Lower bounds for reachability in VASS in fixed dimension

Wojciech Czerwiński
Sławomir Lasota
University of Warsaw

Jerome Leroux
University of Bordeaux

Agata Dubiak
Łukasz Orlikowski
University of Warsaw

Ranko Lazic
University of Warwick

Filip Mazowiecki
MPI Saarbruecken

Infinity’20, online, 2020-07-07
Plan

1. Vector addition systems with states (VASS) and the reachability problem
2. Lower bounds in small fixed dimensions
3. Lower bounds in large fixed dimensions
Many faces of vector addition systems with states

- Vector addition systems with states VASS [Hopcroft, Pansiot '79]:
  \[
  (-1, 1, 0) \xrightarrow{p} (0, 0, -1) \xrightarrow{q} (1, -1, 0)
  \]

- Petri nets [Petri 1962]:

- Counter programs without 0-tests:
  1: loop
  2: loop
  3: \(x \leftarrow 1\) \(y \rightarrow 1\)
  4: loop
  5: \(x \rightarrow 1\) \(y \leftarrow 1\)
  6: \(z \leftarrow 1\)

- VAS [Karp, Miller '69]
- Multiset rewriting
- ...
Counter programs without zero tests

counters are nonnegative integer variables initially all equal zero

a sequence of commands of the form:

\[ \begin{align*}
    x & \mathbf{+=} n & \text{(increment counter } x \text{ by } n) \\
    x & \mathbf{-=} n & \text{(decrement counter } x \text{ by } n) \\
    \text{goto } L & \text{ or } L' & \text{(jump to either line } L \text{ or line } L')
\end{align*} \]

except for the very last command which is of the form:

\[ \text{halt if } x_1, \ldots, x_l = 0 \quad \text{(terminate provided all the listed counters are zero)} \]

abort if \( x < n \)

otherwise abort

Example:

1: \( x' + = 100 \)
2: goto 5 or 3
3: \( x + = 1 \) \( x' - = 1 \) \( y + = 2 \)
4: goto 2
5: halt if \( x' = 0 \).

initially: \( x' = x = y = 0 \)

finally: \( x' = 0 \) \( x = 100 \) \( y = 200 \)
Loop programs

1: \( x' \) += 100
2: \textbf{goto} 5 or 3
3: \( x \) += 1 \( x' \) -= 1 \( y \) += 2
4: \textbf{goto} 2
5: \textbf{halt if} \( x' = 0 \).

1: \( x' \) += 100
2: \textbf{loop}
3: \( x \) += 1 \( x' \) -= 1 \( y \) += 2
4: \textbf{halt if} \( x' = 0 \).
Minsky machines

the conditional jump of Minsky machines

\[
\text{if } x = 0 \text{ then goto } L \text{ else } x -= 1
\]

is simulated by counter program with zero tests:

1: goto 2 or 4
2: zero? x
3: goto L
4: x -= 1
Reachability (and coverability)

**Reachability problem:** given a counter program without zero tests

1: \( x' += 100 \)
2: \text{goto 5 or 3} \)
3: \( x += 1 \quad x' -= 1 \quad y += 2 \)
4: \text{goto 2}
5: \text{halt if } x' = 0.

Can it terminate (execute its halt command)?

**Coverability problem:** given a counter program without zero tests with trivial halt command

1: \( x' += 100 \)
2: \text{goto 5 or 3} \)
3: \( x += 1 \quad x' -= 1 \quad y += 2 \)
4: \text{goto 2}
5: \text{halt.}

Can it terminate (reach its halt command)?
Complexity of reachability (and coverability)

Time/space needed is at least \( F_3(n) = 2^2 \ldots 2^{2^{2^{2\ldots}}} \) \( \in O(n) \)

<table>
<thead>
<tr>
<th>coverability</th>
<th>reachability</th>
</tr>
</thead>
<tbody>
<tr>
<td>lower bound</td>
<td>EXPSPACE</td>
</tr>
<tr>
<td>[Lipton '76]</td>
<td></td>
</tr>
<tr>
<td>upper bound</td>
<td>EXPSPACE</td>
</tr>
<tr>
<td>[Rackoff '78]</td>
<td></td>
</tr>
</tbody>
</table>

