_k$ be the first k primes.
   - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

- Let $p_1, \ldots, p_k$ be the first k primes.
- Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

   - Let $p_1, \ldots, p_k$ be the first k primes.
   - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

   - To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n - 1$.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

   - Let $p_1, \ldots, p_k$ be the first k primes.
   - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

   - To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n - 1$.
   - Instead, check $p_i \nmid n - 2$ AND $p_i \nmid n - 3$ AND $\cdots$ AND $p_i \nmid n - (p_i - 1)$.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

  - Let $p_1, \ldots, p_k$ be the first k primes.
  - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

  - To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n - 1$.
  - Instead, check $p_i \nmid n - 2$ AND $p_i \nmid n - 3$ AND $\cdots$ AND $p_i \nmid n - (p_i - 1)$.

3) Again, use a conjunction of non-divisibility tests to verify that n represents a satisfying assignment.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

   - Let $p_1, \ldots, p_k$ be the first k primes.
   - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

   - To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n - 1$.
   - Instead, check $p_i \nmid n - 2$ AND $p_i \nmid n - 3$ AND $\cdots$ AND $p_i \nmid n - (p_i - 1)$.

3) Again, use a conjunction of non-divisibility tests to verify that n represents a satisfying assignment.

   - A clause $x_1 \vee \neg x_2 \vee x_3$ is satisfied if $n \equiv 1 \bmod 2$ OR $n \equiv 0 \bmod 3$ OR $n \equiv 1 \bmod 5$.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

   - Let $p_1, \ldots, p_k$ be the first k primes.
   - Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

   - To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n - 1$.
   - Instead, check $p_i \nmid n - 2$ AND $p_i \nmid n - 3$ AND $\cdots$ AND $p_i \nmid n - (p_i - 1)$.

3) Again, use a conjunction of non-divisibility tests to verify that n represents a satisfying assignment.

   - A clause $x_1 \vee \neg x_2 \vee x_3$ is satisfied if $n \equiv 1 \bmod 2$ OR $n \equiv 0 \bmod 3$ OR $n \equiv 1 \bmod 5$.
   - Instead, check $2 \cdot 3 \cdot 5 \nmid n - 10$.

Main Contribution

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof approach. Reduce from 3-SAT. Suppose φ has k variables $(x_1, \ldots, x_k)$ and m clauses.

1) Use "Chinese remainder encoding" for SAT.

- Let $p_1, \ldots, p_k$ be the first k primes.
- Let $n \in \mathbb{N}$ such that $n \equiv 0 \bmod p_i \iff x_i$ is false and $n \equiv 1 \bmod p_i \iff x_i$ is true.

2) Use a conjunction of non-divisibility tests to verify that n represents a valid assignment.

- To verify that $n \equiv 0 \bmod p_i$ OR $n \equiv 1 \bmod p_i$, check $p_i \mid n$ OR $p_i \mid n-1$.
- Instead, check $p_i \nmid n-2$ AND $p_i \nmid n-3$ AND $\cdots$ AND $p_i \nmid n-(p_i-1)$.

3) Again, use a conjunction of non-divisibility tests to verify that n represents a satisfying assignment.

- A clause $x_1 \vee \neg x_2 \vee x_3$ is satisfied if $n \equiv 1 \bmod 2$ OR $n \equiv 0 \bmod 3$ OR $n \equiv 1 \bmod 5$.
- Instead, check $2 \cdot 3 \cdot 5 \nmid n - 10$.

Non-Divisibility Testing Simple Linear Path Schemes

Non-Divisibility Testing Simple Linear Path Schemes

Suppose we want to perform a non-divisibility test $\boldsymbol{v} \nmid \boldsymbol{7}$.

Non-Divisibility Testing Simple Linear Path Schemes

Suppose we want to perform a non-divisibility test $\boldsymbol{v} \nmid \mathbf{7}$.

