On the Computational Complexity of Minimal Cumulative Cost Graph Pebbling

Graph Pebbling is Cool and Important!...but Hard

Jeremiah Blocki

Samson Zhou

Motivation

❖ Users tend to pick weak passwords

❖ Server attacks are inevitable

Entity	Year	Records
Yahoo	2013	1,000,000,000
Yahoo	2014	500,000,000
Friend Finder Networks	2016	412,214,295
Massive American business hack including 7-Eleven and Nasdaq	2012	160,000,000
Adobe Systems	2014	152,000,000
eBay	2014	145,000,000
Heartland	2009	130,000,000
Rambler.ru	2012	98,167,935
TK / TJ Maxx	2007	94,000,000
AOL	2004	92,000,000
Anthem Inc.	2015	80,000,000
Sony PlayStation Network	2011	77,000,000
JP Morgan Chase	2014	76,000,000
National Archives and Records Administration (U.S. military veterans' records)	2009	76,000,000
Target Corporation	2014	70,000,000
Home Depot	2014	56,000,000

YAHOO!

OHIO STATE

Gmail™ by Google

Adobe®

AMERICA Online®

ASHLEY MADISON®.COM

BLIZZARD® ENTERTAINMENT

eHarmony®

LastPass ****

ebay

Dropbox

LinkedIn™

PlayStation™

User	Password
Stephen	auhsoJ
Lisa	hsifdrowS
James	1010NO1Z
Harry	sinocarD tupaC
Sarah	auhsoJ

User	Password Hash
Stephen	39e717cd3f5c4be78d97090c69f4e655
Lisa	f567c40623df407ba980bfad6dff5982
James	711f1f88006a48859616c3a5cbcc0377
Harry	fb74376102a049b9a7c5529784763c53
Sarah	39e717cd3f5c4be78d97090c69f4e655

User	Random Salt	Password Hash
Stephen	06917d7ed65c466fa180a6fb62313ab9	b65578786e544b6da70c3a9856cdb750
Lisa	51f2e43105164729bb46e7f20091adf8	2964e639aa7d457c8ec0358756cbffd9
James	fea659115b7541479c1f956a59f7ad2f	dd9e4cd20f134dda87f6ac771c48616f
Harry	30ebf72072134f1bb40faa8949db6e85	204767673a8d4fa9a7542ebc3eceb3a2
Sarah	711f51082ea84d949f6e3efecf29f270	e3afb27d59a34782b6b4baa0c37e2958

Motivation

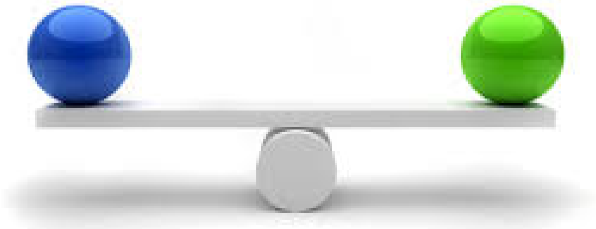
- Users tend to pick weak passwords
- Server attacks are inevitable
- Try to mitigate offline attacks
- Specialized hardware (ASIC) can compute 10^{12} hashes per second.

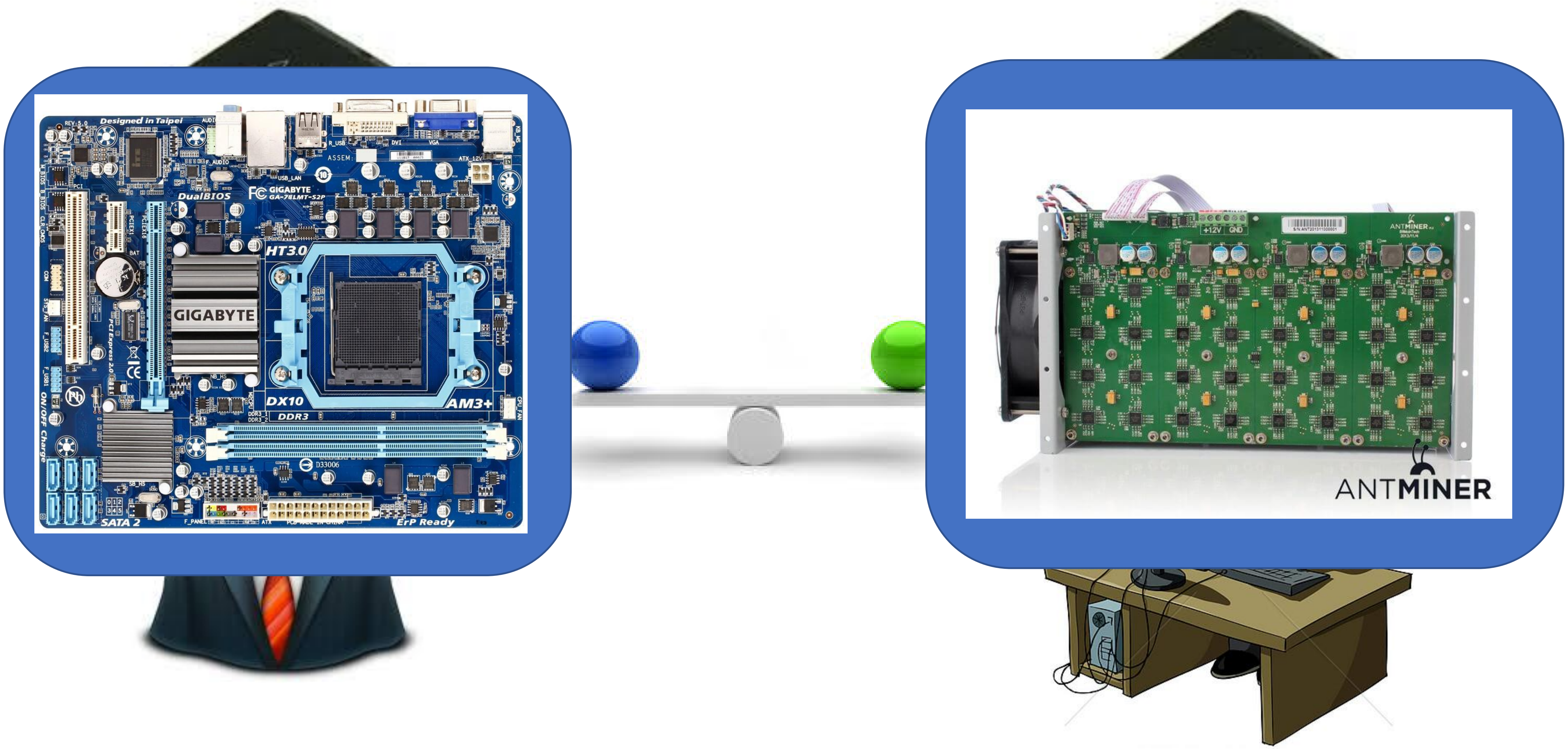
Password Hash Function Goals

- ❖ “Moderately Expensive” to compute
- ❖ Expensive to compute on ASIC
- ❖ Fast and cheap on PC


AT27C010
70J1
0307


AT27C010
70J1
0307



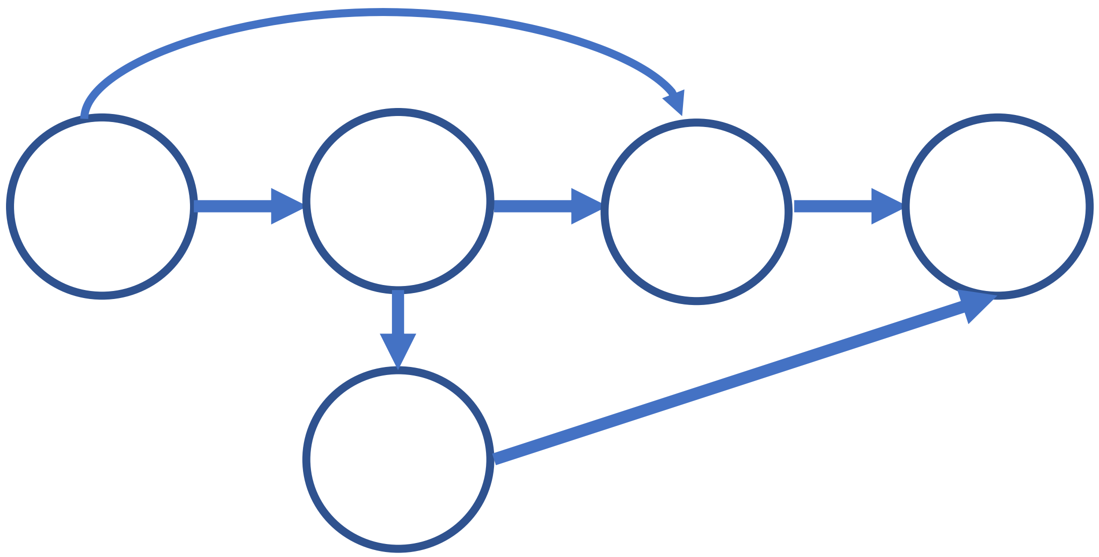

Designed in Taipei
DualBIOS
GIGABYTE
HT3.0
DX10
AM3+
DDR3
SATA 2
ErP Ready
ANTMINER

Memory Hard Functions

- Memory hard functions require comparatively more resources for adversaries to compute
- Data-dependent memory hard functions are susceptible to side-channel attacks
- Data-independent memory hard functions (iMHFs)

