

## Contents

<b>1 Scalably Scheduling Processes with Arbitrary Speedup Curves</b> <b>Jeff Edmonds and Kirk Pruhs</b>	<b>9</b>
<b>2 Three-coloring triangle-free planar graphs in linear time</b> <b>Zdenek Dvorak, Ken-ichi Kawarabayashi and Robin Thomas</b>	<b>9</b>
<b>3 An Efficient Sparse Regularity Concept</b> <b>Amin Coja-Oghlan, Colin Cooper and Alan Frieze</b>	<b>9</b>
<b>4 Exponential Lower Bounds and Integrality Gaps for Lovasz-Schrijver Procedures</b> <b>Toniann Pitassi and Nathan Segerlind</b>	<b>9</b>
<b>5 On the power of two, three and four probes</b> <b>Uriel Feige and Noga Alon</b>	<b>10</b>
<b>6 Dimension detection via slivers</b> <b>Siu-Wing Cheng and Man-Kwun Chiu</b>	<b>10</b>
<b>7 On the Approximability of Dodgson and Young Elections</b> <b>Ioannis Caragiannis, Jason Covey, Michal Feldman, Christopher Homan, Christos Kaklamanis, Nikos Karanikolas, Ariel Procaccia and Jeffrey Rosenschein</b>	<b>10</b>
<b>8 Self-overlapping Curves Revisited</b> <b>David Eppstein and Elena Mumford</b>	<b>11</b>
<b>9 Hardness of embedding simplicial complexes in <math>\mathbb{R}^d</math></b> <b>Jiri Matousek, Martin Tancer and Uli Wagner</b>	<b>11</b>
<b>10 The Assignment Problem in Content Distribution Networks: Unsplittable Hard-Capacitated Facility Location</b> <b>MohammadTaghi Hajiaghayi and Mohammad H. Bateni</b>	<b>11</b>
<b>11 Approximation Algorithms for Restless Bandit Problems</b> <b>Sudipto Guha, Kamesh Munagala and Peng Shi</b>	<b>12</b>
<b>12 Biased Range Trees</b> <b>Vida Dujmovic, John Howat and Pat Morin</b>	<b>12</b>
<b>13 An almost <math>O(\log k)</math>-approximation for <math>k</math>-connected subgraphs</b> <b>Zeev Nutov</b>	<b>12</b>
<b>14 Linear-Time Algorithms for Geometric Graphs with Sublinearly Many Crossings</b> <b>David Eppstein, Michael Goodrich and Darren Strash</b>	<b>13</b>
<b>15 Improved Approximating Algorithms for Directed Steiner Forest</b> <b>Moran Feldman, Guy Kortsarz and Zeev Nutov</b>	<b>13</b>
<b>16 Line Transversals of Convex Polyhedra in <math>\mathbb{R}^3</math></b> <b>Haim Kaplan, Natan Rubin and Micha Sharir</b>	<b>14</b>

<b>17 String Hashing for Linear Probing</b> <b>Mikkel Thorup</b>	<b>14</b>
<b>18 Improved Approximation Bound for Quadratic Optimization Problems with Orthogonality Constraints</b> <b>Anthony Man-Cho So</b>	<b>14</b>
<b>19 Algorithms for Finding an Induced Cycle in Planar Graphs</b> <b>Ken-ichi Kawarabayashi and Yusuke KOBAYASHI</b>	<b>15</b>
<b>20 A quadratic kernel for feedback vertex set</b> <b>Stephan Thomasse</b>	<b>16</b>
<b>21 On risks of using cuckoo hashing with simple universal hash classes</b> <b>Martin Dietzfelbinger and Ulf Schellbach</b>	<b>16</b>
<b>22 Asymptotically optimal frugal colouring</b> <b>Michael Molloy and Bruce Reed</b>	<b>16</b>
<b>23 A Unified Approach to Distance-Two Colouring of Planar Graphs</b> <b>Omid Amini, Louis Esperet and Jan van den Heuvel</b>	<b>17</b>
<b>24 Weighted flow time does not admit <math>O(1)</math>-competitive algorithms</b> <b>Nikhil Bansal and Ho-Leung Chan</b>	<b>17</b>
<b>25 On the bit-complexity of Lempel-Ziv compression</b> <b>Paolo Ferragina, Igor Nitto and Rossano Venturini</b>	<b>17</b>
<b>26 Testing Halfspaces</b> <b>Kevin Matulef, Ryan O'Donnell, Ronitt Rubinfeld and Rocco Servedio</b>	<b>17</b>
<b>27 Multi-Dimensional Online Tracking</b> <b>Ke Yi and Qin Zhang</b>	<b>18</b>
<b>28 Termination Criteria for Solving Concurrent Safety and Reachability Games</b> <b>Krishnendu Chatterjee, Luca de Alfaro and Thomas Henzinger</b>	<b>18</b>
<b>29 Shortest Paths in Directed Planar Graphs with Negative Lengths: a linear-space <math>O(n \log^2 n)</math>-time algorithm</b> <b>Philip Klein, Shay Mozes and Oren Weimann</b>	<b>19</b>
<b>30 Efficient coordination mechanisms for unrelated machine scheduling</b> <b>Ioannis Caragiannis</b>	<b>19</b>
<b>31 Approximating fractional hypertree width</b> <b>Daniel Marx</b>	<b>19</b>
<b>32 Improved Smoothed Analysis of the k-Means Method</b> <b>Bodo Manthey and Heiko Roeglin</b>	<b>20</b>
<b>33 An Improved Approximation Algorithm for the Column Subset Selection Problem</b> <b>Christos Boutsidis, Michael Mahoney and Petros Drineas</b>	<b>20</b>

<b>34 (Un)Expected Behavior of Digital Search Tree Profile</b> <b>Michael Drmota and Wojciech Szpankowski</b>	<b>21</b>
<b>35 A logarithmic approximation for the unsplittable flow on line graphs</b> <b>Nikhil Bansal, Zachary Friggstad, Rohit Khandekar and Mohammad Salavatipour</b>	<b>21</b>
<b>36 Variance-optimal sampling-based estimation of subset sums</b> <b>Edith Cohen, Nick Duffield, Haim Kaplan, Carsten Lund and Mikkel Thorup</b>	<b>21</b>
<b>37 Finding shortest contractible cycles in embedded graphs</b> <b>Sergio Cabello</b>	<b>22</b>
<b>38 On the approximability of the maximum feasible subsystem problem with 0/1-coefficients</b> <b>Khaled Elbassioni, Rajiv Raman, Saurabh Ray and Rene Sitters</b>	<b>22</b>
<b>39 Clustering on Noisy Mixtures</b> <b>S. Charles Brubaker</b>	<b>22</b>
<b>40 Perfect matchings via uniform sampling in regular bipartite graphs</b> <b>Ashish Goel, Michael Kapralov and Sanjeev Khanna</b>	<b>23</b>
<b>41 An <math>\tilde{O}(m)</math> algorithm for constructing a cactus of an undirected graph</b> <b>David Karger and Debmalya Panigrahi</b>	<b>23</b>
<b>42 Efficient Algorithms for Column Subset Selection</b> <b>Joel Tropp</b>	<b>23</b>
<b>43 Improved bounds and new techniques for Davenport-Schinzel sequences and their generalizations</b> <b>Gabriel Nivasch</b>	<b>24</b>
<b>44 Efficient algorithms on sets of permutations, dominance, and real-weighted APSP</b> <b>Raphael Yuster</b>	<b>24</b>
<b>45 High rate fingerprinting codes and the fingerprinting capacity</b> <b>Gabor Tardos and Ehsan Amiri</b>	<b>25</b>
<b>46 Almost all hypergraphs without Fano planes are bipartite</b> <b>Yury Person and Mathias Schacht</b>	<b>25</b>
<b>47 Pairing Heaps with <math>O(\log \log n)</math> decrease Cost</b> <b>Amr Elmasry</b>	<b>25</b>
<b>48 A simpler implementation and analysis of Chazelle’s Soft Heaps</b> <b>Haim Kaplan and Uri Zwick</b>	<b>26</b>
<b>49 Discounted Deterministic Markov Decision Processes and Discounted All-Pairs Shortest Paths</b> <b>Mikkel Thorup, Uri Zwick and Omid Madani</b>	<b>26</b>
<b>50 Efficient algorithms for the 2-gathering problem</b> <b>Alon Shalita and Uri Zwick</b>	<b>26</b>

<b>51 Persistent Homology for Kernels, Images, and Cokernels</b> david cohen-steiner, Herbert Edelsbrunner, John Harer and Dmitriy Morozov	<b>26</b>
<b>52 Clique-width: On the Price of Generality</b> Fedor Fomin, Petr Golovach, Daniel Lokshtanov and Saket Saurabh	<b>27</b>
<b>53 Maximum Independent Set of Rectangles</b> Parinya Chalermsook and Julia Chuzhoy	<b>27</b>
<b>54 On Algorithmic Hypergraph Regularity</b> Brendan Nagle, Annika Poerschke, Vojtech Rdl and Mathias Schacht	<b>27</b>
<b>55 Partitioning Graphs into Balanced Components</b> Robi Krauthgamer, Seffi Naor and Roy Schwartz	<b>28</b>
<b>56 Speed Scaling with an Arbitrary Power Function</b> Nikhil Bansal, Ho-Leung Chan and Kirk Pruhs	<b>28</b>
<b>57 Coloring triangle-free graphs on surfaces</b> Zdenek Dvorak, Daniel Kral and Robin Thomas	<b>28</b>
<b>58 On the Relative Strength of Split, Triangle and Quadrilateral Cuts (Extended Abstract)</b> Amitabh Basu, Pierre Bonami, gerard cornuejols and Francois Margot	<b>29</b>
<b>59 From coding theory to efficient pattern matching</b> Raphael Clifford, Klim Efremenko, Ely Porat and Amir Rothschild	<b>29</b>
<b>60 Fast Edge Orientation for Unweighted Graphs</b> Ramesh Hariharan and Anand Bhalgat	<b>29</b>
<b>61 On the Maximum Quadratic Assignment Problem</b> Viswanath Nagarajan and Maxim Sviridenko	<b>30</b>
<b>62 Natural Algorithms</b> Bernard Chazelle	<b>30</b>
<b>63 Comparison-Based, Time-Space Lower Bounds for Selection</b> Timothy M. Chan	<b>30</b>
<b>64 Optimal Halfspace Range Reporting in Three Dimensions</b> Peyman Afshani and Timothy M. Chan	<b>31</b>
<b>65 Optimality of Belief Propagation for Random Assignment Problem</b> Justin Salez and Devavrat Shah	<b>31</b>
<b>66 Compressed Counting</b> Ping Li	<b>31</b>
<b>67 The Geometry of Binary Search Trees</b> Erik D. Demaine, Dion Harmon, John Iacono, Daniel Kane and Mihai Patrascu	<b>32</b>

