Walks of Given Length in One-Counter Systems

Henry Sinclair-Banks

WPCCS'21 Presentation Monday 13th December

One-Counter Systems

Directed Graph

- Integer weighted edges

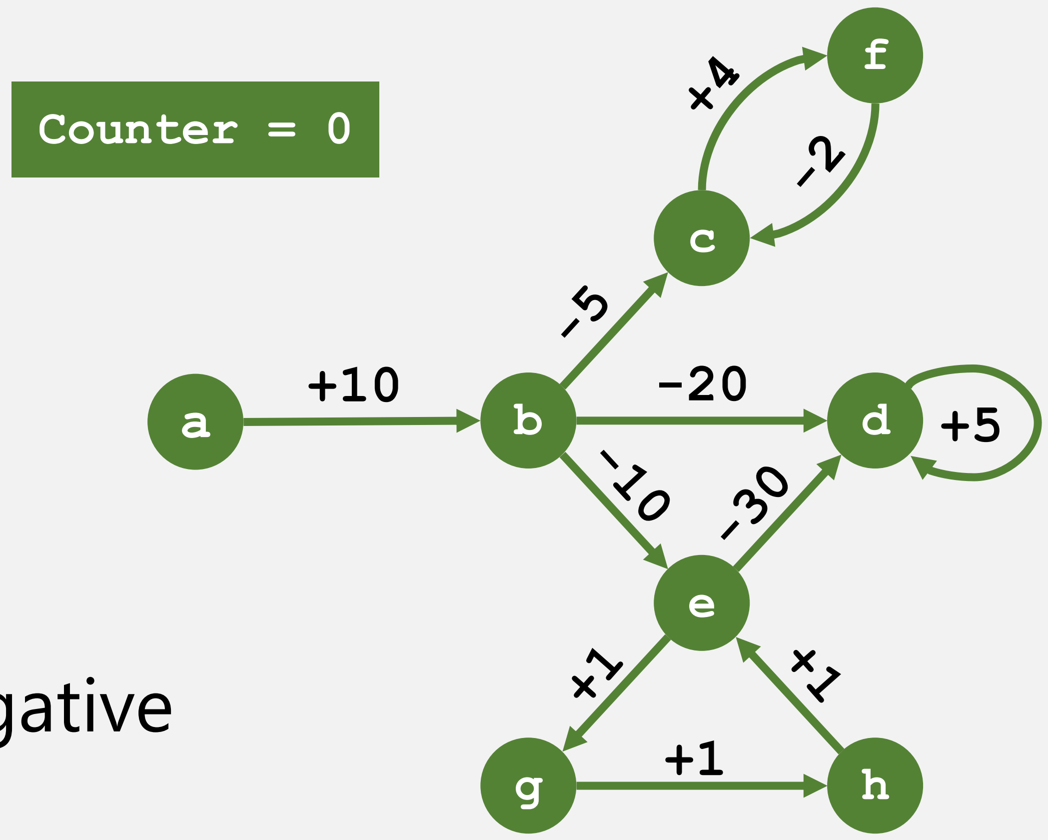
  Represented in binary

- Controls an integer counter

  Counter starts with zero

- Counter must remain non-negative

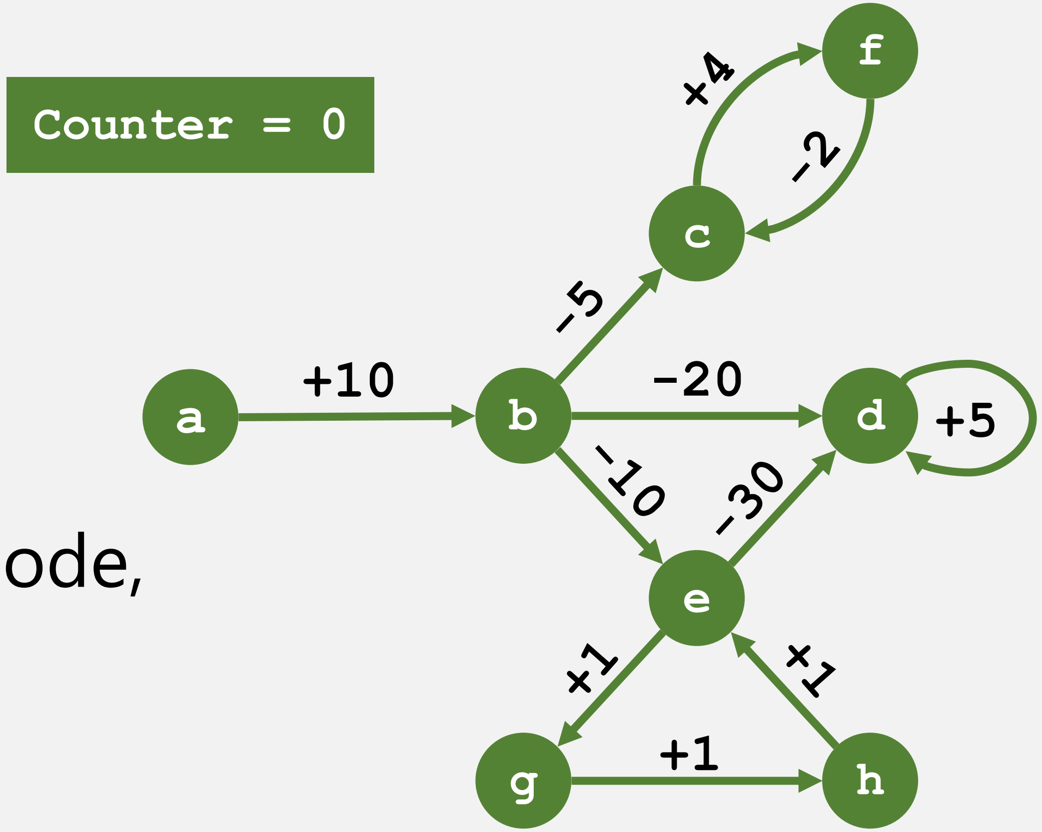
  "Counter condition"



Walks of Given Length

Walks - a sequence of edges