iMHFs

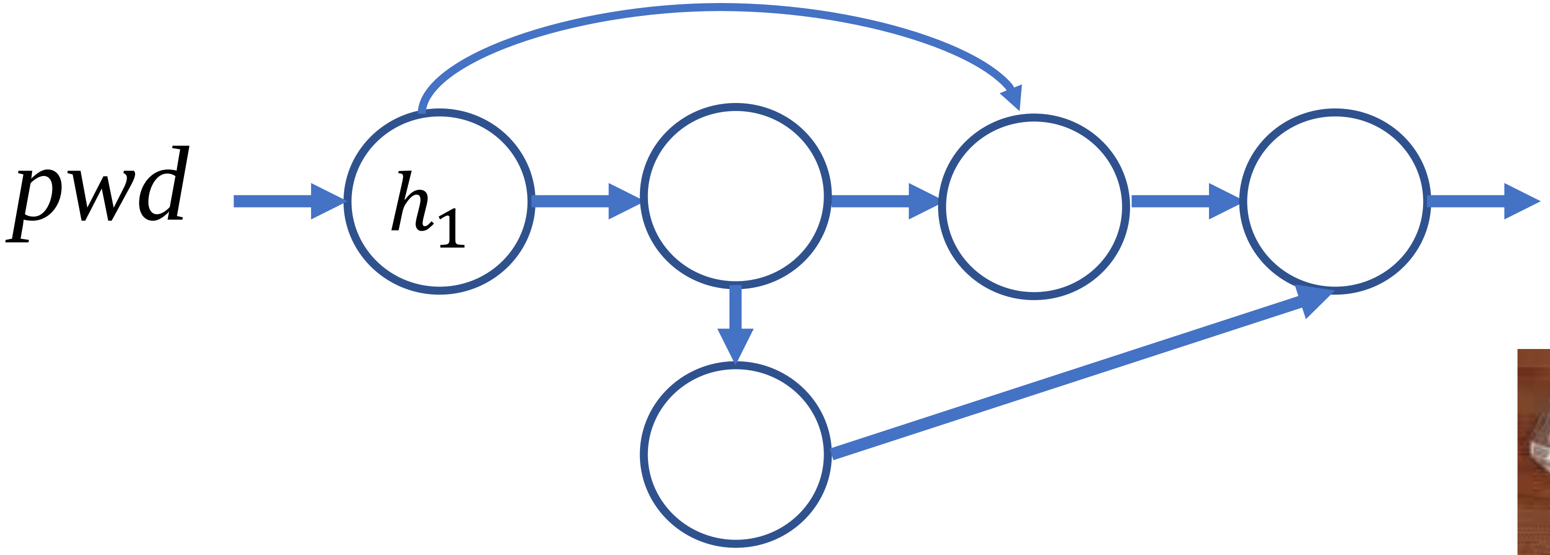
$f_{G,H}$



Hash function: H

iMHFs

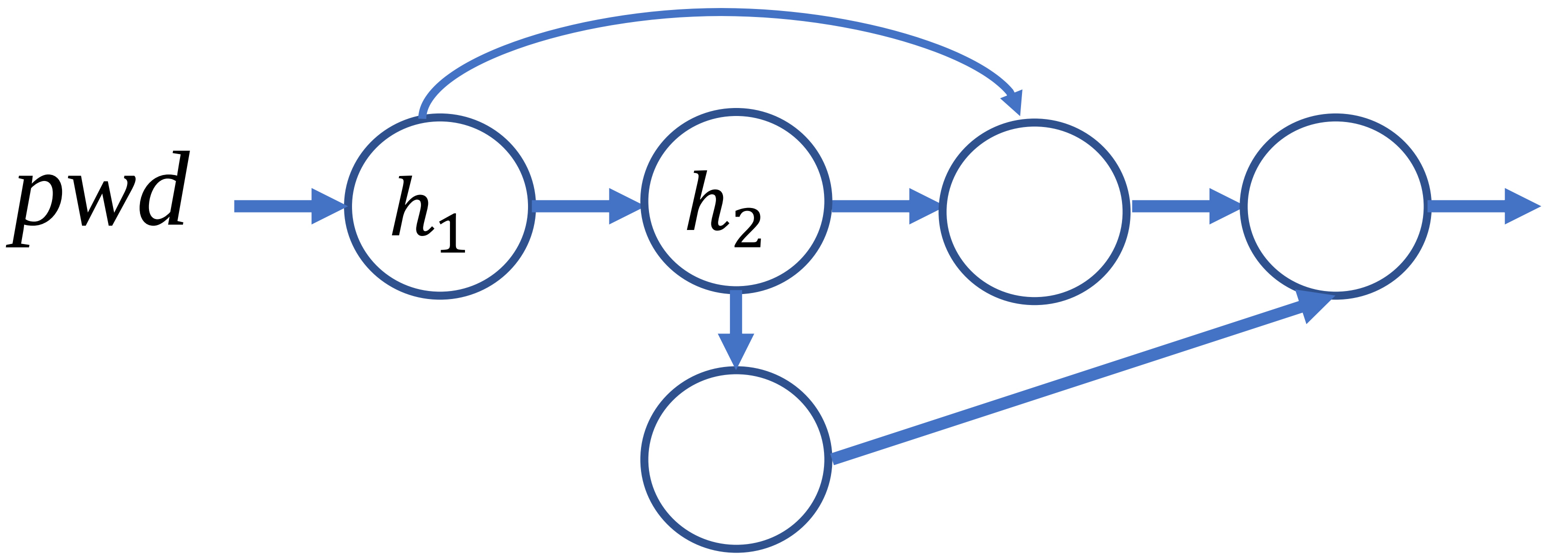
$f_{G,H}$



$$h_1 = H(pwd, salt)$$

iMHFs

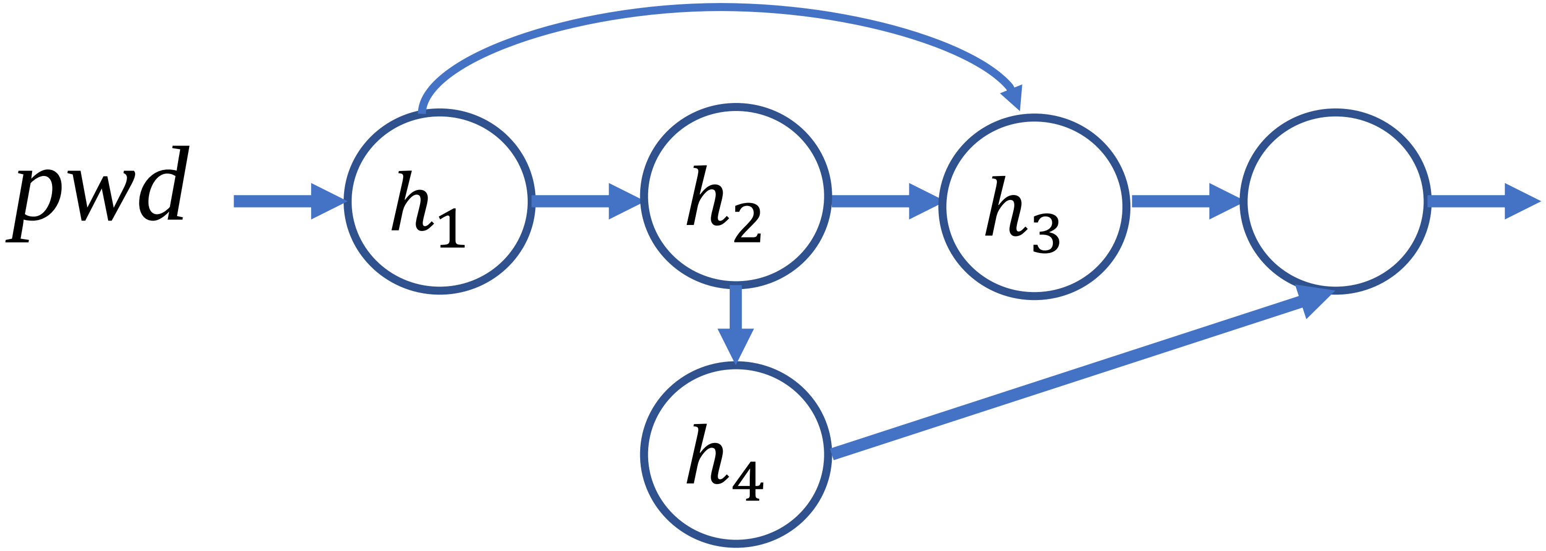
$f_{G,H}$



$$h_2 = H(h_1)$$

iMHFs

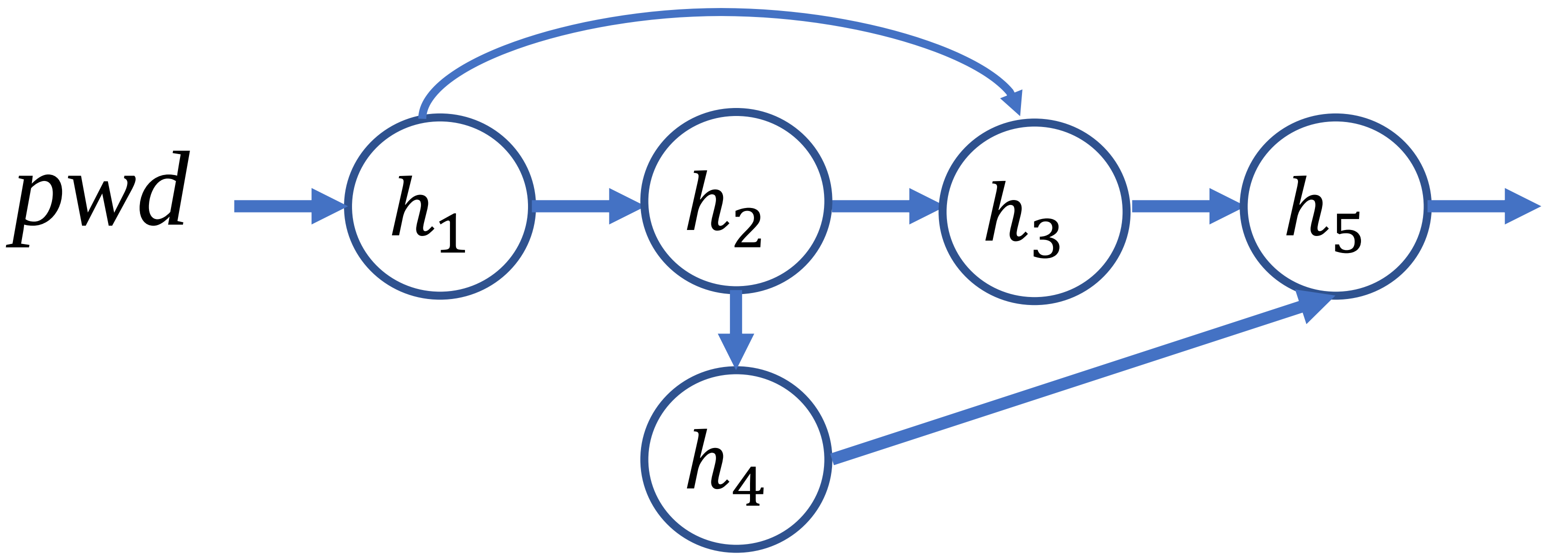
$f_{G,H}$



$h_3 = H(h_1, h_2),\qquad h_4 = H(h_2)$

iMHFs

$f_{G,H}$

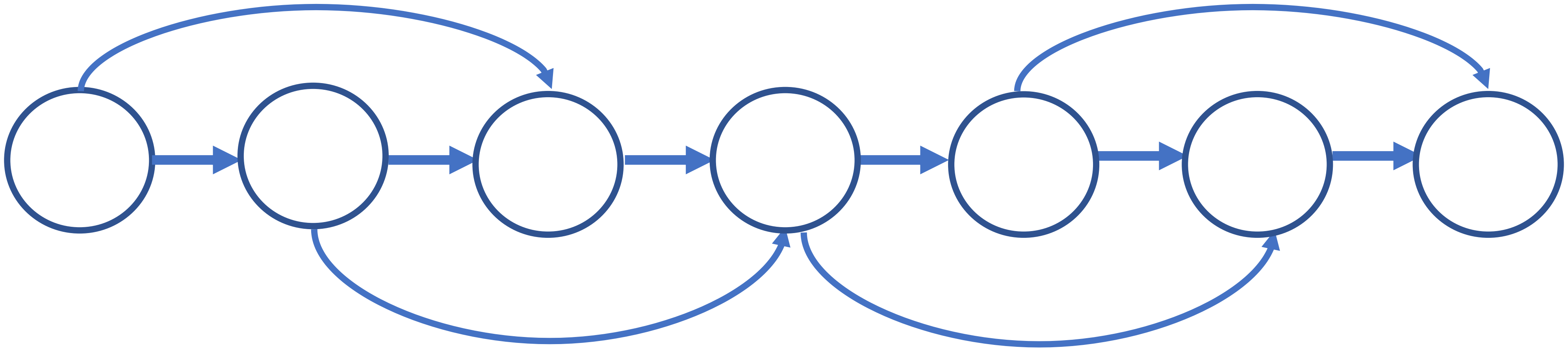


$$h_5 = H(h_3, h_4)$$

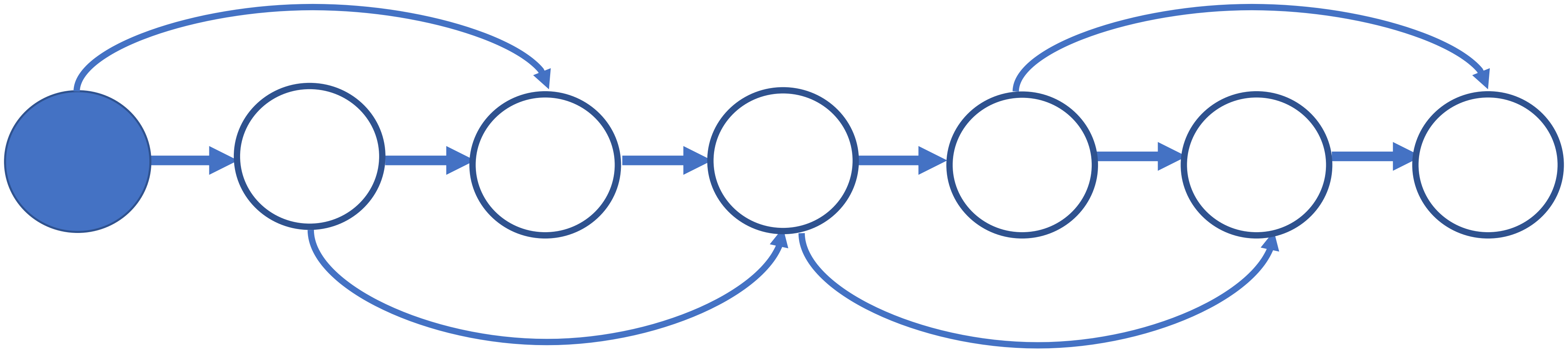
iMHFs

❖ Calculating an iMHF can be modeled as graph pebbling [AS15]

