An Efficient Algorithm for Partial Order Production

Jean Cardinal
ULB/CS

Samuel Fiorini
ULB/Math

Gwenaël Joret
ULB/CS

Raphaël Jungers
UCL/INMA

Ian Munro
Waterloo/CS

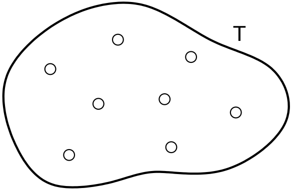
Sorting by Comparisons

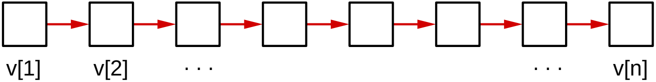
Input: a set T of size n, totally ordered by $\leqslant$

Goal: place the elements of T in a vector v in such a way that

$$v[1] \leqslant v[2] \leqslant \cdots \leqslant v[n]$$

after asking a min number of questions of the form "is $t \leqslant t'$?"





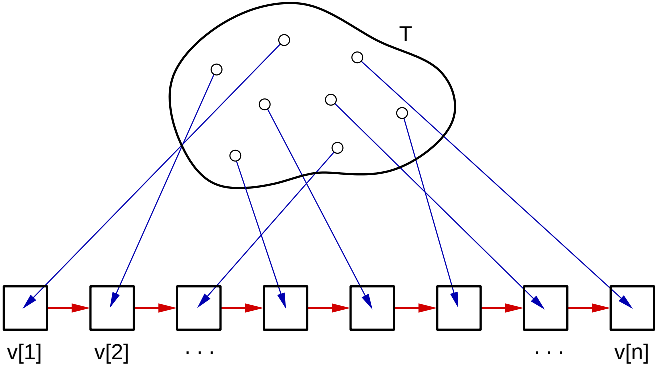
Sorting by Comparisons

Input: a set T of size n, totally ordered by $\leqslant$

Goal: place the elements of T in a vector v in such a way that

$$v[1] \leqslant v[2] \leqslant \cdots \leqslant v[n]$$

after asking a min number of questions of the form "is $t \leqslant t'$?"



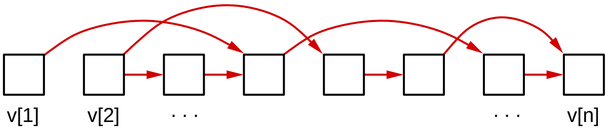
Partial Order Production ("Partial Sorting")

Input: a set T of size n, totally ordered by $\leqslant$
a partial order $\preccurlyeq$ on the set of positions $[n] := \{1, 2, \ldots, n\}$

Goal: place the elements of T in a vector v in such a way that

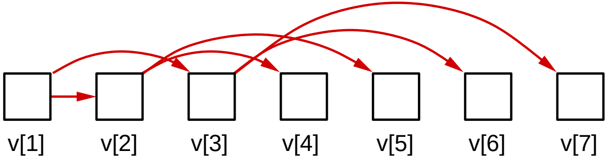
$$v[i] \leqslant v[j] \quad \text{whenever } i \preccurlyeq j$$

after asking a min number of questions of the form "is $t \leqslant t'$?"

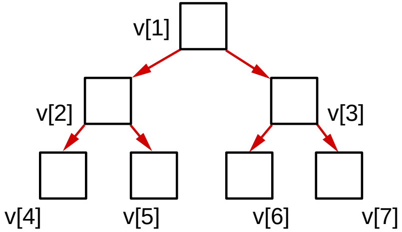


Particular Cases (1/2)

Heap Construction

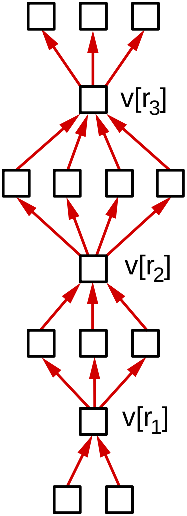


or



Particular Cases (2/2)

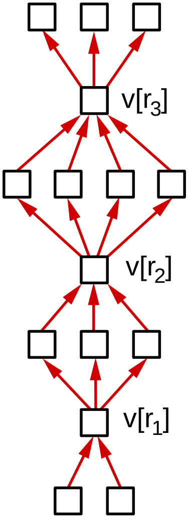
Multiple Selection



Find the elements of rank $r_1, r_2, \ldots, r_k$

Particular Cases (2/2)