<b>68</b>	<b>A Simple Combinatorial Algorithm for Submodular Function Minimization</b> Satoru Iwata and James Orlin	<b>32</b>
<b>69</b>	<b>Towards Computing the Grothendieck Constant</b> Prasad Raghavendra and David Steurer	<b>32</b>
<b>70</b>	<b>A Generic Top-Down Dynamic-Programming Approach to Prefix-Free Coding</b> Mordecai J. Golin, Xiaoming Xu and Jiajin Yu	<b>33</b>
<b>71</b>	<b>Additive Approximation Algorithms for List-Coloring Minor-Closed Class of Graphs</b> Ken-ichi Kawarabayashi	<b>33</b>
<b>72</b>	<b>A Nearly Linear Time Algorithm For The Half Integral Parity Disjoint Paths Packing Problem</b> Ken-ichi Kawarabayashi and Bruce Reed	<b>34</b>
<b>73</b>	<b>Approximating Submodular Functions Everywhere</b> Michel Goemans, Nicholas J.A. Harvey, Satoru Iwata and Vahab Mirrokni	<b>34</b>
<b>74</b>	<b>Stepwise Randomized Combinatorial Auctions Achieve Revenue Monotonicity</b> Baharak Rastegari, Anne Condon and Kevin Leyton-Brown	<b>35</b>
<b>75</b>	<b>Monotone Minimal Perfect Hashing: Searching a Sorted Table with <math>O(1)</math> Accesses</b> Djamal Belazzougui, Paolo Boldi, Rasmus Pagh and Sebastiano Vigna	<b>36</b>
<b>76</b>	<b>Approximate Nearest Neighbors for Affine Subspaces Queries</b> Alexandr Andoni, Piotr Indyk, Robi Krauthgamer and Huy Nguyen	<b>36</b>
<b>77</b>	<b>Collecting Weighted Items from a Dynamic Queue</b> Marcin Bienkowski, Marek Chrobak, Christoph Drr, Mathilde Hurand, Artur Jez, ?ukasz Je? and Grzegorz Stachowiak	<b>36</b>
<b>78</b>	<b>The Uniform Hardcore Lemma via Approximate Bregman Projections</b> Boaz Barak, Moritz Hardt and Satyen Kale	<b>37</b>
<b>79</b>	<b>Transitive-Closure Spanners</b> Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova and David Woodruff	<b>37</b>
<b>80</b>	<b>On the Complexity of Nash Equilibria of Action-Graph Games</b> Constantinos Daskalakis, Grant Schoenebeck, Gregory Valiant and Paul Valiant	<b>38</b>
<b>81</b>	<b>Overcoming the <math>L_1</math> Non-Embeddability Barrier: Algorithms for Product Metrics</b> Alexandr Andoni, Piotr Indyk and Robi Krauthgamer	<b>38</b>
<b>82</b>	<b>Finding repeats in a data-stream</b> Parikshit Gopalan and Jaikumar Radhakrishnan	<b>39</b>
<b>83</b>	<b>Approximate Shared-Memory Counting Despite a Strong Adversary</b> James Aspnes and Keren Censor	<b>39</b>

<b>84 On Low Dimensional Local Embeddings</b> Ittai Abraham, Yair Bartal and Ofer Neiman	<b>39</b>
<b>85 Online Topological Ordering</b> Michael Bender, Jeremy Fineman and Seth Gilbert	<b>40</b>
<b>86 The Johnson-Lindenstrauss lemma almost characterizes Hilbert space, but not quite</b> William Johnson and Assaf Naor	<b>40</b>
<b>87 Parameterized approximation scheme for the multiple knapsack problem</b> Klaus Jansen	<b>40</b>
<b>88 Inserting a Vertex into a Planar Graph</b> Markus Chimani, Carsten Gutwenger, Petra Mutzel and Christian Wolf	<b>40</b>
<b>89 Paging and List Update under Bijective Analysis</b> Spyros Angelopoulos and Pascal Schweitzer	<b>41</b>
<b>90 On stars and Steiner stars II</b> Adrian Dumitrescu, Csaba Toth and G Xu	<b>41</b>
<b>91 Reasoning about Online Algorithms with Weighted Automata</b> Benjamin Aminof, Orna Kupferman and Robby Lampert	<b>42</b>
<b>92 On smoothed k-CNF formulas and the Walksat algorithm</b> Amin Coja-Oghlan, Uriel Feige, Alan Frieze, Michael Krivelevich and Dan Vilenchik	<b>42</b>
<b>93 How hard is it to approximate the best Nash equilibrium?</b> Elad Hazan and Robi Krauthgamer	<b>42</b>
<b>94 The Ratio Index for Budgeted Learning, with Applications</b> Ashish Goel, Sanjeev Khanna and Brad Null	<b>43</b>
<b>95 A universally fastest algorithm for Max 2-Sat, Max 2-CSP, and everything in between</b> Serge Gaspers and Gregory Sorkin	<b>43</b>
<b>96 Decomposition of Multiple Coverings into More Parts</b> Greg Aloupis, Jean Cardinal, Sebastien Collette, Stefan Langerman, David Orden and Pedro Ramos	<b>43</b>
<b>97 Computing the nucleolus of weighted voting games</b> Edith Elkind and Dmitrii Pasechnik	<b>44</b>
<b>98 Maximum Biconnected Subgraphs of Random Planar Graphs</b> Konstantinos Panagiotou and Angelika Steger	<b>44</b>
<b>99 List-Color-Critical Graphs on a Fixed Surface</b> Ken-ichi Kawarabayashi and Bojan Mohar	<b>45</b>
<b>100 Maximizing Submodular Set Functions Subject to Multiple Linear Constraints</b> Ariel Kulik, Hadas Shachnai and Tami Tamir	<b>45</b>

<b>101</b>	<b>Coresets and Approximate Clustering for Bregman Divergences</b> Marcel R. Ackermann and Johannes Blmer	<b>45</b>
<b>102</b>	<b>Online Scheduling to Minimize the Maximum Delay Factor</b> Benjamin Moseley and Chandra Chekuri	<b>46</b>
<b>103</b>	<b>Approximate Clustering without the Approximation</b> Maria-Florina Balcan, Avrim Blum and Anupam Gupta	<b>46</b>
<b>104</b>	<b>Secretary Problems: Weights and Discounts</b> Moshe Babaioff, Michael Dinitz, Anupam Gupta, Nicole Immorlica and Kunal Talwar	<b>47</b>
<b>105</b>	<b>Improved Equilibria via Public Service Advertising</b> Maria-Florina Balcan, Avrim Blum and Yishay Mansour	<b>47</b>
<b>106</b>	<b>Exact Algorithm for Partial Curve Matching via the Frechet Distance</b> Yusu Wang, Kevin Buchin and Maike Buchin	<b>48</b>
<b>107</b>	<b>Approximate Euclidean Shortest Path amid convex obstacles</b> Pankaj K Agarwal, R Sharathkumar and Hai Yu	<b>48</b>
<b>108</b>	<b>Sorting and Selection in Posets</b> Constantinos Daskalakis, Richard M. Karp, Elchanan Mossel, Samantha Riesenfeld and Elad Verbin	<b>48</b>
<b>109</b>	<b>Sorting by Placement and Shift</b> Sergi Elizalde and Peter Winkler	<b>49</b>
<b>110</b>	<b>3-Bit Dictator Testing: 1 vs 5/8</b> Ryan O'Donnell and Yi Wu	<b>49</b>
<b>111</b>	<b>Packing multiway cuts in graphs</b> Siddharth Barman and Shuchi Chawla	<b>49</b>
<b>112</b>	<b>Analysis of Scalar Fields over Point Cloud Data</b> Frederic Chazal, Leonidas Guibas, Steve Oudot and Primoz Skraba	<b>50</b>
<b>113</b>	<b>On the hitting times of quantum versus random walks</b> Frederic Magniez, Ashwin Nayak, Peter Richter and Miklos Santha	<b>50</b>
<b>114</b>	<b>Better Algorithms for Benign Bandits</b> Elad Hazan and Satyen Kale	<b>50</b>
<b>115</b>	<b>An Online Mechanism for Ad Slot Reservations with Cancellations</b> Florin Constantin, Jon Feldman, S Muthukrishnan and Martin Pal	<b>51</b>
<b>116</b>	<b>The cover time of random geometric graphs</b> Colin Cooper and Alan Frieze	<b>51</b>
<b>117</b>	<b>The extended k-tree algorithm</b> Lorenz Minder and Alistair Sinclair	<b>52</b>

<b>118</b>	<b>Constructing Laplace operators from Point Clouds</b> Misha Belkin, jian sun and Yusu Wang	<b>52</b>
<b>119</b>	<b>Online story scheduling for web advertising</b> Anirban Dasgupta, Arpita Ghosh, Hamid Nazerzadeh and Prabhakar Raghavan	<b>52</b>
<b>120</b>	<b>Generating Random Graphs with Large Girth</b> Mohsen Bayati, Andrea Montanari and Amin Saberi	<b>53</b>
<b>121</b>	<b>Improved Approximation Algorithms for Scheduling with Fixed Jobs</b> Florian Diedrich and Klaus Jansen	<b>53</b>
<b>122</b>	<b>Combinatorial Algorithms for Wireless Information Flow</b> Aurore Amaudruz and Christina Fragouli	<b>53</b>
<b>123</b>	<b>Fast Algorithms for (Max,Min)-Matrix Multiplication and Bottleneck Shortest Paths</b> Ran Duan and Seth Pettie	<b>54</b>
<b>124</b>	<b>Appointment Scheduling with Discrete Random Durations</b> Mehmet Begen and Maurice Queyranne	<b>54</b>
<b>125</b>	<b>Expanders via Random Spanning Trees</b> Navin Goyal, Luis Rademacher and Santosh Vempala	<b>54</b>
<b>126</b>	<b>Loopless Generation of Multiset Permutations using a Constant Number of Variables by Prefix Shifts</b> Aaron Williams	<b>55</b>
<b>127</b>	<b>Dual-Failure Distance and Connectivity Oracles</b> Ran Duan and Seth Pettie	<b>55</b>
<b>128</b>	<b>Sequential cavity method for computing limits of the log-partition function for david gamarnik and Dimitry Katz</b>	<b>55</b>
<b>129</b>	<b>Lower Bounds for Succinct Data Structures</b> Alexander Golynski	<b>56</b>
<b>130</b>	<b>Equilibria of Collusive Flow Games are not Unique</b> Umang Bhaskar, Lisa Fleischer, Darrell Hoy and Chien-Chung Huang	<b>56</b>
<b>131</b>	<b>Sampling Biased Lattice Configurations using Exponential Metrics</b> Sam Greenberg, Amanda Pascoe and Dana Randall	<b>56</b>
<b>132</b>	<b>Size Complexity of Volume Meshes vs Surface Meshes</b> Benot Hudson, Gary Miller, Todd Phillips and Don Sheehy	<b>56</b>
<b>133</b>	<b>Combinatorial Algorithms for Nearest Neighbors, Near-Duplicates and Small-World Design</b> Yury Lifshits and Shengyu Zhang	<b>57</b>

## 1 Scalably Scheduling Processes with Arbitrary Speedup Curves

We give a scalable  $((1 + \epsilon)$ -speed  $O(1)$ -competitive) nonclairvoyant algorithm for scheduling jobs with sublinear nondecreasing speed-up curves on multiple processors with the objective of average response time.

## 2 Three-coloring triangle-free planar graphs in linear time

Grötzsch’s theorem states that every triangle-free planar graph is 3-colorable, and several relatively simple proofs of this fact were provided by Thomassen and other authors. It is easy to convert these proofs into quadratic-time algorithms to find a 3-coloring, but it is not clear how to find such a coloring in linear time (Kowalik used a nontrivial data structure to construct an  $O(n \log n)$  algorithm). We design a linear-time algorithm to find a 3-coloring of a given triangle-free planar graph. The algorithm avoids using any complex data structures, which makes it easy to implement. As a by-product we give another simple proof of Grötzsch’s theorem.

## 3 An Efficient Sparse Regularity Concept

Let  $\mathcal{A}$  be a 0/1 matrix of size  $m \times n$ , and let  $p$  be the density of  $\mathcal{A}$  (i.e., the number of ones divided by  $m \cdot n$ ). We show that  $\mathcal{A}$  can be approximated in the cut norm within  $\epsilon \cdot mnp$  by a sum of cut matrices (of rank 1), where the number of summands is independent of the size  $m \cdot n$  of  $\mathcal{A}$ , provided that  $\mathcal{A}$  satisfies a certain boundedness condition. This decomposition can be computed in polynomial time. This result extends the work of Frieze and Kannan [?] to *sparse* matrices. As an application, we obtain efficient  $1 - \epsilon$  approximation algorithms for “bounded” instances of MAX CSP problems.

