Subexponential Lower Bounds for the Simplex Algorithm

Oliver Friedmann

Department of Computer Science,
Ludwig-Maximilians-Universität Munich, Germany.

January 20, 2011

Outline

1. LPs and Simplex algorithm
2. MDPs and Policy iteration
3. Random Edge
4. Random Facet and Least Entered
5. All is well that ends well?

LPs and Simplex algorithm

Outline

1. LPs and Simplex algorithm
   - Linear programming
   - Simplex algorithm
   - Pivoting rules
   - Technique

2. MDPs and Policy iteration

3. Random Edge

4. Random Facet and Least Entered

5. All is well that ends well?

Linear programming

Context

- Optimization of a linear objective subject to linear (in)equality constraints
- One of the most important computational problems
- Simplex Algorithm introduced by Dantzig in 1947 for solving LPs
- Weakly polytime algorithms: Ellipsoid Method by Khachiyan, Interior Point Method by Karmarkar
- No strongly polytime algorithm known

Linear programming

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned} \qquad\qquad \begin{aligned} \min \quad & b^T y \\ \text{s.t.} \quad & A^T y \geq c \end{aligned}$$

- Simplex algorithm (Dantzig 1947)
- Ellipsoid algorithm (Khachiyan 1979)
- Interior-point algorithm (Karmakar 1984)

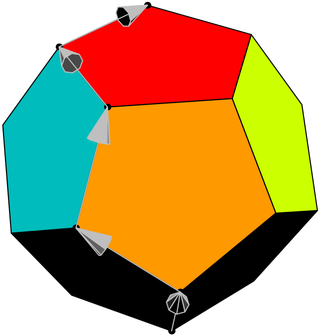
Simplex algorithm

Context

- Introduced by Dantzig in 1947 for solving LPs
- Fixpoint iteration algorithm
- Asymptotic complexity depends on the number of iterations
- Parameterized by pivoting rules
- Hirsch conjecture

Simplex algorithm

Dantzig 1947



Move up, along an edge, to a neighboring vertex, until reaching the top

Pivoting rules

Pivoting rules

Simplex method is parameterized by a pivoting rule.

Pivoting rule = method of chosing adjacent vertices with better objective

- only single-switching
- deterministic vs. randomized
- memorizing vs. oblivious

Deterministic rules

- Dantzig's rule
- Bland's rule
- Lexicographic rule
- many more...

Theorem (Klee-Minty (1972) *et al.*)

All known to require an exponential number of steps in the worst case.

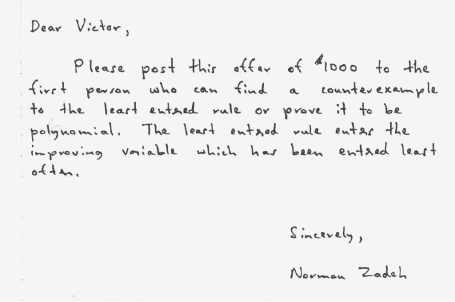
Randomized rules

- Random-Facet: recursively find the optimum (see later) due to Kalai ('92), and Matoušek, Sharir and Welzl ('96)
- Random-Edge: choose a random improving edge

Theorem (Friedmann-Hansen-Zwick (SODA'2011,...))

There are (different) explicit LPs on which Random-Facet *and* Random-Edge *require an expected subexponential number of iterations.*

Memorizing rule



Dear Victor,

Please post this offer of $1000 to the first person who can find a counterexample to the least entered rule or prove it to be polynomial. The least entered rule enters the improving variable which has been entered least often.

Sincerely,

Norman Zadeh

(taken from David Avis' paper)

Theorem (Friedmann (IPCO'2011))

There are explicit LPs on which Least-Entered *requires a subexponential number of iterations.*

Technique

Lower bound technique

1. Lower bounds for Markov decision process policy iteration
2. Induce linear programs
3. Correspondence of simplex algorithm and policy iteration

---

originally : lower bounds for parity games first

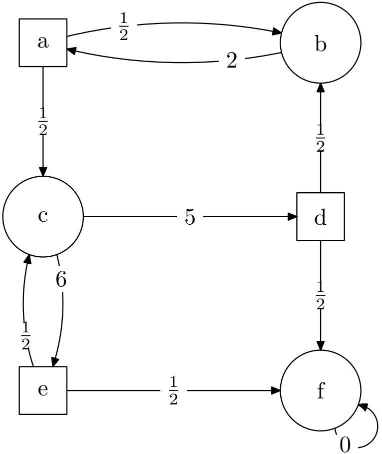
MDPs and Policy iteration

Outline

Markov decision processes

Markov decision process

due to Shapley '53, Bellman '57, Howard '60, ...