Let's construct a 2-SLPS with zero tests that:

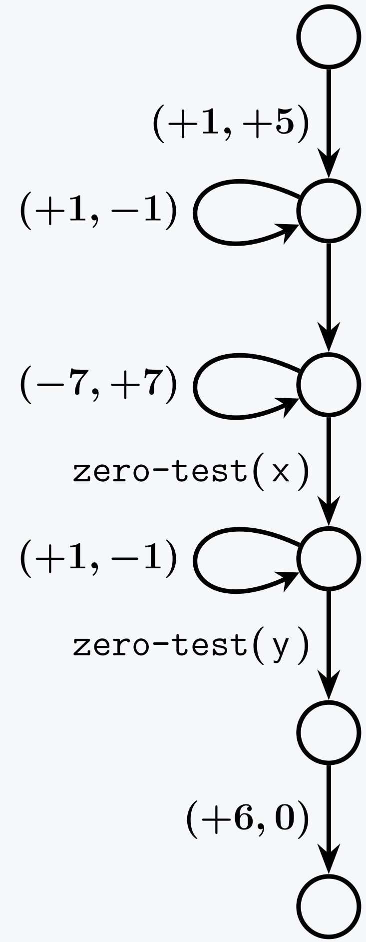
- starts with $\mathsf{x} = \boldsymbol{v}$, $\mathsf{y} = \mathbf{0}$,
- can only be passed if $\boldsymbol{v} \nmid \mathbf{7}$, and
- ends with $\mathsf{x} = \boldsymbol{v}$, $\mathsf{y} = \mathbf{0}$.

Non-Divisibility Testing Simple Linear Path Schemes

Suppose we want to perform a non-divisibility test $\boldsymbol{v} \nmid \mathbf{7}$.

Let's construct a 2-SLPS with zero tests that:

- starts with x $= \boldsymbol{v}$, y $= \mathbf{0}$,
- can only be passed if $\boldsymbol{v} \nmid \mathbf{7}$, and
- ends with x $= \boldsymbol{v}$, y $= \mathbf{0}$.



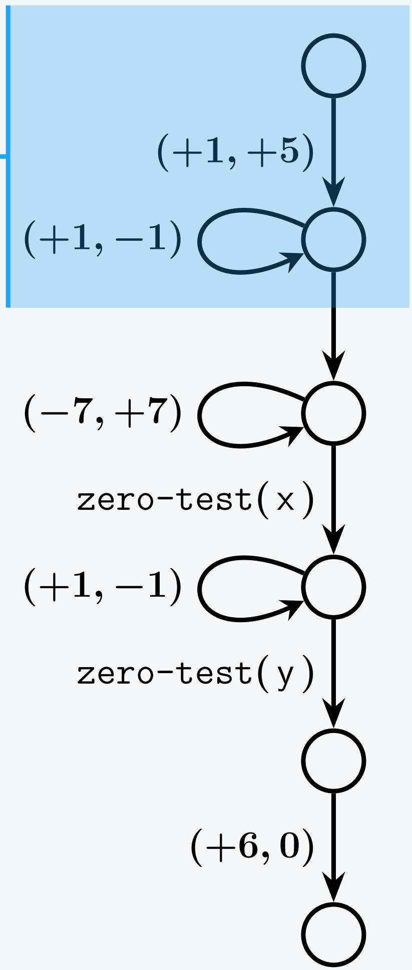
Non-Divisibility Testing Simple Linear Path Schemes

Suppose we want to perform a non-divisibility test $v \nmid 7$.

Let's construct a 2-SLPS with zero tests that:

- starts with x $= v$, y $= 0$,
- can only be passed if $v \nmid 7$, and
- ends with x $= v$, y $= 0$.

(i) Choose $r \in \{1, 2, 3, 4, 5, 6\}$ ...



Non-Divisibility Testing Simple Linear Path Schemes

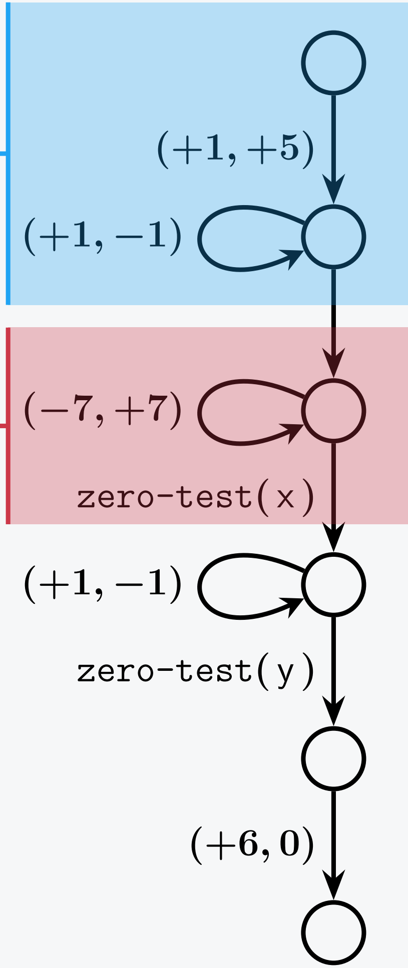
Suppose we want to perform a non-divisibility test $v \nmid 7$.

