Fixed Parameter Algorithms for Kemeny Scores

Frances Rosamond
The University of Newcastle, Au

Joint work with

Nadja Betzler, Jiong Guo, and Rolf Niedermeier. Friedrich-Schiller University, Jena, Germany. Michael Fellows, Univ. of Newcastle, AU.

Parameterized Complexity Workshop

University of Newcastle

30 and 31 March, 2010

Outline

Motivation

The problem

Parameterized complexity and where to find information

Previous results

Parameterized by the Kemeny Score

Parameterized by the Maximum Kendall-Tau Distance

Parmeterized by the Average Kendall-Tau Distance

Summary and Open Problems

The deBorda Experiment

- UK groups: de Borda Institute and New Economics Foundation
- Modified Borda Count-consensus voting
- Internet participation: political science, social science, everyone
- www.opendemocracy.net/deborda

Borda Count: give least favoured candidate one point,, next best two points, …most favoured candidate m points. Candidate elected may not be anyone’s first choice!

Australia uses preferential voting for almost all elections.

Ranked ballot methods

- Ranked ballot methods allow voters to list the candidates in order of preference: first choice, second choice, and so on.

- In Australia, voters are required to rank all of the candidates in order for their ballot to be counted. The effort to reform election abuses led to the widespread use of the Australian ballot, which was adopted in Victoria in 1857, in Great Britain in 1872, and grew increasingly popular in the United States after 1888. The Australian ballot is now used in many European countries and in almost all sections of the United States.

- In US, it gradually replaced earlier methods of voting such as lengthy "tickets" distributed by political parties. In the Australian system all candidates' names are printed on a single ballot and placed in the polling places at public expense, and the printing, distribution, and marking of the ballot are protected by law, thus assuring a secret vote.

electionbuddy.com

A Complete Online Election in 3 Steps

- 1 Create Name your election: Choose a start date/Choose an election type /Design your ballot: add positions, candidates and questions.

- Three types of elections: Preferential Voting (Instant Runoff Voting), First Past the Post or Plurality Voting, and Approval Voting.

  2 Vote Unique keys are generated for each voter and sent out by email

  Each voter casts their ballot anonymously.

- 3 Watch Watch in real-time as your voters cast their ballots and your election results are tallied. Allow voters to see election results by turning on the 'Public URL' option.

LESSON: People are interested. They want to know about voting.

youtube,

[You Can *Vote* However You Like]
3 min 56 sec - 24 Oct 2008
www.youtube.com

- [Homer Simpson tries to *vote* for Obama]
  1 min 21 sec - 29 Sep 2008
  www.youtube.com

LESSON: A lot of interesting voting is going on.

Can vote over just about anything: political representatives, award nominees, dinner tonight, allocations of tasks/resources, …

- MTV Movie Awards 2010 Nominees' *Voting* Now Open! - 9 hours ago. Well now you can *vote* for your favorite movie moments — starting today (March 29) — for the 2010 MTV Movie Awards, which airs live on June 6 from Los **...**

- WALL STREET JOURNAL, The Australian, 16 Feb.

Delhi government cites a need to "build consensus" t[o] ... cultivation of genetically modified eggplant.

LESSON: People are using voting vocabulary "build consensus".

Aggregate Paradox

100 voters: Basketball court or a Gym

Basketball Court Yes but No gym	Basketball Court No Yes gym	Both Yes
49	49	2

Aggregate Decisions Paradox

100 voters: Basketball court or a Gym

Basketball Court Yes but No gym	Basketball Court No Yes gym	Both Yes
49	49	2

Court Yes	Court No	Gym Yes	Gym No
50	49	50	49

Voting separately on each issue gives outcome Yes to both Basketball Court and Gym, even though this option received only 2 votes out of 100.

Manipulation by a voter

- Voter 1 a > b > c
- Voter 2 a > b > c
- Voter 3 b > a > c
- Voter 4 b > a > c
- Voter 5 c > a > b

Manipulation by a voter

- Voter 1 a > b > c
- Voter 2 a > b > c
- Voter 3 b > a > c
- Voter 4 b > a > c
- Voter 5 c > a > b

Voter 5's true vote has no chance.

Better for Voter 5 to vote insincerely and at least get second choice to win.

Voter 5 should change vote to

a > c > b

Many Election Sysems

- Majority
- Condorcet
- Kemeny
- Dodgson
- Young
- Copeland
- Many more

Election

Set of votes V, set of candidates C.

A vote is a ranking (total order) over all candidates.

Example: $C = \{a, b, c\}$

vote 1:	a	>	b	>	c
vote 2:	a	>	c	>	b
vote 3:	b	>	c	>	a

How to aggregate the votes into a "consensus ranking"?

What do we want in a concensus?

A ranking "close" to the votes.

Marquis de Condorcet (Marie Jean Antoine Nicolas Caritat). The Condorcet criterion was discovered independently Ramon Llull in 1299.

Condorcet (1785): Find the candidate who wins in a head to head election against every other candidate individually. The election breaks into a series of **pairwise comparisons** between every candidate and every other candidate.

Kendall-Tau distance between two votes

KT-distance (between two votes v and w)

$$\text{dist}(v, w) = \sum_{\{c,d\} \subseteq C} d_{v,w}(c, d),$$

where $d_{v,w}(c, d)$ is 0 if v and w rank c and d in the same order, else 1.

Find a distance measure between the input objects:

Kendall-Tau distance (KT-dist) counts the number of pairwise disagreements between two rankings.

“Bubble-sort Metric”

Number of inversions

Kendall-Tau distance between two votes

Votes V	**Candidates** C
Voter1:	**a > b > c**
Voter2:	**a > c > b**
Voter3:	**b > c > a**

The KT-distance between vote 1 and vote 2 is 1.

The KT-distance between vote 1 and vote 3 is 2.

Compare pairwise rankings ab, ac, bc.

	ab		**ac**		**bc**		
Vote 1&2:	0	+	0	+	1	=	1
Vote 1&3:	1	+	1	+	0	=	2
Vote 2&3:	1	+	1	+	1	=	3

Kemeny Score for an order

Votes V	**Candidates** C
Voter1:	**a > b > c**
Voter2:	**a > c > b**
Voter3:	**b > c > a**

Order	**a > b > c**			
	ab	**ac**	**bc**	
V1:	0 +	0 +	0	= 0
V2:	0 +	0 +	1	= 1
V3:	1 +	1 +	0	= 2

Kemeny Score = 3

Order	**a > c > b**			
	ab	**ac**	**bc**	
V1:	0 +	0 +	1	= 1
V2:	0 +	0 +	0	= 0
V3:	1 +	1 +	1	= 3

Kemeny Score = 4

Order	**b > a > c**			
	ab	**ac**	**bc**	
V1:	1 +	0 +	0	= 1
V2:	1 +	0 +	1	= 2
V3:	0 +	1 +	1	= 2

Kemeny Score = 5

The **Kemeny Score** for an order is the sum of KT-distances between that order and all votes.

Kemeny Consensus: an order that minimizes the Kemeny Score.

All possible orders for abc:

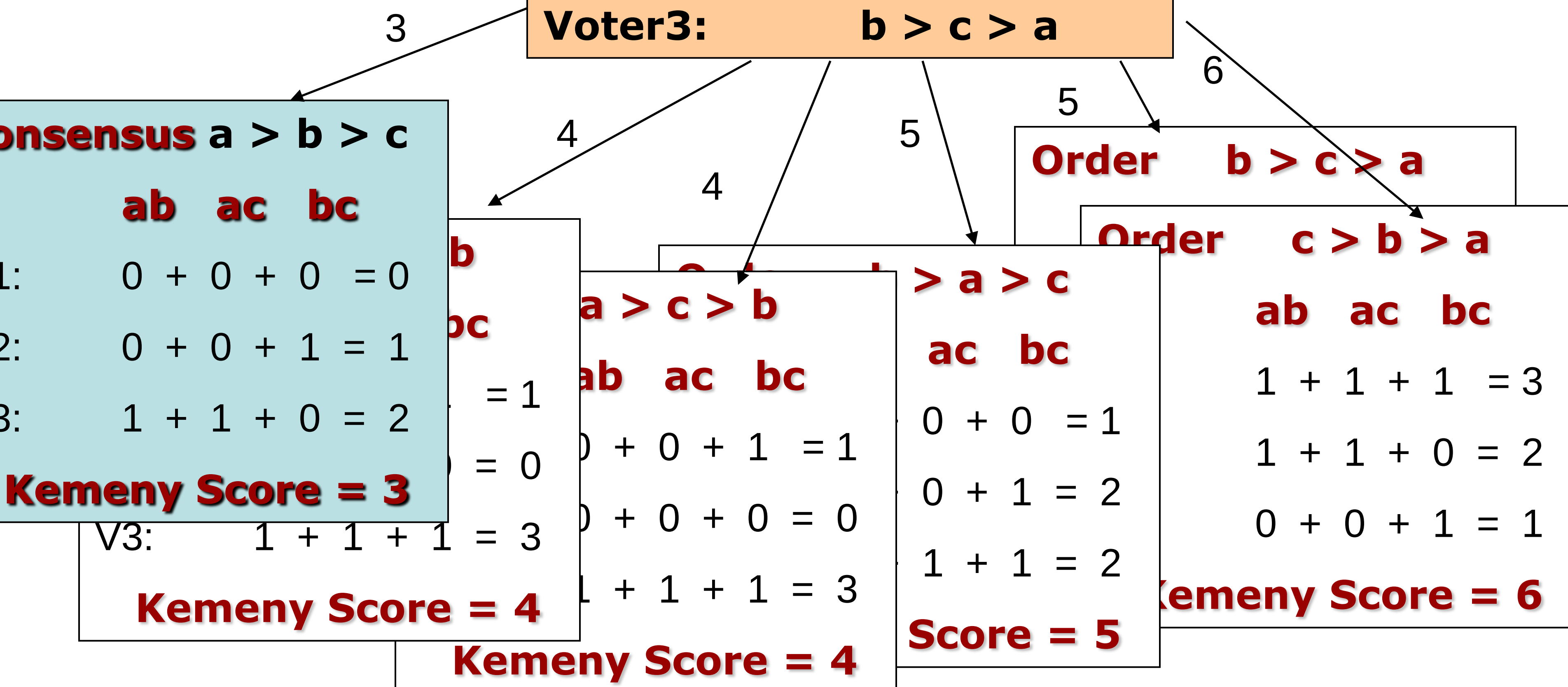
abc, acb, bac, bca, cab, cba

Consider the KT-distances between all votes and all possible orders.