(Valid if counter condition not violated)

Length - the number of edges
(Not the sum of the weights)



<u>INPUT:</u>
$\mathcal{A}$ one-counter system, u start node, v end node, n walk length.
(given in binary)

<u>QUESTION:</u>
Does there exist a valid walk in $\mathcal{A}$ from u to v of length n?

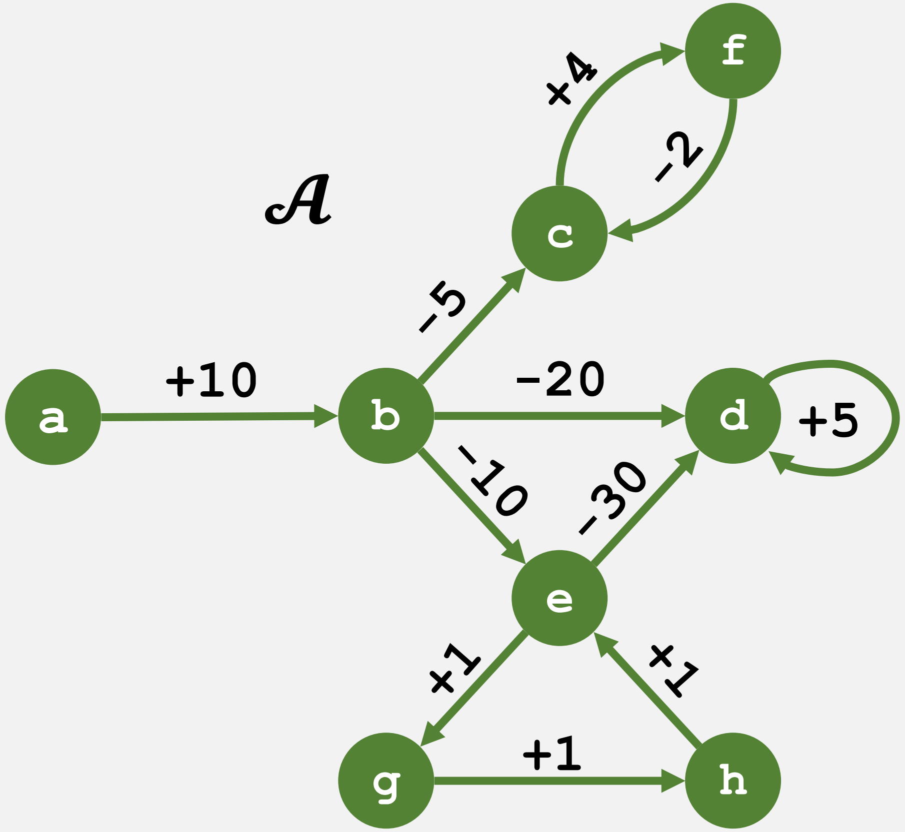
Example & Complexity Gap

"YES" instance:

a start node and **d** end node,
and $n = 42$ walk length.