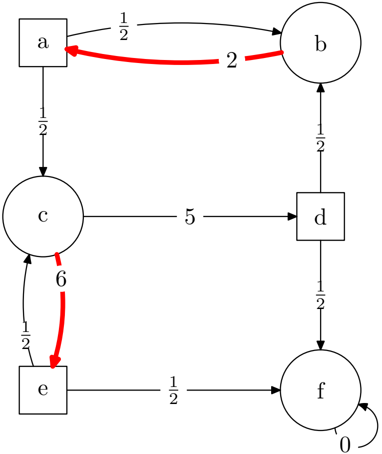


- Controller (circle) chooses outgoing action
- Randomizer (square) determines successive state
- Objective: maximize expected total reward

$$\max \mathbb{E}[\sum_{i=0}^{\infty} r(a_i)]$$

Markov decision process

due to Shapley '53, Bellman '57, Howard '60, ...



- Controller (circle) chooses outgoing action
- Randomizer (square) determines successive state
- Objective: maximize expected total reward

$$\max \mathbb{E}[\sum_{i=0}^{\infty} r(a_i)]$$

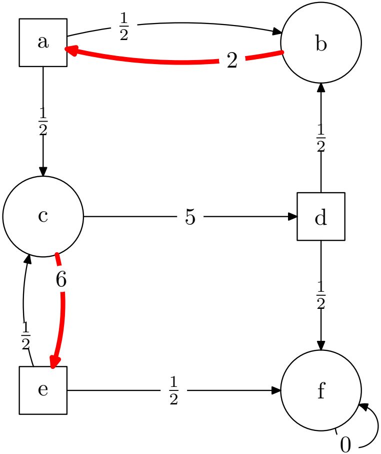
policy σ = choice of an action from each state

Policy iteration

Context

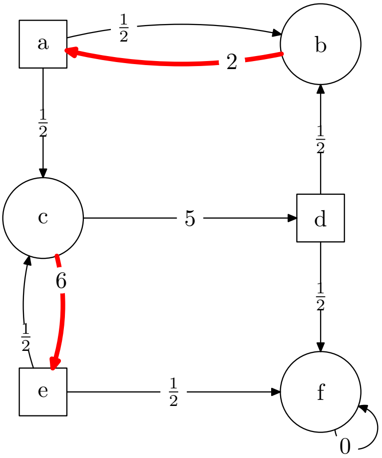
- Introduced by Howard in 1960 to solve Markov Decision Processes
- Transferred to many other areas by several authors
- Fixpoint iteration algorithm
- Asymptotic complexity depends on the number of iterations
- Parameterized by improvement rules

Policy valuations



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c =$$
$$\mathrm{val}_\sigma(b) = 2 + a =$$
$$\mathrm{val}_\sigma(c) = 6 + e =$$
$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(f) =$$

Policy valuations



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c = 14$$
$$\mathrm{val}_\sigma(b) = 2 + a = 16$$
$$\mathrm{val}_\sigma(c) = 6 + e = 12$$
$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f = 8$$
$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f = 6$$
$$\mathrm{val}_\sigma(f) = 0$$

(Improving) Switches

A σ-switch is an edge $e \in E_0 \setminus \sigma$ not chosen by σ.

Facts about switches

- Comparability: $\mathrm{val}_\sigma \trianglelefteq \mathrm{val}_{\sigma[e]}$ or $\mathrm{val}_{\sigma[e]} \trianglelefteq \mathrm{val}_\sigma$ for every σ-switch.
- Easy check: $\mathrm{val}_\sigma \lhd \mathrm{val}_{\sigma[(v,w)]}$ iff $\mathrm{val}_\sigma(\sigma(v)) < \mathrm{val}_\sigma(w)$.

Improving switches: $I(\sigma) = \{e \mid \mathrm{val}_\sigma \lhd \mathrm{val}_{\sigma[e]}\}$

Theorem

- Switching: $\emptyset \subsetneq J \subseteq I(\sigma)$ implies $\mathrm{val}_\sigma \lhd \mathrm{val}_{\sigma[J]}$.
- Optimality: $I(\sigma) = \emptyset$ implies σ is optimal .

Policy iteration algorithm

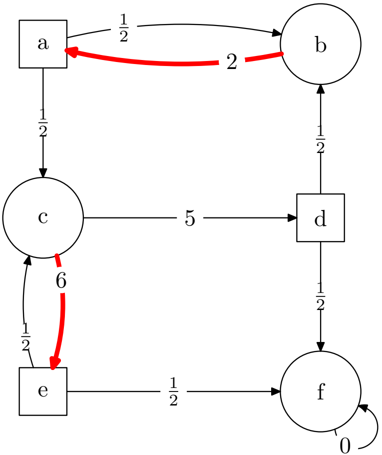
Howard (1960), Hoffman-Karp (1966), Puri (1995), Vöge / Jurdziński (2000)...