Voter1: a > b > c

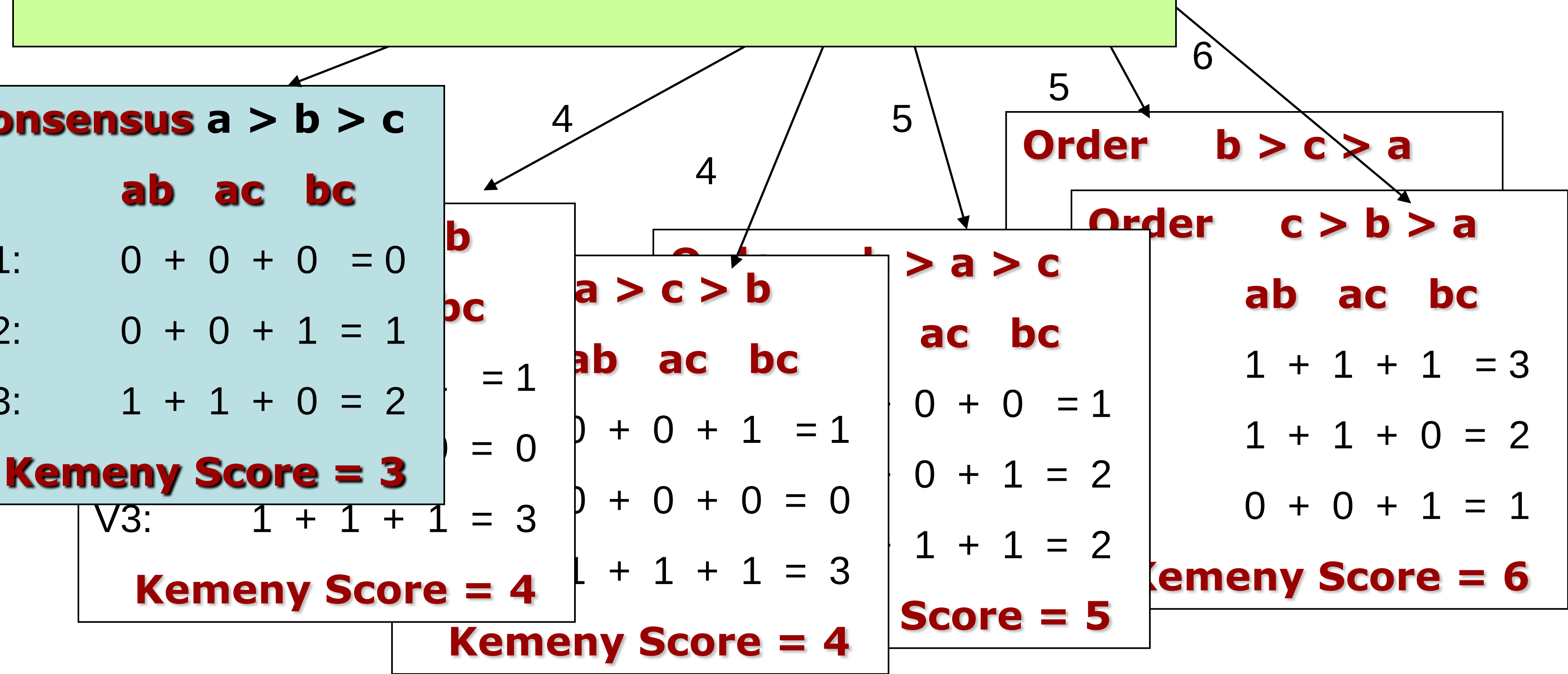
Voter2: a > c > b

Voter3: b > c > a



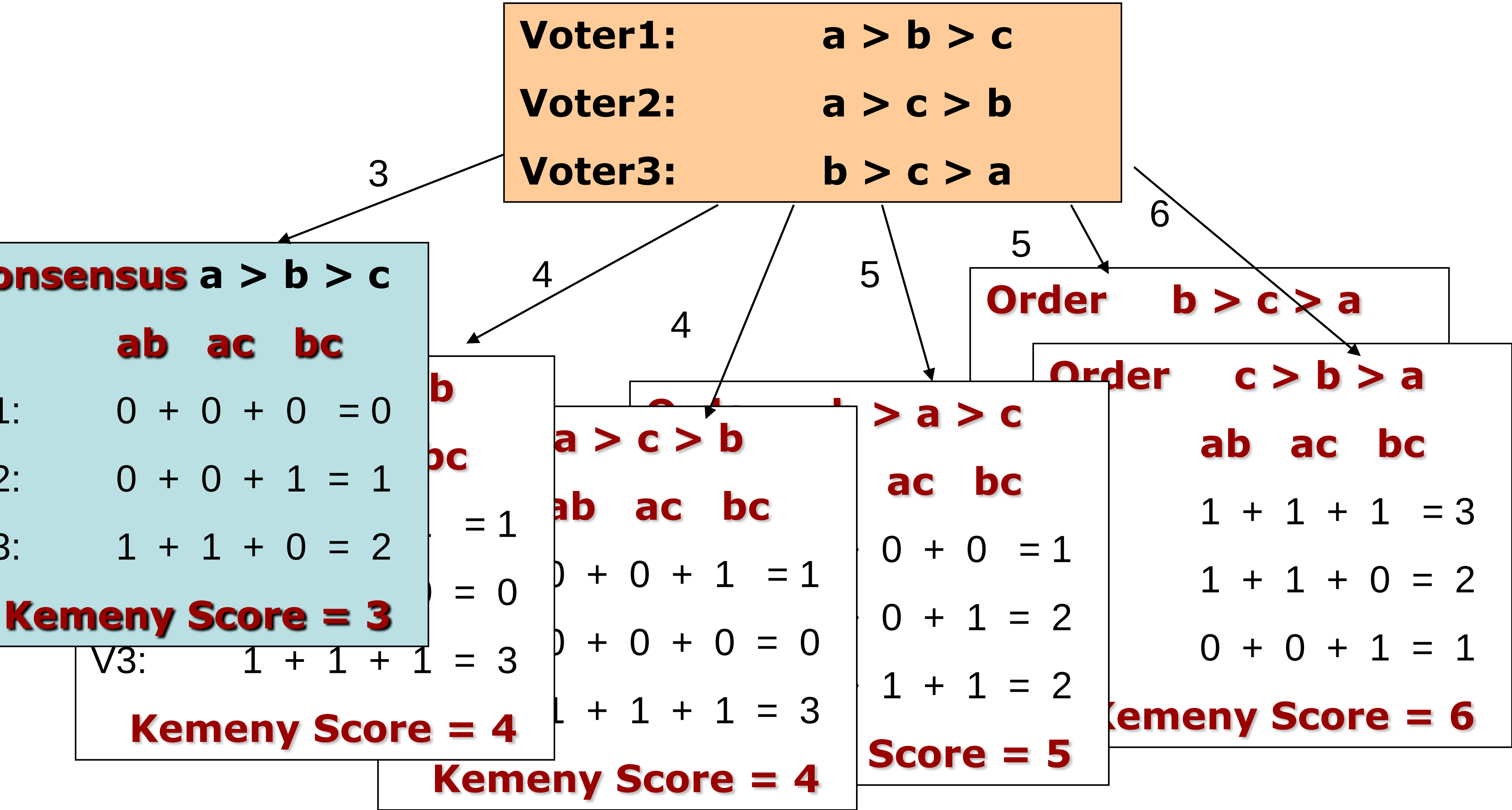
Kemeny Consensus: order that minimizes the score

Lemma [Truchon, Technical Report 1998]
Let a and b be two candidates. If a > b in all votes,then every Kemeny consensus has a > b.



Kemeny Winner: candidate that is ranked first in a Kemeny Consensus

Kemeny Winner: In our example, candidate *a* is Kemeny winner.



John Kemény

John Kemény was born in Budapest, escaped with parents, attended Princeton but left to work at Los Alamos (with Richard Feynman and John von Neumann) during the war. He returned to Princeton, graduated with his BA in 1947, then worked for his doctorate under Alonzo Church. He worked as Einstein's mathematical assistant during graduate school. Kemeny doctorate in 1949, dissertation entitled "Type-Theory vs. Set-Theory.