Time/space needed is at most \( F_\omega(n) \)

see Jerome Leroux’s invited talk on Thursday
2. Lower bounds for reachability in **small** fixed dimensions

\[
\begin{align*}
\text{dimension} &= \text{number of counters:} \\
(-1, 1, 0) &\xrightarrow{p} (0, 0, 0) \\
&\xleftarrow{(0, 0, -1)} q &\xrightarrow{q} (1, -1, 0)
\end{align*}
\]

1: loop
2: loop
3: \(x \leftarrow 1\) \(y \rightarrow 1\)
4: loop
5: \(x \rightarrow 1\) \(y \leftarrow 1\)
6: \(z \leftarrow 1\)
Reachability in dimension 1 and 2

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Unary</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NL*</td>
<td>NP*</td>
</tr>
<tr>
<td></td>
<td>[Valiant, Peterson '75]</td>
<td>[Haase, Kreutzer, Ouaknine, Worrell '09]</td>
</tr>
<tr>
<td>2</td>
<td>NL*</td>
<td>PSPACE*</td>
</tr>
<tr>
<td></td>
<td>[Englert, Lazic, Totzke '16]</td>
<td>[Blondin, Finkel, Goeller, Haase, McKenzie '15]</td>
</tr>
</tbody>
</table>

*complete

- Shortest run has **polynomial** length
- Shortest run has **exponential** length
- Encoding of integers
- Effectively flattable in dimension 2

Upper bounds similar to dimension 2 for every fixed dimension? ...at least for flat counter programs?
Flat = no nested loops

1: loop
2: \( x -= 1 \) \( y += 1 \)
3: loop
4: \( x += 2 \) \( y -= 1 \)
5: \( x += 1 \)
6: loop
7: \( x -= 1 \) \( y += 1 \)
8: loop
9: \( x += 2 \) \( y -= 1 \)

1: loop
2: \( \text{loop} \)
3: \( x -= 1 \) \( y += 1 \)
4: \( \text{loop} \)
5: \( x += 1 \) \( y -= 1 \)
6: \( z -= 1 \)
Shortest runs in small dimensions
[Czerwiński, L., Lazic, Leroux, Mazowiecki ’20]

<table>
<thead>
<tr>
<th>dim 1</th>
<th>unary flat</th>
<th>unary</th>
<th>binary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>poly*</td>
<td>poly*</td>
<td>exp*</td>
</tr>
<tr>
<td>dim 2</td>
<td>poly*</td>
<td>poly*</td>
<td>exp*</td>
</tr>
<tr>
<td>dim 3</td>
<td>exp*</td>
<td>exp*</td>
<td>?</td>
</tr>
<tr>
<td>dim 4</td>
<td></td>
<td></td>
<td>2-exp*</td>
</tr>
<tr>
<td>dim 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*upper bound
*lower bound

Key ingredient: computing **exactly** large numbers
Exponential shortest run in unary flat dim 3

1: \( x += 1 \quad y += 1 \)
2: \( \text{loop} \)
3: \( x += 1 \quad y += 1 \)
4: \( \text{for } i := n \ \text{down to} \ 1 \ \text{do} \)
5: \( \quad \text{loop} \)
6: \( \quad \quad x -= 1 \quad z += 1 \)
7: \( \quad \text{loop} \)
8: \( \quad x += i + 1 \quad z -= i \)
9: \( \quad \text{for } i := n \ \text{down to} \ 1 \ \text{do} \)
10: \( \quad \text{loop} \)
11: \( x -= n + 1 \quad y -= 1 \)

\text{objective:}

- halts iff \( y \) divisible by \( 1, 2, \ldots, n \)
- \( x = (n+1) \cdot y \) iff all multiplications exact
- \( x = (n+1) \cdot y \) divisible by \( 1, 2, \ldots, n \)
- \( x = (n+1) \cdot y \) iff all inner loops iterated maximally