Strategy improvement / Policy iteration

1: **while** σ is not optimal **do**
2: $\quad \sigma \leftarrow \sigma[J]$ for some $\emptyset \subsetneq J \subseteq I(\sigma)$
3: **end while**

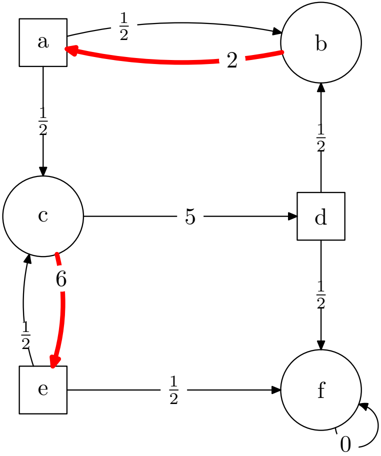
Complexity: depends on the number of iterations .

Policy iteration



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c =$$
$$\mathrm{val}_\sigma(b) = 2 + a =$$
$$\mathrm{val}_\sigma(c) = 6 + e =$$
$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(f) =$$

Policy iteration



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c = 14$$

$$\mathrm{val}_\sigma(b) = 2 + a = 16$$

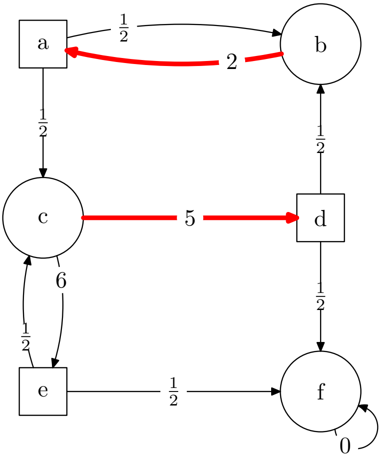
$$\mathrm{val}_\sigma(c) = 6 + e = 12$$

$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f = 8$$

$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f = 6$$

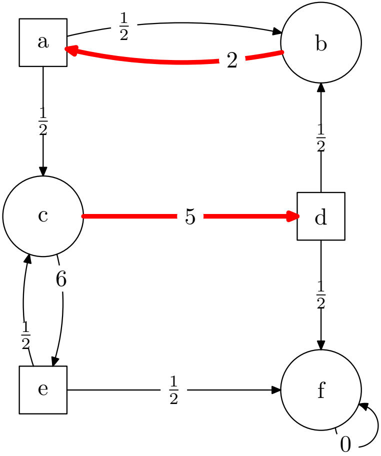
$$\mathrm{val}_\sigma(f) = 0$$

Policy iteration



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c =$$
$$\mathrm{val}_\sigma(b) = 2 + a =$$
$$\mathrm{val}_\sigma(c) = 5 + d =$$
$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f =$$
$$\mathrm{val}_\sigma(f) =$$

Policy iteration



$$\mathrm{val}_\sigma(a) = \frac{1}{2}b + \frac{1}{2}c = 16$$
$$\mathrm{val}_\sigma(b) = 2 + a = 18$$
$$\mathrm{val}_\sigma(c) = 5 + d = 12$$
$$\mathrm{val}_\sigma(d) = \frac{1}{2}b + \frac{1}{2}f = 9$$
$$\mathrm{val}_\sigma(e) = \frac{1}{2}c + \frac{1}{2}f = 7$$
$$\mathrm{val}_\sigma(f) = 0$$

Complexity

Policy Iteration is parameterized by an improvement rule.

Improvement Rule = method of chosing improving switches

- Single-switching vs. Multi-switching
- Deterministic vs. Randomized
- Memorizing vs. Oblivious

Diameter

Question: theoretically possible to have polynomially many iterations?

Let G be a game and n be the number of nodes.

Definition: the diameter of G is the least number of iterations required to solve G

Diameter Theorem

The diameter of G is less or equal to n .

MDPs as LPs

Dual LP of unichain MDP

$$(D) \qquad \begin{array}{ll} \min & \sum_{i \in V} y_i \\ \text{s.t.} & y_i - \sum_{j \in V} p_{j,a} y_j \geq r_a \,,\; i \in V, a \in A_i \end{array}$$

- V - set of all states
- A - set of all actions, A_i - set of actions from state i
- r_a - reward of action a , $p_{i,a}$ - transition probability from action a to state i
- y_i - value of state i

Primal LP of unichain MDP

$$(P) \qquad \begin{array}{ll} \max & \sum_{a\in A} r_a x_a \\ \text{s.t.} & \sum_{a\in A_i} x_a - \sum_{a\in A} p_{i,a} x_a = 1\,,\ i \in V \\ & x_a \ \geq \ 0 \quad , \quad a \in A \end{array}$$

- x_a - expected number of times of taking action a , when starting from all positions

Summary

Pivoting on the induced LP

Theorem

Vertex of bfs in the primal LP corresponds to policy in the MDP. Improving switches correspond to pivoting steps.

Theorem

Optimal solution of the dual LP corresponds to optimal policies in the MDP. Policy corresponds to basic solution of the dual.

Policy iteration vs. Simplex method

	Policy iteration	Simplex method
Pivoting Rules	single- and multi-switch	single-switch only
Diameter	linear	unknown / Hirsch conjecture

Random Edge

Outline

Random edge rule

Random edge rule

Random-Edge rule

Perform single switch arbitrarily at random.