"NO" instance:

a start node and **f** end node,
and $n = 42$ walk length.



NL-hard and in NP

...but not known to be NP-hard or in P

Motivation

Petri Nets and Vector Addition Systems with States:

- Related to short paths in VASS, and thus short paths in Petri Nets.
- Generalises 1-VASS Coverability (known to be in P).

Compressed Algorithmics:

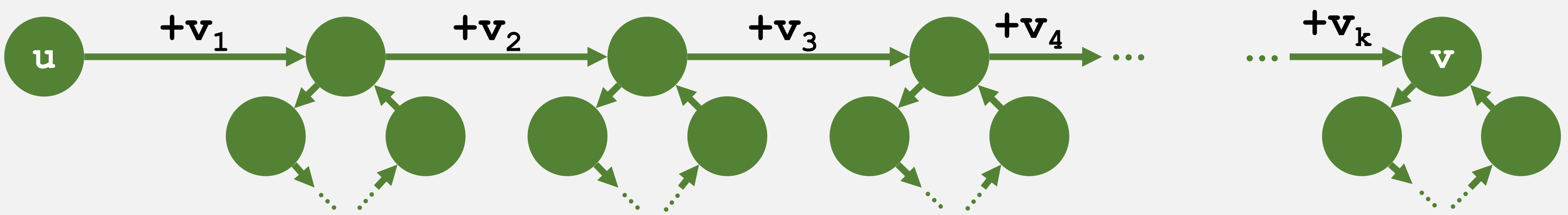
- Length compression applied to stack actions of pushdown automata with a unary stack.

Verification of safety conditions:

- Systems with control flow represented by a single integer variable.

Special Case: Linear Path Schemes

A sequence of non-overlapping cycles connected by a simple path



Cycle 1
Weight w_1
Length l_1

Cycle 2
Weight w_2
Length l_2

Cycle 3
Weight w_3
Length l_3

Cycle k
Weight w_k
Length l_k

Conjecture: For linear path schemes, the given length walks decision problem is in P.

Polynomial Time Algorithm Features

<u>Idea:</u> compute the set of all reachable points. **NP-hard!**

(n, c) is a reachable point if there exists a valid walk of length n ending with counter value c.

<u>New idea:</u> compute some useful subset of reachable points.

Points of interest: c is the greatest counter value achievable for a walk of length n.

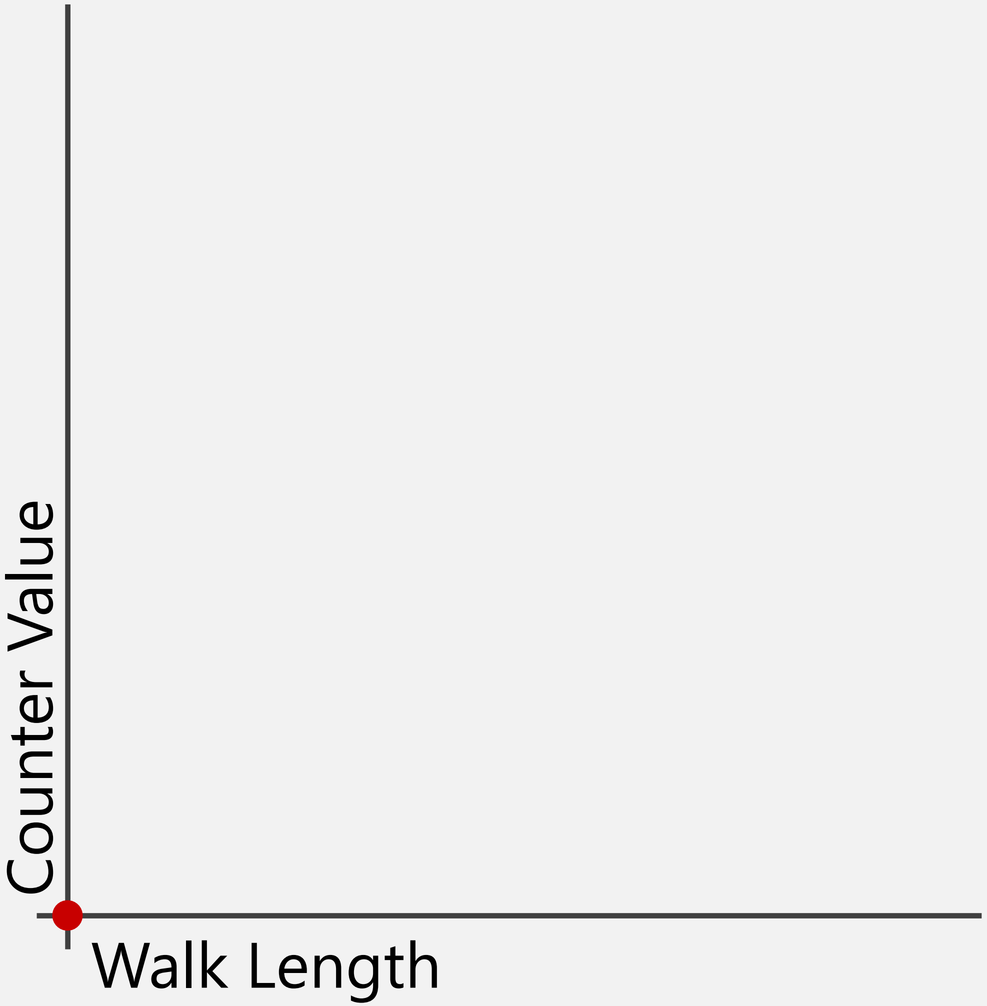
<u>Requirements:</u> careful representation of subset of reachable points.