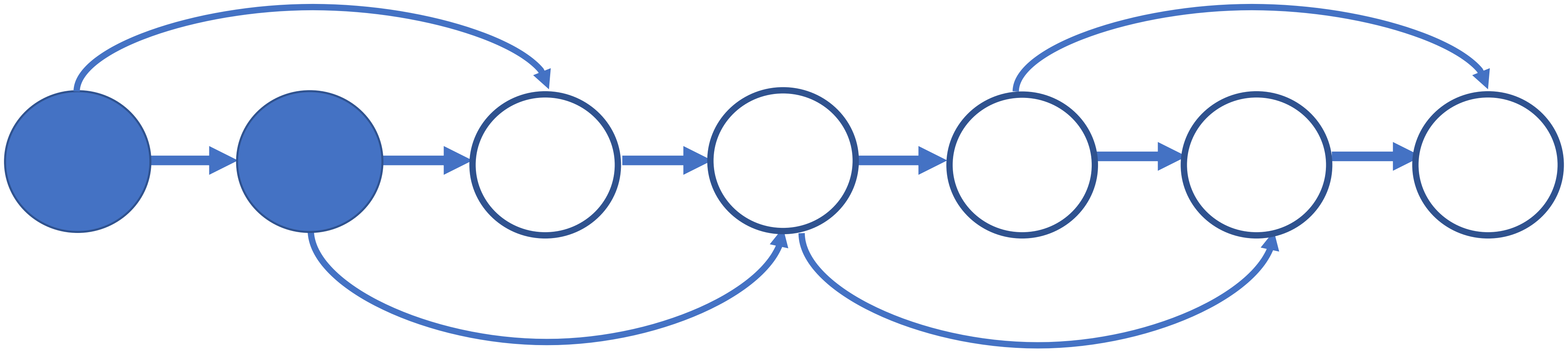
Graph Pebbling



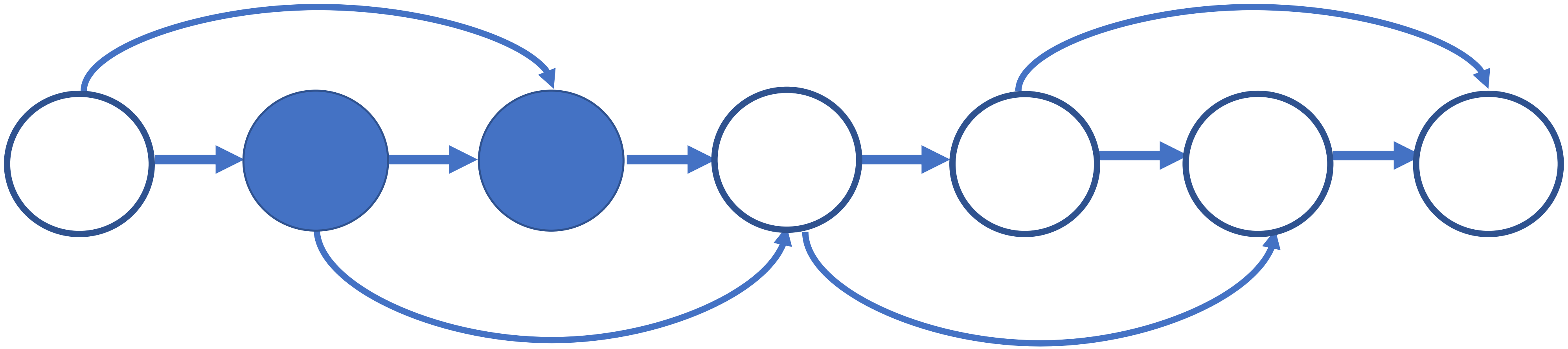
Graph Pebbling



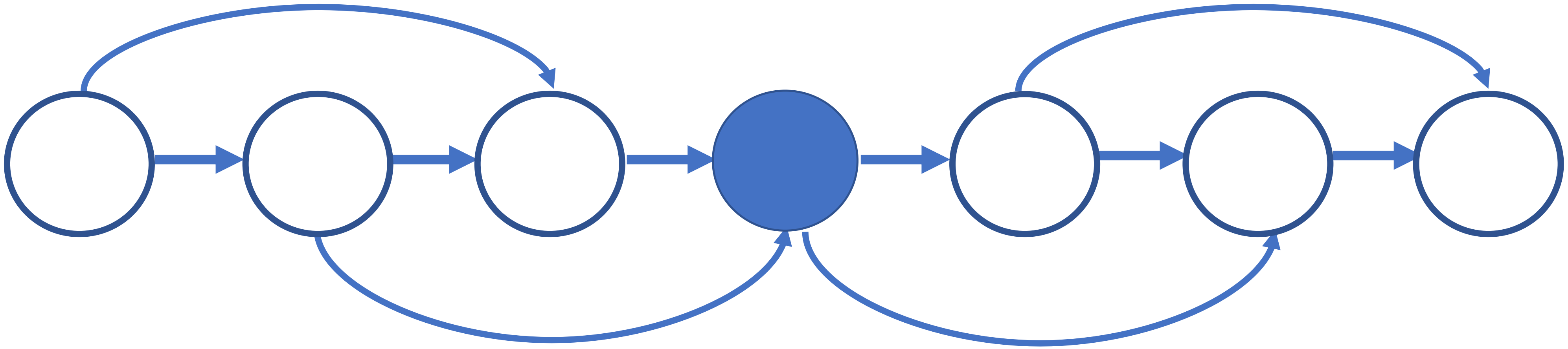
Graph Pebbling



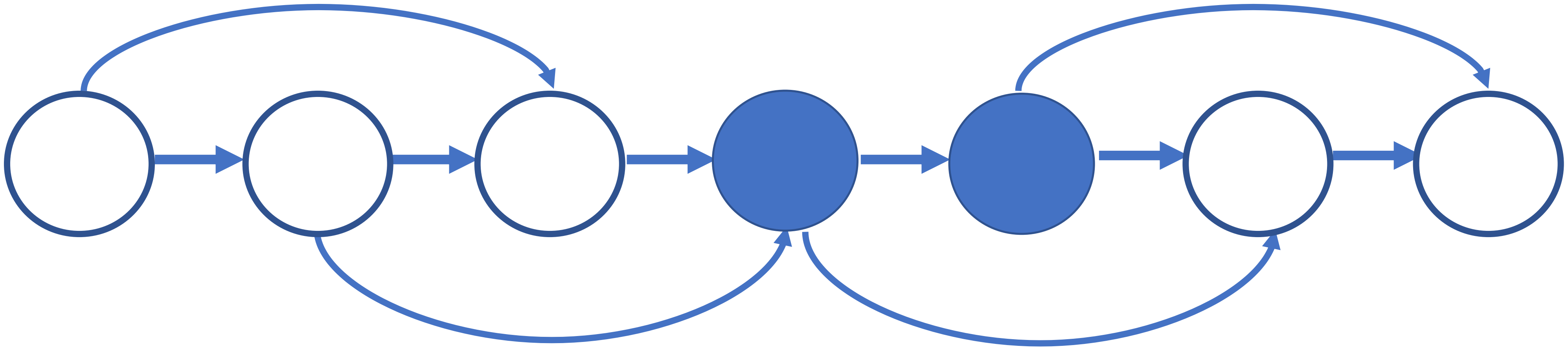
Graph Pebbling



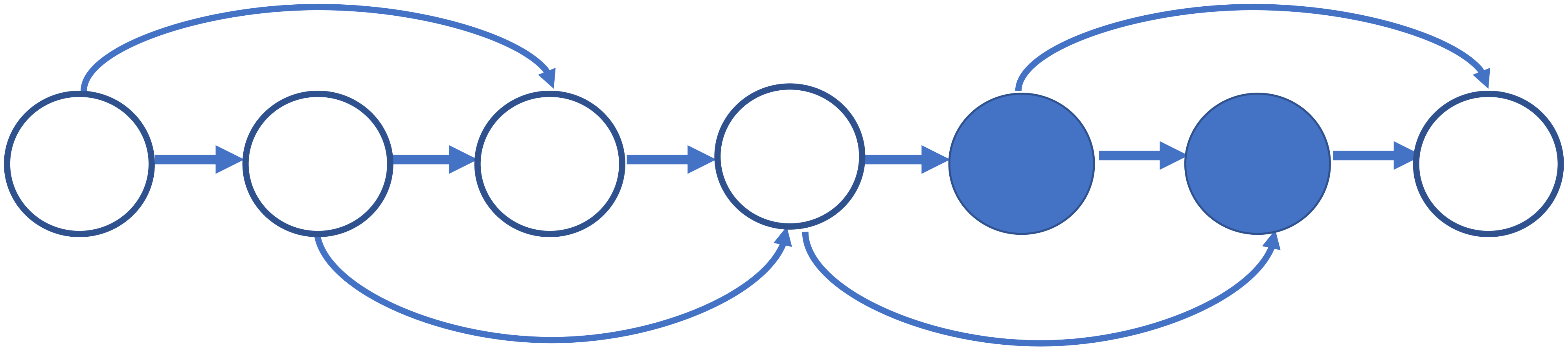
Graph Pebbling



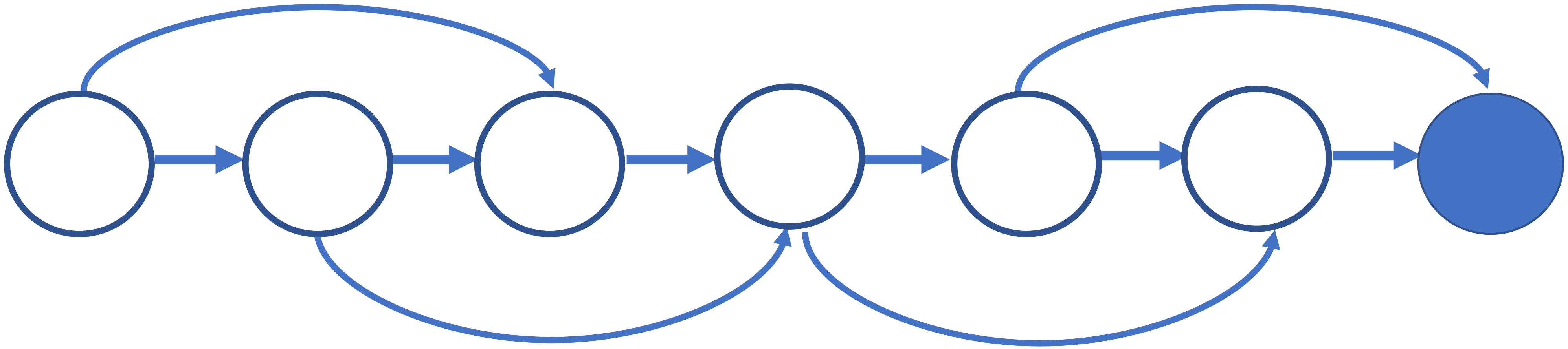
Graph Pebbling



Graph Pebbling



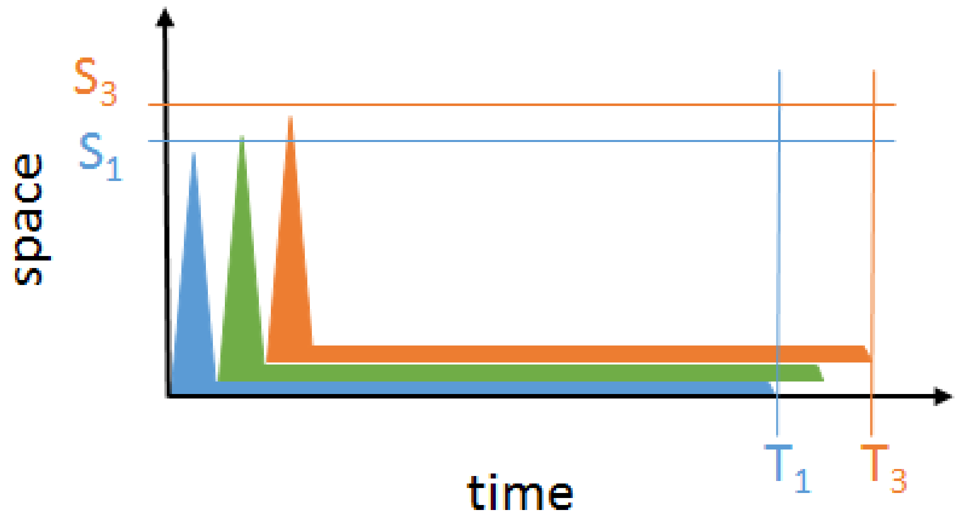
Graph Pebbling



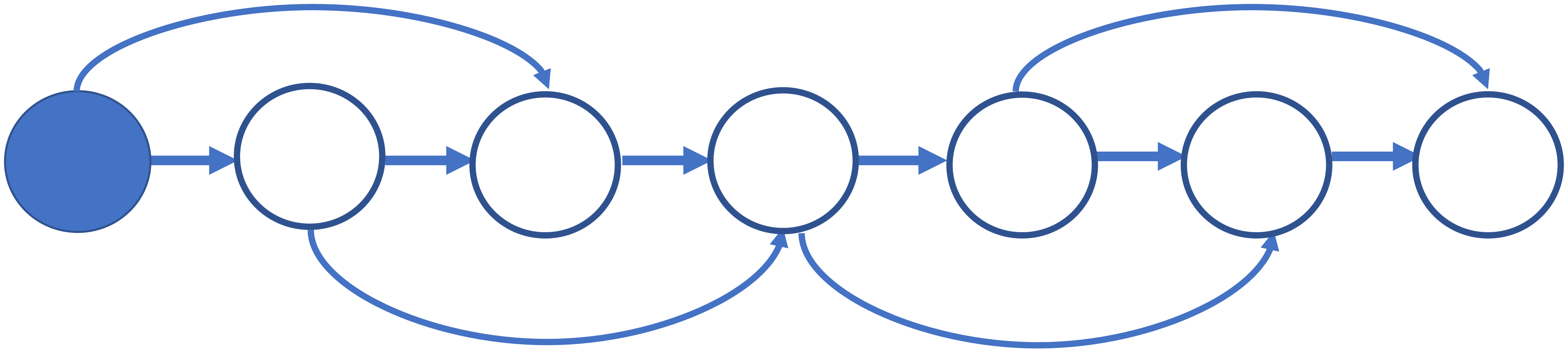
iMHFs

- How to quantify "memory-hardness" of iMHF?
- ST-complexity: maximum number of pebbles × number of steps
  - 2 pebbles × 7 steps = 14
- ST-complexity can scale badly with multiple evaluations [AS15]
- [AS15] ∃ function f such that:

$$ST(\sqrt{n}\ instances\ of\ f) = O(ST(f))$$

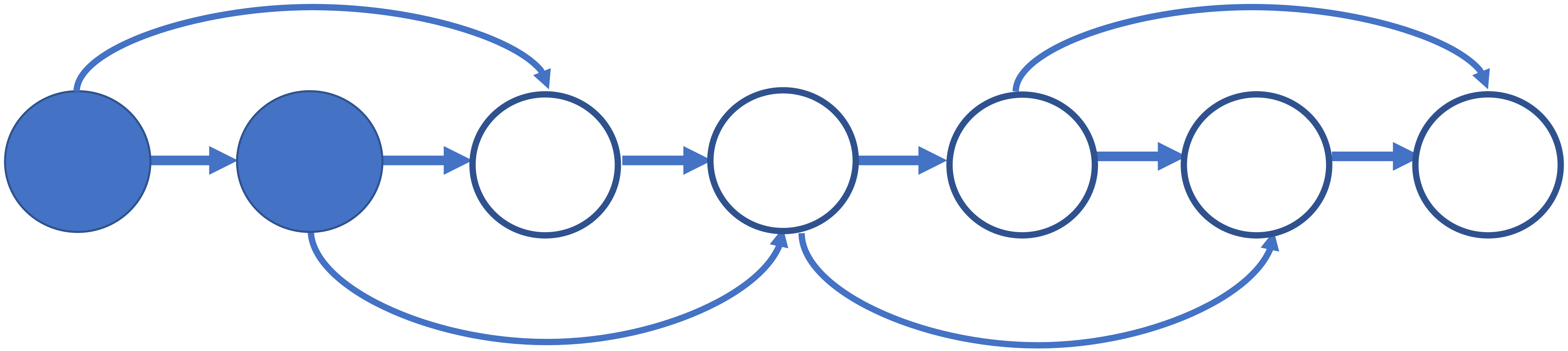


Graph Pebbling



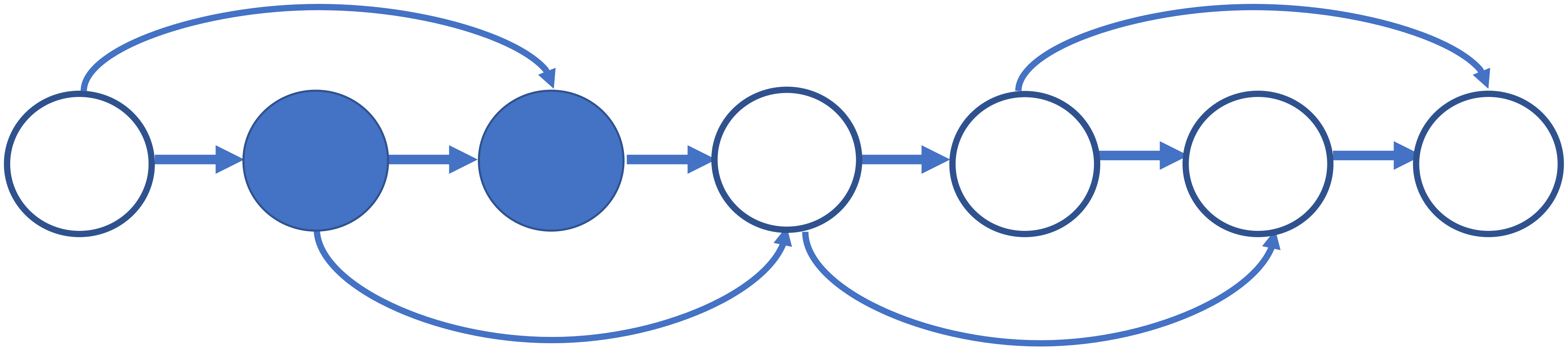
$|P_1| = 1$

Graph Pebbling



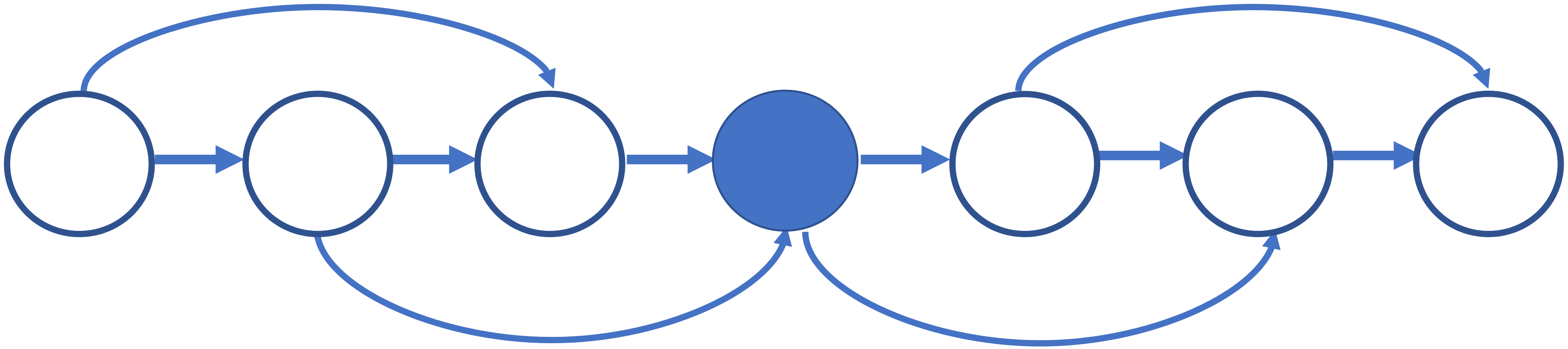
$$|P_1| + |P_2| = 1 + 2 = 3$$

Graph Pebbling



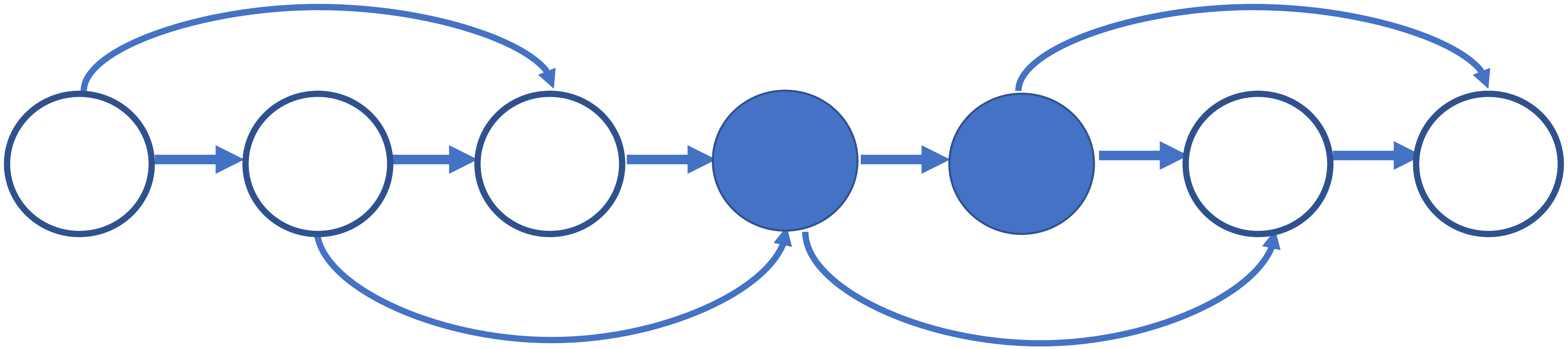
$|P_1| + |P_2| + |P_3| = 3 + 2 = 5$

Graph Pebbling



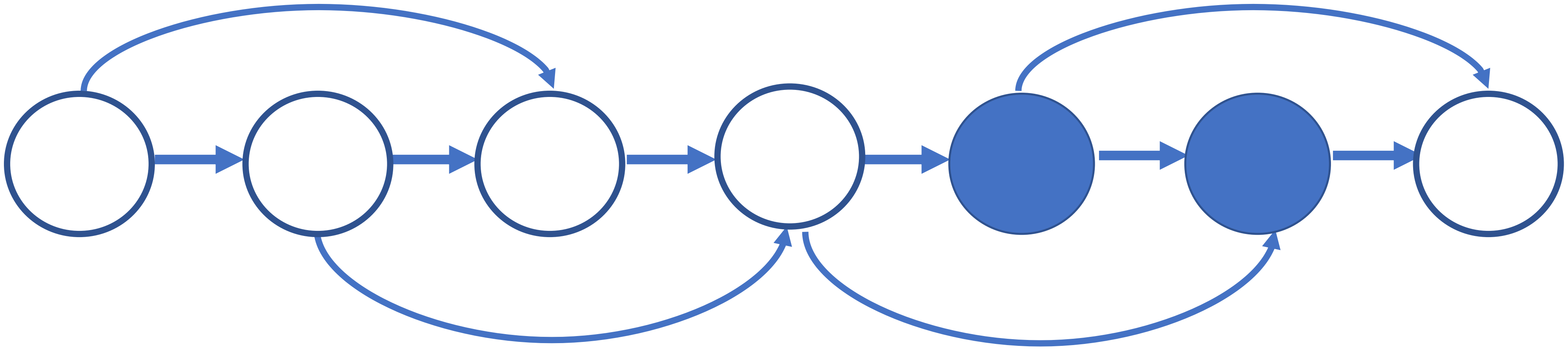
$$\sum_{\{i=1\}}^{4} |P_i| = 5 + 1 = 6$$

Graph Pebbling



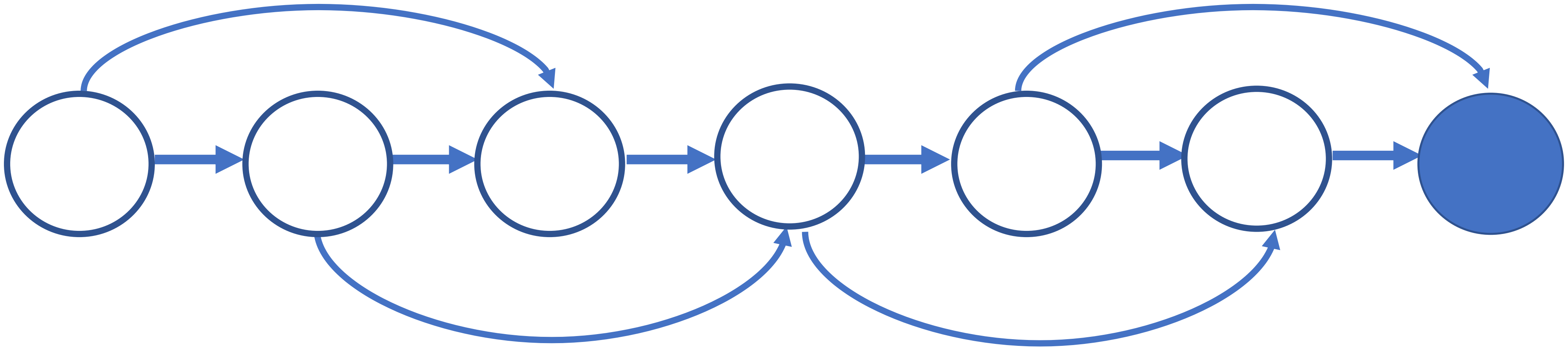
$$\sum_{\{i=1\}}^{5} |P_i| = 6 + 2 = 8$$