## 4 Exponential Lower Bounds and Integrality Gaps for Lovasz-Schrijver Procedures

The matrix cuts of Lovasz and Shrijver are methods for tightening linear relaxations of zero-one programs by the addition of new linear inequalities. We address the question of how many new inequalities are necessary to approximate certain combinatorial problems and to solve certain instances of Boolean satisfiability. Our first result is a size/rank tradeoff for tree-like LS+ refutations, showing that any refutation that has small size also has small rank. This allows us to immediately derive exponential size lower bounds for tree-like LS+ derivations of many unsatisfiable systems of inequalities where prior to our work, only rank lower bounds were known. Unfortunately we show that this tradeoff does not hold more generally for derivations of arbitrary inequalities. We give a simple example showing that derivations can be very small but nonetheless require maximal rank. This rules out a generic argument for obtaining size-based integrality gaps from rank-based gaps. Our second contribution is to show that a modified argument can often be used to prove size-based gaps from rank-based gaps. We illustrate this method by proving size-based integrality gaps for several prominent problems, where again prior to our work only rank-based integrality gaps were known. Our third contribution is to prove new separation results between proof systems. Using our machinery for converting rank-based lower bounds and integrality gaps into size-based lower bounds, we show that tree-like LS+ cannot polynomially simulate tree-like Cutting Planes. We also show that tree-like LS+ cannot simulate Resolution, and in particular, low rank LS+ cannot simulate Resolution. We conclude by examining size/rank tradeoffs behind the LS systems. We show that

for Sherali-Adams and Lasserre systems, size/rank tradeoffs continue to hold, even in the general (non-tree) case.

## 5 On the power of two, three and four probes

An adaptive  $(n, m, s, t)$ -scheme is a deterministic scheme for encoding a vector  $X$  of  $m$  bits with at most  $n$  ones by a vector  $Y$  of  $s$  bits, so that any bit of  $X$  can be determined by  $t$  adaptive probes to  $Y$ . A non-adaptive  $(n, m, s, t)$ -scheme is defined analogously. The study of such schemes arises in the investigation of the static membership problem in the bitprobe model. Answering a question of Buhrman, Miltersen, Radhakrishnan and Venkatesh [SICOMP 2002] we present adaptive  $(n, m, s, 2)$  schemes with  $s < m$  for all  $n$  satisfying  $4n^2 + 4n < m$  and adaptive  $(n, m, s, 2)$  schemes with  $s = o(m)$  for all  $n = o(\log m)$ . We further show that there are adaptive  $(n, m, s, 3)$ -schemes with  $s = o(m)$  for all  $n = o(m)$ , settling a problem of Radhakrishnan, Raman and Rao [ESA 2001], and prove that there are non-adaptive  $(n, m, s, 4)$ -schemes with  $s = o(m)$  for all  $n = o(m)$ . Therefore, three adaptive probes or four non-adaptive probes already suffice to obtain a significant saving in space compared to the total length of the input vector. Lower bounds are discussed as well.

## 6 Dimension detection via slivers

We present a method to estimate the manifold dimension by analyzing the shape of simplices formed by point samples in some small neighborhoods. Approximate tangent spaces at these neighborhoods can also be reported. Let  $P$  be a set of point samples drawn from a manifold  $\mathcal{M} \subset \mathbb{R}^d$  with dimension  $m$  according to a Poisson distribution with parameter  $\lambda$ . Both  $\mathcal{M}$  and  $\lambda$  are unknown to our algorithm. For sufficiently large  $\lambda$ , given any integer  $k \geq 1$ , our algorithm correctly outputs  $m$  in  $O(kdm^3|P|\log|P|)$  time with probability  $1 - O(\lambda^{-k\beta})$  for some constant  $\beta \in (0, 1)$ . It holds with the same probability that the angular error of each approximate tangent space reported is  $O(\epsilon)$ , where  $\epsilon$  is a value depending on  $\mathcal{M}$ ,  $\lambda$  and  $m$  and  $\lim_{\lambda \rightarrow \infty} \epsilon = 0$ . We experimented with a practical variant of our algorithm and demonstrated that it does not require very high sampling density and it is competitive with several previous methods.

## 7 On the Approximability of Dodgson and Young Elections

The voting rules proposed by Dodgson and Young are both designed to find the alternative closest to being a Condorcet winner, according to two different notions of proximity; the score of a given alternative is known to be hard to compute under both rules. In this paper, we put forward two algorithms for approximating the Dodgson score, an LP-based randomized rounding algorithm and a deterministic greedy algorithm, both of which yield an  $O(\log m)$  approximation ratio, where  $m$  is the number of alternatives; we observe that this result is asymptotically optimal, and further prove that our greedy algorithm is optimal up to a factor of 2, unless problems in NP have quasi-polynomial time algorithms. Although the greedy algorithm is computationally superior, we argue that the randomized rounding algorithm has an advantage from a social choice point of view. Further, we demonstrate that computing any reasonable approximation of the ranking produced by Dodgson's rule is NP-hard. This result provides a complexity-theoretic explanation of sharp discrepancies that have been observed in the Social Choice Theory literature when comparing Dodgson elections with simpler voting rules. Finally, we show that the problem of calculating the Young score is NP-hard to approximate by any factor. This leads to an inapproximability result for the Young ranking.

## 8 Self-overlapping Curves Revisited

A surface embedded in space in such a way that each point has a neighborhood within which the surface is a terrain projects to an immersed surface in the plane, the boundary of which is a self-intersecting curve. Under what circumstances can we reverse these mappings algorithmically? Shor and van Wyk considered one such problem, determining whether a curve is the boundary of an immersed disk; they showed that the self-overlapping curves defined in this way can be recognized in polynomial time. We show that several related problems are more difficult: it is NP-complete to determine whether an immersed disk is the projection of a surface embedded in space, or whether a curve is the boundary of an immersed surface in the plane that is not constrained to be a disk. However, when a casing is supplied with a self-intersecting curve, describing which component of the curve lies above and which below at each crossing, we may determine in time linear in the number of crossings whether the cased curve forms the projected boundary of a surface in space. As a related result, we show that an immersed surface with a single boundary curve that crosses itself  $n$  times has at most  $2^{n/2}$  combinatorially distinct spatial embeddings, and we discuss the existence of fixed-parameter tractable algorithms for related problems.

## 9 Hardness of embedding simplicial complexes in $\mathbb{R}^d$

Let  $\text{EMBED}(k,d)$  be the following algorithmic problem: Given a finite  $k$ -dimensional simplicial complex  $K$ , does there exist a (piecewise linear) embedding of  $K$  into  $\mathbb{R}^d$ ? Known results easily imply polynomiality of  $\text{EMBED}(k,2)$  ( $k = 1, 2$ ; the case  $k = 1, d = 2$  is graph planarity) and of  $\text{EMBED}(k,2k)$  for all  $k > 2$  (even if  $k$  is not considered fixed). We observe that the celebrated result of Novikov on the algorithmic unsolvability of recognizing the 5-sphere implies that  $\text{EMBED}(d,d)$  and  $\text{EMBED}(d-1,d)$  are undecidable for any  $d > 4$ . Our main result is NP-hardness of  $\text{EMBED}(2,4)$  and, more generally, of  $\text{EMBED}(k,d)$  for all  $k$  and  $d$  with  $d > 3$  and  $d \geq k \geq (2d - 2)/3$ . These dimensions fall outside the so-called metastable range of a theorem of Haefliger and Weber, which characterizes embeddability using the deleted product obstruction. Our reductions are based on examples, due to Segal, Spiez, Freedman, Krushkal, Teichner, and Skopenkov, showing that outside the metastable range the deleted product obstruction is not sufficient to characterize embeddability.

## 10 The Assignment Problem in Content Distribution Networks: Unsplittable Hard-Capacitated Facility Location

In a Content Distribution Network (CDN), there are  $m$  servers storing the data; each of them has a specific bandwidth. All the requests from a particular client should be assigned to one server, because of the routing protocols used. The goal is to minimize the total cost of these assignments — which is proportional to the distance as well as the request size — while the load on each server is kept below its bandwidth constraint. When each server also has a setup cost, this is an *unsplittable hard-capacitated facility location problem*. As much attention as facility location problems have received, there has been no nontrivial approximation algorithm when we have hard capacities (i.e., there can only be one copy of each facility whose capacity cannot be violated) and demands are unsplittable (i.e., all the demand from a client has to be assigned to a single facility). We observe it is NP-hard to approximate the cost to within any bounded factor. Thus, we relax the capacities to a  $1 + \epsilon$  factor. For the case where capacities are almost uniform, we give a bicriteria  $O(\log n, 1 + \epsilon)$ -approximation algorithm for general metrics and a  $(1 + \epsilon, 1 + \epsilon)$ -approximation algorithm for tree metrics. We can get the same guarantee for non-uniform capacities if we allow quasipolynomial running time. A bicriteria  $(\alpha, \beta)$ -approximation algorithm produces a solution of cost at most  $\alpha$  times

the optimum, while violating the capacities by no more than a  $\beta$  factor. In our algorithm, some clients guess the facility they are assigned to, and facilities decide the size of clients they serve. A straight-forward approach results in exponential running time. Handling this in (quasi-)polynomial time is also of independent interest. When costs do not satisfy metricity, we show that a 1.5 violation of capacities is necessary to obtain any approximation. It is worth noting that our results generalizes bin packing (zero cost matrix and facility costs equal to one), knapsack (single facility with all costs being zero), minimum makespan scheduling for related machines (all costs being zero) and facility location problems.

## 11 Approximation Algorithms for Restless Bandit Problems

In this paper, we consider the restless bandit problem, which is one of the most well-studied generalizations of the celebrated stochastic multi-armed bandit problem in decision theory. In its ultimate generality, the restless bandit problem is known to be PSPACE-Hard to approximate to any non-trivial factor, and little progress has been made on this problem despite its significance in modeling activity allocation under uncertainty. We make progress on this problem by showing that for an interesting and general subclass that we term Monotone bandits, a surprisingly simple and intuitive greedy policy yields a factor 2 approximation. Such greedy policies are termed index policies, and are popular due to their simplicity and their optimality for the stochastic multi-armed bandit problem. The Monotone bandit problem strictly generalizes the stochastic multi-armed bandit problem, and naturally models multi-project scheduling where the state of a project becomes increasingly uncertain when the project is not scheduled. As a consequence we are able to significantly improve previous results on special cases, as well as provide algorithms for generalizations. It is worth noting that the previous algorithm(s) do not apply to Monotone bandits and a new analysis was essential to this progress. The notion of Monotone bandits is similar to (but not quite the same as) the notion of a Partially Observable Markov Decision Process (POMDP). This and related restless bandit problems serve as models for wireless scheduling, unmanned aerial vehicle (UAV) routing, and machine replenishment, among others. We develop several novel techniques in the design and analysis of the index policy. Our algorithm proceeds by introducing a novel "balance" constraint to the dual of a well-known LP relaxation to the restless bandit problem. This is followed by a structural characterization of the optimal solution by using both the exact primal as well as dual complementary slackness conditions. This yields an interpretation of the dual variables as potential functions from which we derive the index policy and the associated analysis. In addition, our techniques are general and of independent interest in other stochastic optimization contexts, and we provide one such example.

## 12 Biased Range Trees

A data structure, called a *biased range tree*, is presented that preprocesses a set  $S$  of  $n$  points in  $\mathbb{R}^2$  and a query distribution  $D$  for 2-sided orthogonal range counting queries. The expected query time for this data structure, when queries are drawn according to  $D$ , matches, to within a constant factor, that of the optimal comparison tree for  $S$  and  $D$ . The memory and preprocessing requirements of the data structure are  $O(n \log n)$ .