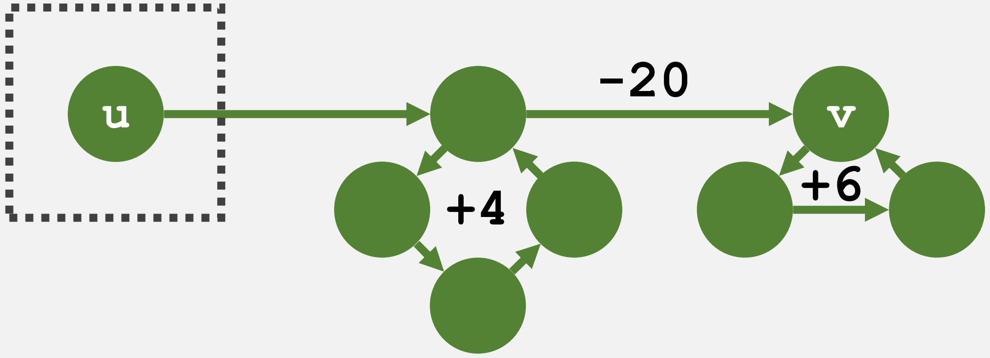
Succinct: poly-sized subset output & Efficient: easy to query target walk lengths.

Dynamic programming approach.