Graph Pebbling



$$\sum_{\{i=1\}}^{6} |P_i| = 8 + 2 = 10$$

Graph Pebbling



$$cc(G) = \sum_{\{i=1\}}^{7} |P_i| = 10 + 1 = 11$$

iMHFs

- [AS15] CC amortizes well:
  - $CC(n\ copies\ of\ f) = n \times CC(f)$

$cc(G)$	Lower Bound	Upper Bound
Argon2i [BDK15]	$\Omega(n^{1.75})$ [BZ17]	$O(n^{1.767})$ [BZ17]
DRSample [ABH17]	$\Omega\left(\frac{n^2}{\log n}\right)$ [ABH17]	$O\left(\frac{n^2 \log\log n}{\log n}\right)$ [ABH17]

Gap of 500,000

$7.3 \ast 10^{-6} \ast \frac{n^2}{\log n}$

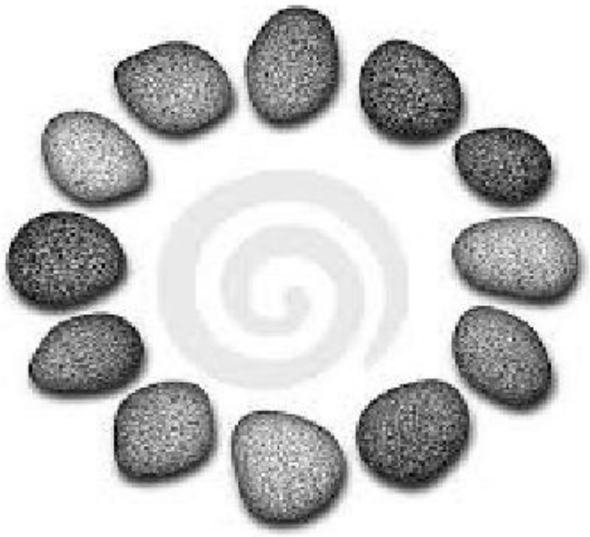
$\frac{n^2}{6}$

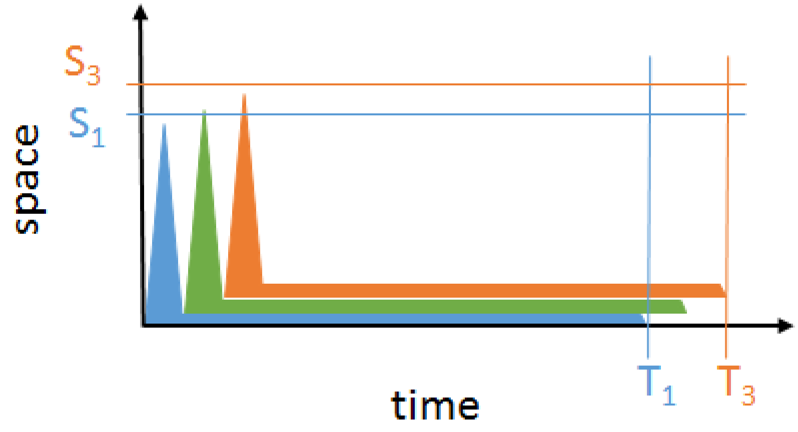
Review

❖ Data-independent memory hard functions require comparatively more resources for adversaries to compute

❖ Calculating an iMHF can be modeled as graph pebbling

❖ Cumulative complexity better model than space-time complexity





Main Result

- Computing $cc(G)$ is NP-hard!

- Implication: Cryptanalysis of memory-hard functions is provably hard

Bounded 2-Linear Covering

- Given n variables $x_1, x_2, \dots, x_n$, integers $m \le k$, and k equations of the form $x_i + c = x_j$, can we find m assignments so that all equations are satisfied?
- $x_1 + 2 = x_2, \quad x_2 + 3 = x_3, \quad x_1 + 6 = x_3$
- $x_1 + 5 = x_2, \quad x_2 + 1 = x_3, \quad x_1 + 5 = x_3$
- $m = 2$ $(k = 6)$
- Assignment 1: $x_1 = 1, x_2 = 3, x_3 = 6$

Bounded 2-Linear Covering

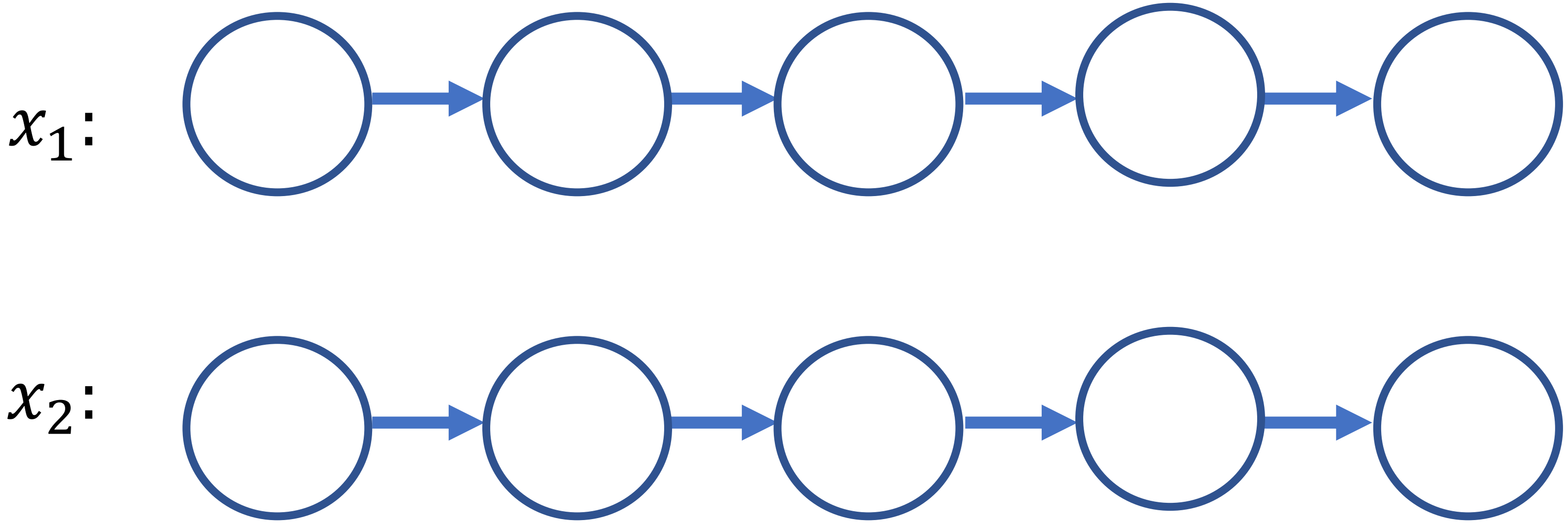
- Given n variables $x_1, x_2, \dots, x_n$, integers $m \leq k$, and k equations of the form $x_i + c = x_j$, can we find m assignments so that all equations are satisfied?
- $x_1 + 2 = x_2, \quad x_2 + 3 = x_3, \quad x_1 + 6 = x_3$
- $x_1 + 5 = x_2, \quad x_2 + 1 = x_3, \quad x_1 + 5 = x_3$
- $m = 2$
- Assignment 1: $x_1 = 1, x_2 = 3, x_3 = 6$
- Assignment 2: $x_1 = 1, x_2 = 6, x_3 = 7$

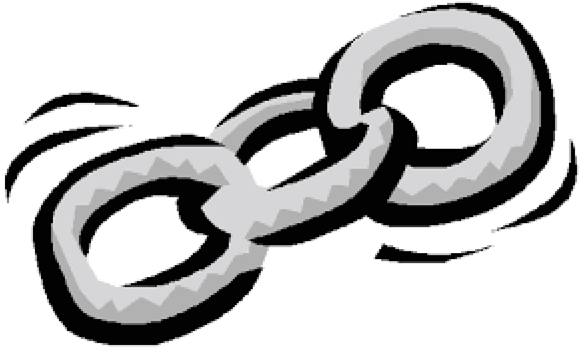
Bounded 2-Linear Covering

- Given n variables $x_1, x_2, \ldots, x_n$, integers $m \leq k$, and k equations of the form $x_i + c = x_j$, can we find m assignments so that all equations are satisfied?
- $x_1 + 2 = x_2,\quad x_2 + 3 = x_3,\quad x_1 + 6 = x_3$
- $x_1 + 5 = x_2,\quad x_2 + 1 = x_3,\quad x_1 + 5 = x_3$
- $m = 2$
- Assignment 1: $x_1 = 1, x_2 = 3, x_3 = 6$
- Assignment 2: $x_1 = 1, x_2 = 6, x_3 = 7$
- **Our Theorem**: B2LC is NP-complete (reduction from 3PARTITION)

Reductions

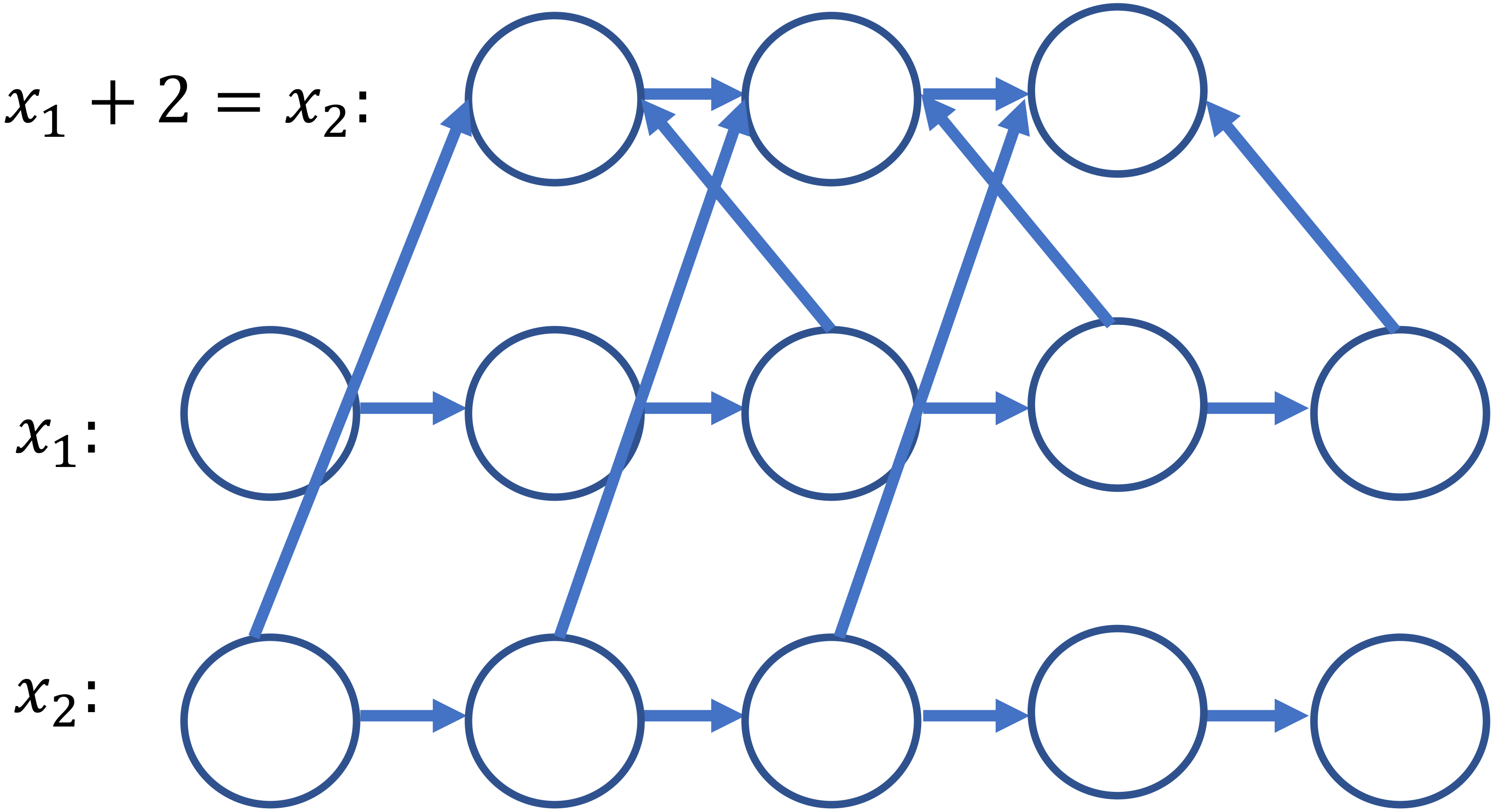
GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE





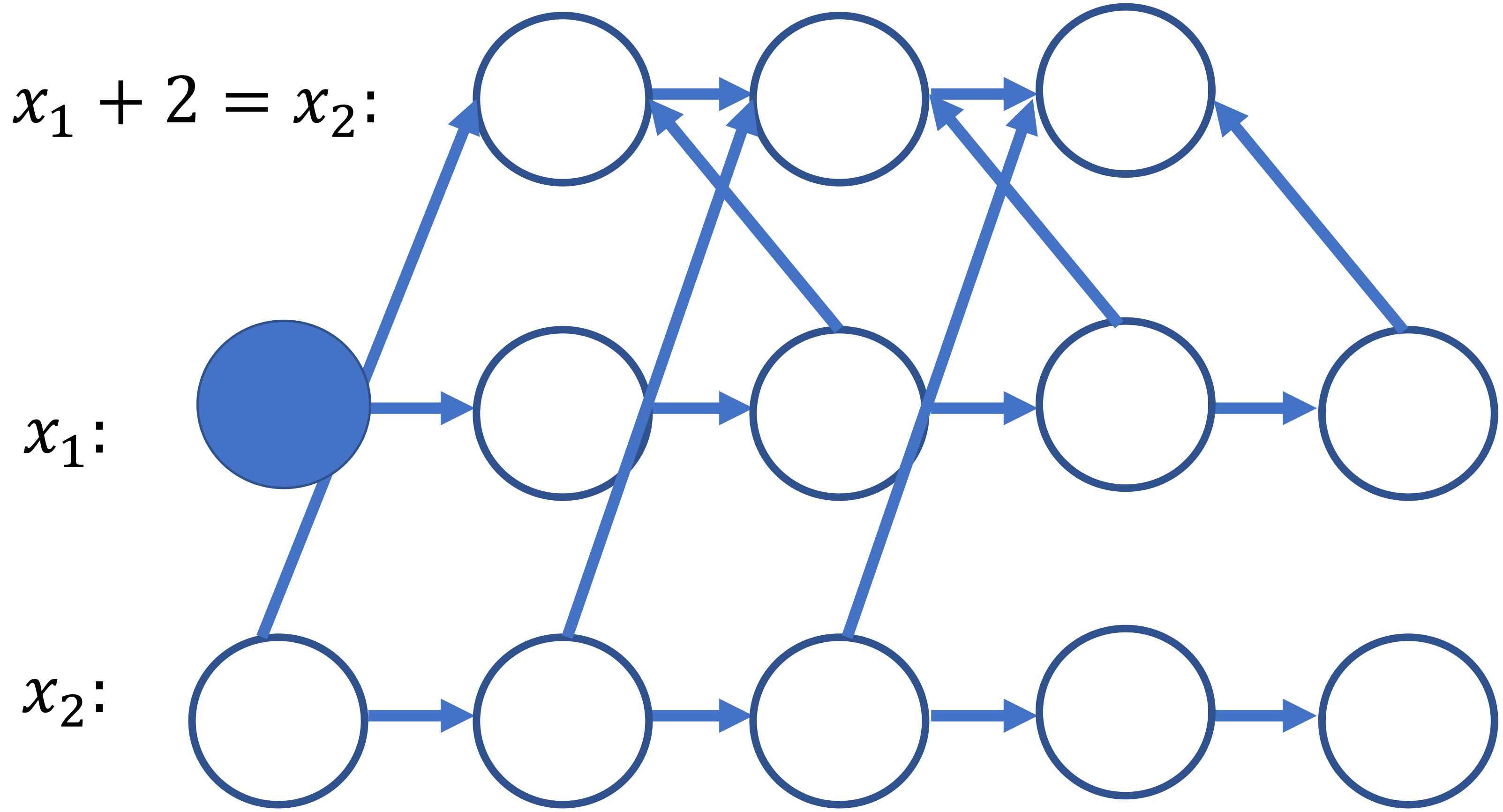
Reductions

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL.
Otherwise, Pebbling Cost is LARGE