Kemeny became President of Dartmouth, which became coed (after 203 years of single-sex education). While president, he taught two courses a year, never missing a class.

He ,with Thomas Kurtz, invented BASIC. Kemeny made Dartmouth a pioneer in student use of computers, equating computer literacy with reading literacy.

American colleges and universities experienced a tumultuous period of student protest, Dartmouth enjoyed relative calm due in large part to Kemeny's appeal to students and his practice **of seeking consensus** on vital college issues.

He invented the Kemeny Election scheme around 1958.

Decision Problem

KEMENY SCORE

Input: An election (V, C) and a positive integer k.

Parameter: k.

Question: Is the Kemeny Score of (V, C) at most k?

KEMENY WINNER

Input: An election (V, C) a distinguished candidate c.

Question: Is there a Kemeny consensus in which c wins?

Looking for a consensus list that minimizes the sum of the distances to the given votes.

Results **1** - **10** of about **28,500** for **fpt complexity**. (0.2

search : fpt complexity

Search Results

1. [Parameterized *complexity* - Wikipedia, the free encyclopedia]

The corresponding ***complexity*** **class is called *FPT*. For example, there is an algorithm which solves the vertex cover problem in O(kn + 1.274k) time, ...**

***en.wikipedia.org/wiki/Parameterized_complexity** - 23k - Cached - Similar pages -*

2. [Schloss Dagstuhl : Seminar Homepage]

Parameterized ***complexity*** **is a new and promising approach to the central issue of how ... (called *FPT* complexity). ...**

***www.dagstuhl.de/de/programm/kalender/semhp/?semnr=01311** - 16k - Cached - Similar pages*

3. [*Complexity* Zoo:F - Qwiki]

Lists of related classes: Communication ***Complexity*** **- Hierarchies - Nonuniform .... Same as *FPT* except that the algorithm can vary with the parameter k ...**

***qwiki.stanford.edu/wiki/Complexity_Zoo:F** - 39k - Cached - Similar pages -*

4. [Parameterized *Complexity*: an Overview]

File Format: **PDF/Adobe Acrobat** - View as HTML

5 Feb 2007 ... Note: If the number of states in M is bounded by a constant, the. problem is in *FPT*. Parameterized *Complexity* - an **Overview 21 / 30 ...**

www.ens-lyon.fr/LIP/MC2/files/Uffe_Flarup.pdf

5. [PhD Proposal Parameterized *complexity* - reduction ...]

File Format: **PDF/Adobe Acrobat** - View

Of course, to obtain such a *complexity*, the choice of the parameter is ... theoretical point of view, ...

***www.lirmm.fr/~paul/sujet-FPT.pdf** - S...*

6. [+ + "]

extended '.he ***FPT*** **concept with the Chinese remainder theorem. The gen... reduce the. *FPT* complexity ...**

***ieeexplore.ieee.org/iel6/29/26194/01164749.pdf?arnumber** ...*

by SC Pei - 1985 - Cited by 3 - Related a...

1 - 10 of 57,400 for

1. [Parameterized complexity - W...]

... class **FPT**, and the early name of the theory of paramete... corresponding **complexity** class is called **FPT**. **...en.wi...**

2. [Parameterized Proof Complexity: a Co...]

187k - Adobe PD...

... treatment of parameterized **complexity** classes and **fpt**-redu... hierarchy are distinct from **FPT**. **...eccc.hpi-web...**

3. [Descriptive Complexity...]

285k - Adobe PDF - **View as html** **FPT** stands for the class of fixed...

... a third support for parameterized c**omplexit**theory**..**w...

4. [Genus Characterizes the Complexity ...]

242k - Adobe PD...

Denote by FPT the class of all fixed. parameter tractable proble... First note that by our assumption on **...facweb.cs.**...

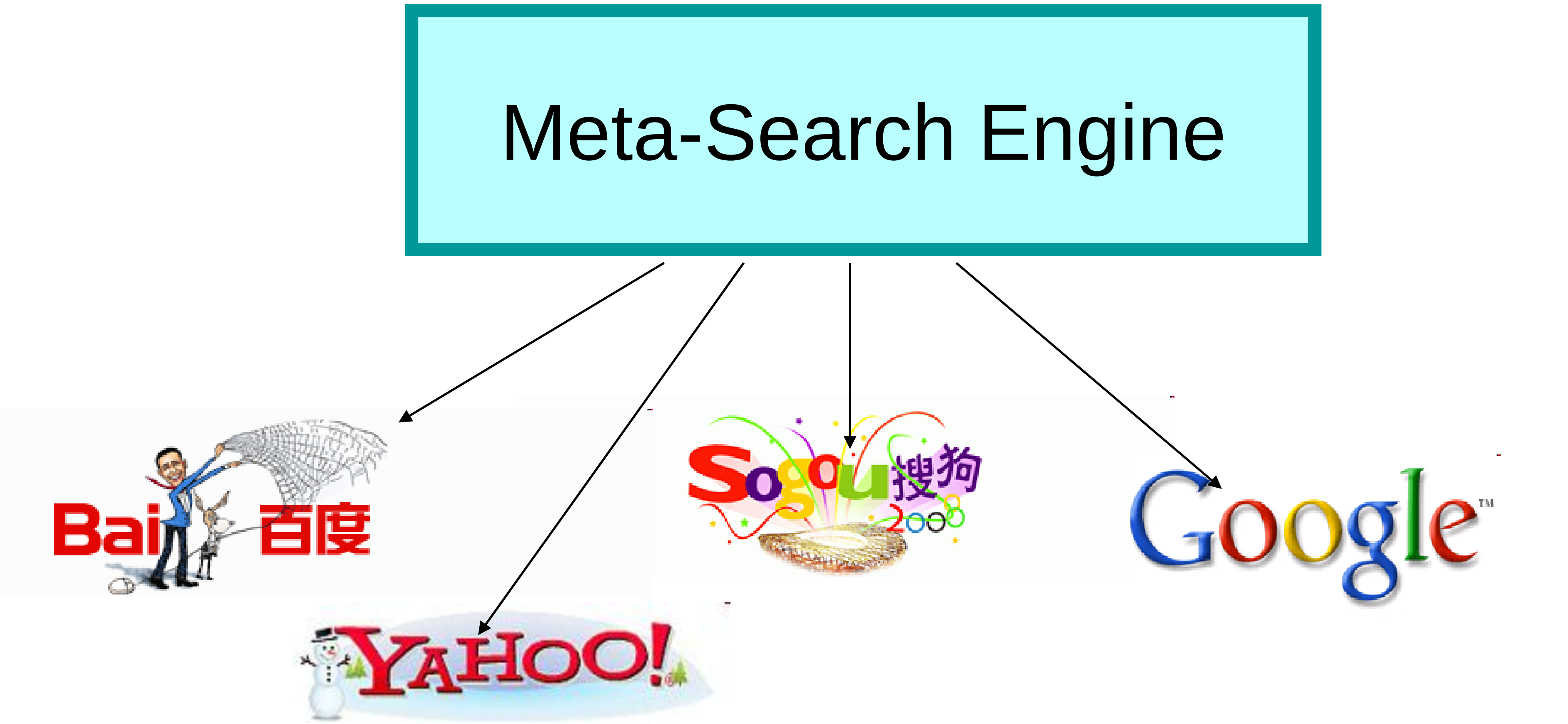
... class FPT ...

Researc...

Paramete...

How do search engines RANK 57,000 web pages?

Which are first 6...



How to make an aggregrate ranking--“concensus”?

Voters = Internet search engines. Candidates = webpages

Search engines: few voters

Web pages:

Huge number of "candidates"

Rank aggregation methods for the web

C. Dwork, R. Kumar, M. Naor, D. Sivakumar

Voters = Internet search engines. Candidates = webpages

Search engines: few voters

Web pages:

Huge number of “candidates”

Method of Kemeny minimizes the total disagreement between several input rankings and their aggregation. Unfortunately, computing optimal solutions based on Kemeny is NP-hard, even for only 4 rankings.

Known results

- KEMENY SCORE is NP-complete (even for 4 votes)
  [DWORK & AL. WWW 2001]
- KEMENY WINNER is $P^{NP}_{||}$-complete
  [HEMASPAANDRA & AL. TCS 2005]

Algorithms:

- randomized factor 11/7-approximation [AILON & AL. STOC 2005]
- factor 8/5-approximation [VAN ZUYLEN & WILLIAMSON WAOA 2007]
- PTAS [KENYON-MATHIEU & SCHUDY STOC 2007]
- greedy, branch and bound
  [CONITZER & AL. AAAI 2006],[DAVENPORT & KALAGNANAM AAAI 2004]

Import ideas from CS to solve questions from social choice

Can we design a voting protocol that makes it impossible for a voter to cheat?

Computer Science: Design and analysis of algorithms

Maybe not, but the problem may be intractable, and therefore an acceptable risk.

OK, we can take the risk

These results

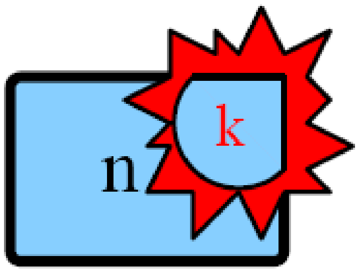
- Nadja Betzler, Mike Fellows, Jiong Guo, Rolf Niedermeier, FR. Fixed-parameter algorithms for Kemeny rankings. *Theoretical Computer Science*, 410 (45): 4554-4570, 2009.