Multiple Selection

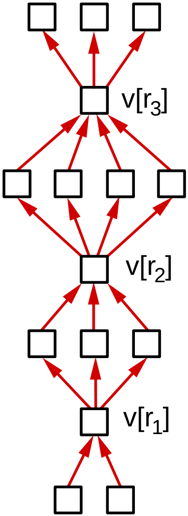


Find the elements of rank $r_1, r_2, \ldots, r_k$

Target poset $P := ([n], \preccurlyeq)$ is a weak order

Particular Cases (2/2)

Multiple Selection



Find the elements of rank $r_1, r_2, \ldots, r_k$

Target poset $P := ([n], \preccurlyeq)$ is a weak order

$\exists$ near-optimal algorithm (Kaligosi, Mehlhorn, Munro and Sanders, 05)

Worst Case Lower Bounds

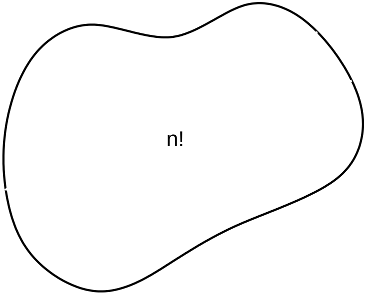
Well known fact. *For Sorting by Comparisons:*

$$\text{worst case \#comparisons} \geq \lg n!$$

Fact. (Schönage 76, Aigner 81) *For Partial Order Production:*

$$\text{worst case \#comparisons} \geq \underbrace{\lg n! - \lg e(P)}_{=:\, LB}$$

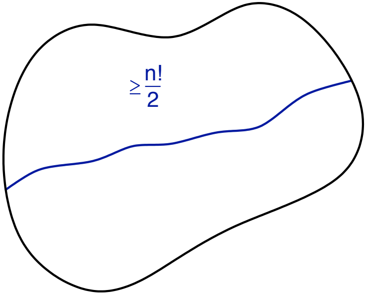
where $e(P) :=$ # linear extensions of P



Fact. (Schönage 76, Aigner 81) *For Partial Order Production:*

$$\text{worst case \#comparisons} \geq \underbrace{\lg n! - \lg e(P)}_{=: LB}$$

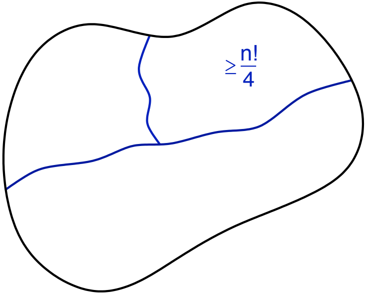
where $e(P) := \#$ linear extensions of P



Fact. (Schönage 76, Aigner 81) *For Partial Order Production:*

$$\text{worst case } \#\text{comparisons} \geq \underbrace{\lg n! - \lg e(P)}_{=:\, LB}$$

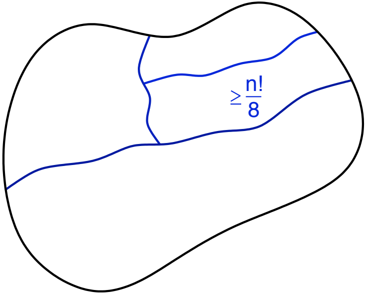
where $e(P) := \#$ linear extensions of P



Fact. (Schönage 76, Aigner 81) *For Partial Order Production:*

$$\text{worst case \#comparisons} \geq \underbrace{\lg n! - \lg e(P)}_{=:\, LB}$$

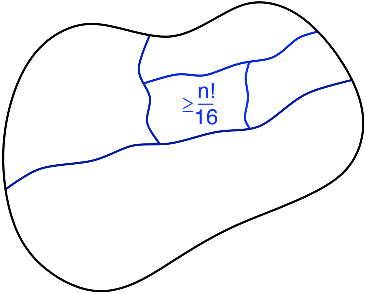
where $e(P) := \#$ linear extensions of P



Fact. (Schönage 76, Aigner 81) *For Partial Order Production:*

$$\text{worst case } \#\text{comparisons} \geq \underbrace{\lg n! - \lg e(P)}_{=:\, LB}$$

where $e(P) := \#$ linear extensions of P



$$|\text{leaf set}| \leq e(P) \Longrightarrow \#\text{comparisons} \geq \lg \frac{n!}{e(P)} = LB$$

Problem History

1976 Schönage defined POP problem

Problem History

1976 Schönage defined POP problem

1981 Aigner studied POP problem

Problem History

1976 Schönage defined POP problem

1981 Aigner studied POP problem

1985 two surveys: Bollobás & Hell, and Saks.
Saks conjectured that $\exists$ algorithm for POP problem
s.t. worst case #comparisons $= O(LB) + O(n)$

Problem History

1976 Schönage defined POP problem

1981 Aigner studied POP problem

1985 two surveys: Bollobás & Hell, and Saks.
Saks conjectured that $\exists$ algorithm for POP problem
s.t. worst case #comparisons $= O(LB) + O(n)$

1989 Yao solved Saks' conjecture, stated open problems

Our Result

There exists a $O(n^3)$ algorithm for the POP problem s.t.