\( x <= \frac{n+1}{n} \cdot \frac{n}{n-1} \cdot \ldots \cdot \frac{3}{2} \cdot \frac{2}{1} \cdot y \)

program size \( O(n^2) \), shortest run \( 2^{O(n^c)} \)
Complexity lower bounds for reachability

[Czerwiński, L., Lazic, Leroux, Mazowiecki ’19, ’20]
[Dubiak ’20]

<table>
<thead>
<tr>
<th></th>
<th>unary flat</th>
<th>unary</th>
<th>binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>dim 1</td>
<td>NL*</td>
<td>NL*</td>
<td>NP*</td>
</tr>
<tr>
<td>dim 2</td>
<td>NL* NL</td>
<td>NL* NL</td>
<td>PSPACE</td>
</tr>
<tr>
<td>dim 3</td>
<td>NL</td>
<td>NL</td>
<td>PSPACE</td>
</tr>
<tr>
<td>dim 4</td>
<td>NL</td>
<td>NL</td>
<td>PSPACE</td>
</tr>
<tr>
<td>dim 5</td>
<td>NP* NL</td>
<td>NL</td>
<td>PSPACE</td>
</tr>
<tr>
<td>...</td>
<td>NP* NL</td>
<td>NL</td>
<td>PSPACE</td>
</tr>
<tr>
<td>dim 13</td>
<td>NP* NL</td>
<td>PSPACE</td>
<td>EXPSPACE</td>
</tr>
<tr>
<td>dim 14</td>
<td>NP* NL</td>
<td>EXPSPACE</td>
<td>2-EXPSPACE</td>
</tr>
<tr>
<td>dim 15</td>
<td>NP* NL</td>
<td>2-EXPSPACE</td>
<td>3-EXPSPACE</td>
</tr>
<tr>
<td>...</td>
<td>... NL</td>
<td>... NL</td>
<td>PSPACE</td>
</tr>
</tbody>
</table>

*complete
*hard

What about coverability?

[Dubiak '20]
3. Lower bounds for reachability in large fixed dimensions
Parametric lower bound for reachability
[Czerwiński, L., Lazic, Leroux, Mazowiecki ’19]
[Czerwiński, L., Orlikowski ’??]

<table>
<thead>
<tr>
<th>unary flat</th>
<th>unary</th>
<th>binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>dim 13</td>
<td>PSPACE*</td>
<td></td>
</tr>
<tr>
<td>dim 14</td>
<td>EXPSPACE*</td>
<td></td>
</tr>
<tr>
<td>dim 15</td>
<td>2-EXPSPACE*</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Minsky machine $M$ of size $n$
with counters bounded by

\[
\left\{ 2^{h+1} \right\} \quad \{ \ldots \} \quad 2 \quad \ldots \quad n \]

Counter program of size $O(n+h)$
with $O(h) = h+13$ counters

Counter program of size $O(n+\log h)$
with $O(\log h)$ counters

Improved reduction
Parametric lower bound for reachability
[Czerwiński, L., Lazic, Leroux, Mazowiecki ’19]
[Czerwiński, L., Orlikowski ’??]

Reachability problem for programs of size $n$ with $d$ counters needs space at least

$$2^{2^2 \cdots 2^2} \{O(d) \} \quad d - 13$$

improved lower bound:

$$2^{2^2 \cdots 2^2} \{2O(d) \} \quad 2^{(d-13)/3}$$
Exponential amplifier

1:  \( i += 1 \quad x += 1 \quad y += 1 \quad b += 1 \quad c += 1 \quad d += 1 \)
2:  \textbf{loop}
3:  \( x += 1 \quad y += 1 \quad c += 1 \quad d += 1 \)
4:  \textbf{loop}
5:    \textbf{loop}
6:      \( c -= 1 \quad c' += 1 \)
7:    \textbf{loop at most} b \textbf{times}
8:      \( x -= i \quad d -= i \quad x' += i + 1 \)
9:    \textbf{loop}
10: \( b -= 1 \quad b' += i + 1 \)
11: \textbf{loop}
12: \( b' -= 1 \quad b += 1 \)
13: \textbf{loop}
14: \( c' -= 1 \quad c += 1 \)
15: \textbf{loop at most} b \textbf{times}
16: \( x' -= 1 \quad x += 1 \quad d += 1 \)
17: \( i += 1 \)
18: \textbf{zero?} \( i \)
19: \textbf{loop}
20: \( x -= i \quad y -= 1 \)
21: \textbf{halt if} \( y = 0 \)