Let's construct a 2-SLPS with zero tests that:

- starts with $x = v$, $y = 0$,
- can only be passed if $v \nmid 7$, and
- ends with $x = v$, $y = 0$.

(i) Choose $r \in \{1, 2, 3, 4, 5, 6\}$ ...

(ii) ... such that $7 \mid v + r$.



Non-Divisibility Testing Simple Linear Path Schemes

Suppose we want to perform a non-divisibility test $v \nmid 7$.

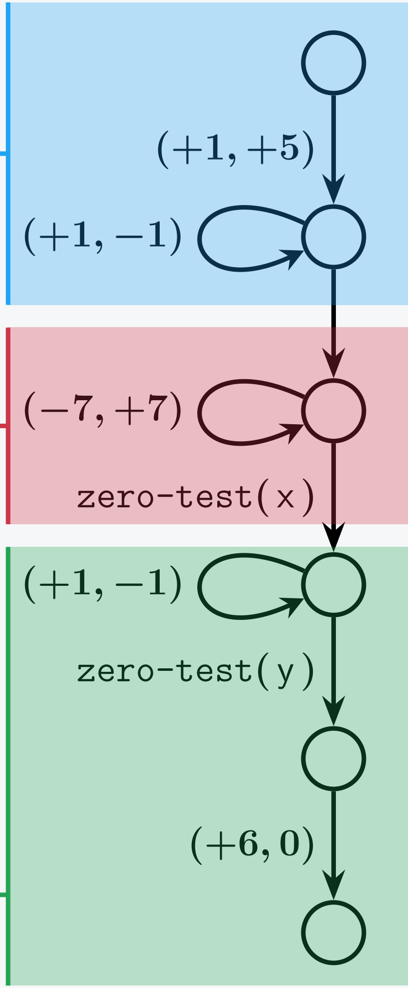
Let's construct a 2-SLPS with zero tests that:

- starts with x $= v$, y $= 0$,
- can only be passed if $v \nmid 7$, and
- ends with x $= v$, y $= 0$.

(i) Choose $r \in \{1, 2, 3, 4, 5, 6\}$ ...

(ii) ... such that $7 \mid v + r$.

(iii) Restore x $= v$, y $= 0$.



Simulating Zero Tests

Simulating Zero Tests

Lemma 2.2 (Controlling Counter Technique). *Let $\mathcal{Z}$ be a d-VASS with zero tests and let $s(\mathbf{x}), t(\mathbf{y})$ be two configurations. Suppose $\mathcal{Z}$ has the property that on any accepting run from $s(\mathbf{x})$ to $t(\mathbf{y})$, at most m zero tests are performed on each counter. Then there exists a $(d+1)$-VASS $\mathcal{V}$ and two configurations $s'(\mathbf{0}), t'(\mathbf{y}')$ such that:*

(1) $s(\mathbf{x}) \xrightarrow{*}_{\mathcal{Z}} t(\mathbf{y})$ *if and only if* $s'(\mathbf{0}) \xrightarrow{*}_{\mathcal{V}} t'(\mathbf{y}')$,

(2) $\mathcal{V}$ *can be constructed in* $\mathcal{O}((size(\mathcal{Z}) + \|x\|) \cdot (m+1)^d)$ *time, and*

(3) $\|\mathbf{y}'\| \leq \|\mathbf{y}\|$.

Moreover, if $\mathcal{Z}$ is a flat VASS or a (simple) linear path scheme in which no zero-testing transition lies on a cycle, then $\mathcal{V}$ can be assumed to be a flat VASS or a (simple) linear path scheme, respectively.