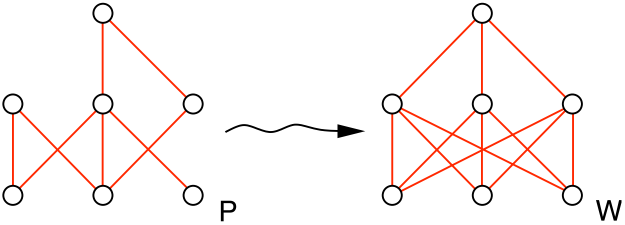
$$\text{worst case } \#\text{comparisons} = LB + o(LB) + O(n)$$

Improvements over Yao's algorithm:

- overall complexity is polynomial
- smaller number of comparisons

A Simple Plan

1. Extend the target poset P to a weak order W
2. Solve the problem for W using Multiple Selection algorithm



Key Tool: the Entropy of a Graph

The entropy of $G = (V, E)$ equals:

$$H(G) := \min_{x \in STAB(G)} -\frac{1}{n} \sum_{v \in V} \lg x_v$$

where $STAB(G) :=$ stable set polytope of G

Key Tool: the Entropy of a Graph

The entropy of $G = (V, E)$ equals:

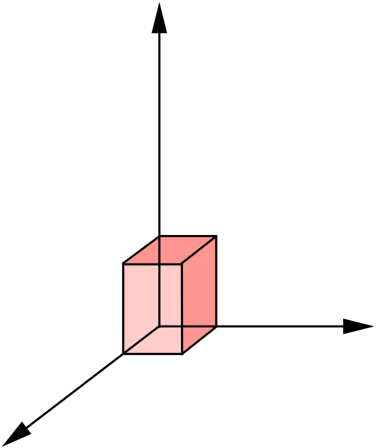
$$H(G) := \min_{x \in STAB(G)} -\frac{1}{n} \sum_{v \in V} \lg x_v$$

where $STAB(G) :=$ stable set polytope of G

- Introduced in information theory by J. Körner (73)
- Graph invariant with lots of applications (mostly in TCS)
  - bounds for perfect hashing
  - circuit lower bounds for monotone Boolean functions
  - sorting under partial information (Kahn and Kim 95)
  - ...

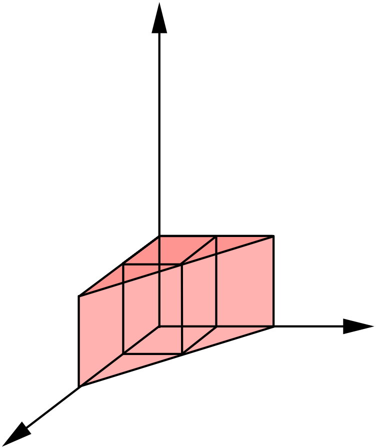
Lemma. (Kahn and Kim 95)

$$\underbrace{-n\,H(G)}_{=\lg \mathrm{V}ol(Box)} \leq \lg \mathrm{Vol}(STAB(G)) \leq \underbrace{n \lg n - \lg n! - n\,H(G)}_{=\lg \mathrm{V}ol(Simplex)}$$



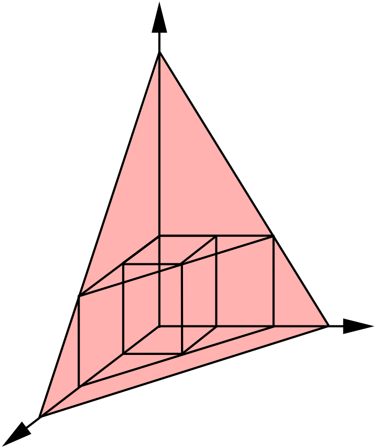
Lemma. (Kahn and Kim 95)

$$\underbrace{-n\,H(G)}_{=\lg Vol(Box)} \le \lg \mathrm{Vol}(STAB(G)) \le \underbrace{n \lg n - \lg n! - n\,H(G)}_{=\lg Vol(Simplex)}$$