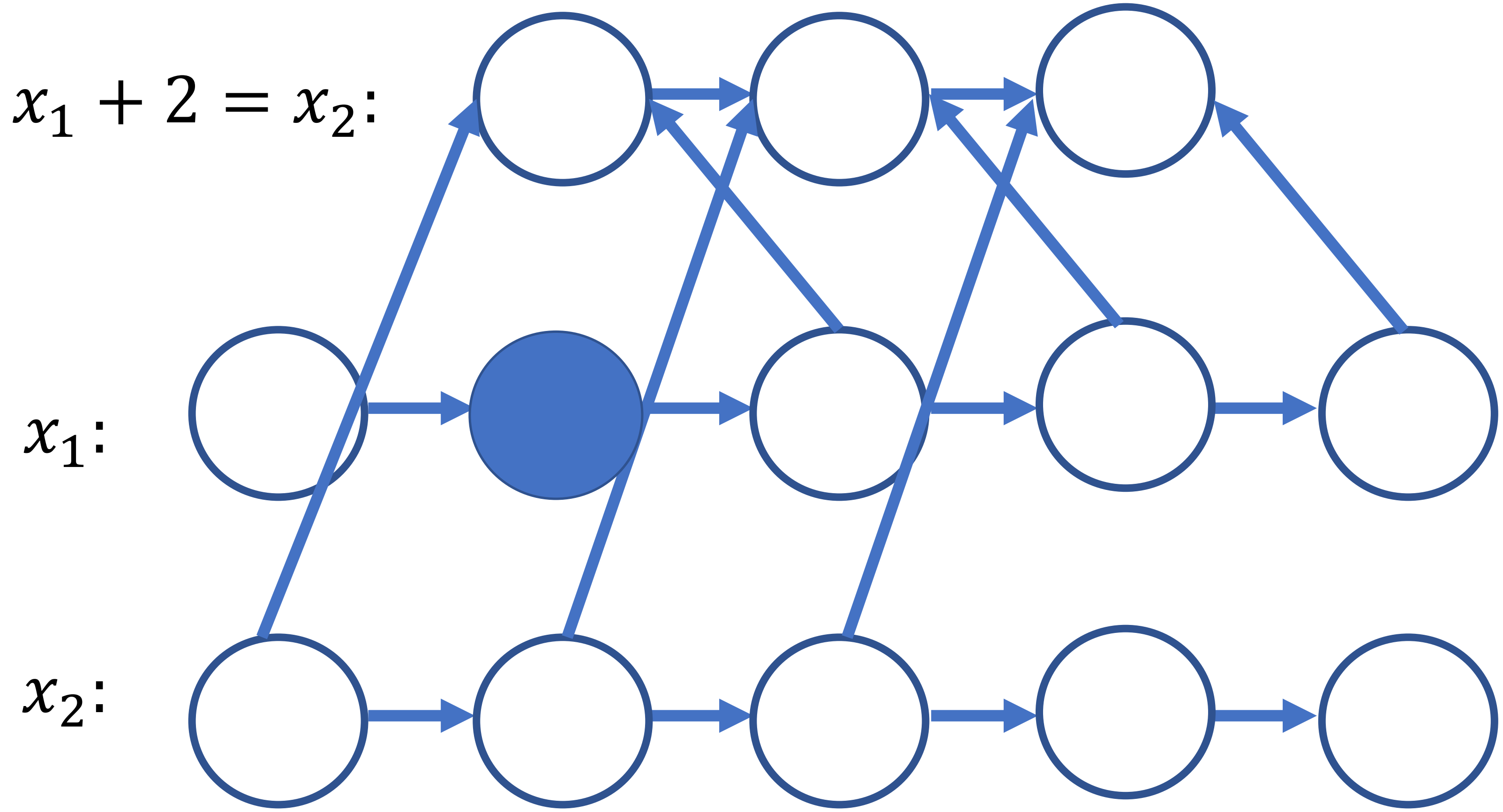
Honest Pebbling

GOAL: If m assignments for k equations,
then Pebbling Cost is SMALL.
Otherwise, Pebbling Cost is LARGE



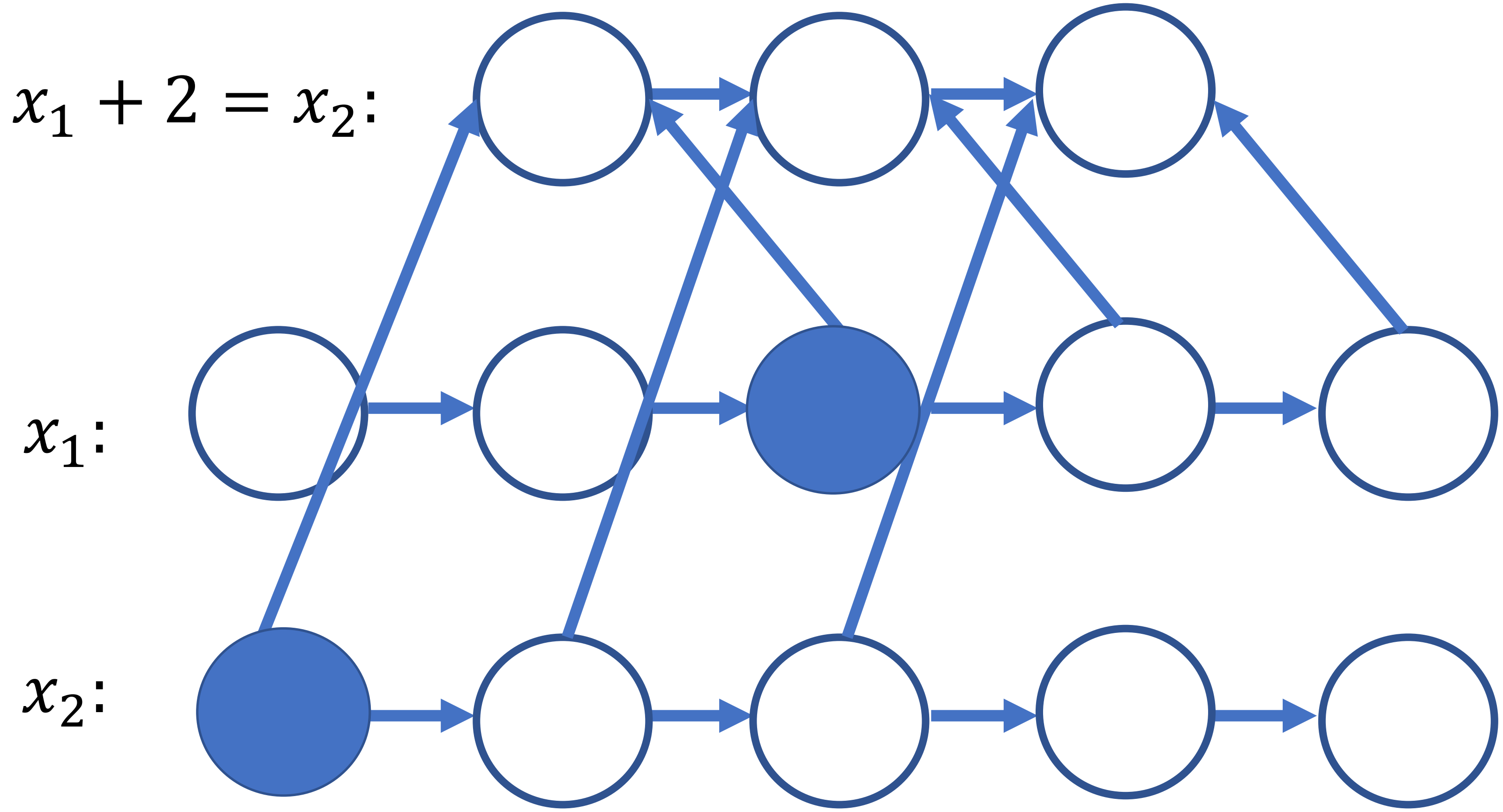
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL.
Otherwise, Pebbling Cost is LARGE



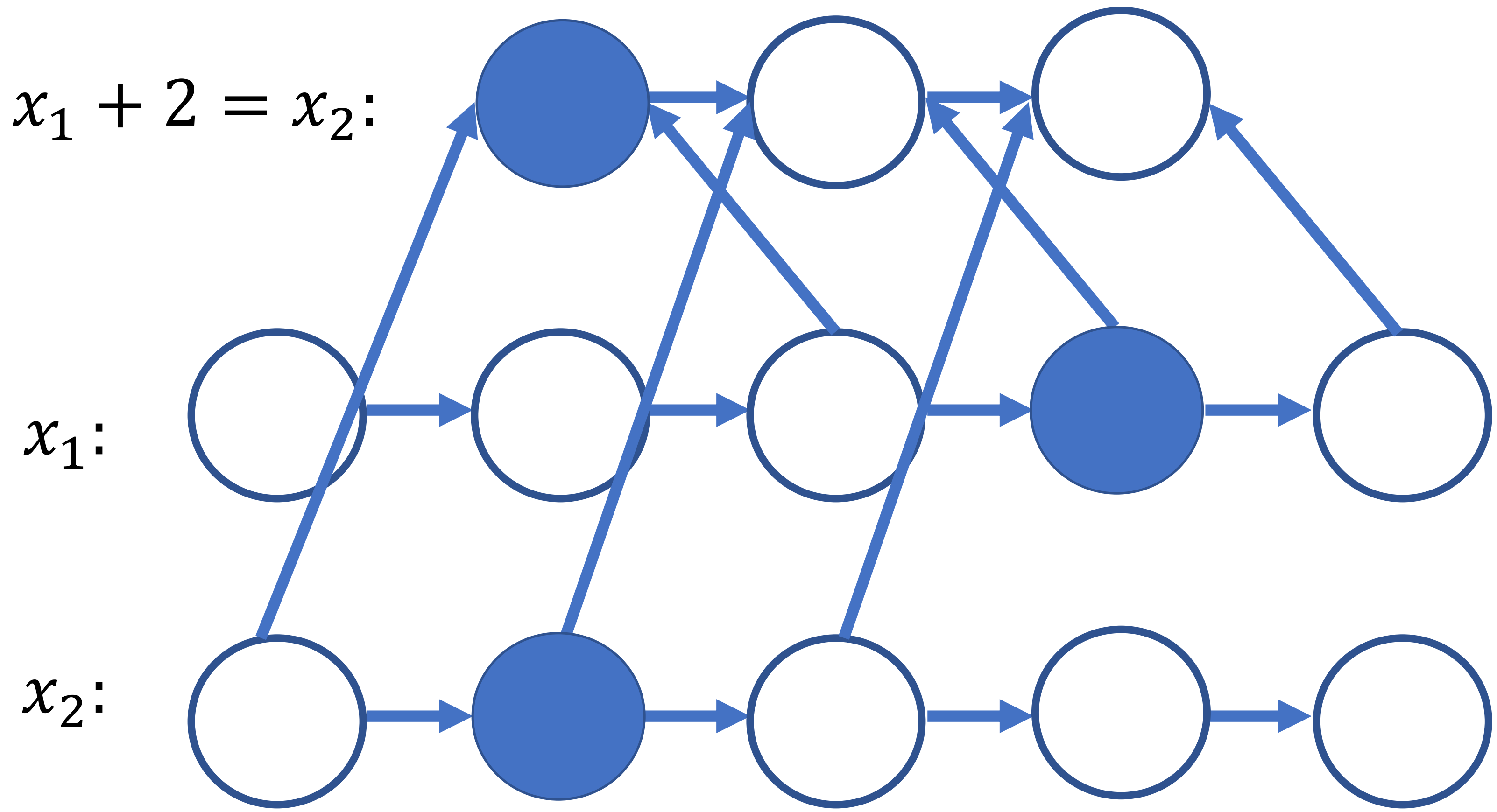
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



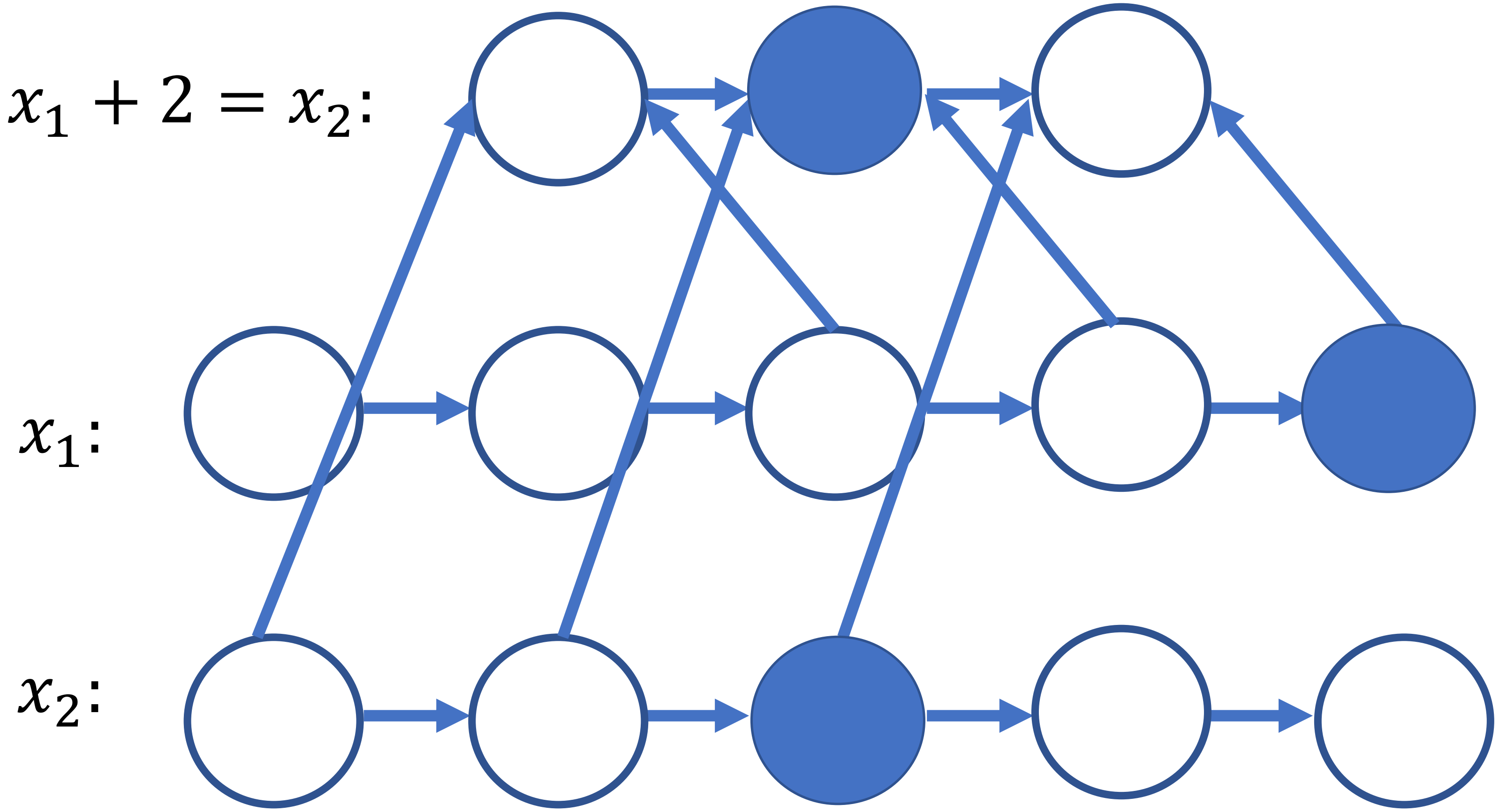
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL.
Otherwise, Pebbling Cost is LARGE