## 13 An almost $O(\log k)$ -approximation for $k$ -connected subgraphs

We give an  $O(\log k \cdot \log \frac{n}{n-k})$ -approximation algorithm for the  $k$ -Connected Subgraph problem, that seeks to find a directed/undirected  $k$ -connected spanning subgraph of minimum cost. Our ratio is

$O(\log k)$ , unless  $k = n - o(n)$ . Previously, the best known approximation guarantees for this problem were  $O(\log^2 k)$  for directed/undirected graphs [Fakcharoenphol and Laekhanukit, STOC 2008], and  $O(\log k)$  for undirected graphs with  $k \leq \sqrt{n/2}$  [Cheriyān, Vempala, and Vetta, STOC 2002]. As in previous work, we consider the augmentation problem of increasing at minimum cost the connectivity of a given graph  $J$  by 1 (a  $\rho$ -approximation for it is used to derive an  $O(\rho \cdot \log k)$ -approximation for  $k$ -Connected Subgraph). Fakcharoenphol and Laekhanukit showed that this augmentation problem admits an  $O(\log t)$ -approximation algorithm, where  $t$  is the number of minimal "violated" sets in  $J$ . However, we may have  $t = \Theta(n)$ , so this gives only an  $O(\log n)$ -approximation. We show that a modification of the primal-dual algorithm of Ravi and Williamson can be used to add an edge set of cost  $\leq \text{OPT}$  to get  $t \leq \frac{2n}{n-k}$ . Combined with the algorithm of Fakcharoenphol and Laekhanukit, this gives the ratio  $O(\log \frac{n}{n-k})$  for the augmentation problem, which is  $O(1)$ , unless  $k = n - o(n)$ .

## 14 Linear-Time Algorithms for Geometric Graphs with Sublinearly Many Crossings

We provide linear-time algorithms for geometric graphs with sublinearly many crossings. That is, we provide algorithms running in  $O(n)$  time on connected geometric graphs having  $n$  vertices and  $k$  crossings, where  $k$  is smaller than  $n$  by an iterated logarithmic factor. Specific problems we study include Voronoi diagrams and single-source shortest paths. Our algorithms all run in linear time in the standard comparison-based computational model; hence, we make no assumptions about the distribution or bit complexities of edge weights, nor do we utilize unusual bit-level operations on memory words. Instead, our algorithms are based on a *planarization* method that "zeroes in" on edge crossings, together with methods for extending planar separator decompositions to geometric graphs with sublinearly many crossings. Incidentally, our planarization algorithm also solves an open computational geometry problem of Chazelle for triangulating a self-intersecting polygonal chain having  $n$  segments and  $k$  crossings in linear time, for the case when  $k$  is sublinear in  $n$  by an iterated logarithmic factor.

## 15 Improved Approximating Algorithms for Directed Steiner Forest

We consider the  $k$ -Directed Steiner Forest ( $k$ -DSF) problem: given a directed graph  $G = (V, E)$  with edge costs, a collection  $D \subseteq V \times V$  of ordered node pairs, and an integer  $k \leq |D|$ , find a minimum cost subgraph  $H$  of  $G$  that contains an  $st$ -path for (at least)  $k$  pairs  $(s, t) \in D$ . When  $k = |D|$ , we get the Directed Steiner Forest (DSF) problem. The best known approximation ratios for these problems are:  $\tilde{O}(k^{2/3})$  for  $k$ -DSF by Charikar et al. [?], and  $O(k^{1/2+\epsilon})$  for DSF by Chekuri et al. [?]. We improve these approximation ratios as follows. For DSF we give an  $O(n^\epsilon \cdot \min\{n^{4/5}, m^{2/3}\})$ -approximation scheme using a novel LP-relaxation that seeks to connect pairs with "cheap" paths. This is the first sub-linear (in terms of  $n = |V|$ ) approximation ratio for the problem; all previous algorithm had ratio  $\Omega(n^{1+\epsilon})$ . For  $k$ -DSF we give a simple greedy  $O(k^{1/2+\epsilon})$ -approximation algorithm. This improves the best known ratio  $\tilde{O}(k^{2/3})$  by Charikar et al. [?], and (almost) matches in terms of  $k$  the best ratio known for the undirected variant [?]. Even when used for the particular case of DSF, our algorithm favorably compares to the one of [?], which repeatedly solves linear programs and uses complex space and time consuming transformations. Our algorithm is much simpler and faster, since it essentially reduces  $k$ -DSF to a variant of the Directed Steiner Tree problem. The simplification is due to a new notion of "junction star-tree" – a union of an in-star and an out-branching having the same root, which is of independent interest.

## 16 Line Transversals of Convex Polyhedra in $\mathbb{R}^3$

We establish a bound of  $O(n^2k^{1+\epsilon})$ , for any  $\epsilon > 0$ , on the combinatorial complexity of the set  $\mathcal{T}$  of line transversals of a collection  $\mathfrak{P}$  of  $k$  convex polyhedra in  $\mathbb{R}^3$  with a total of  $n$  facets, and present a randomized algorithm which computes the boundary of  $\mathcal{T}$  in comparable expected time. Thus, when  $k \ll n$ , the new bounds on the complexity (and construction cost) of  $\mathcal{T}$  improve upon the previously best known bounds, which are nearly cubic in  $n$ . To obtain the above result, we study the set  $\mathcal{TL}$  of line transversals which emanate from a fixed line  $\ell_0$ , establish an almost tight bound of  $O(nk^{1+\epsilon})$  on the complexity of  $\mathcal{TL}$ , and provide a randomized algorithm which computes  $\mathcal{TL}$  in comparable expected time. Slightly improved combinatorial bounds for the complexity of  $\mathcal{TL}$ , and comparable improvements in the cost of constructing this set, are established for two special cases, both assuming that the polyhedra of  $\mathfrak{P}$  are pairwise disjoint: the case where  $\ell_0$  is disjoint from the polyhedra of  $\mathfrak{P}$ , and the case where the polyhedra of  $\mathfrak{P}$  are unbounded in a direction parallel to  $\ell_0$ .

## 17 String Hashing for Linear Probing

Hash tables for strings and other complex objects are central to the analysis of data, and they are directly built into high level programming languages such as python, and perl. Linear probing is one of the most popular implementations of hash tables in practice. We store all keys in a single array. A new key is hashed. If a different key is in the position hashed to, we scan next positions sequentially until either the key is found, or we hit an empty spot concluding that the key is new. This is compact and it is fast because we typically get all positions considered in single cache-line, and that also holds for deletions. However, linear probing has been known to be a bit unreliable in practice. Analyzing the performance of linear probing with universal hash functions was raised as an open problem by Carter and Wegman in 1979. At STOC'07, Pagh et al. studied the expected number of linear probes with worst-case data sets. They showed that with the standard implementation of 2-universal hashing, the expected number of probes could be  $\Omega(\log n)$ . A worst-case is one or two intervals — something that could very well appear in practice, possibly explaining the experienced unreliability. On the positive side, they showed that with 5-universal hashing, the expected number of probes is constant. Unfortunately, we do not have 5-universal hashing for, say, variable length strings. When we want to do such complicated hashing on a complicated domain, the generic standard solution is that we first do collision free hashing (w.h.p.) into a simpler intermediate domain, and second do the complicated hash function on this intermediate domain. Our contribution is that for an expected constant number of linear probes, it suffices that each key has  $O(1)$  expected collisions with the first hash function, as long as the second hash function is 5-universal. This means that the intermediate domain can be  $n$  times smaller. Such a smaller intermediate domain typically means that the both the first and the second hash function run at least twice as fast. This doubling of hashing speed follows not just for variable length strings, but for most domains bigger than 32-bit integers, e.g., fixed length strings and even 64-bit integers.

## 18 Improved Approximation Bound for Quadratic Optimization Problems with Orthogonality Constraints

In this paper we consider approximation algorithms for a class of quadratic optimization problems that contain orthogonality constraints, i.e. constraints of the form  $X^T X = I$ , where  $X \in \mathbb{R}^{m \times n}$  is the optimization variable. This class of problems, which we denote by (QP-OC), is quite general and captures several well-studied problems in the literature as special cases. In a recent work, Nemirovski [Math. Prog. 109:283–317, 2007] gave the first non-trivial approximation algorithm for

(QP-OC). His algorithm is based on semidefinite programming and has an approximation guarantee of  $O((m+n)^{1/3})$ . We improve upon this result by providing the first logarithmic approximation guarantee for (QP-OC). Specifically, we show that (QP-OC) can be approximated to within a factor of  $O(\ln(\max\{m, n\}))$ . The main technical tool used in the analysis is a concentration inequality for the spectral norm of a Rademacher sum of matrices, which we develop using tools from functional analysis. As a by-product, we resolve in the affirmative a conjecture of Nemirovski concerning the typical spectral norm of a sum of certain random matrices. The aforementioned concentration inequality also has ramifications in the design of so-called safe tractable approximations of chance constrained optimization problems. In particular, we use it to simplify and improve a recent result of Ben-Tal and Nemirovski [Manuscript, 2007] concerning certain chance constrained linear matrix inequality systems.

## 19 Algorithms for Finding an Induced Cycle in Planar Graphs

In this paper, we consider the problem for finding an induced cycle passing through  $k$  given vertices, which we call the induced cycle problem. The significance of finding induced cycles stems from the fact that precise characterization of perfect graphs would require structures of graphs without an odd induced cycle, and its complement. There has been huge progress in the recent years, especially, the Strong Perfect Graph Conjecture was solved by Chudnovsky, Robertson, Seymour and Thomas. Concerning recognition of perfect graphs, there had been a long-standing open problem for detecting an odd hole and its complement, and finally this was solved by Chudnovsky, Cornuéjols, Liu, Seymour and Vuskovic. Unfortunately, the problem of finding an induced cycle passing through two given vertices is NP-complete in a general graph. However, if the input graph is constrained to be planar and  $k$  is fixed, then the induced cycle problem can be solved in polynomial time. In particular, an  $O(n^2)$  time algorithm is given for the case  $k = 2$  by McDiarmid, Reed, Schrijver and Shepherd, where  $n$  is the number of vertices of the input graph. Our main results in this paper are to improve their result in the following sense. (1) The number of vertices  $k$  is allowed to be non-trivially super constant number, up to  $k = o((\frac{\log n}{\log \log n})^{\frac{2}{3}})$ . More precisely, when  $k = o((\frac{\log n}{\log \log n})^{\frac{2}{3}})$ , then the ICP in planar graphs can be solved in  $O(n^{2+\epsilon})$  time for any  $\epsilon > 0$ . (2) The time complexity is linear if the given graph is planar and  $k$  is fixed. (3) The above results are extended to graphs embedded in a fixed surface. We note that the linear time algorithm (the second result) is independent from the first result. Let us observe that if  $k$  is as a part of the input, then the problem is still NP-complete. So we need to impose some condition on  $k$ . Our proofs of the above results basically follow the same line of the proof of the disjoint paths problem by Robertson and Seymour, together with Reed, Robertson, Schrijver, Seymour's result, which improves the time complexity of the algorithm of the disjoint paths by Robertson and Seymour ( $O(n^3)$  time algorithm) to linear time when an input graph is planar. However, our cycle must be induced, so some of arguments must be extended to induced paths, which needs much more involved arguments. In addition, we are also interested in the case when  $k$  is as a part of the input. Therefore, we need to sharpen the function of  $k$ . This needs nontrivial amount of work, since both Robertson-Seymour's proof and Reed, Robertson, Schrijver, Seymour's proof do not care much about sharpening the hidden constant of  $k$ . These proofs just guarantee the existence of the function of  $k$ , therefore, it is highly expensive, and nonpractical. Our proofs, though, give fairly small function of  $k$ . Therefore, we believe that this result may be viewed as a much more practical result. Price to pay is to need to analyze the structure of planar graphs and graphs embedded in a fixed surface more closely.

## 20 A quadratic kernel for feedback vertex set

We prove that given an undirected graph  $G$  on  $n$  vertices and an integer  $k$ , one can compute in polynomial time in  $n$  a graph  $G'$  with at most  $5k^2 + k$  vertices and an integer  $k'$  such that  $G$  has a feedback vertex set of size at most  $k$  iff  $G'$  has a feedback vertex set of size at most  $k'$ . This result improves a previous  $O(k^{11})$  kernel of Burrage et al. [?], and a more recent cubic kernel of Bodlaender [?]. This problem was communicated by Fellows in [?].