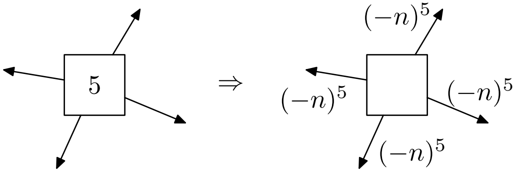
Context

- Abstract lower bound: $2^{\Omega(\sqrt[3]{n})}$ (Matoušek, Szabó 2006)

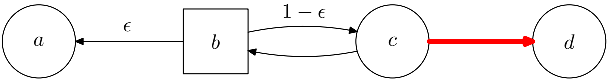
Motivation

Construction principles

1. Process always reaches the sink (unichain MDPs)
2. Priority rewards on the nodes

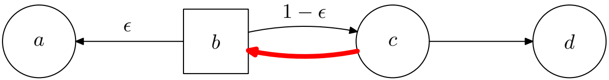


Cycle gadgets



$$\mathrm{val}_\sigma(b) \approx \mathrm{val}_\sigma(d)$$

Cycle gadgets



$$\mathrm{val}_\sigma(b) = \mathrm{val}_\sigma(a)$$

Binary Counting - How does it work?

101011

Binary Counting - How does it work?

101011

↓

Setting

↓

101111

Binary Counting - How does it work?

$$\texttt{101}\color{blue}{\texttt{0}}\color{black}{\texttt{11}}$$

$$\downarrow$$

Setting

$$\downarrow$$

$$\texttt{101}\color{red}{\texttt{1}}\color{green}{\texttt{11}}$$

$$\downarrow$$

Resetting

$$\downarrow$$

$$\texttt{101}\color{red}{\texttt{1}}\color{black}{\texttt{00}}$$

Binary Counting - How does it work?

101011

↓

Setting

↓

101111

↓

Resetting

↓

101100

↓

Activating

↓

101100

Binary Counters

Binary Counting proceeds in three phases.

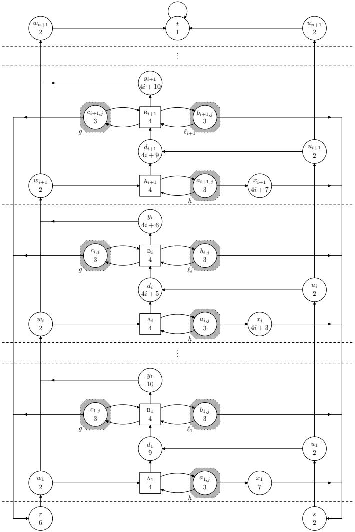
1. Set the least unset bit
2. Reset all lower (inactive) bits
3. Activate recently set bit

How to separate active from inactive bits? By access control .

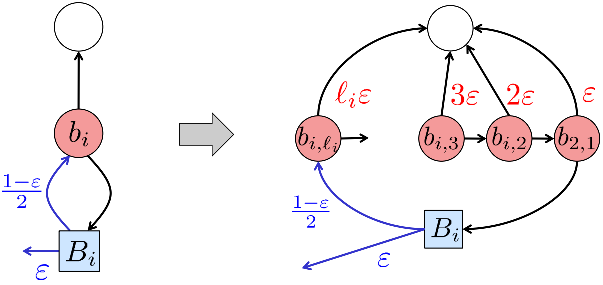
- Attach access control structure to every bit
- Set bits are activated
- Recently set bit is not activated
- After resetting lower bits, recently set bit is activated