- New results:
  - N. Simjour. Improved parameterized algorithms for the Kemeny Aggregation problem. In Proc. 4th IWPEC, v 5917 of LNCS, pp 312-323, 2009.)
  - Nadja Betzler, Jiong Guo, Christian Komusiewicz, and Rolf Niedermeier. Average Parameterization and Partial Kernelization for Computing Medians. *Proceedings of 9th LATIN, 2010*.

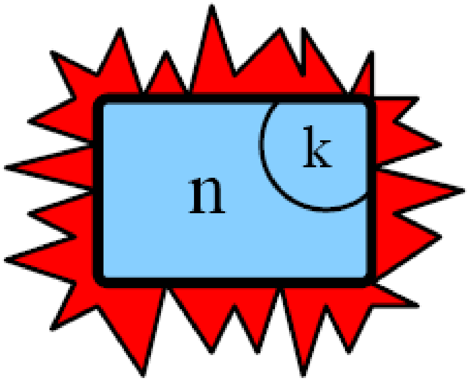
Parameterized Complexity

Given a NP-hard problem with input size n and a parameter k

Basic idea: Confine the combinatorial explosion to k



instead of



Definition

A problem of size n is called *fixed parameter tractable* with respect to a parameter k if it can be solved in $f(k) \cdot n^{O(1)}$ time.

Or additively, in $O(f(k) + n^c)$ time.

Parameterized Complexity

$$O(f(k)n^c)$$

$$O(f(k) + n^c)$$

- Database query size tends to be much smaller than the size of the entire database.
- The number of candidates in an election may be much smaller than the number of voters.
- Number of web pages is large compared to the number of search engines.
- Parameters recognize different size magnitudes.

Parameterizations of KEMENY SCORE

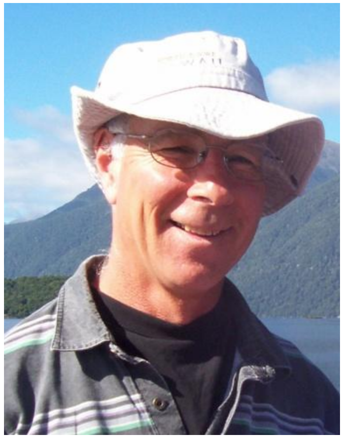
1. *Parameter "the Kemeny Score"*. Solvable in $O(1.53^k + m^2 n)$ time, where k denotes the Kemeny Score of the given election.

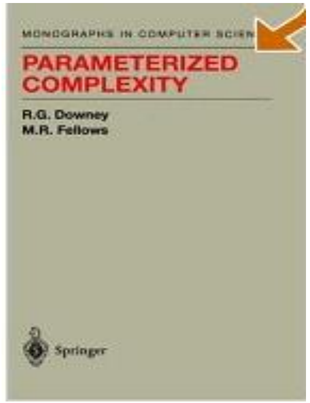
2. *Parameter "the maximum KT-distance between any two input votes".* Solvable in $O((3d + 1)!\ d\ (\log d)\ m\ n)$ time.

3. *Parameter "the number of candidates".* Solvable in $O(2^m m^2 n)$ time.

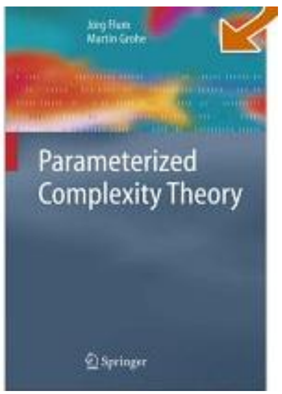
4. *Parameter "the maximum range of candidate positions".* Solvable in $O^*(32^r)$ time.

5. *Parameter "the average KT-distance".* Solvable in $O^*(16^{aveKT})$ time.

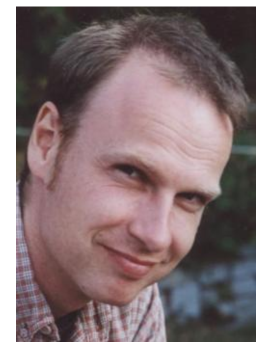




Downey-Fellows: Parameterized Complexity, Springer, 1999



Flum-Grohe: Parameterized Complexity Theory, Springer, 2006.



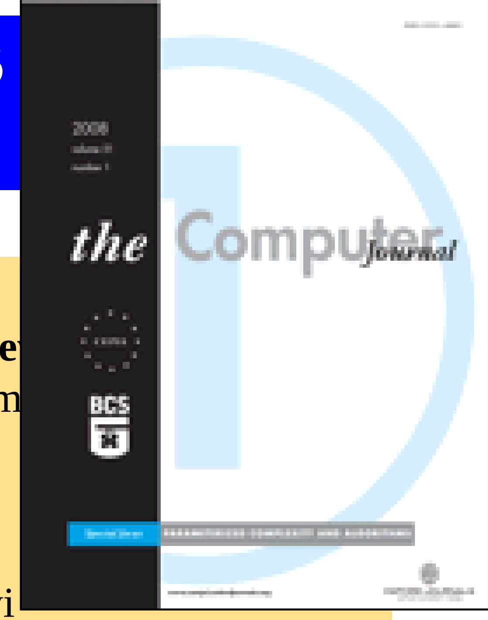
Niedermeier: Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.

the Computer *journal*

volume 51, number 1
January 2008

the Computer *journal*

volume 51, number 3
May 2008



Hans L. Bodlaender and Arie M. C. A. Koster
Combinatorial Optimization on Graphs of Bounded Tree

Liming Cai, Xiuzhen Huang, Chunmei Liu, Frances Rosam
and Yinglei Song
Parameterized Complexity and Biopolymer Sequence Comparison

Erik D. Demaine and MohammadTaghi Hajiaghayi
The Bidimensionality Theory and Its Algorithmic Applications

Georg Gottlob and Stefan Szeider
Fixed-Parameter Algorithms For Artificial Intelligence, Constraint Satisfaction and Database Problems

Petr Hlineny, Sang-il Oum, Detlef Seese, and Georg Gottlob
Width Parameters Beyond Tree-width and their Applications

Gregory Gutin and Anders Yeo
Some Parameterized Problems On Digraphs

Panos Giannopoulos, Christian Knauer, and Sue Whitesides
Parameterized Complexity of Geometric Problems

Iris van Rooij and Todd Wareham
Parameterized Complexity in Cognitive Modeling: Foundations, Applications and Opportunities

FPT NEWS, THE PARAMETERIZED COMPLEXITY NEWSLETTER

email: Frances.Rosamond@newcastle.edu.au
if you would like to receive the Newsletter.

The Parameterized Complexity WIKI is located at

http://www.fpt.wikidot.com

Problem	$f(k)$	kernel	Ref
Vertex Cover	1.2738^k	$2k$	1
Feedback Vertex Set	5^k	k^3	2
Planar DS	$2^{15.13\sqrt{k}}$	$67k$	3
1-Sided Crossing Min	1.4656^k		4
Max Leaf	6.75^k	$4k$	5
Directed Max Leaf	$2^{O(k \log k)}$	?	6
Set Splitting	2^k	$2k$	7
Nonblocker	2.5154^k	$5k/3$	8
3-D Matching	2.77^{3k}		9
Edge Dominating Set	2.4181^k	$8k^2$	10
k-Path*	4^k	no $k^{O(1)}$	11
Convex Recolouring	4^k	$O(k^2)$	12
VC-max degree 3	1.1899^k		13
Clique Cover	$2^{\Delta k}$	2^k	14
Clique Partition		2^k	15
Cluster Editing	1.83^k	$4k$	16
Steiner Tree	2^k		17
3-Hitting Set	2.076^k	$O(k^2)$	18
Minimum Fill/ Interval Completion	$O(k^{2k}n^3m)$		19

The f(k) and kernel races

fpt.wikidot.com

1) Kemeny score is FPT parameterized by the score

Concept of Dirtyness: Dirty Pair

Dirty pair

Two candidates a and b form a *dirty pair* if in V there is one vote with $a > b$ and another vote with $b > a$.

Example:

vote 1:	a	>	b	>	c	>	d
vote 2:	a	>	d	>	b	>	c
vote 3:	b	>	c	>	a	>	d

In optimal consensus: "$a > d$" and "$b > c$"

dirty pairs: a,b and a,c and b,d and c,d

Plan: Find reduction rules (preprocessing rules), and search tree bounded by f(k).
Algorithms often interleave reductions and branching.

Rule1: Delete all candidates not in a "dirty pair".

Parameter: Kemeny score

After Reduction Rule: The Election has at most 2k candidates.
If there are more than 2k candidates after Rule 1, then NO.

v1 : a > b ...
v2 : b > a...
v3 : ... b > c ...
v4 : ... c > b ...
v5 : ... c > d...
v3 : ... d > c ...
. : ...
. : ...
vn : ...

In an optimal consensus, every dirty pair contributes at least one to the score. A Kemeny score of "k" means there can be at most only k dirty *pairs*; i.e., 2k candidates.

Rule 2: If there are more than 2k votes identical to a consensus list ℓ, return YES, otherwise, return NO.