## 21 On risks of using cuckoo hashing with simple universal hash classes

Background: Cuckoo hashing, introduced by Pagh and Rodler (“Cuckoo hashing”, ESA 2001 and J. Algorithms 51, 2004), is a strategy for maintaining hash tables for keys from  $\mathcal{U}$ ,  $|\mathcal{U}| = N$ , so that lookups take constant time in the worst case. The data structure consists of two tables of size  $m$  each, and it uses two hash functions  $h_1, h_2: \mathcal{U} \rightarrow [m]$ . For the scheme to work it is necessary and sufficient that  $m \geq (1 + \epsilon)n$  for an arbitrary constant  $\epsilon > 0$ , where  $n$  is the number of keys stored. Pagh and Rodler’s analysis establishes expected amortized constant time for insertion; for the analysis to work it is required that the hash functions are  $c \log n$ -wise independent. In experiments, cuckoo hashing works very well with weaker hash function classes. However, Pagh and Rodler report on experimental results that indicate that cuckoo hashing might not work well in combination with the “multiplicative class” (which consists of functions  $h_a: [2^k] \rightarrow [2^l]$  of the form  $h_a(x) = ((\alpha x) \bmod 2^k) \div 2^{k-l}$ , for  $0 < \alpha < 2^k$  odd, and is close to “2-universal”). They state that they do not have an explanation for this phenomenon. In 2008, Mitzenmacher and Vadhan (“Why simple hash functions work: exploiting the entropy in a data stream”, SODA 2008) proved that if a 2-universal hash class  $H$  is used and the key set exhibits a certain degree of randomness, and further technical conditions are fulfilled, then the combination of the key set and a hash function chosen at random from  $H$  will behave very close to full randomness, in a technical sense. Our results: 1) We show that if cuckoo hashing with  $2^l = m = 2n$  is employed (this table size is way above the threshold sufficient for the standard analysis), then all function pairs from the multiplicative hash class will work badly with high probability for fully random key sets of size  $n$ , if  $n/N > N^{1-\gamma}$  for some constant  $\gamma > 0$ . On the one hand, this explains the experimental results obtained by Pagh and Rodler and justifies a warning against using this simple class, on the other hand, it shows that one of the “further technical conditions” of the Mitzenmacher/Vadhan result, namely the requirement that key sets must be relatively sparse in  $\mathcal{U}$ , is really necessary for their result to be true. 2) We show that cuckoo hashing with a standard almost 2-wise independent class of hash functions (functions of the form  $h_{a,b} = ((\alpha x + b) \bmod p) \bmod m$ ,  $p \geq |\mathcal{U}|$  a prime number) exhibits a similar behavior as the class in 1), again in the case where the key set is relatively dense in  $\mathcal{U}$ . This is true even in the case when the two hash functions use different prime moduli. Further related work: Cohen and Kane (“Bounds on the independence required for cuckoo hashing”, unpublished manuscript, 2005, [http://web.mit.edu/dankane/www/Independence2-, 3-, and even 5-wise independent hash families](http://web.mit.edu/dankane/www/Independence2-,3-,and%20even%205-wise%20independent%20hash%20families), for which cuckoo hashing has high failure probability. However, these families are quite contrived and far from being common.

## 22 Asymptotically optimal frugal colouring

We prove that every graph with maximum degree  $D$  can be properly  $(D+1)$ -coloured so that no colour appears more than  $O(\log D / \log \log D)$  times in the neighbourhood of any vertex. This is best possible up to the constant factor in the  $O(-)$  term. We also provide an efficient algorithm to produce such a colouring.

## 23 A Unified Approach to Distance-Two Colouring of Planar Graphs

In this paper we introduce the notion of  $(A, B)$ -colouring of a graph: For given vertex sets  $A, B$ , this is a colouring of the vertices in  $B$  so that both adjacent vertices and vertices with a common neighbour in  $A$  receive different colours. This concept generalises the notion of colouring the square of graphs and of cyclic colouring of plane graphs. We prove a general result which implies asymptotic versions of Wegner’s and Borodin’s Conjecture on these two colourings. Using a recent approach of Havet *et al.*, we reduce the problem to edge-colouring of multigraphs and then use Kahn’s result that the list chromatic index is close from the fractional chromatic index. Our results are based on a strong structural lemma for planar graphs which also implies that the size of a clique in the square of a planar graph of maximum degree  $\Delta$  is at most  $\frac{3}{2}\Delta$  plus a constant.

## 24 Weighted flow time does not admit $O(1)$ -competitive algorithms

We consider the classic online scheduling problem of minimizing the total weighted flow time on a single machine with preemptions. Here, each job  $j$  has an arbitrary arrival time  $r_j$ , weight  $w_j$  and size  $p_j$ , and given a schedule its flow time is defined as the duration of time since its arrival until it completes its service requirement. The first non-trivial algorithms with poly-logarithmic competitive ratio for this problem were obtained relatively recently, and it was widely believed that the problem admits a constant factor competitive algorithm. In this paper, we show an  $\omega(1)$  lower bound on the competitive ratio of any deterministic online algorithm. Our result is based on a novel gap amplification technique for online algorithms. Starting with a trivial lower bound of 1, we give a procedure to improve the lower bound sequentially, while ensuring at each step that the size of the instance increases relatively modestly.

## 25 On the bit-complexity of Lempel-Ziv compression

One of the most famous and investigated lossless data-compression schemes is the one introduced by Lempel and Ziv about 30 years ago. This compression scheme is known as “dictionary-based compressor” and consists of squeezing an input string by replacing some of its substrings with (shorter) codewords which are actually pointers to a dictionary of phrases built as the string is processed. Surprisingly enough, although many fundamental results are nowadays known about the speed and effectiveness of this compression process, “we are not aware of any parsing scheme that achieves optimality when the LZ-dictionary is in use under any constraint on the codewords other than being of equal length” [N. Rajpoot and C. Sahinalp. Handbook of Lossless Data Compression, chapter Dictionary-based data compression. Academic Press, 2002. pag. 159]. Here optimality means to achieve the minimum number of bits in compressing each individual input string, without any assumption on its generating source. In this paper we investigate three issues pertaining to the bit-complexity of LZ-based compressors, and we design algorithms which achieve bit-optimality in the compressed output size by taking efficient/optimal time and optimal space. These theoretical results will be sustained by some experiments that will compare our novel LZ-based compressors against the most popular compression tools (like Gzip, Bzip) and state-of-the-art compressors (like the Compression Booster [Ferragina-Giancarlo-Manzini-Sciortino, J.ACM 52(4), 2005]).

## 26 Testing Halfspaces

This paper addresses the problem of testing whether a Boolean-valued function  $f$  is a halfspace, i.e. a function of the form  $f(x) = \text{sgn}(w \cdot x - \theta)$ . We consider halfspaces over the continuous domain

$\mathbb{R}^n$  (endowed with the standard multivariate Gaussian distribution) as well as halfspaces over the Boolean cube  $\{-1, 1\}^n$  (endowed with the uniform distribution). In both cases we give an algorithm that distinguishes halfspaces from functions that are  $\epsilon$ -far from any halfspace using only  $\text{poly}(\frac{1}{\epsilon})$  queries, independent of the dimension  $n$ . Two simple structural results about halfspaces are at the heart of our approach for the Gaussian distribution: the first gives an exact relationship between the expected value of a halfspace  $f$  and the sum of the squares of  $f$ 's degree-1 Hermite coefficients, and the second shows that any function that approximately satisfies this relationship is close to a halfspace. We prove analogous results for the Boolean cube  $\{-1, 1\}^n$  (with Fourier coefficients in place of Hermite coefficients) for balanced halfspaces in which all degree-1 Fourier coefficients are small. Dealing with general halfspaces over  $\{0, 1\}^n$  poses significant additional complications and requires other ingredients. These include “cross-consistency” versions of the results mentioned above for pairs of halfspaces with the same weights but different thresholds; new structural results relating the largest degree-1 Fourier coefficient and the largest weight in unbalanced halfspaces; and algorithmic techniques from recent work on testing juntas [?].

## 27 Multi-Dimensional Online Tracking

We propose and study a new class of online problems, which we call *online tracking*. Suppose an *observer*, say Alice, observes a multi-valued function  $f : \mathbb{Z}^+ \rightarrow \mathbb{Z}^d$  over time in an online fashion, i.e., she only sees  $f(t)$  for  $t \leq t_{\text{now}}$  where  $t_{\text{now}}$  is the current time. She would like to keep a *tracker*, say Bob, informed of the current value of  $f$  at all times. Under this setting, Alice could send new values of  $f$  to Bob from time to time, so that the current value of  $f$  is always within a distance of  $\Delta$  to the last value received by Bob. We give competitive online algorithms whose communication costs are compared with the optimal offline algorithm that knows the entire  $f$  in advance. We also consider variations of the problem where Alice is allowed to send “predictions” to Bob, to further reduce communication for well-behaved functions. These online tracking problems have a variety of application ranging from sensor monitoring, location-based services, to publish/subscribe systems.

## 28 Termination Criteria for Solving Concurrent Safety and Reachability Games

We consider concurrent games played on graphs. At every round of a game, each player simultaneously and independently selects a move; the moves jointly determine the transition to a successor state. Two basic objectives are the safety objective to stay forever in a given set of states, and its dual, the reachability objective to reach a given set of states. We present in this paper a strategy improvement algorithm for computing the value of a concurrent safety game, that is, the maximal probability with which player 1 can enforce the safety objective. The algorithm yields a sequence of player-1 strategies which ensure probabilities of winning that converge monotonically to the value of the safety game. Our result is significant because the strategy improvement algorithm provides, for the first time, a way to approximate the value of a concurrent safety game from below. Since a value iteration algorithm, or a strategy improvement algorithm for reachability games, can be used to approximate the same value from above, the combination of both algorithms yields a method for computing a converging sequence of upper and lower bounds for the values of concurrent reachability and safety games. Previous methods could approximate the values of these games only from one direction, and as no rates of convergence are known, they did not provide a practical way to solve these games.

## 29 Shortest Paths in Directed Planar Graphs with Negative Lengths: a linear-space $O(n \log^2 n)$ -time algorithm

We give an  $O(n \log^2 n)$ -time, linear-space algorithm that, given a directed planar graph with positive and negative arc-lengths and no negative-length cycles, and given a node  $s$ , finds the distances from  $s$  to all nodes. The best previously known algorithm requires  $O(n \log^3 n)$  time and  $O(n \log n)$  space.

## 30 Efficient coordination mechanisms for unrelated machine scheduling

We present three new coordination mechanisms for scheduling  $n$  selfish jobs on  $m$  unrelated machines. A coordination mechanism aims to mitigate the impact of selfishness of jobs on the efficiency of schedules by defining a local scheduling policy on each machine. The scheduling policies induce a game among the jobs and each job prefers to be scheduled on a machine so that its completion time is minimum given the assignments of the other jobs. We consider the maximum completion time among all jobs as the measure of the efficiency of schedules. The approximation ratio of a coordination mechanism quantifies the efficiency of pure Nash equilibria (price of anarchy) of the induced game. Our mechanisms are deterministic, local, and preemptive in the sense that the scheduling policy does not necessarily process the jobs in an uninterrupted way and may introduce some idle time. Our first coordination mechanism has approximation ratio  $O(\log m)$  and always guarantees that the induced game has pure Nash equilibria to which the system converges in at most  $n$  rounds. This result improves a recent bound of  $O(\log^2 m)$  due to Azar, Jain, and Mirrokni and, similarly to their mechanism, our mechanism uses a global ordering of the jobs according to their distinct IDs. Next we study the intriguing scenario where jobs are anonymous, i.e., they have no IDs. In this case, coordination mechanisms can only distinguish between jobs that have different load characteristics. Our second mechanism handles anonymous jobs and has approximation ratio  $O\left(\frac{\log m}{\log \log m}\right)$  although the game induced is not a potential game and, hence, the existence of pure Nash equilibria is not guaranteed by potential function arguments. However, it provides evidence that the known lower bounds for non-preemptive coordination mechanisms could be beaten using preemptive scheduling policies. Our third coordination mechanism also handles anonymous jobs and has a nice “cost-revealing” potential function. Besides in proving the existence of equilibria, we use this potential function in order to upper-bound the price of stability of the induced game by  $O(\log m)$ , the price of anarchy by  $O(\log^2 m)$ , and the convergence time to  $O(\log^2 m)$ -approximate assignments by a polynomial number of best-response moves. Our third coordination mechanism is the first that handles anonymous jobs and simultaneously guarantees that the induced game is a potential game and has bounded price of anarchy.