Construction

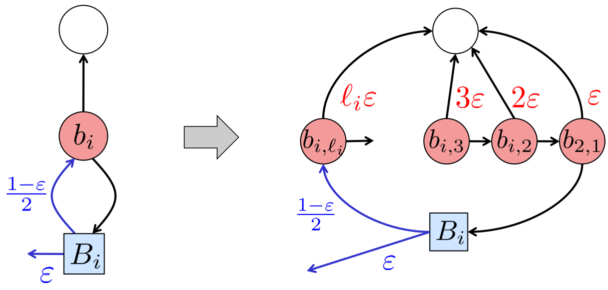
Full construction



Cycle Gadget

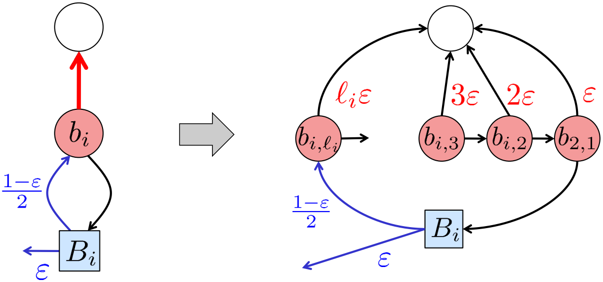


Cycle Gadget



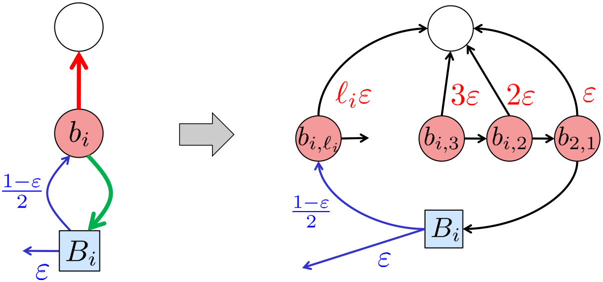
Cycles close one edge at a time

Cycle Gadget



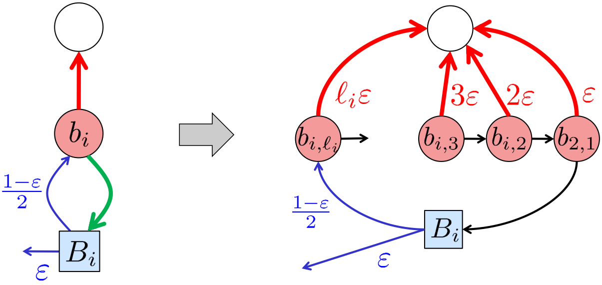
Cycles close one edge at a time

Cycle Gadget



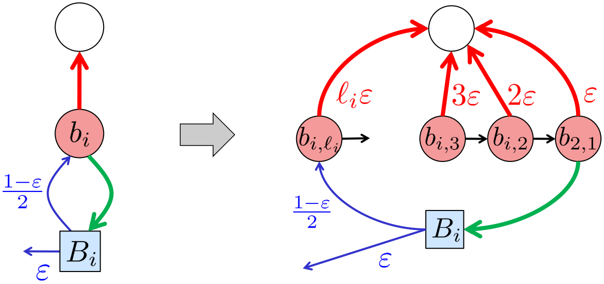
Cycles close one edge at a time

Cycle Gadget



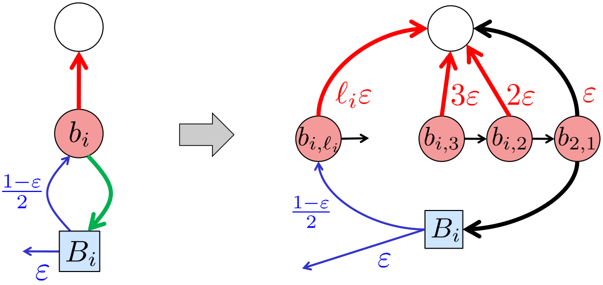
Cycles close one edge at a time

Cycle Gadget



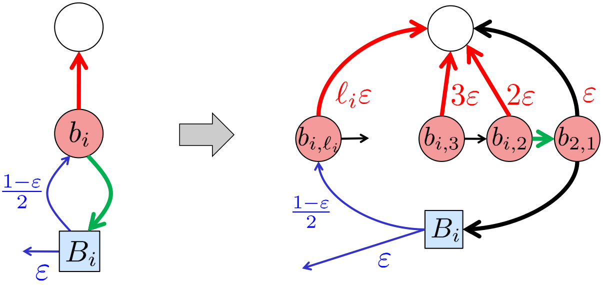
Cycles close one edge at a time

Cycle Gadget



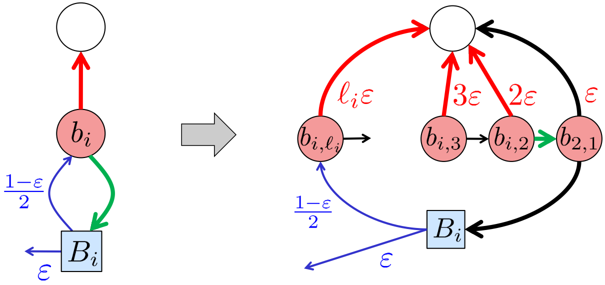
Cycles close one edge at a time

Cycle Gadget



Cycles close one edge at a time

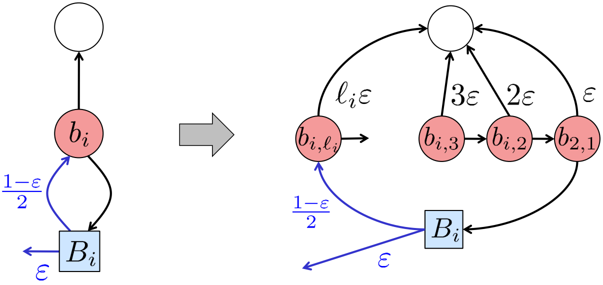
Cycle Gadget



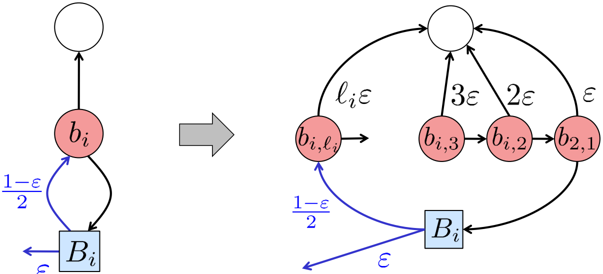
Cycles close one edge at a time

Shorter cycles close faster

Cycle Gadget

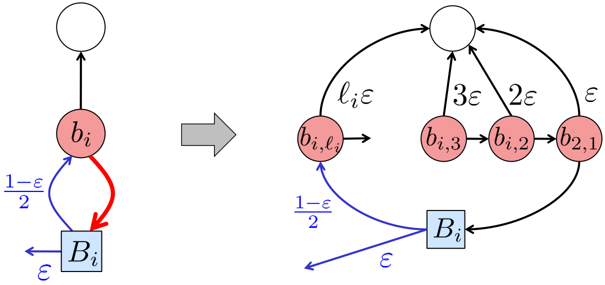


Cycle Gadget



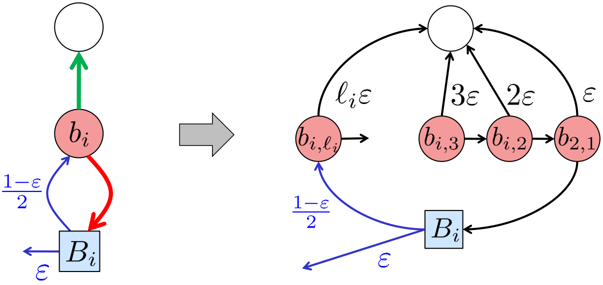
Cycles open "simultaneously"

Cycle Gadget



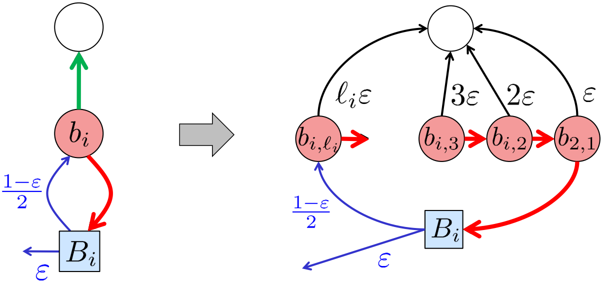
Cycles open “simultaneously”

Cycle Gadget



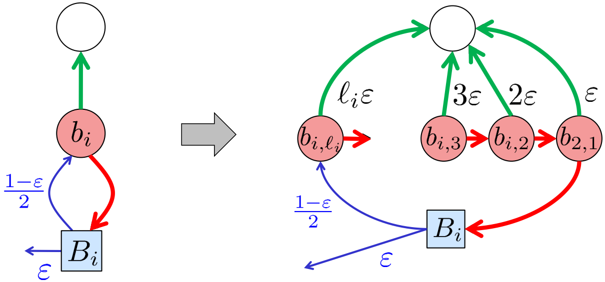
Cycles open "simultaneously"

Cycle Gadget