Parameter: Kemeny score

The score of ℓ is at most k

v1 : a > b ...
v2 : b > a...
v3 : ... b > c ...
v4 : ... c > b ...
v5 : ... c > d...
v3 : ... d > c ...
. : ...
. : ...

vn : ...

k dirty pairs, (2k votes)

k votes identical to the consensus list

consensus list ℓ

Exhaustively apply Rules 1 and 2.

Parameter: Kemeny score

Rule 1: Delete all candidates not in a "dirty pair".
Rule 2: If there are more than $2k$ votes identical to a consensus list ℓ, return YES if the score of ℓ is at most k; otherwise, return NO.

A YES instance of KEMENY SCORE has at most $2k$ candidates and $2k$ votes.

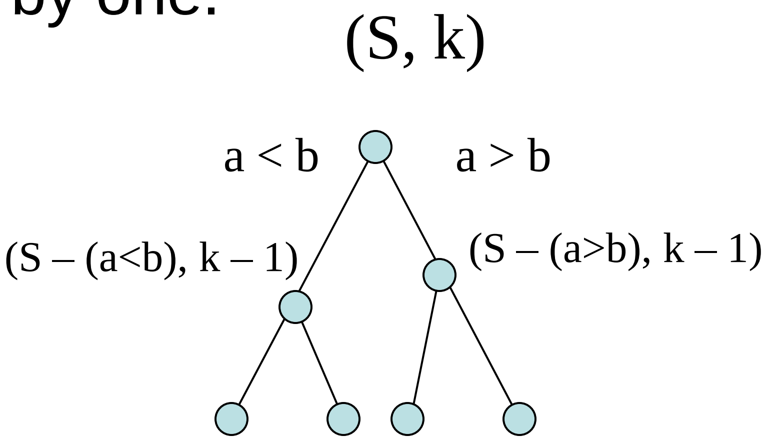
The Rules can be computed in $O(m^2 n)$ time.

Bounded Search Tree

Parameter: Kemeny score

A dirty pair "increases" the Kemeny score at least by 1.

At each search tree node, branch into the two possible relative orders of the dirty pair, and in each case decrease the parameter by one.



e f g consensus list ℓ

All non flip candidates have their relative order with respect to all other candidates already fixed.

Branching vector (1,1)
Gives banching number : 2
i.e., 2^k run time

Branching into dirty pairs results in a search tree of size $O(2^k)$.

Improvement by branching into *Dirty Triples*

Three candidates form a *dirty triple* if they occur in at least two dirty pairs.

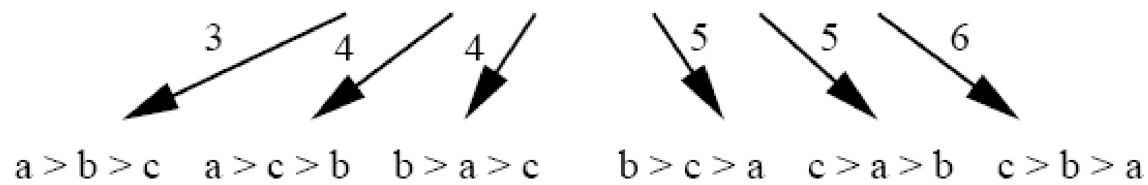
Example: {a, b, c} form a dirty triple for these 3 votes.

Case 1: dirty triple with 3 dirty pairs

vote 1: $a > b > c$

vote 2: $a > c > b$

vote 3: $b > c > a$



branching vector: $(3, 4, 4, 5, 5, 6) \Rightarrow$ branching number: 1.52

Case 2: dirty triple with 2 dirty pairs

branching vector: $(3, 3, 2) \Rightarrow$ branching number: 1.53

Case 3: no more dirty triples

for every dirty pair: make a “majority decision”
(and decrease the parameter accordingly)

KEMENY SCORE can be solved in $O(1.53^k + m^2 n)$ time, where k is the Kemeny Score.

Outline of the Dynamic Programming Algorithm

•The search tree algorithm enumerates all dirty pairs and then branches according to the dirty triples.

•At a search tree node, in each case of the branching, an order of the candidates involved in the dirty triples processed at this node is fixed and maintained in a set.

•This order represents the relative positions of these candidates in the Kemeny Consensus sought. The parameter is decreased according to this order.

•Since every order of two candidates in a dirty pair decreases the parameter by at least one, the height of the search tree is upper-bounded by k.

•At each node of the search tree we exhaustively apply the data reduction rules (interleave kernelization and search trees).

2) Parameter: "the maximum KT-distance between any two input votes"

Voter1:	**a > b > c**
Voter2:	**a > c > b**
Voter3:	**c > b > a**

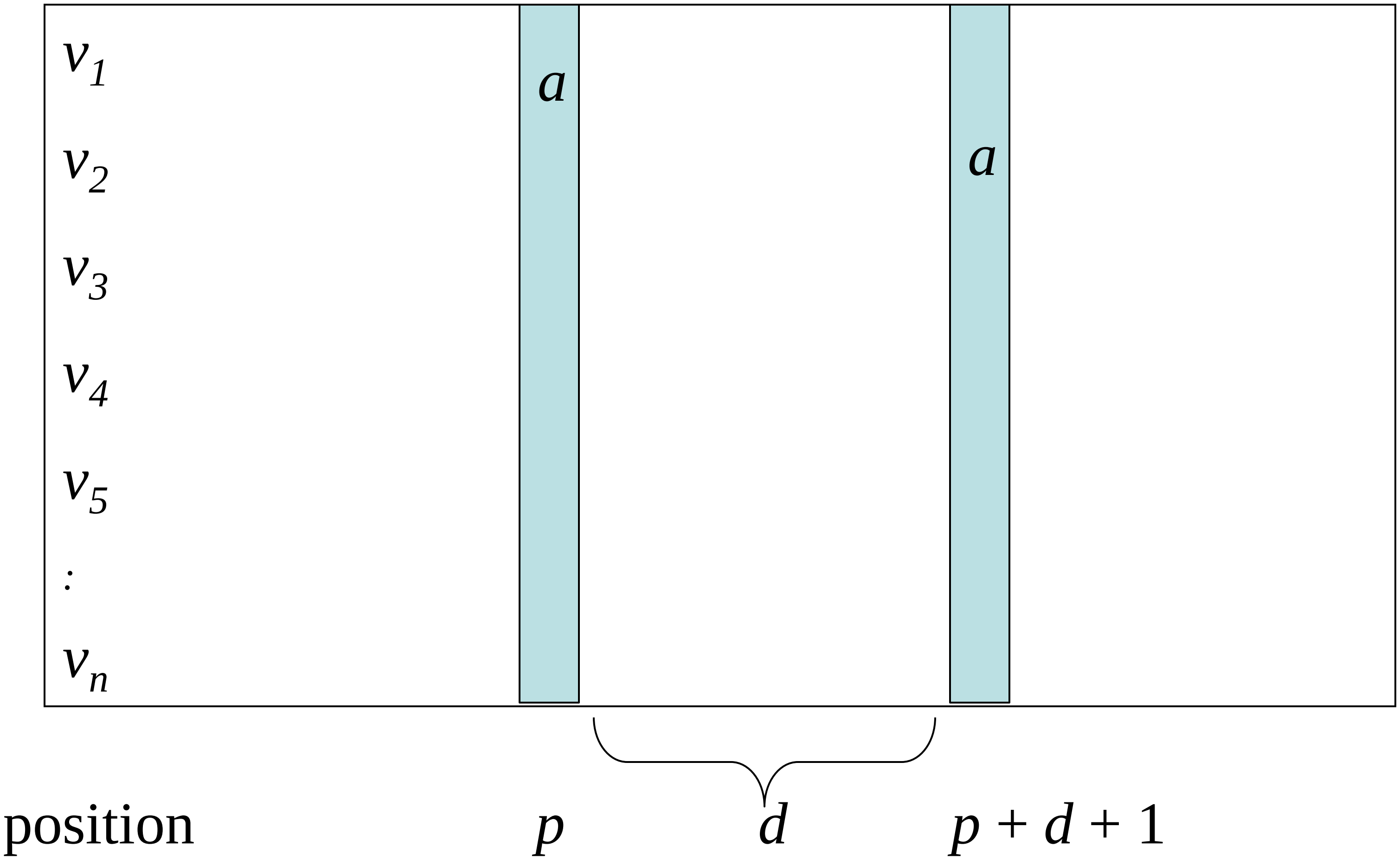
Input Votes	Candidate pairs ab	ac	bc	KT-distance between voters
Vote1&2	0	0	1	1
Vote2&3	1	1	0	2
Vote1&3	1	1	1	3

d = Maximum KT-distance between any two votes is 3.