Sequentially considers each cycle, then edge, then cycle, …

Algorithm Behaviour by Example

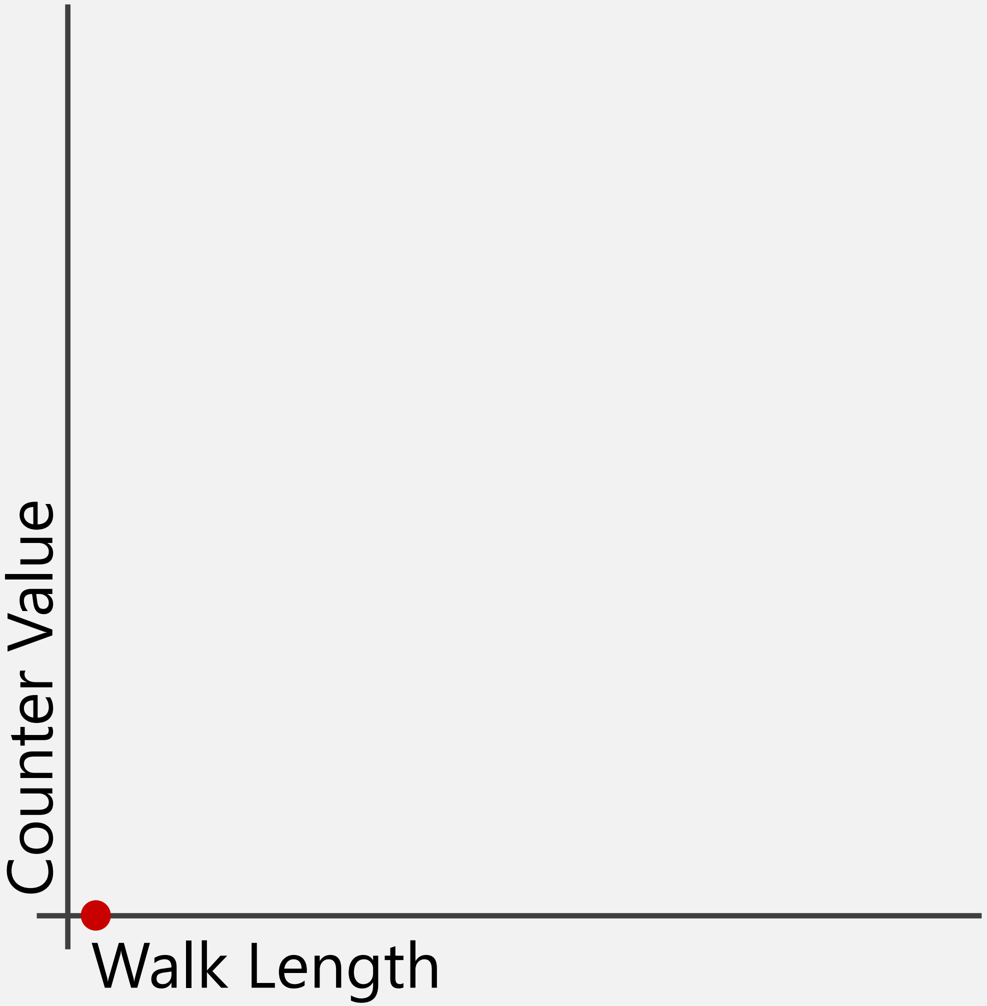


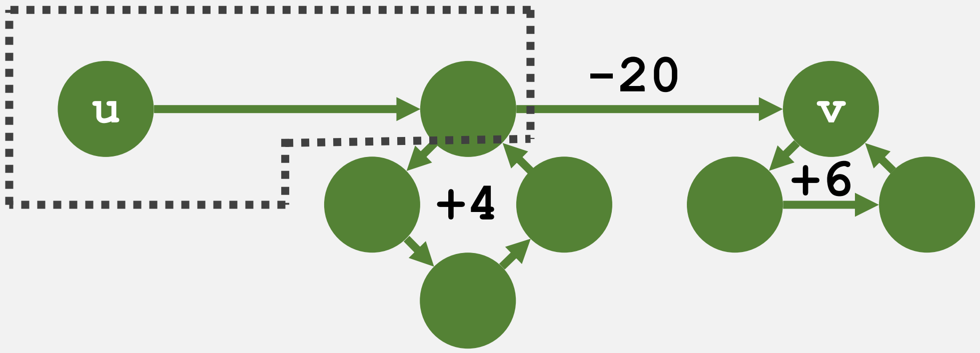


Initialisation:

Only $(0,0)$ is reachable.