Cycles open “simultaneously”

Cycle Gadget



Cycles open “simultaneously”

Increment phases: from 101011 to 101100

1. Setting: $\begin{cases} B_k \text{ counting cycle closes} \\ C_k \text{ helper cycle closes} \end{cases}$
2. Resetting: $\begin{cases} U \text{ lane realigns} \\ A_i \text{ and } B_i \text{ cycles } (i < k) \text{ open} \end{cases}$
3. Activating: $\begin{cases} A_k \text{ access cycle closes} \\ W \text{ lane realigns} \\ C_i \text{ cycles of unset bits open} \end{cases}$

Analysis

Cycles (opening and closing) and lanes compete with each other.

Supposed candidate has to win with high probability .

Solution: increase length of higher cycles, resulting in $\mathcal{O}(n^4)$ vertices.

Work in progress: improved construction.

Lower bound

Theorem (F.-Hansen-Zwick (2011))

The number of improving step performed by RANDOM-EDGE *on the MDPs (and LPs), which contain* $\mathcal{O}(n^4)$ *vertices and edges, is* $2^{\Omega(n)}$.

Random Facet and Least Entered

Outline

1. LPs and Simplex algorithm
2. MDPs and Policy iteration
3. Random Edge
4. Random Facet and Least Entered
   - Random Facet on LPs and MDPs
   - Random Facet construction
   - Least Entered
5. All is well that ends well?