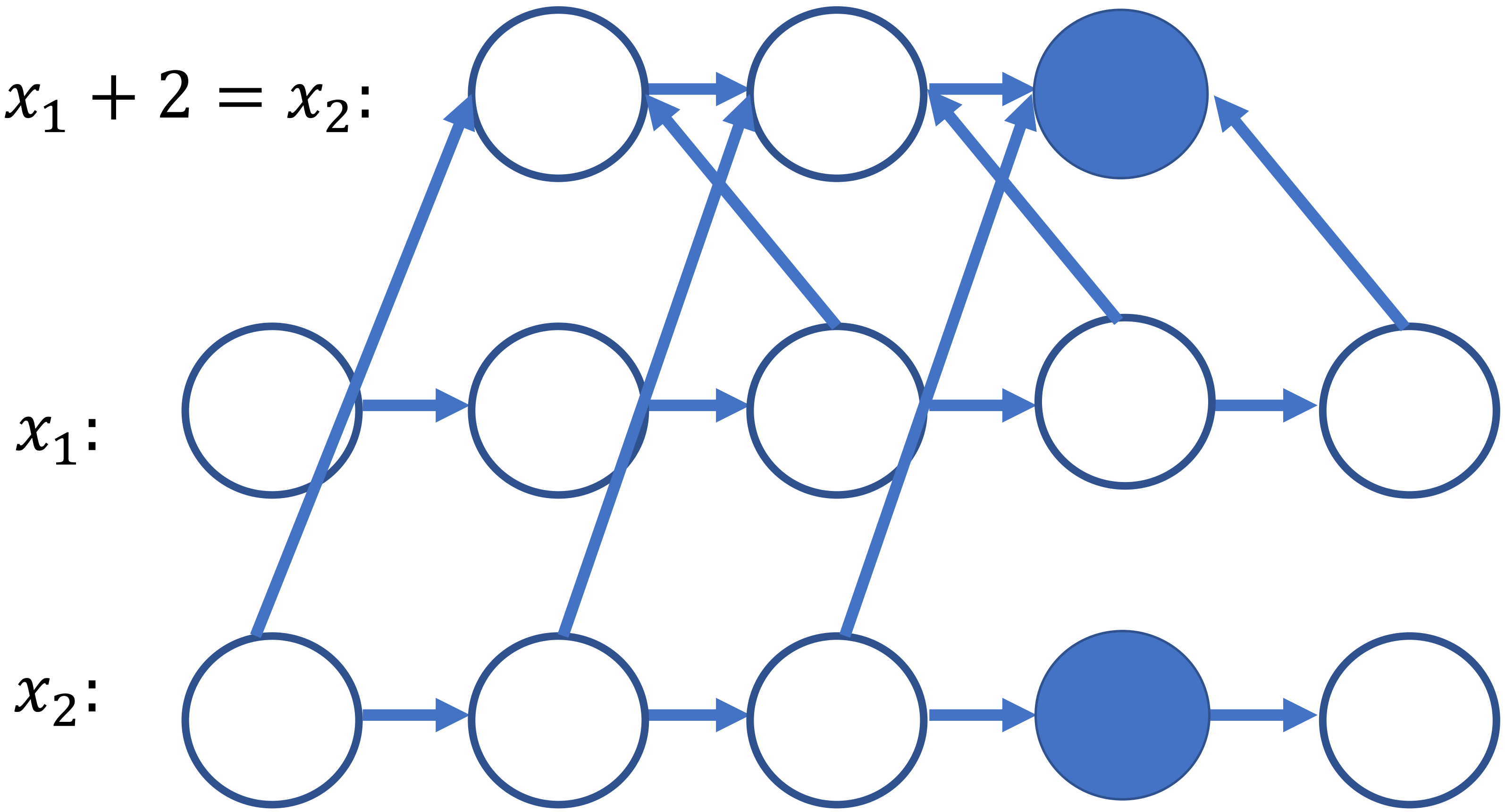
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



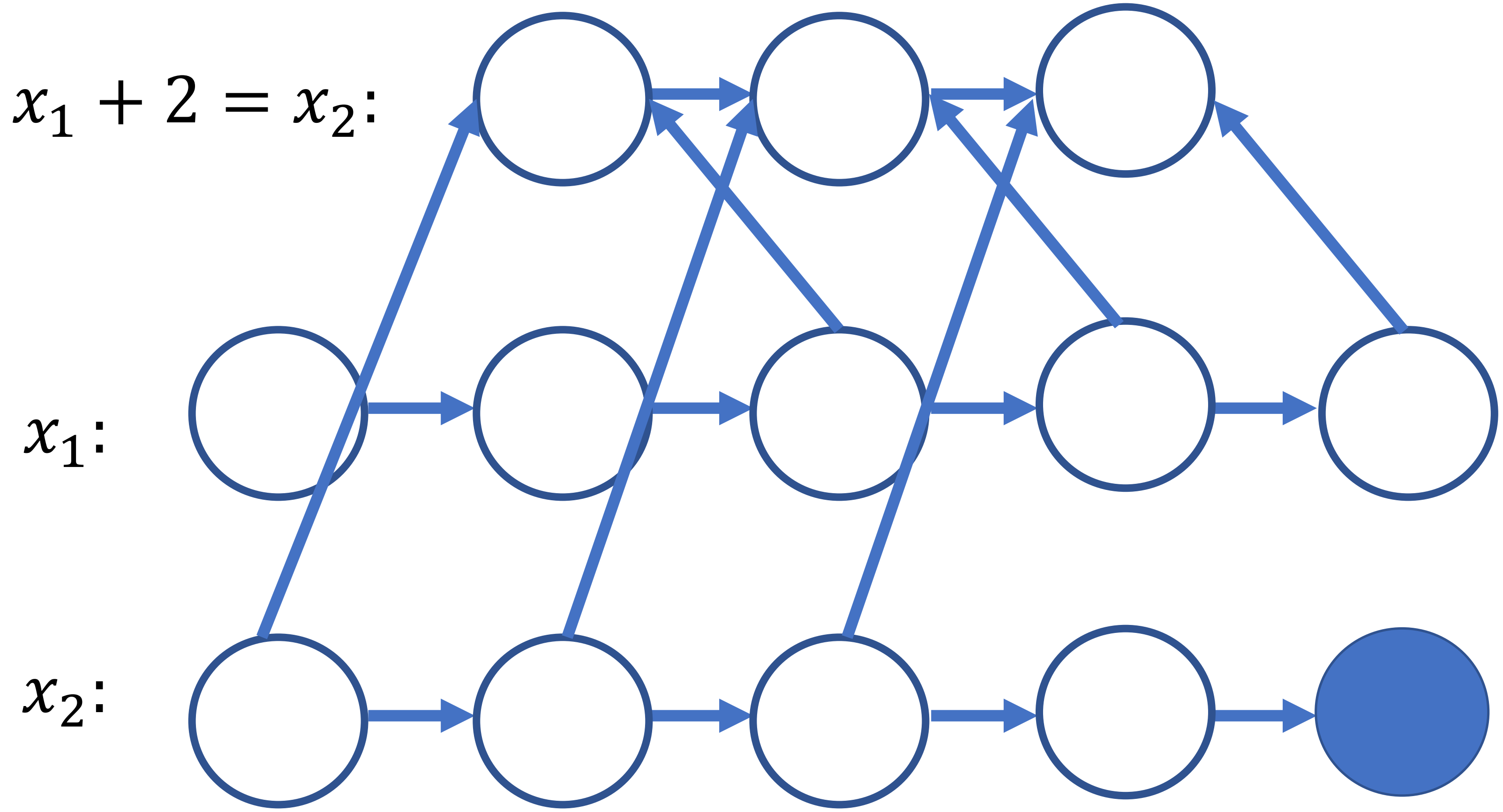
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



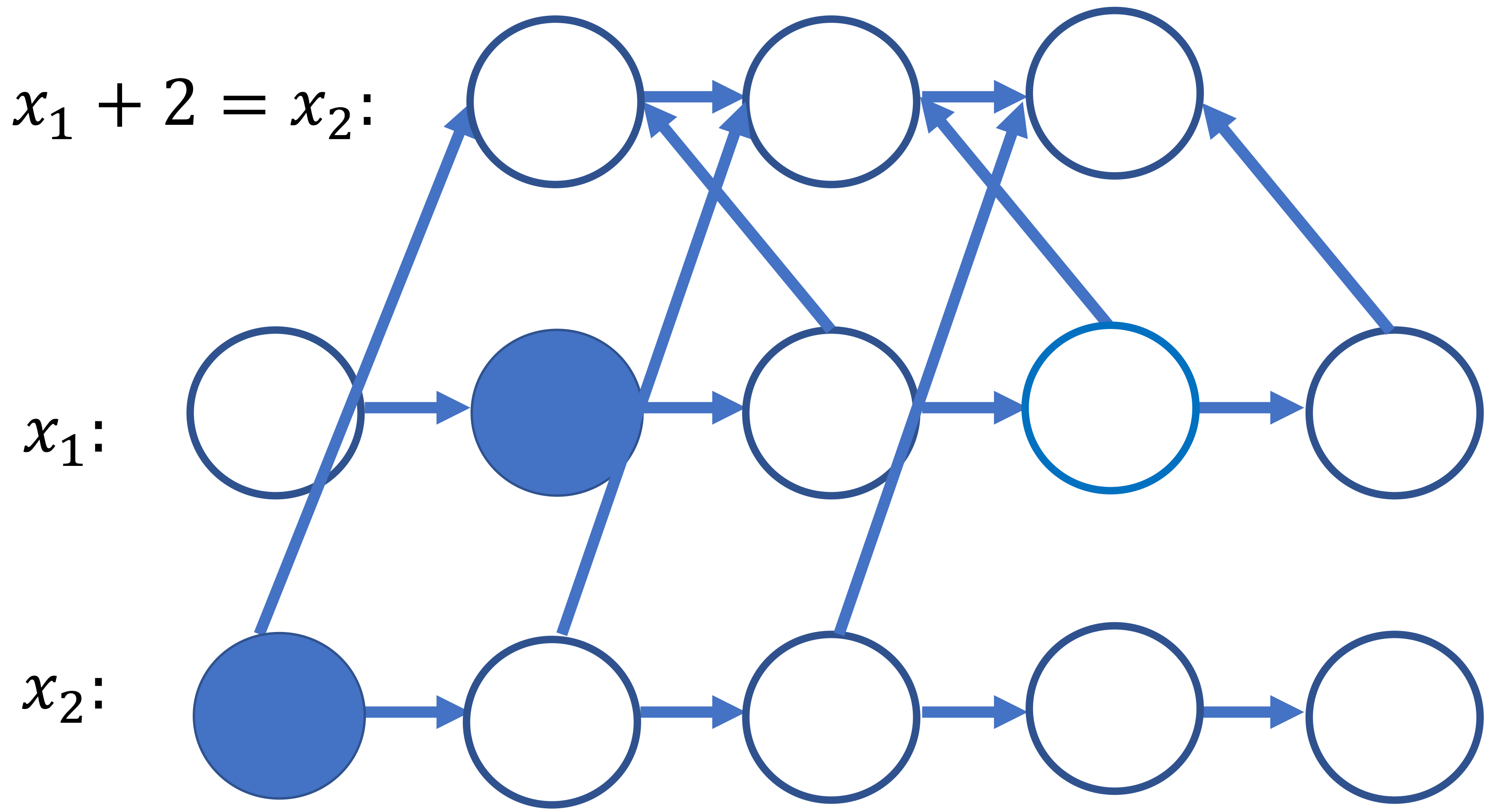
Honest Pebbling

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



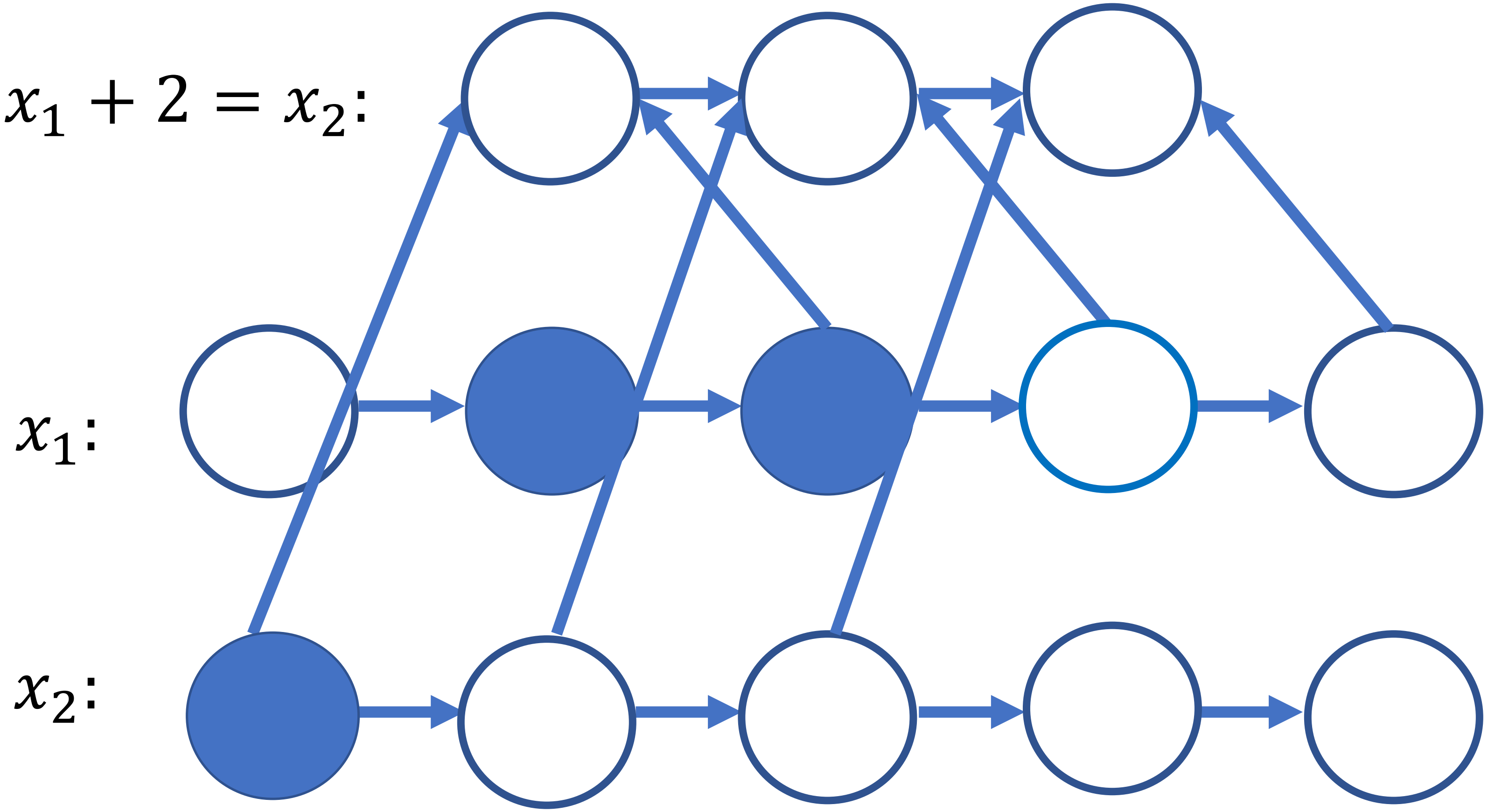
Cheater!

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



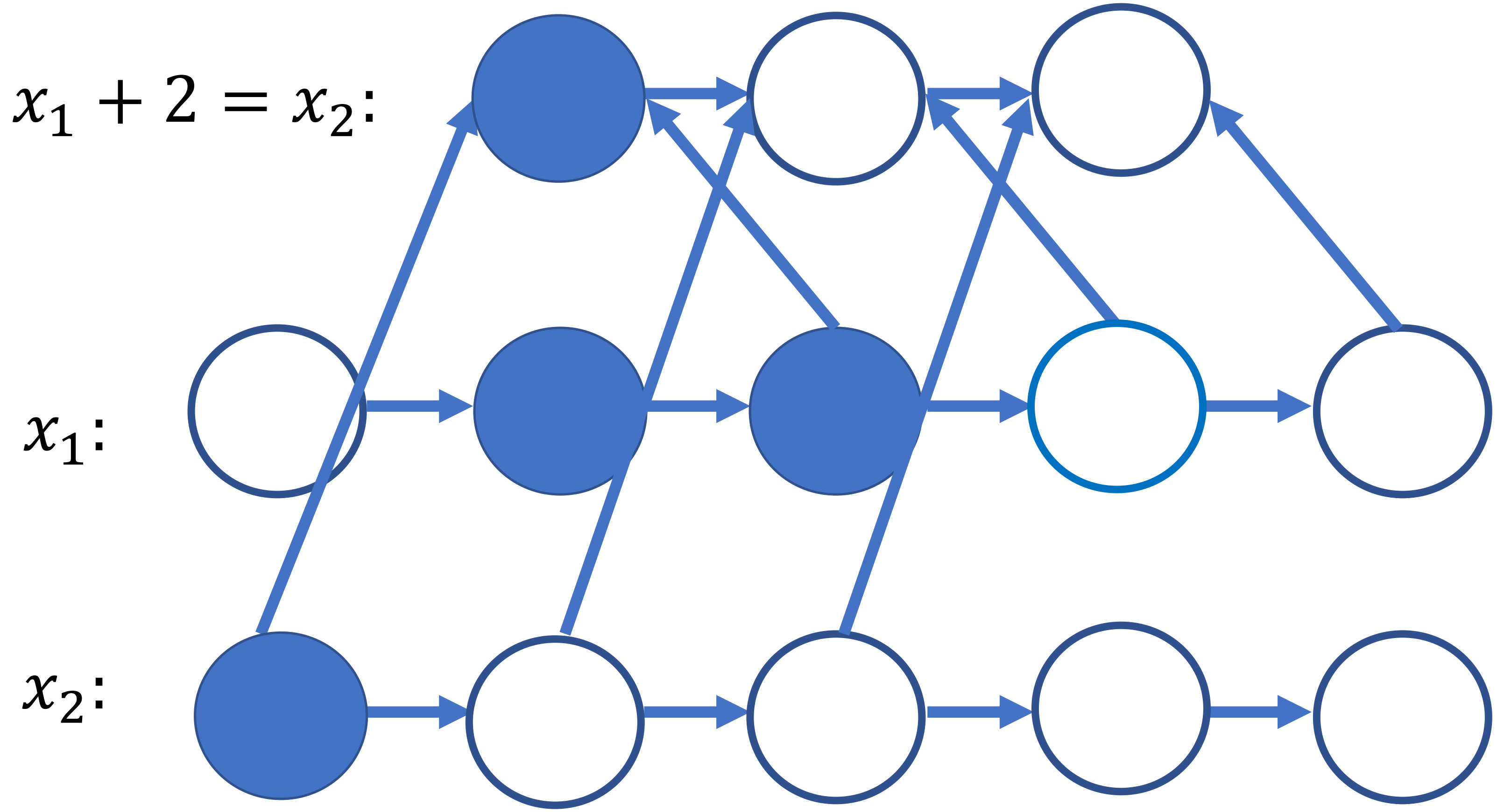
Cheater!