Lemma. (Kahn and Kim 95)

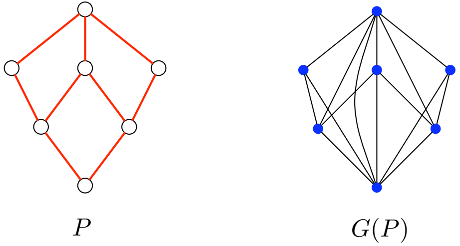
$$\underbrace{-n\,H(G)}_{=\lg Vol(Box)} \leq \lg \mathrm{Vol}(STAB(G)) \leq \underbrace{n \lg n - \lg n! - n\,H(G)}_{=\lg Vol(Simplex)}$$



Comparability Graphs and Entropy

$G(P) :=$ comparability graph of target poset P

$H(P) := H(G(P))$



Lemma. (Stanley 86) $\mathrm{Vol}\Big(STAB\big(G(P)\big)\Big) = \dfrac{e(P)}{n!}$

Corollary. $n\,H(P) - n \lg e \leq LB \leq n\,H(P)$

Weak Order Extensions $\rightarrow$ Colorings

Observation.

Every weak order extension W of P gives a coloring of $G(P)$

$$\Downarrow$$

Want: "good" coloring of $G(P)$

$$\begin{aligned} W \text{ extends } P \quad &\Longrightarrow \quad STAB\big(G(P)\big) \supseteq STAB\big(G(W)\big) \\ &\Longrightarrow \quad H(P) \leq H(W) \end{aligned}$$

Intuition.

$H(W)$ should be as small as possible

$$\Downarrow$$

The class sizes should be distributed as unevenly as possible

Greedy Colorings and Greedy Points

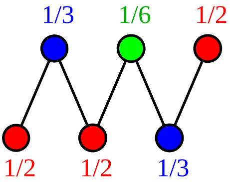
For G = perfect graphs

Iteratively remove a maximum stable set from G

$\rightsquigarrow$ sequence $S_1, S_2, \ldots, S_k$ of stable sets

- Gives greedy coloring (k colors, ith color class = S_i)
- Also gives greedy point:

$$\tilde{x} := \sum_{i=1}^{k} \frac{|S_i|}{n} \cdot \chi^{S_i} \quad \in \quad STAB(G)$$



Theorem. *Let G be a perfect graph on n vertices and denote by $\tilde{g}$ the entropy of an arbitrary greedy point $\tilde{x} \in STAB(G)$. Then*

$$\tilde{g} \leq \frac{1}{1-\delta}\left(H(G) + \lg \frac{1}{\delta}\right)$$

for all $\delta > 0$, and in particular

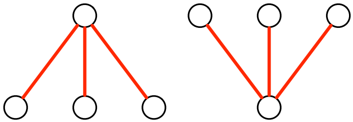
$$\tilde{g} \leq H(G) + \lg H(G) + O(1).$$

Proof idea. Dual fitting, using min-max relation

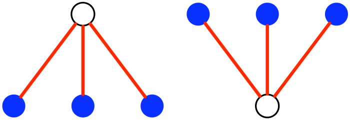
$$H(G) + H(\bar{G}) = \lg n$$

due to Csiszár, Körner, Lovász, Marton and Simonyi (90) $\square$

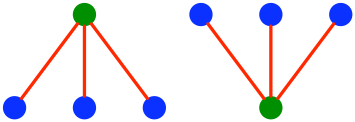
Colorings $\not\to$ Weak Order Extensions



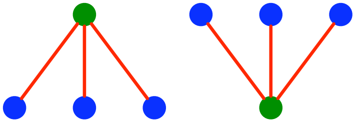
Colorings $\not\to$ Weak Order Extensions