Algorithm Behaviour by Example

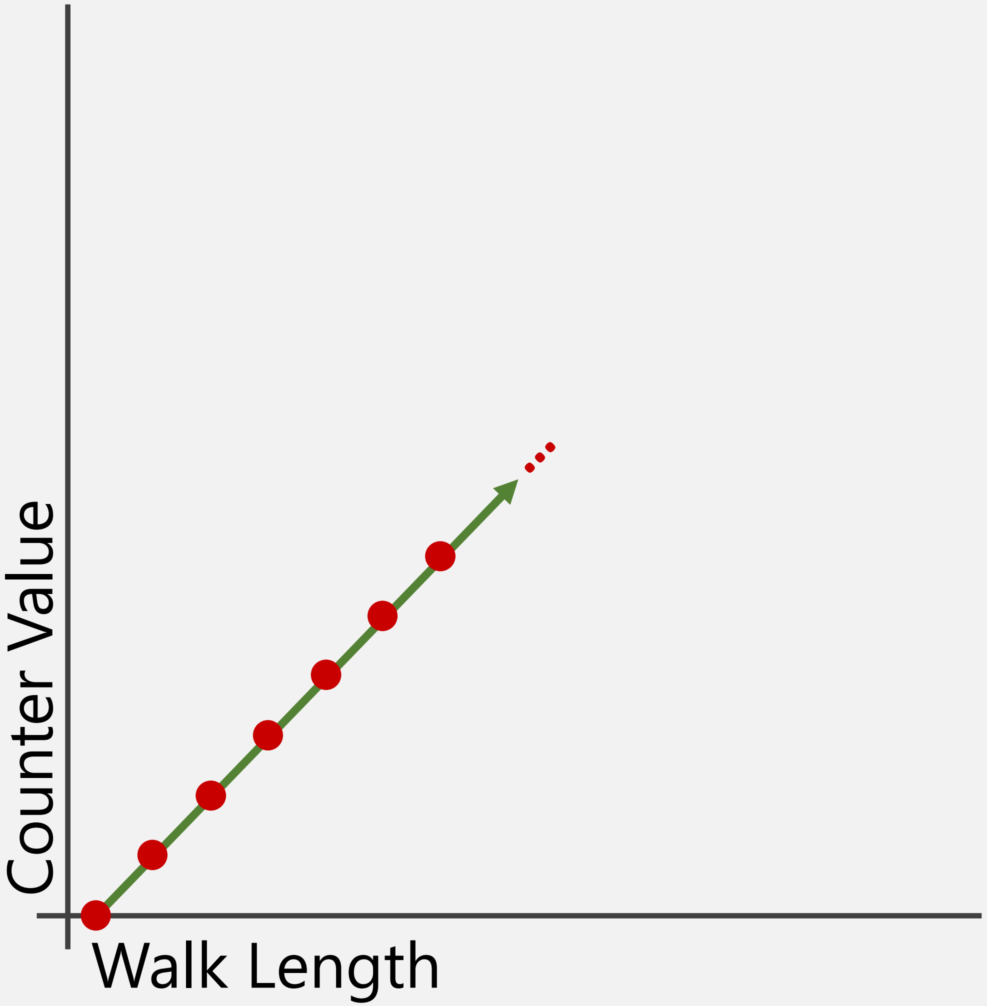


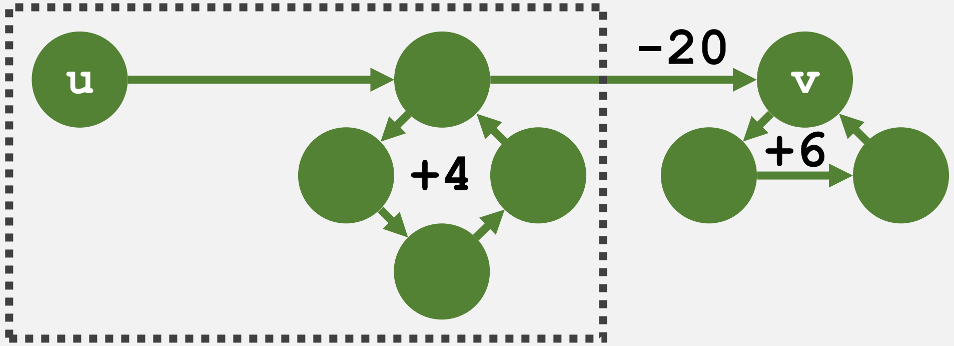


<u>After first edge:</u>

Walk length increments, so (1,0) is reachable.