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE

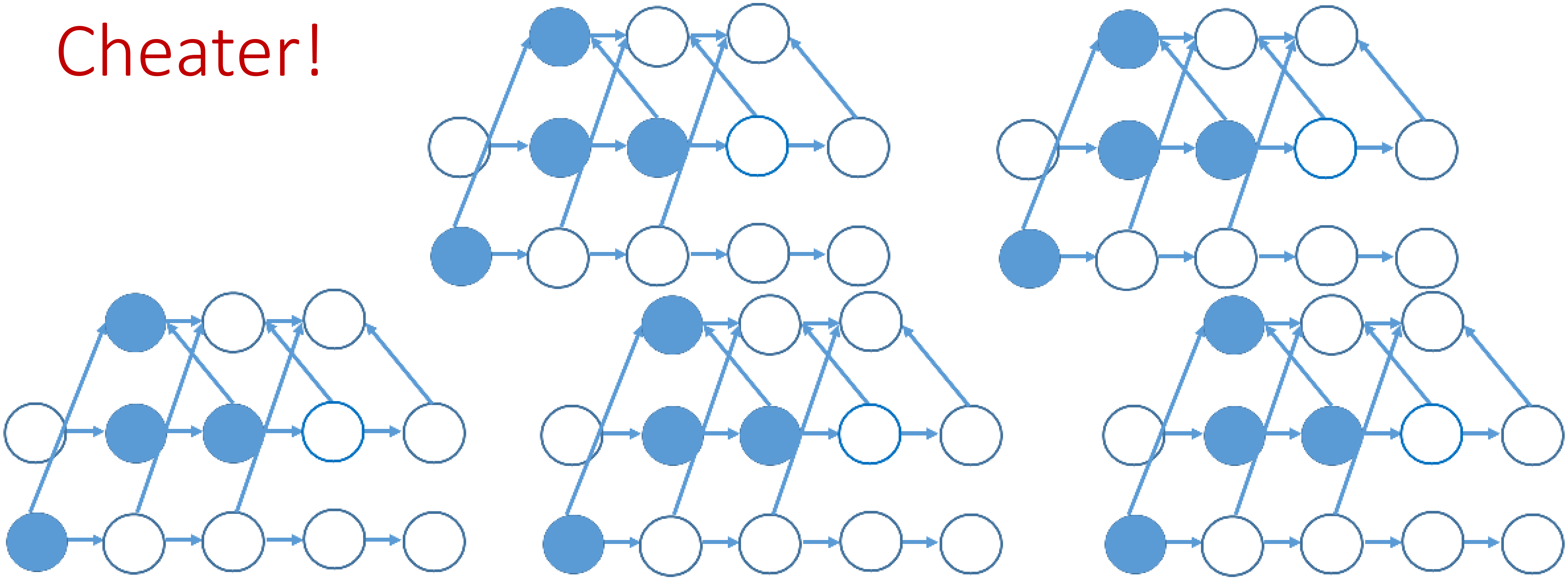


Cheater!

GOAL: If m assignments for k equations, then Pebbling Cost is SMALL. Otherwise, Pebbling Cost is LARGE



Cheater!



Lemma 4 *If the* B2LC *instance has a valid solution, then* $\Pi^{\parallel}_{cc}(G_{B2LC}) \leq \tau cmn + 2cmn + 2ckm + 1$.

Lemma 5 *If the* B2LC *instance does not have a valid solution, then* $\Pi^{\parallel}_{cc}(G_{B2LC}) \geq \tau cmn + \tau$.

Reductions

Figure 5 shows an example of a reduction in its entirety when $\tau = 1$.

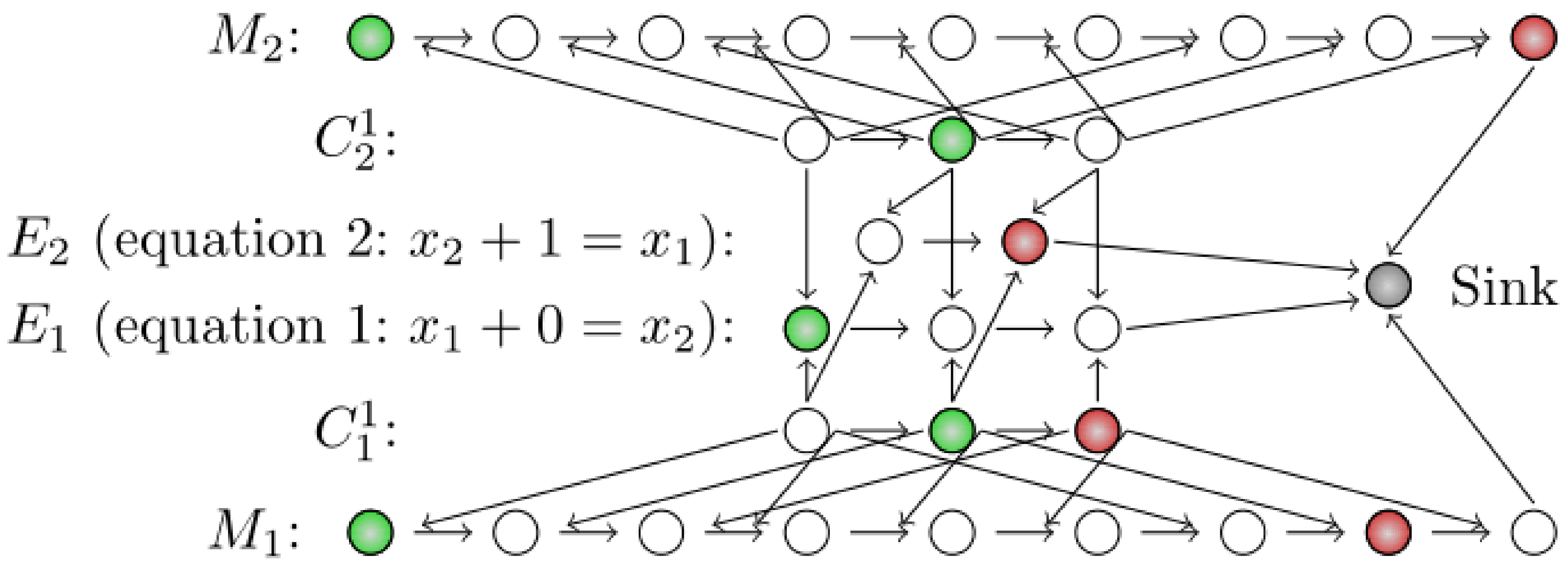
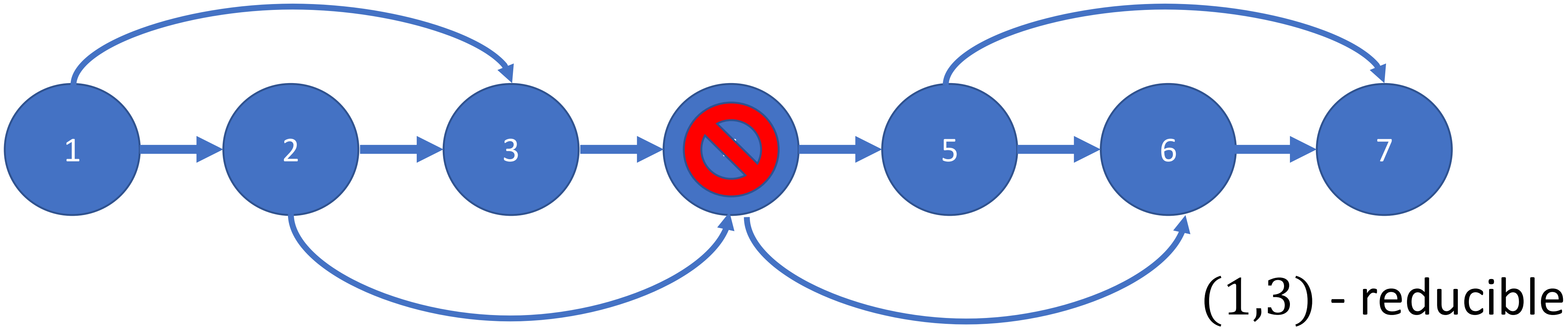


Fig. 5. An example of a complete reduction G_{B2LC}, again $m = 3$ and $c = 3$. The green nodes represent the pebbled vertices at time step 2 while the red nodes represent the pebbled vertices at time step 10.

Graph Reducibility

❖ We say that a directed acyclic graph G is (e, d)-reducible if there exists a set S of e nodes such that $G - S$ has depth at most d.



Graph Reducibility

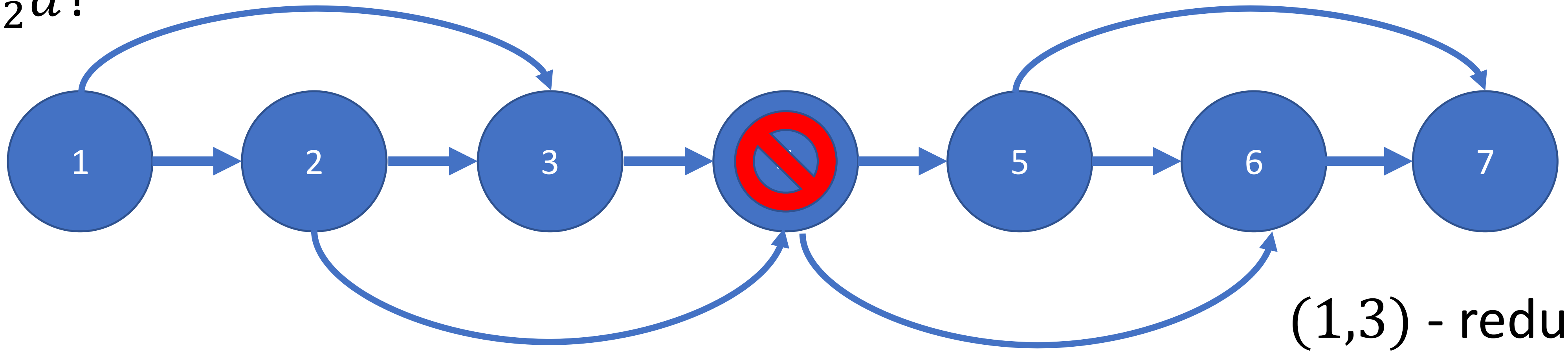
- Can deduce $cc(G)$ from (e, d)-reducible!
- Depth-robustness is a *necessary* condition for secure iMHFs (AB16)
  - There exists attack with $E_R(A) = O\left(en + \sqrt{n^3 d}\right)$, which is $o(n^2)$ for $e, d = o(n)$.
- Depth-robustness is a *sufficient* condition for secure iMHFs (ABP16)
  - $cc(G) \geq ed$

Summary

- We show computing $cc(G)$ is NP-hard (as is computing $st(G)$).
- Integer Program for $cc(G)$ has $\Omega\left(\frac{n}{\log n}\right)$ integrality gap.
  - Evidence that a problem is hard to approximate
- We show that given e, d, it is NP-hard to determine whether a graph is (e, d)-reducible (even for graphs with bounded degree).
- Given d, it is hard to:
  - Approximate e to a factor of 1.3 (minimum Vertex Cover).
  - Approximate e to a factor of 2 (Unique Games Conjecture).
- An optimal cumulative cost pebbling of a graph may take more than n steps.

Open Questions

- Does there exist an algorithm to approximate $cc(G)$?
- Do there exist constants c_1, c_2 so that given an (e, d)-reducible graph, we can a set S of $c_1 e$ nodes such that $G - S$ has depth at most $c_2 d$?

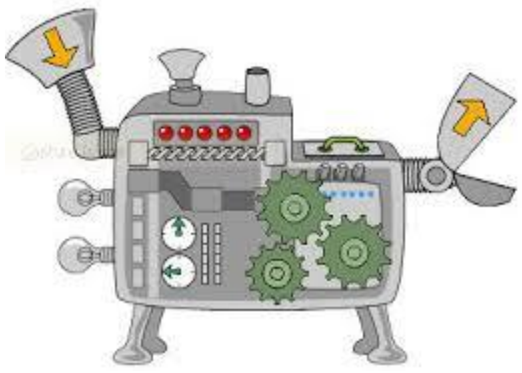


$(1,3)$ - reducible

- Even $O(\log n)$ approximation helps!

рахмат
danke
謝謝
ngiyabonga
spas
teşekkür ederim
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
blagodaram
dank je
misaotra
paldies
grazzi
matondo
спасибі
kiitos
dankie
thank you
gracias
хвала
nanni
nandri
dhanyavad
mauruuru
bedankt
hvala
köszönöm
aciu
dankon
akun
asante
manana
obrigada
murakoze
tenki
enkosi
bayarlalaa
gràçie
djiere dieuf
tau
mochchakkeram
дякую
mamnun
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
go raibh maith agat
chokrane
sobodi
děkuji
sagolun
sukriya
kop khun krap
taiku
obrigado
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
najis tuke
ありがとう
tanemirt
grazie
rahmet
kam sah hamnida
rahmat
terima kasih
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерси
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci

Questions?



(1) Variables: For $1 \le v \le n$ and $0 \le t \le n^2$,

  (a) Integer Program: $x_v^t \in \{0, 1\}$

  (b) Relaxed Linear Program: $0 \le x_v^t \le 1$

(2) Goal (minimize cumulative pebbling cost): $\min \sum_{v \in V} \sum_{t=0}^{n^2} x_v^t$.

(3) Constraint 1 (Must Finish): $\sum_{t=0}^{n^2} x_n^t \ge 1$.

(4) Constraint 2 (No Pebbles At Start): $\sum_{v>0} x_v^0 \le 0$.

(5) Constraint 3 (Pebbling Is Valid): For all v s.t $|\mathsf{Parents}(v)| \ge 1$ and $0 \le t \le n^2 - 1$ we have

$$x_v^{t+1} \le x_v^t + \frac{\sum_{v' \in \mathsf{Parents}(v)} x_{v'}^t}{|\mathsf{Parents}(v)|} \ .$$

Fig. 5: Integer Program for Pebbling.