Random Facet on LPs and MDPs

The algorithm

due to Kalai ('92), and Matoušek, Sharir and Welzl ('96)

procedure Random-Facet(H,B)
  if $H = B$ **then**
    return B
  else
    Choose $h \in H \setminus B$ at random
    $B' \leftarrow$ Random-Facet($H \setminus \{h\}, B$)
    if h is violated by B' **then**
      $B'' \leftarrow$ Basis($B' \cup \{h\}$)
      return Random-Facet(H, B'')
    else
      return B'
    end if
  end if
end procedure

Random Facet for LPs

procedure Random-Facet(G,σ)
  if $E_0 = \sigma$ **then**
    return σ
  else
    Choose $e \in E_0 \setminus \sigma$ at random
    $\sigma' \leftarrow$ Random-Facet($G \setminus \{e\}, \sigma$)
    if $\mathrm{val}_{\sigma'} \lhd \mathrm{val}_{\sigma'[e]}$ **then**
      $\sigma'' \leftarrow \sigma'[e]$
      return Random-Facet(G, σ'')
    else
      return σ'
    end if
  end if
end procedure

Random Facet for MDPs

Context

Known results

- Upper bound: $2^{\mathcal{O}(n)}$ (Kalai 1992)
- Abstract lower bound: $2^{\Omega(\sqrt{n})}$ (Matoušek 1994)

Theorem (F.-Hansen-Zwick (2011))

Random-Facet *for MDPs and LPs is subexponential.*

Random Facet construction

Randomized counting

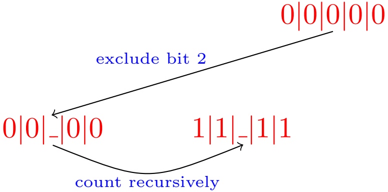
0|0|0|0|0

Randomized counting

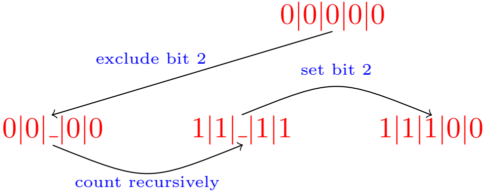
$$0|0|0|0|0$$

$$0|0|0|0|0 \xrightarrow{\text{exclude bit 2}} 0|0|\_|0|0$$

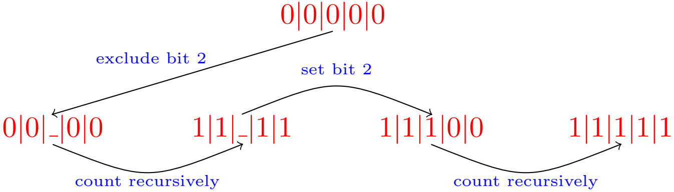
Randomized counting



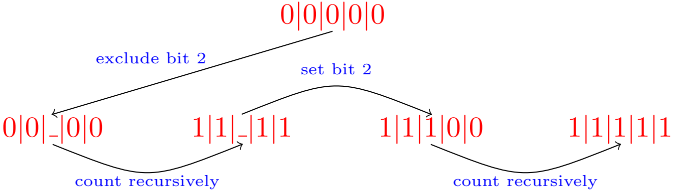
Randomized counting



Randomized counting

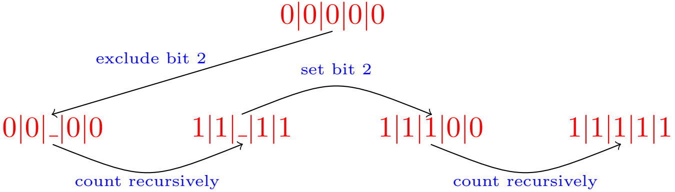


Randomized counting



Analysis

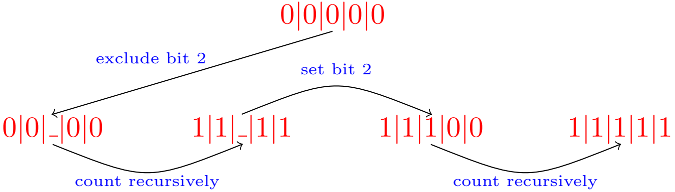
Randomized counting



Analysis

- Counting 0|0|_|0|0 equivalent to counting 0|0|0|0

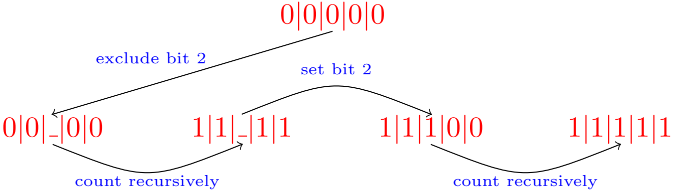
Randomized counting



Analysis

- Counting 0|0|_|0|0 equivalent to counting 0|0|0|0
- Counting 1|1|1|0|0 equivalent to counting 0|0

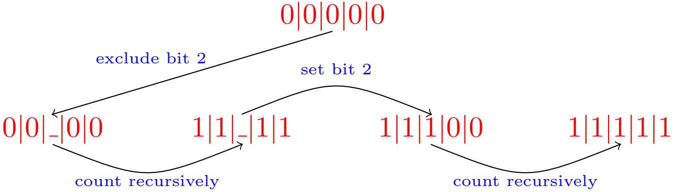
Randomized counting



Analysis

- Counting 0|0|_|0|0 equivalent to counting 0|0|0|0
- Counting 1|1|1|0|0 equivalent to counting 0|0
- Recurrence $f(n) = f(n-1) + \frac{1}{n}\sum_{i=0}^{n-1} f(i)$