Candidate Positions

Parameter: Maximum KT-distance

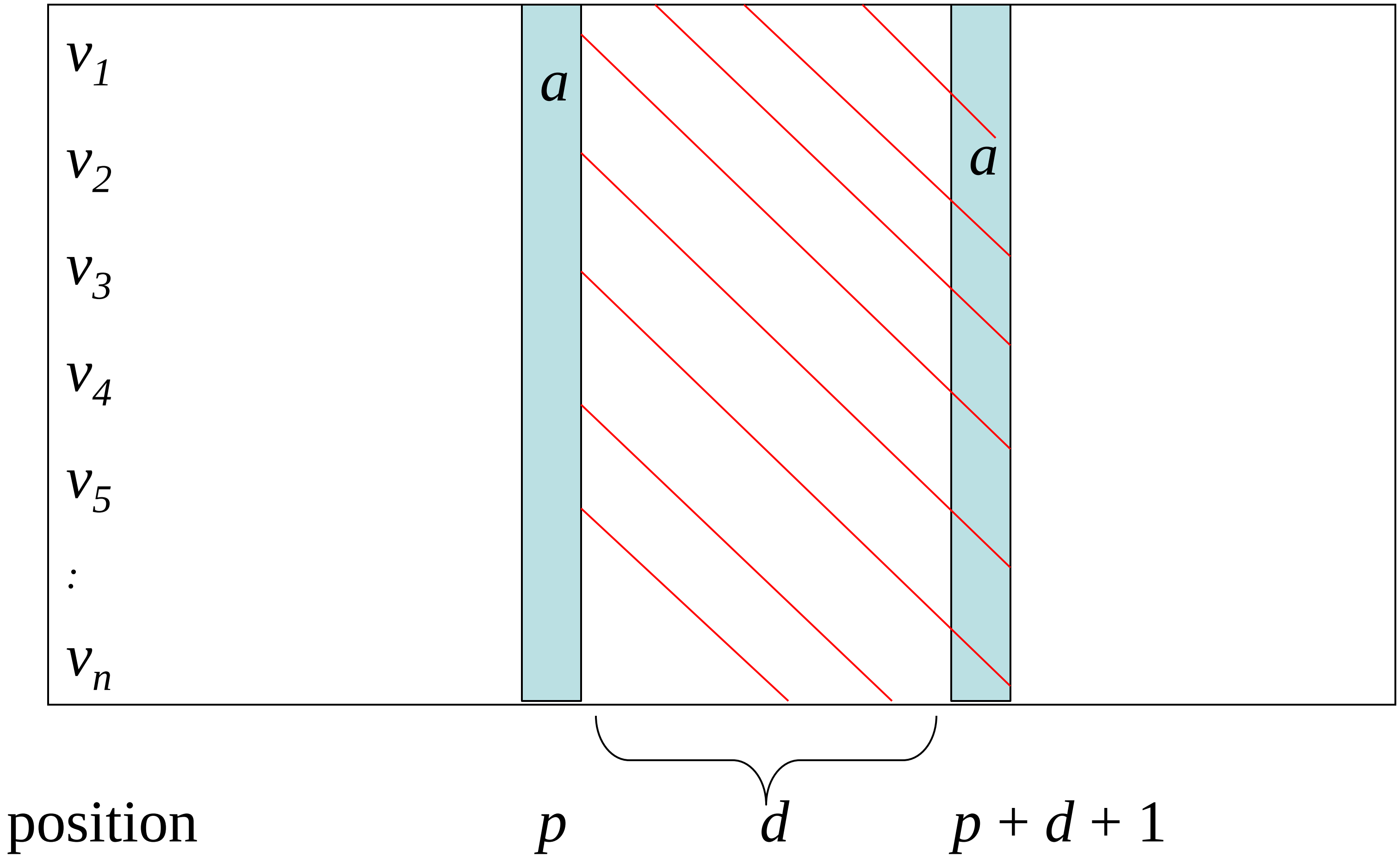
Lemma. In an input instance with maximum KT-distance d, the *positions* of a candidate in two votes differ by at most d.



Candidate Positions

Parameter: Maximum KT-distance

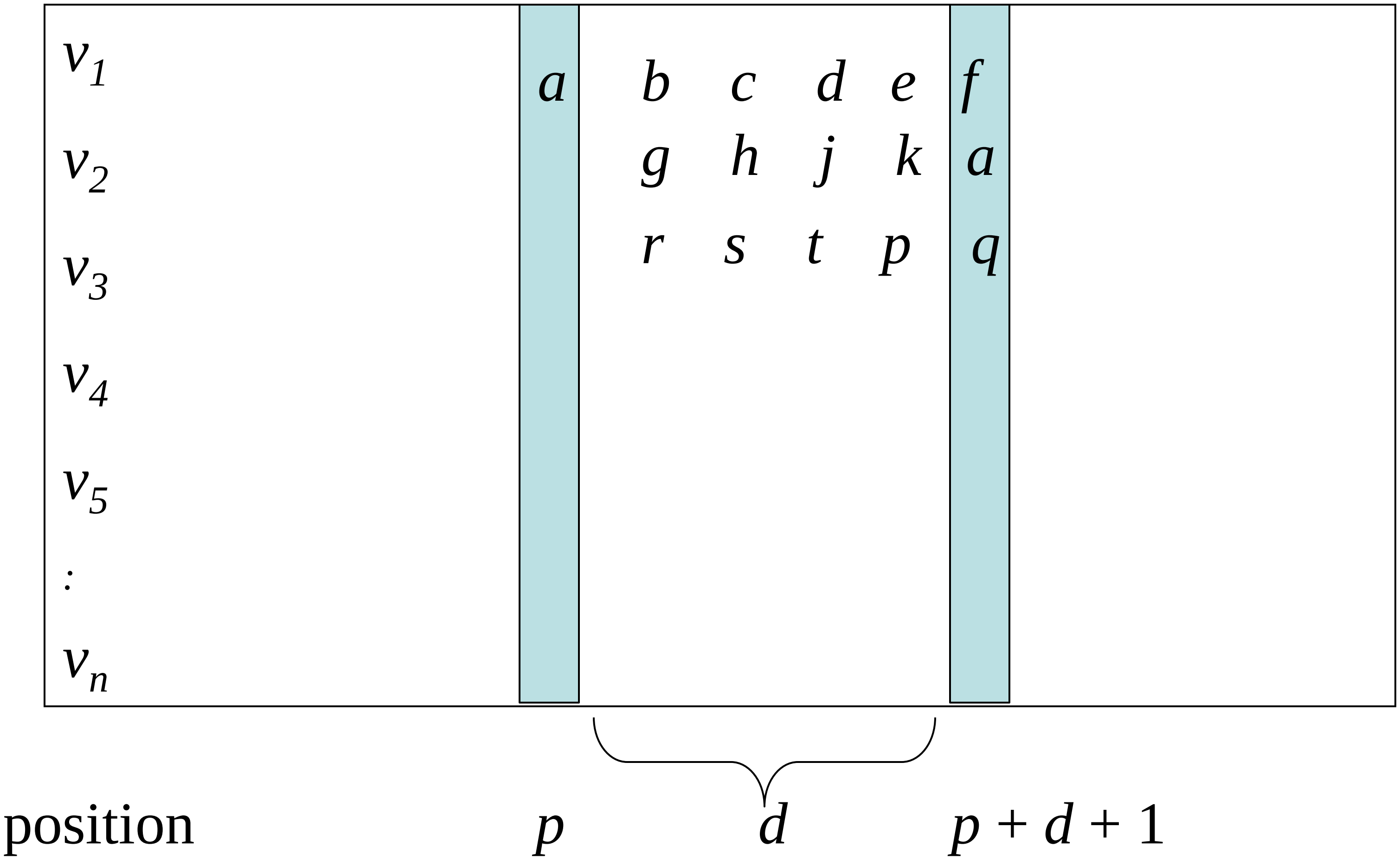
Lemma. In an input instance with maximum KT-distance d, every block of width d + 1 contains at most 3d + 1 candidates.



Candidate Positions

Parameter: Maximum KT-distance

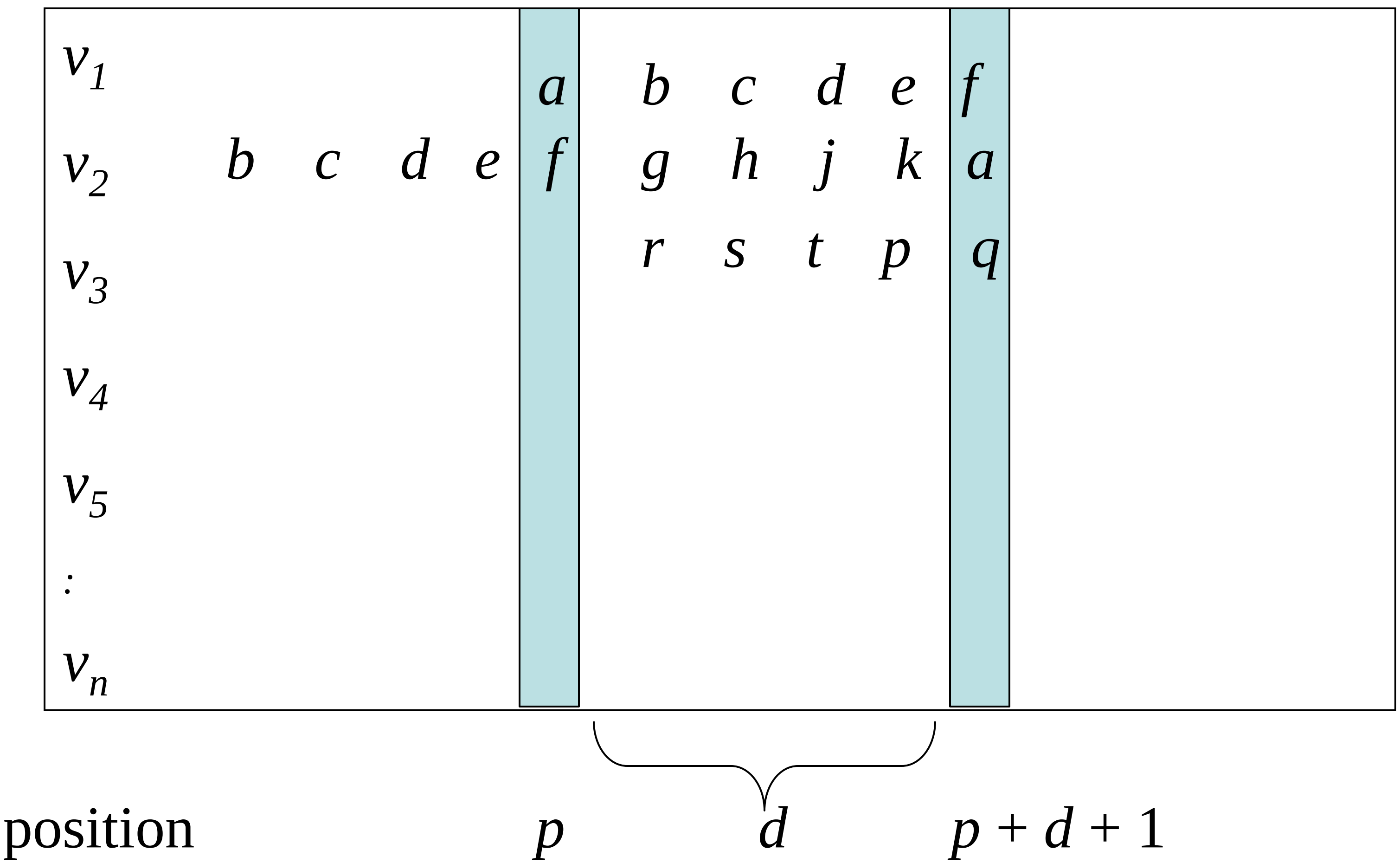
Lemma. In an input instance with maximum KT-distance d, every block of width d + 1 contains at most 3d + 1 candidates.