## 31 Approximating fractional hypertree width

Fractional hypertree width is a hypergraph measure similar to tree width and hypertree width. Its algorithmic importance comes from the fact that, as shown in previous work [?], constraint satisfaction problems (CSP) and various problems in database theory are polynomial-time solvable if the input contains a bounded-width fractional hypertree decomposition of the hypergraph of the constraints. In this paper, we show that for every  $w \geq 0$ , there is a polynomial-time algorithm that, given a hypergraph  $H$  with fractional hypertree width at most  $w$ , computes a fractional hypertree decomposition of width  $O(w^3)$  for  $H$ . This means that polynomial-time algorithms relying on bounded-width fractional hypertree decompositions no longer need to be given a decomposition explicitly in the input, since an appropriate decomposition can be computed in polynomial time. Therefore, if  $\mathcal{H}$  is a class

of hypergraphs with bounded fractional hypertree width, then constraint satisfaction restricted to instances whose structure is in  $\mathcal{H}$  is polynomial-time solvable. This makes bounded fractional hypertree width the most general known hypergraph property that makes CSP polynomial-time solvable.

## 32 Improved Smoothed Analysis of the k-Means Method

The k-means method is a widely used clustering algorithm. One of its distinguished features is its speed in practice. Its worst-case running-time, however, is exponential, leaving a gap between practical and theoretical performance. Arthur and Vassilvitskii (FOCS 2006) aimed at closing this gap, and they proved a bound of  $\text{poly}(n^k, 1/\sigma)$  on the smoothed running-time of the k-means method, where  $n$  is the number of data points and  $\sigma$  is the standard deviation of the Gaussian perturbation. This bound, though better than the worst-case bound, is still much larger than the running-time observed in practice. We improve the smoothed analysis of the k-means method by showing two upper bounds on the expected running-time of k-means. First, we prove that the expected running-time is bounded by a polynomial in  $n^{\sqrt{k}}$  and  $1/\sigma$ . Second, we prove an upper bound of  $k^{kd} \text{poly}(n, 1/\sigma)$ , where  $d$  is the dimension of the data space. The polynomial is independent of  $k$  and  $d$ , and we obtain a polynomial bound for the expected running-time for  $k, d \in O(\sqrt{\log n / \log \log n})$ . Finally, we show that k-means runs in smoothed polynomial time for one-dimensional instances.

## 33 An Improved Approximation Algorithm for the Column Subset Selection Problem

We consider the problem of selecting the “best” subset of *exactly*  $k$  columns from an  $m \times n$  matrix  $A$ . In particular, we present and analyze a novel two-stage algorithm that runs in  $O(\min\{mn^2, m^2n\})$  time and returns as output an  $m \times k$  matrix  $C$  consisting of exactly  $k$  columns of  $A$ . In the first stage (the *randomized* stage), the algorithm randomly selects  $O(k \log k)$  columns according to a judiciously-chosen probability distribution that depends on information in the top- $k$  right singular subspace of  $A$ . In the second stage (the *deterministic* stage), the algorithm applies a deterministic column-selection procedure to select and return exactly  $k$  columns from the set of columns selected in the first stage. Let  $C$  be the  $m \times k$  matrix containing those  $k$  columns, let  $P_C$  denote the projection matrix onto the span of those columns, and let  $A_k$  denote the “best” rank- $k$  approximation to the matrix  $A$  as computed with the singular value decomposition. Then, we prove that

$$\|A - P_C A\|_T \leq O\left(k^{\frac{3}{4}} \log^{\frac{1}{2}}(k) (n - k)^{\frac{1}{4}}\right) \|A - A_k\|_T,$$

with probability at least 0.7. This spectral norm bound improves upon the best previously-existing result (of Gu and Eisenstat [?]) for the spectral norm version of this Column Subset Selection Problem. We also prove that

$$\|A - P_C A\|_F \leq O\left(k\sqrt{\log k}\right) \|A - A_k\|_F,$$

with the same probability. This Frobenius norm bound is only a factor of  $\sqrt{k \log k}$  worse than the best previously existing existential result and is roughly  $O(\sqrt{k!})$  better than the best previous algorithmic result (both of Deshpande et al. [?]) for the Frobenius norm version of this Column Subset Selection Problem.

## 34 (Un)Expected Behavior of Digital Search Tree Profile

A digital search tree (DST) – one of the most fundamental data structure on words – is a digital tree in which keys (strings, words) are stored directly in (internal) nodes. Such trees find myriad of applications from the popular Lempel-Ziv’78 data compression scheme to distributed hash tables. The profile of a DST measures the number of nodes at the same distance from the root; it is a function of the number of stored strings and the distance from the root. Most parameters of DST (e.g., height, fill-up, shortest path) can be expressed in terms of the profile. However, from the inception of DST analysis of the profile has been elusive and it has become a prominent open problem in the area of analysis of algorithms. We make here the first, but decisive step, towards solving this problem. We present a precise analysis of the average profile when stored strings are generated by a biased memoryless source. The main technical difficulty of analyzing the profile lies in solving a sophisticated recurrence equation. We present such a solution for the Poissonized version of the problem (i.e., when the number of stored strings is generated by a Poisson distribution) in the Mellin transform domain. To accomplish it, we introduce a novel functional operator that allows us to express the solution in an explicit form, and then using analytic algorithmic tools we extract asymptotic behavior of the profile as a function of the number of stored strings and the distance from the root. This analysis is surprisingly demanding but once it is carried out it reveals unusually intriguing and interesting behavior. The average profile undergoes several phase transitions when moving from the root to the longest path. It first resembles a full tree until it abruptly starts growing polynomially. Furthermore, the expected profile is oscillating in a range where profile grows polynomially. Such a behavior is quite unexpected for most shape parameters of random trees, except recently analyzed profiles of tries which are another incarnations of digital trees. Our results are derived by methods of analytic algorithmics such as generating functions, Mellin transform, Poissonization and de-Poissonization, the saddle-point method, singularity analysis and uniform asymptotic analysis.

## 35 A logarithmic approximation for the unsplittable flow on line graphs

We consider the unsplittable flow problem on a line. In this problem, we are given a set of  $n$  tasks, each specified by a start time  $s_i$ , and end time  $t_i$ , a demand  $d_i > 0$ , and a profit  $p_i > 0$ . A task, if accepted, requires  $d_i$  units of “bandwidth” from time  $s_i$  to  $t_i$  and accrues a profit of  $p_i$ . For every time  $t$ , we are also specified the available bandwidth  $c_t$ , and the goal is to find a subset of tasks with maximum profit subject to the bandwidth constraints. We present the first polynomial-time  $O(\log n)$ -approximation algorithm for this problem. This significantly advances the state-of-the-art, as no polynomial-time  $o(n)$ -approximation was known previously. Previous results for this problem were known only in more restrictive settings. In particular, either if instance satisfies the so-called “no-bottleneck” assumption:  $\max_i d_i \leq \min_t c_t$ , or else if the ratio of both the maximum to the minimum demands and the maximum to minimum capacities are polynomially (or quasi-polynomially) bounded in  $n$ . Our result, on the other hand, does not require any of these assumptions. Our algorithm is based on a combination of dynamic programming and rounding a natural linear programming relaxation for the problem. While there is an  $\Omega(n)$  integrality gap known for this LP relaxation, our key idea is to exploit certain structural properties of the problem to show that instances that are bad for the LP can in fact be handled using dynamic programming.

## 36 Variance-optimal sampling-based estimation of subset sums

From a high volume stream of weighted items, we want to maintain a generic sample of a certain limited size  $k$  that we can later use to estimate the total weight of arbitrary subsets. This is the classic

context of on-line reservoir sampling, thinking of the generic sample as a reservoir. We present an efficient reservoir sampling scheme,  $\text{VarOpt}_k$ , that dominates all previous schemes in terms of estimation quality.  $\text{VarOpt}_k$  provides variance optimal unbiased estimation of subset sums. More precisely, if we have seen  $n$  items of the stream, then for any subset size  $m$ , our scheme based on  $k$  samples minimizes the average variance over all subsets of size  $m$ . In fact, it is the first online scheme with this property and the optimality is against any off-line sampling scheme tailored for the concrete set of items seen: no off-line scheme based on  $k$  samples can perform better than our on-line scheme when it comes to average variance over any subset size. In addition to optimal average variance, our scheme provides tighter worst-case bounds on the variance of particular subsets than previously possible, even by an offline scheme. It is efficient, handling each new item of the stream in  $O(\log k)$  time, which is optimal even on the word RAM. Finally, it is particularly well suited for combination of samples from different streams in a distributed setting.

### 37 Finding shortest contractible cycles in embedded graphs

We give a polynomial-time algorithm to find a shortest contractible cycle (i.e. a closed walk without repeated vertices) in a graph embedded in a surface. This answers a question posed by Mohar and Thomassen. In contrast, we show that finding a shortest contractible cycle through a given vertex is NP-hard. We also show that finding a shortest separating cycle in an embedded graph is NP-hard.

### 38 On the approximability of the maximum feasible subsystem problem with 0/1-coefficients

Given a system of inequalities  $l_i \leq a_i^T x \leq u_i$ , where  $a_i \in \{0, 1\}^n$ , and  $l_i, u_i \in \mathbb{R}_+$ , for  $i = 1, \dots, m$ , we consider the problem of finding the largest subsystem for which there exists a feasible solution  $x \geq 0$ . We give several approximability and inapproximability results, parameterized by the value of  $L = \max\{l_1, \dots, l_m\}$ . In particular, we show that, under reasonable complexity assumptions, the problem cannot be approximated within a factor of  $O(\log^\mu n)$ , for some positive  $\mu$ , even when  $l_i = u_i = 1$  for all  $i$ , and within a factor of  $O(n^{1/3-\epsilon})$ , for an arbitrarily small  $\epsilon > 0$ , if  $L$  is arbitrarily large. On the way, we also obtain an  $\Omega(n^{1/3-\epsilon})$ -inapproximability threshold for the *maximum induced matching problem* on bipartite graphs. On the positive side, we consider  $(\alpha, \beta)$ -approximations, i.e., of finding a subsystem with size at least  $|\text{OPT}|/\alpha$ , which has a feasible solution violating the right-hand side by a factor of at most  $\beta$ . For general 0/1-coefficient matrices, we give a polynomial-time  $(O(\log(nm)), 1+\epsilon)$ -approximation for the case when  $L = \text{poly}(n, m)$ . For interval matrices, we give a quasi-polynomial time  $(1, 1+\epsilon)$ -approximation when  $L = \text{poly}(n, m)$  and a polynomial time  $(O(\log^2 n \log \log(nL)), 1+\epsilon)$ -approximation in the general case, and complement the first result by showing that, when no violation is allowed, the problem is APX-hard. Clearly all the above results apply to systems of equations ( $l_i = u_i$ ) with 0/1-coefficients. As further applications, we show how these results apply to a recently-studied profit-maximizing pricing problem, and to the problem of drawing interval graphs given length constraints on the intervals.

### 39 Clustering on Noisy Mixtures

This paper presents a polynomial algorithm for learning mixtures of log-concave distributions in  $\mathbb{R}^n$  in the presence of malicious noise points. That is, each sample is corrupted with some small probability and replaced by a point about which we can make no assumptions. A key element of the algorithm is Robust Principle Components Analysis (PCA), which is less susceptible to corruption

by noisy points. Where noise may cause standard PCA to collapse well-separated mixture components so that they are indistinguishable, Robust PCA preserves the distance between some of the components, making a partition possible. It then recurses on each half of the mixture until every component is isolated. The success of this algorithm requires only a  $O(\log n)$  factor increase in the required separation between components of the mixture compared to the noiseless case.