[Czerwiński and Orlikowski '21] [Chistikov, Czerwiński, Mazowiecki, Orlikowski, S., and Węgrzycki '24]

Simulating Zero Tests

Lemma 2.2 (Controlling Counter Technique). *Let $\mathcal{Z}$ be a d-VASS with zero tests and let $s(\mathbf{x}), t(\mathbf{y})$ be two configurations. Suppose $\mathcal{Z}$ has the property that on any accepting run from $s(\mathbf{x})$ to $t(\mathbf{y})$, at most m zero tests are performed on each counter. Then there exists a $(d+1)$-VASS $\mathcal{V}$ and two configurations $s'(\mathbf{0}), t'(\mathbf{y}')$ such that:*

(1) $s(\mathbf{x}) \xrightarrow{*}_{\mathcal{Z}} t(\mathbf{y})$ *if and only if* $s'(\mathbf{0}) \xrightarrow{*}_{\mathcal{V}} t'(\mathbf{y}')$,

(2) $\mathcal{V}$ *can be constructed in* $\mathcal{O}((\mathit{size}(\mathcal{Z}) + \|x\|) \cdot (m+1)^d)$ *time, and*

(3) $\|\mathbf{y}'\| \leq \|\mathbf{y}\|$.

Moreover, if $\mathcal{Z}$ is a flat VASS or a (simple) linear path scheme in which no zero-testing transition lies on a cycle, then $\mathcal{V}$ can be assumed to be a flat VASS or a (simple) linear path scheme, respectively.

[Czerwiński and Orlikowski '21] [Chistikov, Czerwiński, Mazowiecki, Orlikowski, S., and Węgrzycki '24]

Takeway message: A "small" number of zero tests can be simulated by an additional counter.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP. For NP-hardness:

- Reduce from 3-SAT and use Chinese remainder encoding.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP. For NP-hardness:

- Reduce from 3-SAT and use Chinese remainder encoding.

- Obtain an equivalent conjunction of non-divisibility tests.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP. For NP-hardness:

- Reduce from 3-SAT and use Chinese remainder encoding.
- Obtain an equivalent conjunction of non-divisibility tests.
- For each non-divisibility test, construct the corresponding unary 2-SLPS with zero tests.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP. For NP-hardness:

- Reduce from 3-SAT and use Chinese remainder encoding.

- Obtain an equivalent conjunction of non-divisibility tests.

- For each non-divisibility test, construct the corresponding unary 2-SLPS with zero tests.

- Prepend a $\mathsf{x}\,\boldsymbol{+1}$ self-loop to allow the assignment value $\mathsf{x} = \boldsymbol{v}$ to be guessed.

Recap – Proof of Main Theorem

Theorem. Reachability in unary 3-SLPS is NP-complete.

Proof sketch: Recall that reachability in (binary) flat VASS is in NP. For NP-hardness:

- Reduce from 3-SAT and use Chinese remainder encoding.

- Obtain an equivalent conjunction of non-divisibility tests.

- For each non-divisibility test, construct the corresponding unary 2-SLPS with zero tests.

- Prepend a $\mathsf{x}+1$ self-loop to allow the assignment value $\mathsf{x} = v$ to be guessed.

- Use the controlling counter technique to obtain unary 3-SLPS for the SAT instance.

The Tractability Border of Reachability in Simple Vector Addition Systems with States

The Tractability Border of Reachability in Simple Vector Addition Systems with States

Theorem. Reachability in unary 3-SLPS is NP-complete.

The Tractability Border of Reachability in Simple Vector Addition Systems with States

Theorem. Reachability in unary 3-SLPS is NP-complete.

Theorem. Reachability in unary *ultraflat* 4-VASS is NP-complete.

The Tractability Border of Reachability in Simple Vector Addition Systems with States

Theorem. Reachability in unary 3-SLPS is NP-complete.

Theorem. Reachability in unary *ultraflat* 4-VASS is NP-complete.

Theorem. Reachability in *unitary* inverse-Ackermann-dimensional SLPS is NP-complete.

The Tractability Border of Reachability in Simple Vector Addition Systems with States

Theorem. Reachability in unary 3-SLPS is NP-complete.

Theorem. Reachability in unary *ultraflat* 4-VASS is NP-complete.

Theorem. Reachability in *unitary* inverse-Ackermann-dimensional SLPS is NP-complete.

Theorem. Reachability in unary 2-SLPS with *binary encoded initial and target configurations* is in P.

The Tractability Border of Reachability in Simple Vector Addition Systems with States

Theorem. Reachability in unary 3-SLPS is NP-complete.

Theorem. Reachability in unary *ultraflat* 4-VASS is NP-complete.

Theorem. Reachability in *unitary* inverse-Ackermann-dimensional SLPS is NP-complete.

Theorem. Reachability in unary 2-SLPS with *binary encoded initial and target configurations* is in P.

Thank You!

Presented by Henry Sinclair-Banks, University of Warwick, UK

KIT, Karlsruhe, Germany

Karlsruher Institut für Technologie

henry.sinclair-banks.com