Algorithm Behaviour by Example

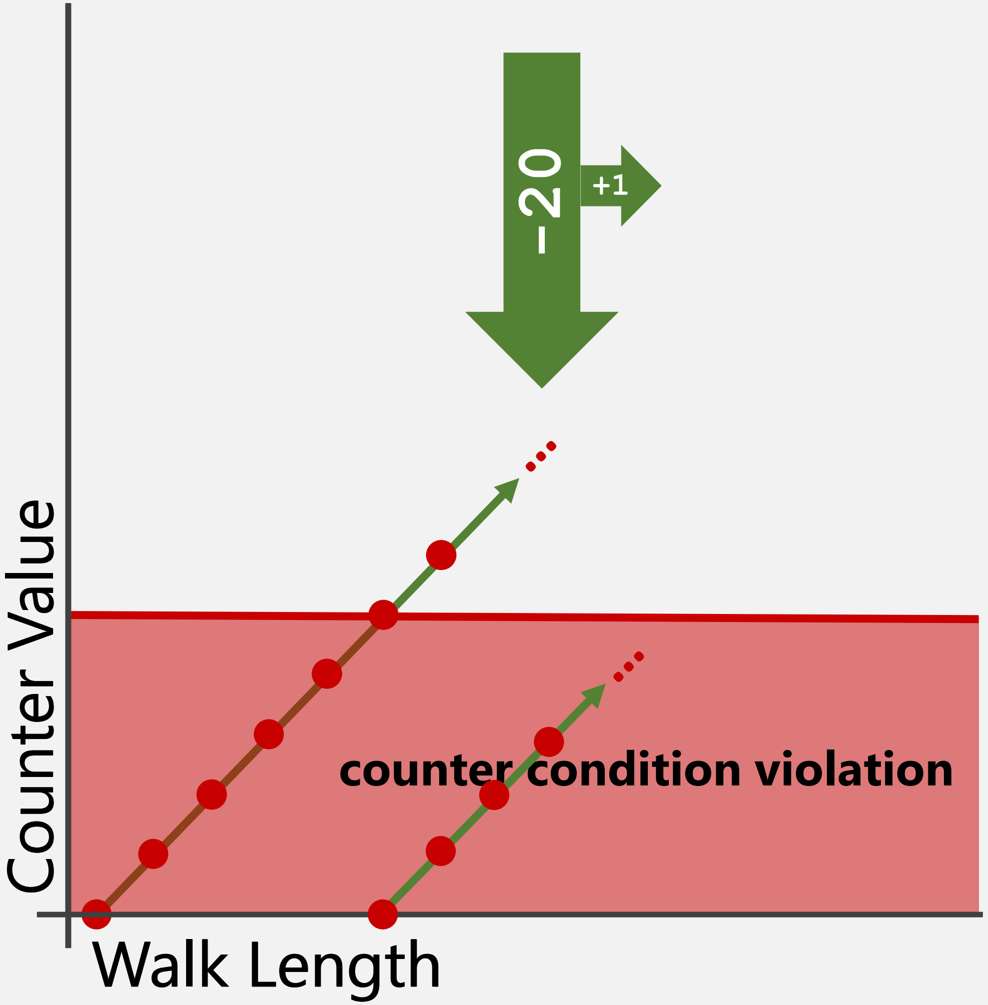


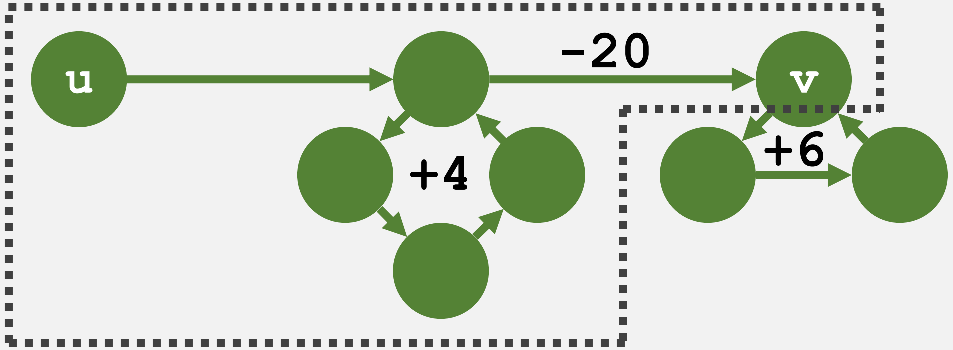


After first cycle:

$$\left.\begin{matrix}(1,0)\\(5,4)\\(9,8)\\\ldots\end{matrix}\right\}\text{period } +(4,4)$$