Candidate Positions

Parameter: Maximum KT-distance

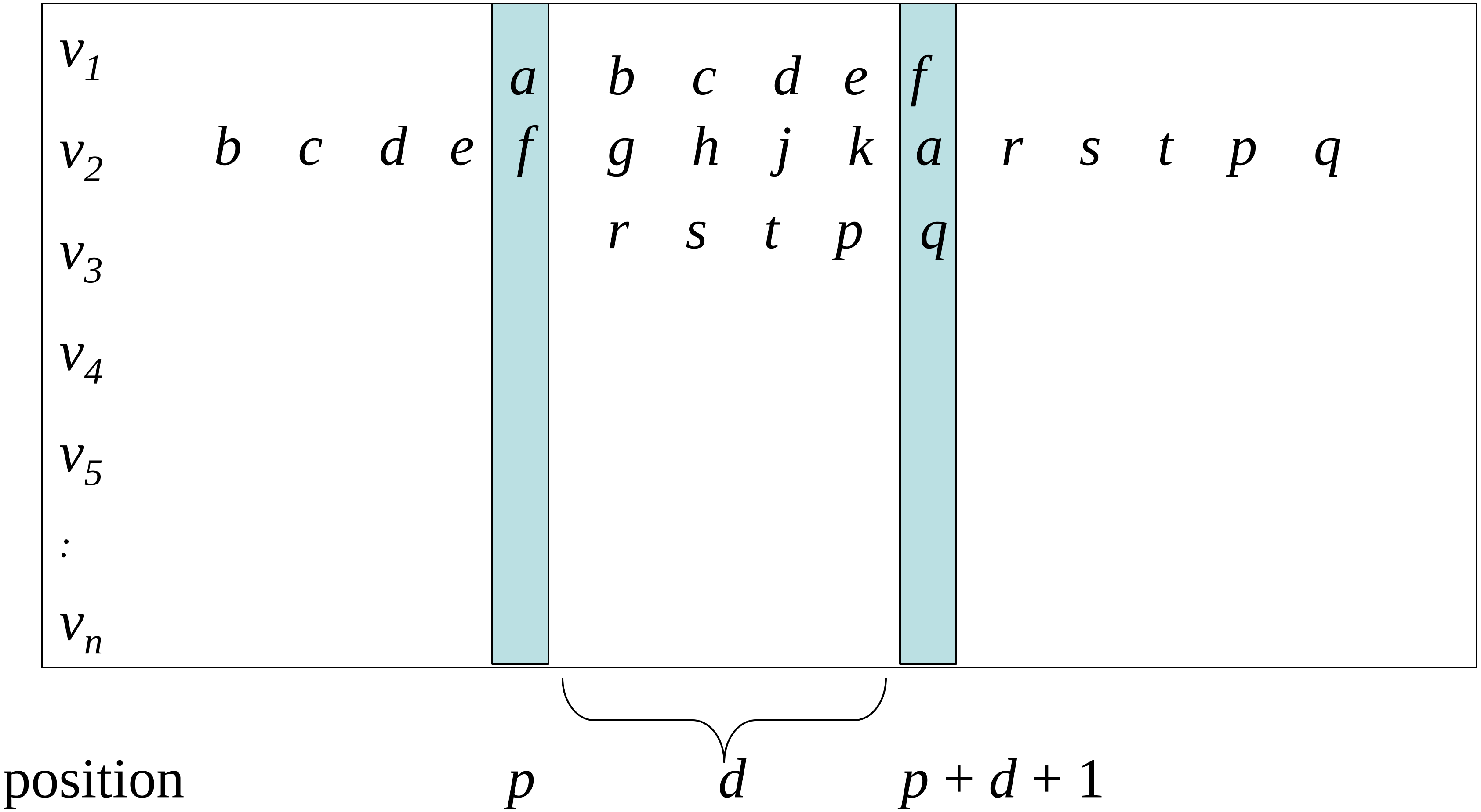
Lemma. In an input instance with maximum KT-distance d, every block of width d + 1 contains at most 3d + 1 candidates.



Candidate Positions

Parameter: Maximum KT-distance

Lemma. In an input instance with maximum KT-distance d, every block of width d + 1 contains at most 3d + 1 candidates.



KEMENY SCORE can be solved in $O((3d+1)!\, d \log d\, m\, n)$ time with d the maximum KT-distance between two input votes

Outline of Dynamic Programming algorithm

- Range of each candidate is bounded by a *block* of size d.
- Block size is bounded by $3d + 1$.

Algorithm:

- Iterate left to right over the blocks of votes.
- Try all possible orders of the candidates within a block.
- There are $(3d + 1)!$ possible orders of a block.

3) Parameter "the average KT-distance"

Definition

For an election (V, C) the average KT-distance d_a is defined as

$$d_a := \frac{1}{n(n-1)} \cdot \sum_{\{u,v\} \in V, u \neq v} \text{KT-dist}(u, v).$$

Sum over all pairwise distances divided by the number of pairs.

Average range of candidate *Parameter: Average KT-distance*

Position	0	1	2	3	4	5
v_1	*a*	*b*	*c*	*d*	*e*	*f*
v_2	*b*	*c*	*d*	*e*	*f*	*a*
v_3	*c*	*a*	*b*	*f*	*e*	*d*
v_4	*c*	*a*	*b*	*f*	*e*	*d*

Maximum Range is
Range(a): 5 – 0 =5

Vote: v_1 v_2 v_3 v_4

ave pos(a): 0 + 5 + 1 + 1 = 7/4

ave pos(b): 1 + 0 + 2 + 1 = 4/4

ave pos(c): 2 + 1 + 0 + 0 = 3/4

ave pos(d): 3 + 2 + 5 + 5 = 15/4

ave pos(e): 4 + 3 + 4 + 4 = 11/4

ave pos(f): 5 + 4 + 3 + 3 = 15/4

Average range: 55/4

In every Kemeny consensus, every candidate can only assume a number of consecutive positions bounded by da.

Example 1: small range, large number of candidates and average distance	Example 2: small average distance, large number of candidates and range

$a > c > b > e > d > f \ldots$	$a > b > c > d > e > f \ldots$
$b > a > c > d > e > f \ldots$	$b > c > d > e > f > a \ldots$
$b > c > a > e > f > d \ldots$	$a > b > c > d > e > f \ldots$

- Number of candidates m ($O^*(2^m)$)
- Maximum range r of candidate positions in the input votes ($O^*(32^r)$)
- Average distance of the input votes ($O^*(16^{d_a})$)

Lesson: Check the size of the parameter, then use the appropriate strategy for your problem.