Theorem 9 *Let* G *be with constant indegree* δ*. Then there is a fractional solution to our LP Relaxation (of the Integer Program in Figure 5) with cost at most* $3n$*.*

Reductions

- Reducing 3-PARTITION to B2LC
- Reducing B2LC to $cc(G)$

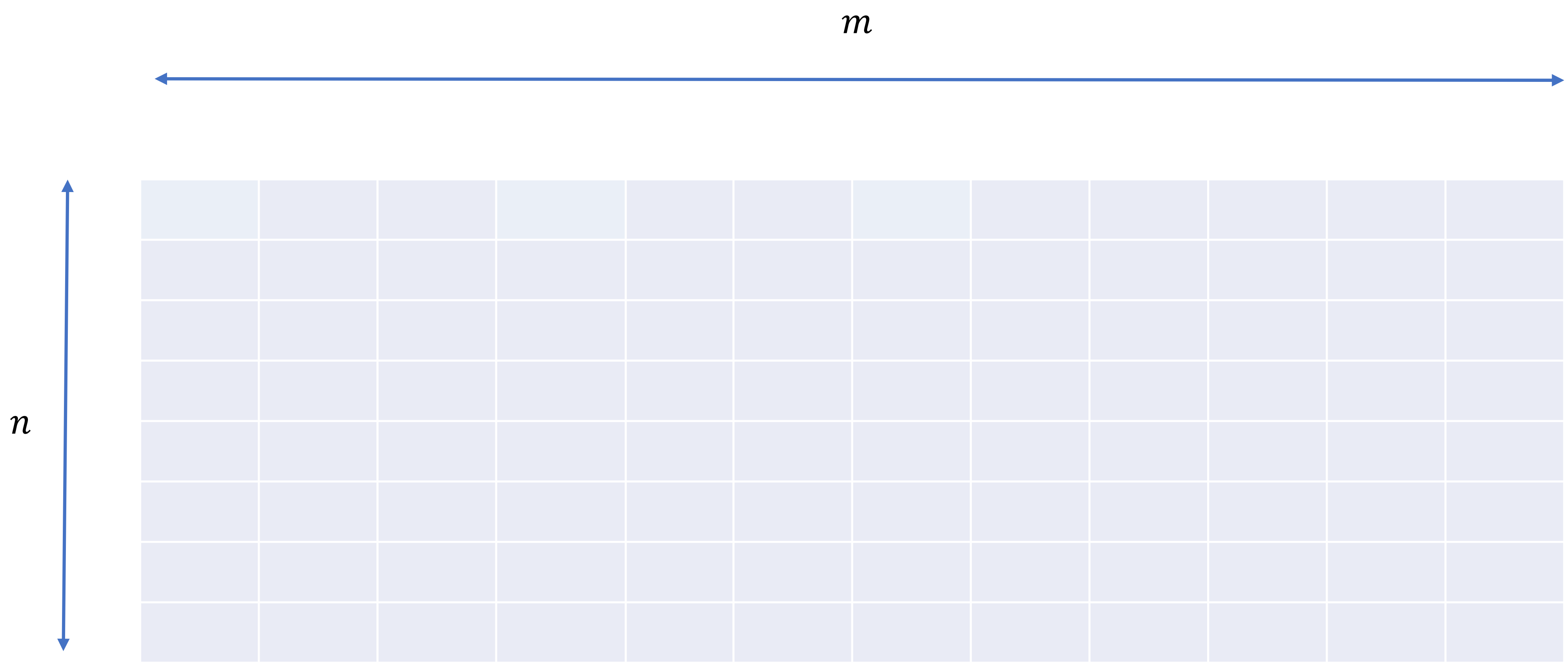
Reductions

$$T = \sum_{\{i=1\}}^{m} s_m$$

$$\begin{aligned}
x_1 + s_1 = x_2, \quad & x_2 + s_2 = x_3, \quad \ldots, \quad x_m + s_m = x_{m+1}, \\
x_1 + 0 = x_2, \quad & x_2 + 0 = x_3, \quad \ldots, \quad x_m + 0 = x_{m+1}, \\
x_1 + T = x_2, \quad & x_2 + T = x_3, \quad \ldots, \quad x_m + T = x_{m+1}, \\
x_1 + 2T = x_2, \quad & x_2 + 2T = x_3, \quad \ldots, \quad x_m + 2T = x_{m+1}, \\
x_1 + (n-2)T = x_2, \quad & x_2 + (n-2)T = x_3, \quad \ldots, \quad x_m + (n-2)T = x_{m+1},
\end{aligned}$$

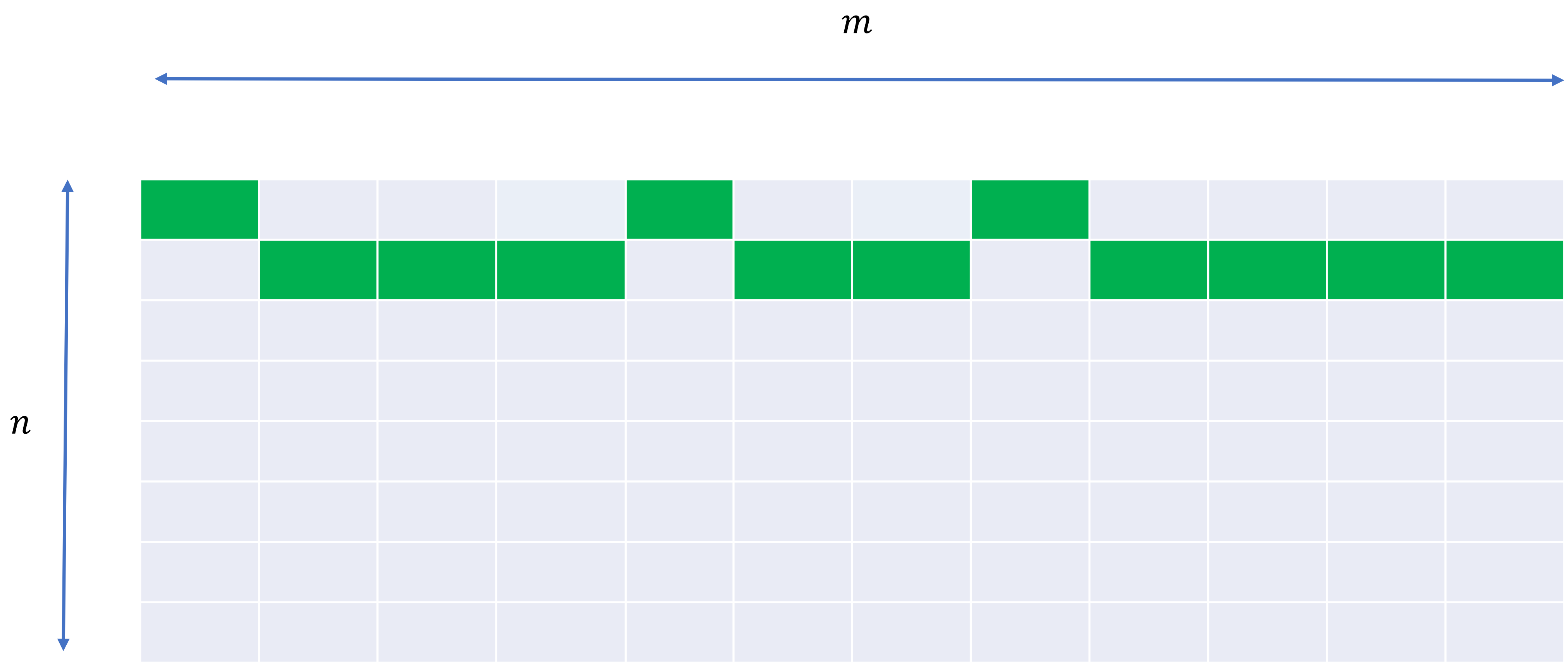
$$x_1 + \frac{T}{n} + 3(i-1)(n-2)T = x_{m+1},$$

Reductions



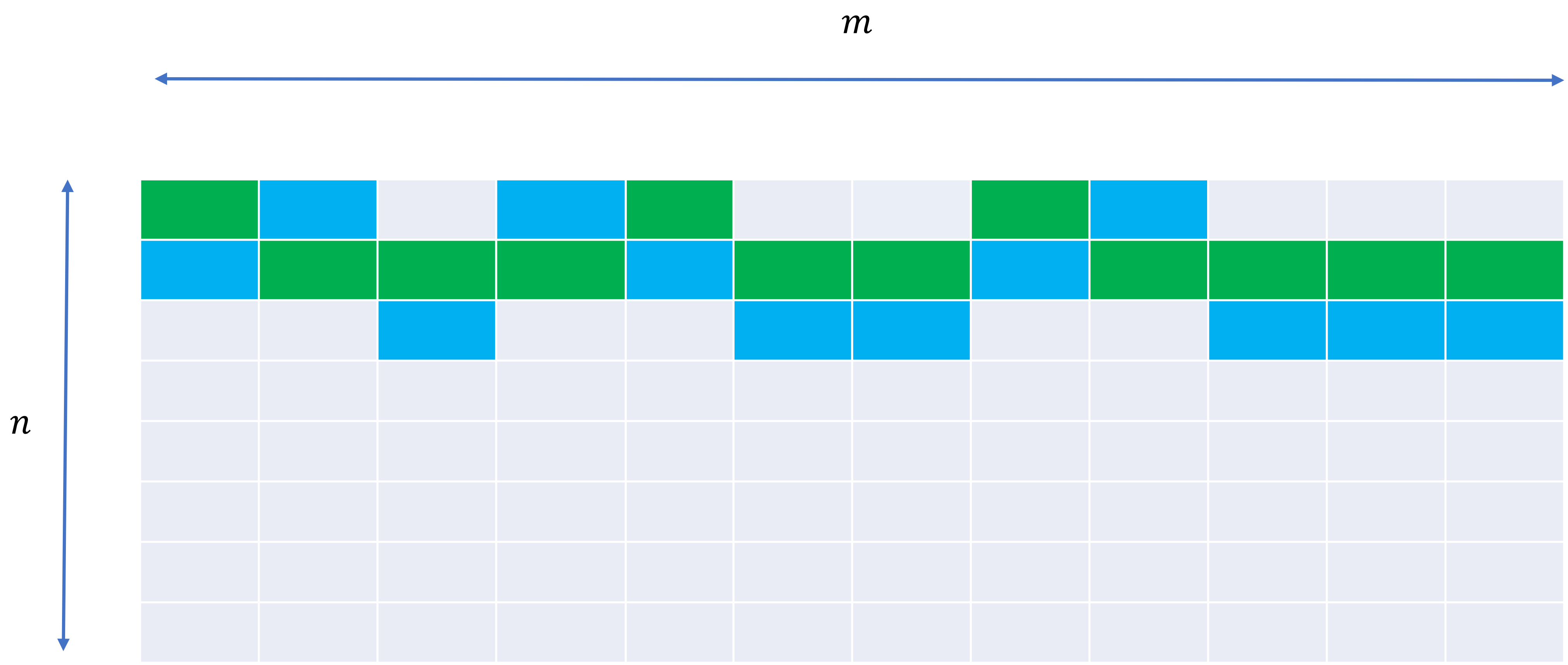
$m = 3n$ integers in original 3-PARTITION instance

Reductions



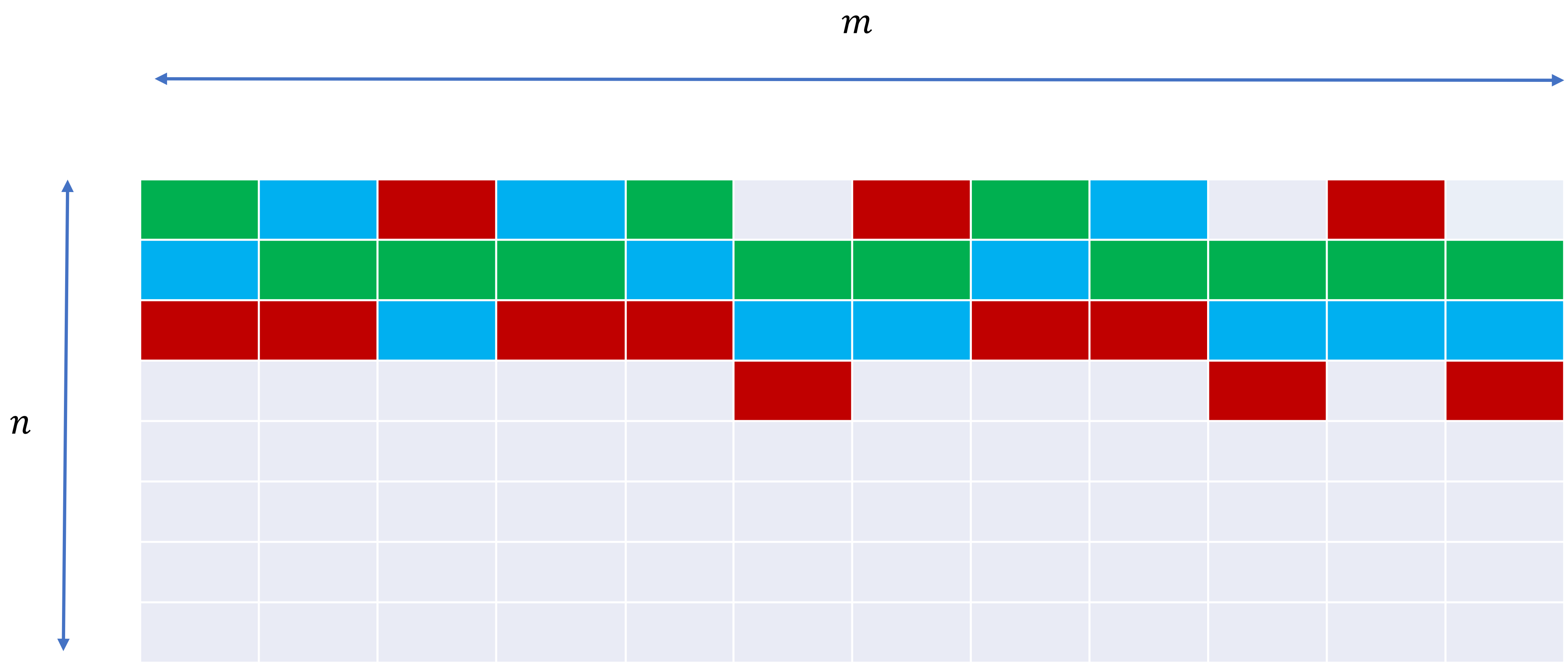
$m = 3n$ integers in original 3-PARTITION instance

Reductions



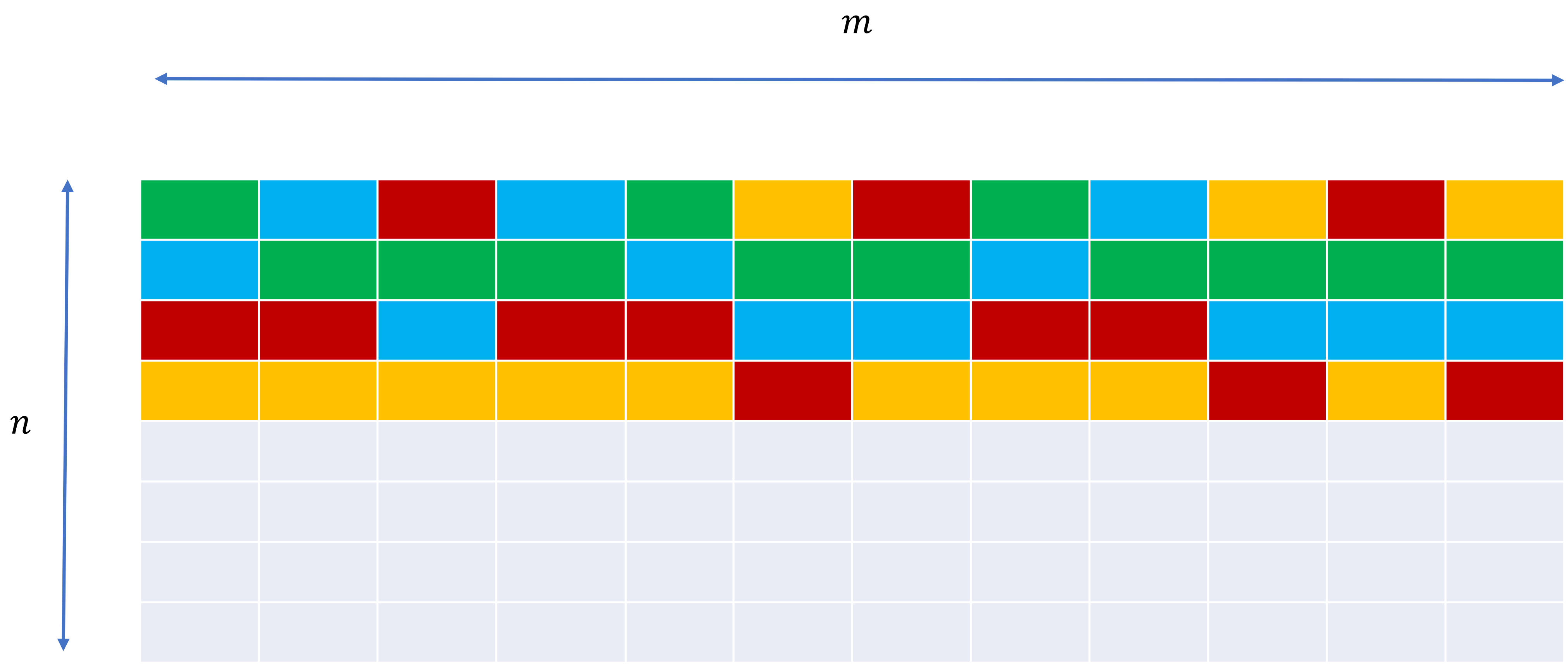
$m = 3n$ integers in original 3-PARTITION instance

Reductions



$m = 3n$ integers in original 3-PARTITION instance

Reductions



$m = 3n$ integers in original 3-PARTITION instance

3-PARTITION

- Given set of $3n$ integers, can we partition them into n sets, each with the same sum?
- $\{1,2,4,5,6,7,8,11,13\}$
  - $\{2,4,13\} \rightarrow 19$ $\{1,7,11\} \rightarrow 19$ $\{5,6,8\} \rightarrow 19$
- $\{1,2,3,4,6,7,9,10,11\}$

Reductions

- ✓ Reducing 3-PARTITION to B2LC
- ❖ Reducing B2LC to $cc(G)$