## 40 Perfect matchings via uniform sampling in regular bipartite graphs

Regular bipartite graphs occur in many real life applications, including scheduling, routing, and task assignment. In particular, finding a perfect matching in a regular bipartite graph is a well-studied problem. The first non-trivial algorithm, with running time  $O(mn)$ , dates back to König’s work in 1916 (here  $m = nd$  is the number of edges in the graph,  $2n$  the number of vertices, and  $d$  the degree of each node). The currently most efficient algorithm takes time  $O(m)$ , and is due to Cole, Ost, and Schirra. In this paper, we further improve this running time to  $O(\min\{m, n^{1.75} \ln n\})$ . We obtain this improvement by proving a uniform sampling theorem: if we sample each edge in a  $d$ -regular bipartite graph independently with a probability  $p = O(\frac{n \ln n}{d^2})$  then the resulting graph has a perfect matching with high probability. The proof involves a sequential decomposition procedure combined with an application of Karger’s sampling theorem in each piece. Running the  $O(m\sqrt{n})$  algorithm (due to Hopcroft and Karp) for finding maximum matchings in bipartite graphs on the sampled graph yields the stated running time. We provide an infinite family of instances to show that the sampling result is tight up to polylogarithmic factors. Thus any super-polylogarithmic improvements to our running time must either employ a different sampling procedure or improve the Hopcroft-Karp algorithm for the case of nearly regular bipartite graphs.

## 41 An $\tilde{O}(m)$ algorithm for constructing a cactus of an undirected graph

We present an  $\tilde{O}(m)$  algorithm for constructing the *cactus* data structure of an undirected graph, which captures all the global minimum edge cuts in the graph in  $O(n)$  space. This represents a fundamental improvement over the previous best time bounds ( $\tilde{O}(n^2)$  due to Karger and Stein) since the number of mincuts in an undirected graph can be  $\tilde{\omega}(m)$ , but has to be  $O(n^2)$ . Thus, unlike earlier algorithms, our algorithm cannot afford to spend even  $O(1)$  time per mincut. Further, our result closes the gap between the time required for finding a single mincut ( $\tilde{O}(m)$  due to Karger) and that for (implicitly) finding all the mincuts. Similar to the previous best algorithm, our algorithm is Monte-Carlo, ie gives the correct answer with high probability but does not give a certificate for checking correctness.

## 42 Efficient Algorithms for Column Subset Selection

Given a fixed matrix, the problem of column subset selection requests a column submatrix that has favorable spectral properties. Most research from the algorithms and numerical linear algebra communities focuses on a variant called rank-revealing QR, which seeks a well-conditioned collection of columns that spans the (numerical) range of the matrix. The functional analysis literature contains another strand of work on column selection whose algorithmic implications have not been explored. In particular, a celebrated result of Bourgain and Tzafriri demonstrates that each matrix with normalized columns contains a large column submatrix that is exceptionally well conditioned. Unfortunately, standard proofs of this result cannot be regarded as algorithmic. This paper presents a randomized, polynomial-time algorithm that produces the submatrix promised by Bourgain and

Tzafiri. The method involves random sampling of columns, followed by a matrix factorization that exposes the well-conditioned subset of columns. This factorization, which is due to Grothendieck, is regarded as a central tool in modern functional analysis. The primary novelty in this work is an algorithm, based on eigenvalue minimization, for constructing the Grothendieck factorization. These ideas also result in a novel approximation algorithm for the  $(\infty, 1)$  norm of a matrix, which is generally NP-hard to compute exactly. As an added bonus, this work reveals a surprising connection between matrix factorization and the famous MAXCUT semidefinite program.

### 43 Improved bounds and new techniques for Davenport-Schinzel sequences and their generalizations

We present several new results regarding  $\lambda_s(n)$ , the maximum length of a Davenport-Schinzel sequence of order  $s$  on  $n$  distinct symbols. First, we prove that  $\lambda_s(n) \leq n * 2^{1/t! \alpha(n)^t + O(\alpha(n)^{t-1})}$  for  $s \geq 4$  even, and  $\lambda_s(n) \leq n * 2^{1/t! \alpha(n)^t \log_2 \alpha(n) + O(\alpha(n)^t)}$  for  $s \geq 3$  odd, where  $t = \lfloor (s-2)/2 \rfloor$ , and  $\alpha(n)$  denotes the inverse Ackermann function. This constitutes an improvement for  $s \geq 6$ , since the previous bounds, by Agarwal, Sharir, and Shor (1989), had a leading coefficient of 1 instead of  $1/t!$  in the exponent. The bounds for even  $s \geq 6$  are now tight up to lower-order terms in the exponent. These new bounds result from a small improvement on the technique of Agarwal et al. More importantly, we also present a new technique for deriving upper bounds for  $\lambda_s(n)$ . This new technique is based on some recurrences very similar to the ones used by the author, together with Alon, Kaplan, Sharir, and Smorodinsky (SODA 2008), for the problem of stabbing interval chains with  $j$ -tuples. With this new technique we: (1) re-derive the upper bound of  $\lambda_3(n) \leq 2n\alpha(n) + O(n\sqrt{\alpha(n)})$  (first shown by Klazar, 1999); (2) re-derive our own new upper bounds for general  $s$ ; and (3) obtain improved upper bounds for the generalized Davenport-Schinzel sequences considered by Adamec, Klazar, and Valtr (1992). Regarding lower bounds, we show that  $\lambda_3(n) \geq 2n\alpha(n) - O(n)$  (the previous lower bound (Sharir and Agarwal, 1995) had a coefficient of  $1/2$ ), so the coefficient 2 is tight. We also present a simpler variant of the construction of Agarwal, Sharir, and Shor that achieves the known lower bounds of  $\lambda_s(n) \geq n * 2^{1/t! \alpha(n)^t - O(\alpha(n)^{t-1})}$  for  $s \geq 4$  even.

### 44 Efficient algorithms on sets of permutations, dominance, and real-weighted APSP

Sets of permutations play an important role in the design of some efficient algorithms. In this paper we design two algorithms that manipulate sets of permutations. Both algorithms, each solving a different problem, use fast matrix multiplication techniques to achieve a significant improvement in the running time over the naive solutions. A set of permutations  $P \subset S_n$  is *expanding* if the product of any two elements of  $P$  yields a distinct permutation. Stated otherwise,  $|P^2| = |P|^2$  where  $P^2 \subset S_n$  is the set of products of two elements of  $P$ . Our first result is a randomized algorithm that computes  $|P^2|$  and hence decides if  $P$  is expanding. The algorithm also produces a table that, for any  $\sigma_1, \sigma_2, \sigma_3, \sigma_4 \in P$ , answers the query  $\sigma_1\sigma_2 = \sigma_3\sigma_4$  in  $\tilde{O}(1)$  time. The algorithm uses, among other ingredients, a combination of fast matrix multiplication and polynomial identity testing. In particular, for  $|P| = n$  our algorithm runs in  $O(n^\omega)$  time where  $\omega < 2.376$  is the matrix multiplication exponent. We note that the naive deterministic solution for this problem requires  $\Theta(n^3)$  time. For a set of permutations  $P \subset S_n$  we say that  $i$   $k$ -dominates  $j$  if the number of permutations  $\pi \in P$  for which  $\pi(i) < \pi(j)$  is  $k$ . The *dominance matrix* of  $P$  is the  $n \times n$  matrix  $D_P$  where  $D_P(i, j) = k$  if and only if  $i$   $k$ -dominates  $j$ . We give an efficient algorithm for computing  $D_P$  using fast *rectangular* matrix multiplication. In particular, when  $|P| = n$  our algorithm runs in  $O(n^{2.684})$  time. Computing the dominance matrix of permutations

is computationally equivalent to the dominance problem in computational geometry. In particular, our algorithm slightly improves upon a well-known  $O(n^{(\omega+3)/2})$  time algorithm of Matousek for the dominance problem. Permutation dominance is used, together with a few other ingredients, to obtain a truly sub-cubic algorithm for the *All Pairs Shortest Paths* (APSP) problem in real-weighted directed graphs, where the number of distinct weights emanating from each vertex is  $O(n^{0.338})$ . A very special case of this algorithm implies an  $O(n^{2.842})$  time algorithm for real vertex-weighted APSP, which slightly improves a recent result of Chan [STOC-07].

## 45 High rate fingerprinting codes and the fingerprinting capacity

Including a unique code in each copy of a distributed document is an effective way of fighting intellectual piracy. Codes designed for this purpose that are secure against collusion attacks are called fingerprinting codes. In this paper we consider fingerprinting with the marking assumption and design codes that achieve much higher rates than previous constructions. We conjecture that these codes attain the maximum possible rate (the fingerprinting capacity) for any fixed number of pirates. We prove new upper bounds for the fingerprinting capacity that are not far from the rate of our codes. We introduce the novel model of weak fingerprinting codes where one pirate should be caught only if the identity of all other pirates are revealed. We construct fingerprinting codes in this model with improved rates but our upper bound on the rate still applies. In fact, these improved codes achieve the fingerprinting capacity of the weak model by a recent upper bound. Using analytic techniques we compare the rates of our codes in the standard model and the rates of the optimal codes in the weak model. To our surprise these rates asymptotically agree, that is, their ratio tends to 1 as  $t$  goes to infinity. Although we cannot prove that each one of our codes in the standard model achieves the fingerprinting capacity, this proves that asymptotically they do.

## 46 Almost all hypergraphs without Fano planes are bipartite

The hypergraph of the Fano plane is the unique hypergraph with 7 triples on 7 vertices in which every pair of vertices is contained in a unique triple. It is not 2-colorable, but becomes so on deleting any hyperedge. We show that taking uniformly at random a 3-uniform hypergraph  $H$  on  $n$  vertices not containing the Fano plane,  $H$  turns out to be bipartite (or 2-colorable) with probability at least  $1 - 2^{-\Omega(n^2)}$ . To prove this result we will study structural properties of Fano plane-free hypergraphs. In particular, we prove that only with exponentially small probability ( $2^{-\Omega(n^3)}$ ) they are  $\epsilon$ -far from being bipartite. These results seem to be interesting in the light of the well-known fact, that to decide whether a given 3-uniform hypergraph is bipartite is NP-complete. On the other hand, the property of being bipartite is known to be testable. We hope that understanding better the structure of (hyper-)graph properties might be useful for average case analysis of running time of coloring algorithms on restricted hypergraph classes, and also advantageous in developing efficient property testers.

## 47 Pairing Heaps with $O(\log \log n)$ decrease Cost

We give a variation of the pairing heaps for which the time bounds for all the operations match the lower bound proved by Fredman for a family of similar self-adjusting heaps. Namely, our heap structure requires  $O(1)$  for insert and find-min,  $O(\log n)$  for delete-min, and  $O(\log \log n)$  for decrease-key and meld (all the bounds are in the amortized sense except for find-min).

## 48 A simpler implementation and analysis of Chazelle’s Soft Heaps

Chazelle (JACM 47(6), 2000) devised an approximate meldable priority queue data structure, called SOFT HEAPS, and used it to obtain the fastest known deterministic comparison-based algorithm for computing minimum spanning trees, as well as some new algorithms for selection and approximate sorting problems. If  $n$  elements are inserted into a collection of soft heaps, then up to  $\epsilon n$  of the elements still contained in these heaps, for a given *error parameter*  $\epsilon$ , may be CORRUPTED, i.e., have their keys artificially increased. In exchange for allowing these corruptions, each soft heap operation is performed in  $O(\log \frac{1}{\epsilon})$  amortized time. Chazelle’s soft heaps are derived from the BINOMIAL HEAPS data structure in which each priority queue is composed of a collection of BINOMIAL TREES. We describe a simpler and more direct implementation of soft heaps in which each priority queue is composed of a collection of standard BINARY trees. Our implementation has the advantage that no CLEAN-UP operations similar to the ones used in Chazelle’s implementation are required. We also present a concise and unified potential-based amortized analysis of the new implementation.