Colorings $\not\to$ Weak Order Extensions



Colorings $\not\to$ Weak Order Extensions



Weak order extensions of $P \to$ colorings of $G(P)$

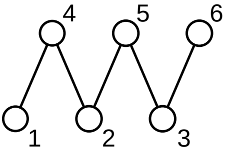
$\not\leftarrow$

$\Longrightarrow$ need to "uncross" our greedy colorings

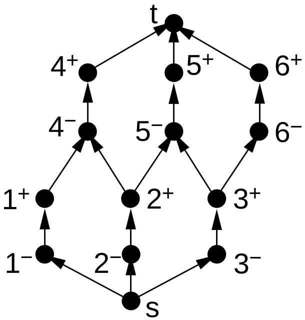
Uncrossing a Greedy Coloring

$D = D(P) :=$ auxiliary network with source s, sink t

$D = (N(D), A(D))$



$\rightsquigarrow$

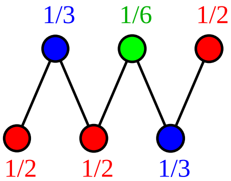


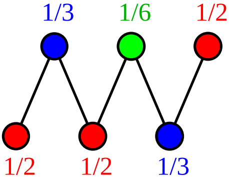
$$
\begin{array}{llrcll}
\text{(H-potential)} & \min & \multicolumn{4}{l}{-\dfrac{1}{n}\displaystyle\sum_{v \in V} \lg x_v} \\
 & \text{s.t.} & x_v & = & y_{v^+} - y_{v^-} & \forall v \in V \\
 & & y_a & \leqslant & y_b & \forall (a,b) \in A(D) \\
 & & y_s & = & 0 & \\
 & & y_t & = & 1 &
\end{array}
$$

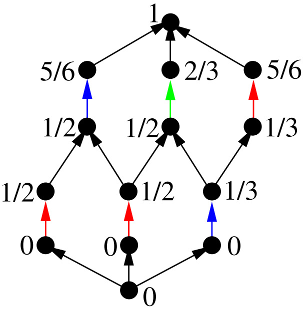
Find potential $\tilde{y}$ for greedy point $\tilde{x}$ (by DP)

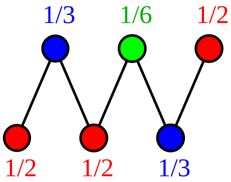
We get:

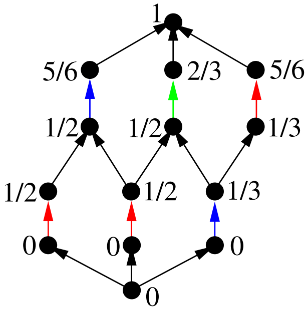
- collection of open intervals $\{\, (\tilde{y}_{v^-}, \tilde{y}_{v^+}) \,\}_{v \in V}$
- interval order I extending P, with $H(I)$ close to $H(P)$

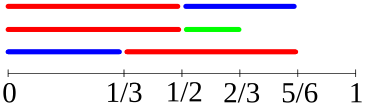

1/3
1/6
1/2
1/2
1/2
1/3

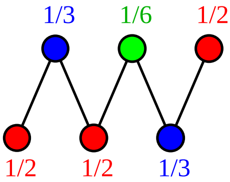

1/3
1/6
1/2
1/2
1/2
1/3

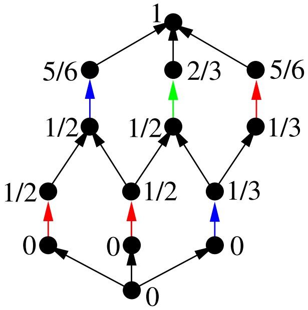

1
5/6
2/3
5/6
1/2
1/2
1/3
1/2
1/2
1/3
0
0
0
0

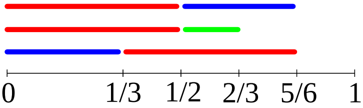

1/3
1/6
1/2
1/2
1/2
1/3

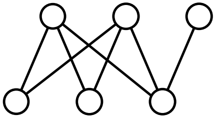

1
5/6
2/3
5/6
1/2
1/2
1/3
1/2
1/2
1/3
0
0
0
0


0
1/3
1/2
2/3
5/6
1


1/3 1/6 1/2
1/2 1/2 1/3


1
5/6 2/3 5/6
1/2 1/2 1/3
1/2 1/2 1/3
0 0 0
0


0 1/3 1/2 2/3 5/6 1



Main Steps of our Algorithm

1. $P \overset{greedy+DP}{\hookrightarrow} I$
2. $I \overset{greedy}{\hookrightarrow} W$
3. Use Multiple Selection algorithm of Kaligosi *et al.* on W

Theorem. *The algorithm above solves the POP problem, in $O(n^3)$ time, after performing at most*

$$LB + o(LB) + O(n)$$

comparisons

Further Result & Open Questions

Tightness result:

- Any algorithm reducing the POP problem to Multiple Selection can be forced to perform

$$LB + \Omega(n \lg \lg n)$$

  comparisons for some P with $H(P) \approx \frac{1}{2} \lg n$

Open questions:

- Is there an algorithm performing $LB + O(n)$ comparisons?
- What about Partial Order Production under Partial Information?

Thank You!

P.S.: The paper is available on ArXiv