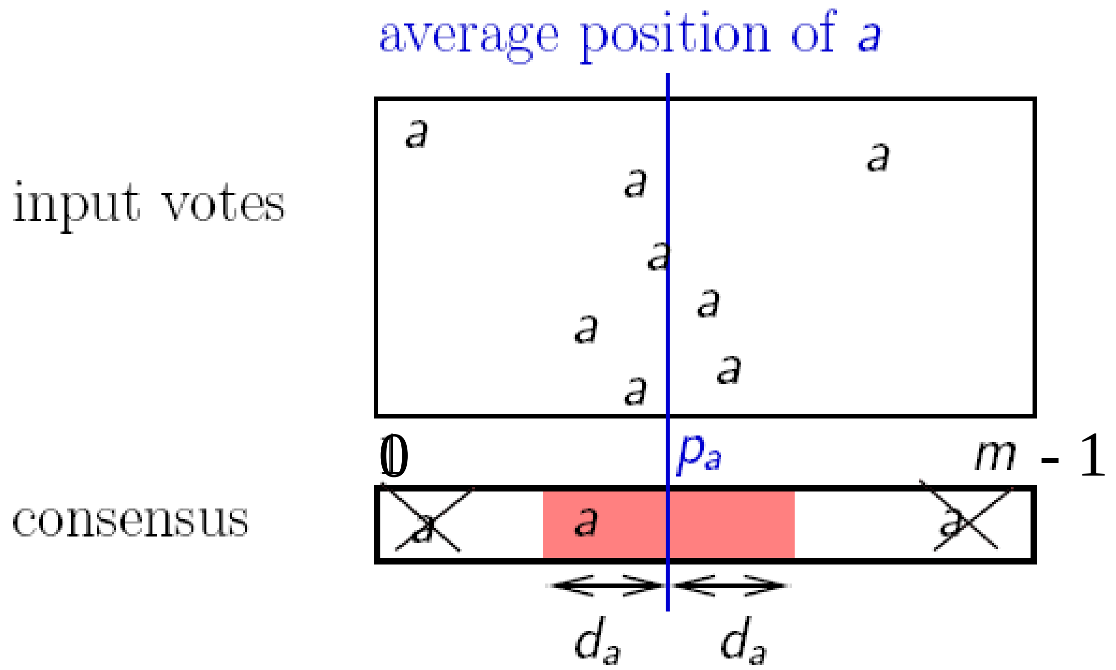
Position of candidate in optimal consensus

Parameter: average KT-distance

Let the average position of a candidate c be $p_a(c)$.

Lemma

Let d_a be the average KT-distance of an election (V, C). Then, in every optimal Kemeny consensus l, for every candidate $c \in C$ we have $p_a(c) - d_a < l(c) < p_a(c) + d_a$.



Bounded number of candidates per position

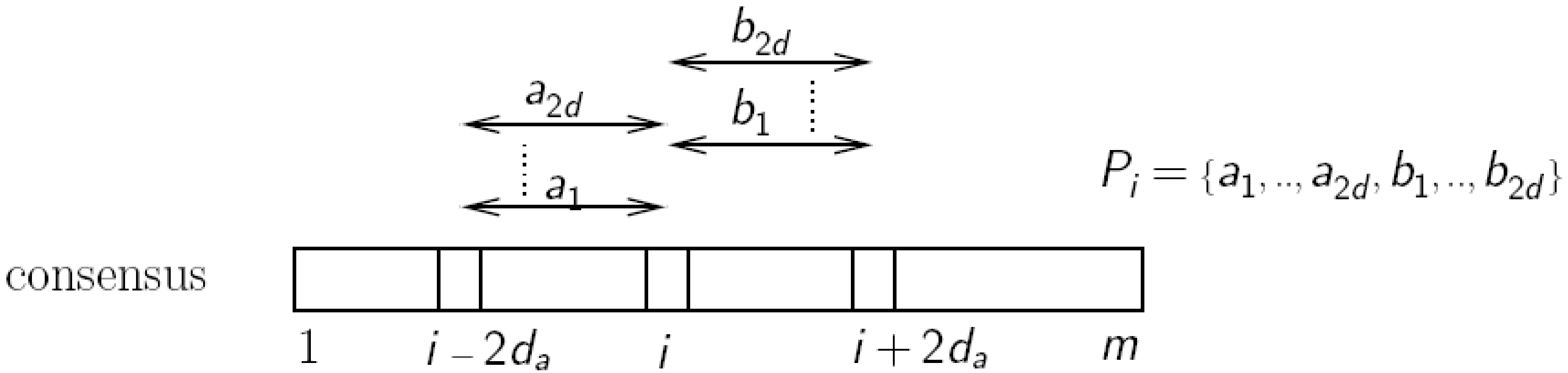
Parameter: average KT-distance

For a position i, let P_i denote the set of candidates that can assume i in an optimal consensus.

Lemma

Let d_a be the average KT-distance of an election (V, C). For a position i, we have $|P_i| \leq 4 \cdot d_a$.

Proof: Position "range" of every candidate is at most $2 \cdot d_a$.



Every candidate of P_i must have a position smaller than $i + 2d_a$ and greater than $i - 2d_a$.

Dynamic Programming

Parameter: average KT-distance

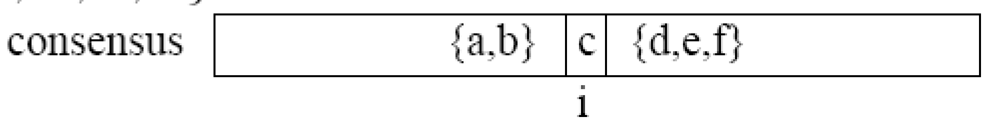
Position i, a candidate $c \in P_i$, a subset of candidates $P'_i \subseteq P_i \setminus \{c\}$

Definition

$T(i, c, P'_i) :=$ optimal partial Kemeny score if c has position i and all candidates of P'_i have positions smaller than i

$P_i = \{a, b, c, d, e, f\}$

$P'_i = \{a, b\}$



Computation of partial Kemeny scores:

- Overall Kemeny score can be decomposed (just a sum over all votes and pairs of candidates)
- Relative orders between c and all other candidates are already fixed

KEMENY SCORE can be solved in $O^*(16^{d_a})$ time, with d_a the average KT-distance

Outline of the Dynamic Programming algorithm

➢T(I, c, P'_i) is defined an *optimum partial Kemeny Score* if ‘c’ has position ‘i’ and all other candidates in P_i have positions smaller than i.

➢Computation of partial Kemeny Score: Overall KS can be decomposed. Relative orders between c and all other candidates are already fixed.

n votes

m candidates

$P_i = \{a, b, c, d, e, f\}$

consensus	$\{a, b\}$	c	$\{d, e, f\}$
		i	

We have $|P_i| \leq 4d_a$, thus there are at most 2^{4d_a} subsets of P_i.

$\Rightarrow$ Table size is bounded by $16^{d_a} \cdot \text{poly}(n, m)$.

Parameterizations of KEMENY SCORE

1. *Parameter "the Kemeny Score"*. Solvable in $O(1.53^k + m^2 n)$ time, where k denotes the Kemeny Score of the given election.
2. *Parameter "the maximum KT-distance between any two input votes".* Solvable in $O((3d + 1)!\ d\ (\log d)\ mn)$ time.
3. *Parameter "the number of candidates".* Solvable in $O(2^m m^2 n)$ time.
4. *Parameter "the maximum range of candidate positions".* Solvable in $O^*(32^r)$ time.
5. *Parameter "the average KT-distance".* Solvable in $O^*(16^{aveKT})$ time.

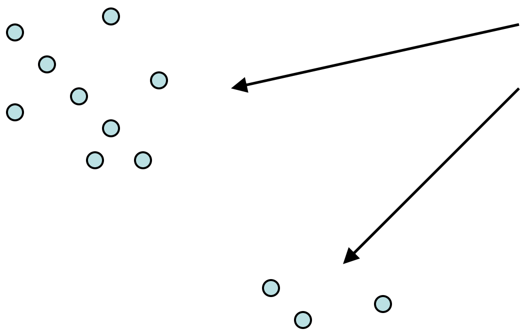
Work done by Nadja Betzler, Jiong Guo, Rolf Niedermeier. Friedrich-Schiller University, Jena, Germany. Michael Fellows, Frances Rosamond. Newcastle, AU.

KEMENY SCORE Ties & Incomplete Votes

	Kemeny Score	with Ties	Incomplete Votes
Kemeny score k	kernelization $O^*(1.53^k)$	kernelization $O^*(1.76^k)$	$O^*((1.48k)^k)$
max. KT-dist. d	$O^*((3d+1)!)$	$O^*((6d+2)!)$	NP-h for $d=0$
# candidates m	$O^*(2^m)$	$O^*(2^m)$	$O^*(2^m)$
# votes n	NP-h for $n=4$	NP-h for $n=4$	NP-h for $n=4$

Open Problems and Future Directions

Notice that the order of the websites given by various search engines (Google, Yahoo, Bai) is almost the same. This implies a small average distance between the sites.



Average distance between points is small. There may be a few outliers. In Geometry called a “one-center” problem. There could be a 2-center problem.

LESSON: Relate the number of dirty elements (involved in dirty pairs) with average distance. Find reduction rules. Bound the number of dirty elements by a function of average distance in reduced instances.

Open Problems and Future Directions

- Investigate typical values for average KT-distance.
- Investigate "similar on average" in other voting systems/settings.
- Consider aggregate parameters.
- The majority of kernelization technique/development has been done with graph problems. Continue to find techniques/tools for consensus type problems. (B, G, K, N: a "partial kernelization").
- There is a branching algorithm for KEMENY SCORE which runs in $O((5.823)^d \text{poly}(n, m))$ time. (N. Simjour. Improved parameterized algorithms for the Kemeny Aggregation problem. In Proc. 4th IWPEC, v 5917 of LNCS, pp 312-323, 2009.) How can branching continued to be improved?
- The "*maximum distance of the input rankings from the solution*" is an upper bound for the average distance. In contrast, "*the average distance of the input rankings from the solution*" is a lower bound for the "*average distance between the input rankings*". Open to investigate the parameterized complexity with respect to this parameter.

Thank you