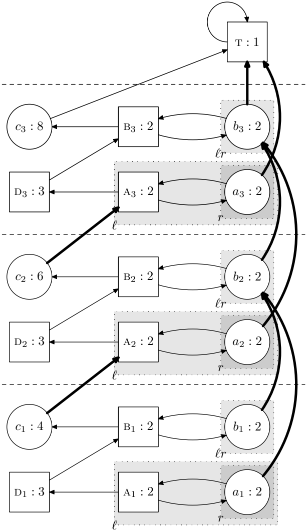
Randomized counting



Analysis

- Counting 0|0|_|0|0 equivalent to counting 0|0|0|0
- Counting 1|1|1|0|0 equivalent to counting 0|0
- Recurrence $f(n) = f(n-1) + \frac{1}{n}\sum_{i=0}^{n-1} f(i)$
- Complexity $f(n) \longrightarrow \frac{e^{2\cdot\sqrt{n}-\frac{1}{2}}}{2\cdot\sqrt{\pi}\cdot n^{\frac{1}{4}}}$ for $n \to \infty$

Simplified construction (for parity games)



Lower bound

Theorem (F.-Hansen-Zwick (2011))

The number of improving step performed by Random-Facet *on the MDPs (and LPs) is* $2^{\Omega(\sqrt{n}/\log(n))}$.

Least Entered

Zadeh's pivoting rule

Zadeh's Least-Entered rule

Perform single switch that has been applied least often.

Fair counting

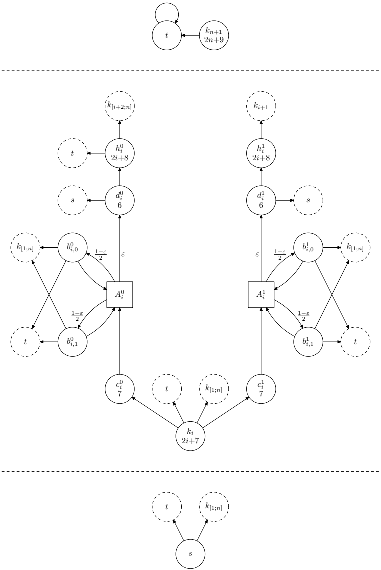
Problem:

- Flipping higher bits happens less often than flipping lower bits
- Zadeh's rule switches higher bits before they are supposed to be switched

Solution:

- Represent every bit by two representatives
- Only one representative is actively working
- Inactive representative switches back and forth to catch up with the rest
- Both representatives change roles after flipping the represented bit

Full construction



Lower bound

Theorem (Friedmann (2011))

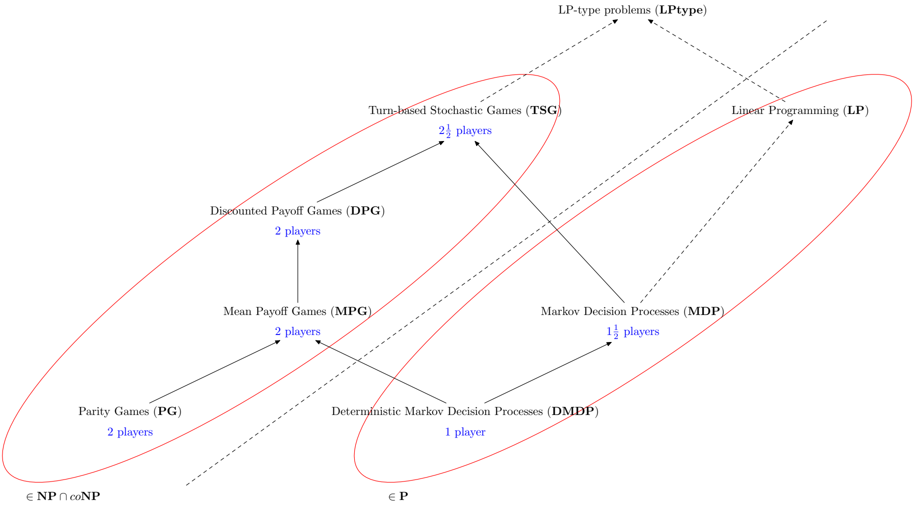
The number of improving step performed by Least-Entered *on the MDPs, which contain* $\mathcal{O}(n^2)$ *vertices and edges, is* $2^{\Omega(n)}$.

All is well that ends well?

Concluding remarks

- Game-theortic perspective helpful for the construction of lower bounds
- Lower bounds transfer to many other classes of determined games
- RANDOM-EDGE lower bound can be used as lower bound for SWITCH-HALF

Relation to other games



Open problems

- Polytime algorithm for two-player games and the like
- Strongly polytime algorithm for LPs (and MDPs)
- Resolving the Hirsch conjecture
- Find game-theoretic model with unresolved diameter bounds

The slide usually called “the end”.

Thank you for listening!