Algorithm Behaviour by Example

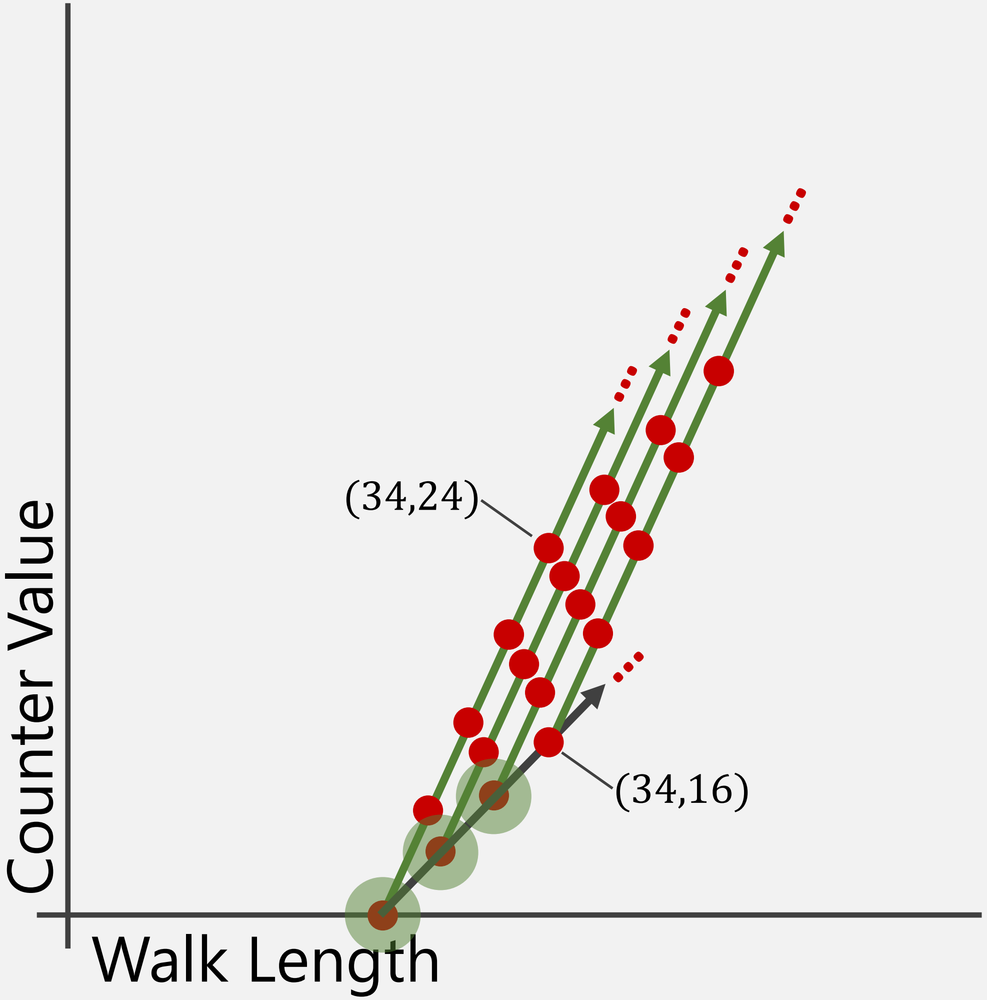


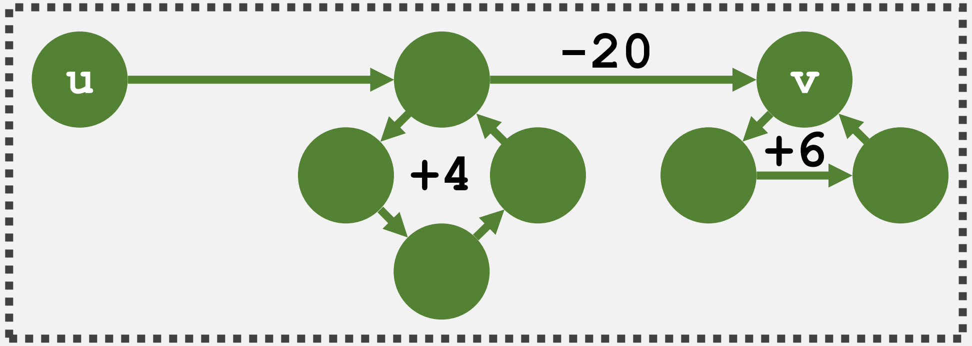


After second edge:

$$\left.\begin{matrix}(22,0)\\(26,4)\\(30,8)\\\ldots\end{matrix}\right\}\text{ period still }+(4,4)$$

Algorithm Behaviour by Example





After second cycle:

First period $+(4,4)$

Second period $+(3,6)$ *...better!*

Extensions

*****_Case when several first cycles are negative is in progress_

Beyond linear path schemes

Series parallel based, directed acyclic graph based, arbitrary.

Beyond walks of a given length

Other decision problems, e.g.: “is there a walk of every length?”.

Questions?

“Walks of Given Length in One-Counter Systems”
by Henry Sinclair-Banks

WPCCS’21 Presentation Monday 13th December