\( x \rightarrow x' \): starting with \( x>0 \) and \( x' = 0 \), computes \( x' \) exponentially larger than \( x \) (if \( x = 0 \) at the end)

if so, also all other counters are forcibly 0 at the end

\( <\text{body}> \)
Composing exponential amplifiers

\[ x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \]

Relay-race

\[ \text{simulate } M \text{ using } x_3 \]

halt if \( x_0, x_1, x_2, x_3 = 0 \)

| \hline
<table>
<thead>
<tr>
<th>unary flat</th>
<th>unary</th>
<th>binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>dim 13</td>
<td>PSPACE*</td>
<td></td>
</tr>
<tr>
<td>dim 14</td>
<td>EXPSPACE*</td>
<td></td>
</tr>
<tr>
<td>dim 15</td>
<td>2-EXPSPACE*</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

...
Relay-race

[Diagram showing a relay race with variables x0, x1, x2, x3, x4, x5, x6, x7, each with a condition to halt if x0, x1, x2, x3, x4, x5, x5, x7 = 0]

**Objective:** decrease the number of counters to logarithmic
Counter recycling?

halt if x0, x1, x2, x3, x4, x5, x7 = 0
halt if \(x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7, y_0, y_1, y_2, y_3, ... = 0\)

 supervisors

\[x_0 + x_1 = y_0\]

\[x_4 + x_5 = y_2\]

\[y_1 = x_2 + x_3\]

\[y_3 = x_6 + x_7\]
Supervisors

\[ x_0 + x_1 = y_0 \]

\[ x_1 + x_2 = y_2 \]

\[ y_1 = x_2 + x_0 \]

\[ y_3 = x_0 + x_1 \]

halt if \( x_0, x_1, x_2, y_0, y_1, y_2, y_3, \ldots = 0 \)
halt if $x_0, x_1, x_2, y_0, y_1, y_2, z_0, z_1, \ldots = 0$

and so on…
Summary

<table>
<thead>
<tr>
<th></th>
<th>unary flat</th>
<th>unary</th>
<th>binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>dim 1</td>
<td><strong>NL</strong>*</td>
<td><strong>NL</strong>*</td>
<td><strong>NP</strong>*</td>
</tr>
<tr>
<td>dim 2</td>
<td><strong>NL</strong>*</td>
<td><strong>NL</strong>*</td>
<td><strong>PSPACE</strong>*</td>
</tr>
<tr>
<td>dim 3</td>
<td><strong>NP</strong>*</td>
<td><strong>PSPACE</strong>*</td>
<td><strong>EXPSPACE</strong>*</td>
</tr>
<tr>
<td>dim 4</td>
<td><strong>NP</strong>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dim 5</td>
<td><strong>NP</strong>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td><strong>NP</strong>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dim 13</td>
<td><strong>NP</strong>*</td>
<td><strong>PSPACE</strong>*</td>
<td><strong>EXPSPACE</strong>*</td>
</tr>
<tr>
<td>dim 14</td>
<td><strong>NP</strong>*</td>
<td><strong>EXPSPACE</strong>*</td>
<td><strong>2-EXPSPACE</strong>*</td>
</tr>
<tr>
<td>dim 15</td>
<td><strong>NP</strong>*</td>
<td><strong>2-EXPSPACE</strong>*</td>
<td><strong>3-EXPSPACE</strong>*</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

I recruit for a fully-funded PhD position in the NCN grant

Reachability problem for $d$-dimensional VASS of size $n$ requires space at least $2^{2^2 \ldots n} \in 2^{O(d)}$

*upper bound
*lower bound

*complete
*hard

thank you!