## 49 Discounted Deterministic Markov Decision Processes and Discounted All-Pairs Shortest Paths

We present an  $O(mn + n^2 \log n)$ -time algorithm for finding optimal strategies for discounted, infinite-horizon, Deterministic Markov Decision Processes (DMDP), where  $n$  is the number of vertices (or states) and  $m$  is the number of edges (or actions). This improves a recent  $O(mn^2)$ -time algorithm of Andersson and Vorobyov. We also present algorithms for finding discounted all-pairs shortest paths.

## 50 Efficient algorithms for the 2-gathering problem

Pebbles are placed on some vertices of a directed graph. Is it possible to move each pebble along at most one edge of the graph so that in the final configuration no pebble is left alone? We give an  $O(mn)$ -time algorithm for solving this problem, which we call the 2-GATHERING problem, where  $n$  is the number of vertices and  $m$  is the number of edges of the graph. If such a 2-gathering is not possible, the algorithm finds a solution that minimizes the number of ‘lonely’ pebbles. The 2-gathering problem forms a non-trivial generalization of the non-bipartite matching problem and it is solved by extending the augmenting paths technique used to solve matching problems. Polynomial time algorithms for the 2-gathering problem can be obtained via reductions to the  $\{K_2, K_3\}$ -packing problem studied by Hell and Kirkpatrick, the general factor problem studied by Cornuejols, and to the simplex matching problem studied recently by Anshelevich and Karagiozova. The resulting algorithms, however, are much slower than the new algorithm presented.

## 51 Persistent Homology for Kernels, Images, and Cokernels

Motivated by the measurement of local homology and of functions on noisy domains, we extend the notion of persistent homology to sequences of kernels, images, and cokernels of maps induced by inclusions in a filtration of pairs of spaces. Specifically, we note that persistence in this context is well defined, we prove that the persistence diagrams are stable, and we explain how to compute them.

## 52 Clique-width: On the Price of Generality

Many hard problems can be solved efficiently when the input is restricted to graphs of bounded treewidth. By the celebrated result of Courcelle, every decision problem expressible in monadic second order logic is fixed parameter tractable when parameterized by the treewidth of the input graph. Moreover, for every fixed  $k \geq 0$ , such problems can be solved in linear time on graphs of treewidth at most  $k$ . In particular, this implies that basic problems like Dominating Set, Graph Coloring, Clique, and Hamiltonian Cycle are solvable in linear time on graphs of bounded treewidth. A significant amount of research in graph algorithms has been devoted to extending this result to larger classes of graphs. It was shown that some of the algorithmic meta-theorems for treewidth can be carried over to graphs of bounded clique-width. Courcelle, Makowsky, and Rotics proved that the analogue of Courcelle's result holds for graphs of bounded clique-width when the logical formulas do not use edge set quantifications. Despite of its generality, this does not resolve the parameterized complexity of many basic problems concerning edge subsets (like Edge Dominating Set), vertex partitioning (like Graph Coloring), or global connectivity (like Hamiltonian Cycle). There are various algorithms solving some of these problems in polynomial time on graphs of clique-width at most  $k$ . However, these are not fixed parameter tractable algorithms and have typical running times  $O(n^{f(k)})$ , where  $n$  is the input length and  $f$  is some function. It was an open problem, explicitly mentioned in several papers, whether any of these problems is fixed parameter tractable when parameterized by the clique-width, i.e. solvable in time  $O(g(k)n^c)$ , for some function  $g$  and a constant  $c$  not depending on  $k$ . In this paper we resolve this problem by showing that Edge Dominating Set, Hamiltonian Cycle, and Graph Coloring are  $W[1]$ -hard parameterized by clique-width. This shows that the running time  $O(n^{f(k)})$  of many clique-width based algorithms is essentially the best we can hope for (up to a widely believed assumption from parameterized complexity, namely  $FPT \neq W[1]$ )—the price we pay for generality.

## 53 Maximum Independent Set of Rectangles

We study the Maximum Independent Set of Rectangles problem (MISR): given a collection  $R$  of  $n$  axis-parallel rectangles, find a maximum-cardinality subset of disjoint rectangles. MISR is a special case of the classical Maximum Independent Set problem, where the input is restricted to intersection graphs of axis-parallel rectangles. Due to its many applications, ranging from map labeling to data mining, MISR has received a significant amount of attention from various research communities. Since the problem is NP-hard, the main focus has been on designing approximation algorithms. Several groups of researchers have independently suggested  $O(\log n)$ -approximation algorithms for MISR, and this remained the best currently known approximation factor for the problem. The main result of our paper is an  $O(\log \log n)$ -approximation algorithm for MISR. Our algorithm combines existing approaches for solving special cases of the problem, in which the input set of rectangles is restricted to containing only specific intersection types, with new insights into the combinatorial structure of sets of intersecting rectangles in the plane. We also consider a generalization of MISR to higher dimensions, where rectangles are replaced by  $d$ -dimensional hyper-rectangles. This problem is known as Maximum Independent Set on  $d$ -box graphs. Our results for MISR imply an  $O((\log n)^{d-2} \log \log n)$  approximation algorithm for this problem, improving upon the best previously known  $O((\log n)^{d-1})$ -approximation.

## 54 On Algorithmic Hypergraph Regularity

Thomason and Chung, Graham, and Wilson were the first to systematically study quasi-random graphs and hypergraphs, and proved that several properties of random graphs imply each other in

a deterministic sense. Their concepts of quasi-randomness match the notion of regularity from the earlier Szemerédi Regularity Lemma. In contrast, there exists no "natural" hypergraph regularity lemma matching the notions of quasi-random hypergraphs considered by those authors. Here, we study several notions of quasi-randomness for 3-uniform hypergraphs which correspond to the regularity lemmas of Frankl and Rödl, Gowers and Haxell, Nagle and Rödl. We establish an equivalence among the three notions of regularity of these lemmas. Since the regularity lemma of Haxell et al. is algorithmic, we obtain algorithmic versions of the lemmas of Frankl-Rödl (a special case thereof) and Gowers as corollaries. As a further corollary, we show that the special case of the Frankl-Rödl lemma (which we can make algorithmic) suffices for an application of its corresponding Counting Lemma.

## 55 Partitioning Graphs into Balanced Components

We consider the *k*-balanced partitioning problem, where the goal is to partition the vertices of an input graph  $G$  into  $k$  equally sized components, while minimizing the total weight of the edges connecting different components. We allow  $k$  to be part of the input and denote the cardinality of the vertex set by  $n$ . This problem is a natural and important generalization of well-known graph partitioning problems, including *minimum bisection* and *minimum balanced cut*. We present a bi-criteria approximation algorithm that achieves an approximation factor of  $O(\sqrt{\log n \log k})$ , which matches or improves over previous algorithms for all relevant values of  $k$ . Our algorithm uses a semidefinite relaxation which combines  $\ell_2^2$  metrics with spreading metrics. Surprisingly, we show that the integrality gap of the semidefinite relaxation is  $\Omega(\log k)$  even for large values of  $k$  (e.g.,  $k = n^{\Omega(1)}$ ), implying that the dependence on  $k$  of the approximation factor is necessary. This is in contrast to the situation with previous approximation algorithms for  $k$ -balanced partitioning which are based on linear programming relaxations and where the approximation factor is independent of  $k$ .

## 56 Speed Scaling with an Arbitrary Power Function

All of the theoretical speed scaling research to date has assumed that the power function, which expresses the power consumption  $P$  as a function of the processor speed  $s$ , is of the form  $P = s^\alpha$ , where  $\alpha > 1$  is some constant. Motivated in part by technological advances, we initiate a study of speed scaling with arbitrary power functions. We consider the problem of minimizing the total flow plus energy. Our main result is a  $(3 + \epsilon)$ -competitive algorithm for this problem, that holds for essentially any power function. We also give a  $(2 + \epsilon)$ -competitive algorithm for the objective of fractional weighted flow plus energy. Even for power functions of the form  $s^\alpha$ , it was not previously known how to obtain competitiveness independent of  $\alpha$  for these problems. We also introduce a model of allowable speeds that generalizes all known models in the literature.

## 57 Coloring triangle-free graphs on surfaces

Gimbel and Thomassen asked whether the 3-colorability of a triangle-free graph drawn on a fixed surface can be tested in polynomial time. We settle the question by giving a linear-time algorithm for every surface. When combined with previous results this gives a linear-time algorithm to compute the chromatic number of a triangle-free graph drawn on a fixed surface. Our algorithm is based on a structure theorem that for a triangle-free graph drawn on a surface  $S$  guarantees the existence of (and can be converted to a linear-time algorithm to find) a subgraph  $H$ , whose size depends only on  $S$ , such that there is an easy test whether a 3-coloring of  $H$  extends to a 3-coloring of  $G$ . The test is based on a topological obstruction, called the "winding number" of a 3-coloring. To prove the

structure theorem we make use of disjoint paths with specified ends (a.k.a. integral multicommodity flows) to find a 3-coloring. If the input triangle-free graph  $G$  drawn in  $S$  is 3-colorable we can find a 3-coloring in quadratic time, and if  $G$  quadrangulates  $S$  then we can find the 3-coloring in linear time. The latter algorithm requires two ingredients that may be of independent interest: a generalization of a data structure of Kowalik and Kurowski to weighted graphs and a speedup of a disjoint paths algorithm of Robertson and Seymour to linear time.

## 58 On the Relative Strength of Split, Triangle and Quadrilateral Cuts (Extended Abstract)

Integer programs defined by two equations with two free integer variables and nonnegative continuous variables have three types of nontrivial facets: split, triangle or quadrilateral inequalities. In this paper, we compare the strength of these three families of inequalities. In particular we study how well each family approximates the integer hull. We show that, in a well defined sense, triangle inequalities provide a good approximation of the integer hull. The same statement holds for quadrilateral inequalities. On the other hand, the approximation produced by split inequalities may be arbitrarily bad.

## 59 From coding theory to efficient pattern matching

We consider the classic problem of pattern matching with few mismatches in the presence of promiscuously matching wildcard symbols. Given a text  $t$  of length  $n$  and a pattern  $p$  of length  $m$  with optional wildcard symbols and a bound  $k$ , our algorithm finds all the locations for which the pattern matches the text with Hamming distance at most  $k$ . The algorithm we present is deterministic and runs in  $\tilde{O}(kn)$  time, returning the location of each mismatch at no extra cost. The solutions we develop borrow from the tool set of algebraic coding theory and provide a new framework in which to tackle approximate pattern matching problems.

## 60 Fast Edge Orientation for Unweighted Graphs

We consider an unweighted undirected graph with  $n$  vertices,  $m$  edges, and edge connectivity  $2k$ . The *weak edge orientation problem* requires that the edges of this graph be oriented so the resulting directed graph is at least  $k$  edge connected. Nash-Williams proved the existence of such orientations and subsequently Frank [?], Gabow [?], and Nagamochi-Ibaraki gave algorithmic constructions. All of these algorithm took time at least quadratic in  $n$ . We provide the first sub-quadratic (in  $n$ ) algorithm for this problem. Our algorithm takes  $\tilde{O}(nk^4 + m)$  time. This improves the previous best bounds of  $\tilde{O}(n^2k^2 + m)$  by Gabow [?] and  $\tilde{O}(n^2m)$  by Nagamochi-Ibaraki when  $k \leq \sqrt{n}$ . Indeed, many real networks have  $k \ll n$ . Our algorithm uses the fast edge splitting paradigm introduced by Bhalgat et al. [?]. We seek to split out a large fraction of the vertices, recurse on the resulting graph, and then put back the split-off vertices. The main challenge we face is that only vertices with even degree may be split-off in an undirected graph and there may not be any such vertex in the current graph. The edge orientation algorithms of Gabow and Nagamochi-Ibaraki as well as Frank's proof are based on showing the existence of at least two even degree vertices (in fact, vertices with degree  $2k$ ) in a  $2k$  minimally connected graph. We generalize this to show that in any edge minimal  $2k$  connected graph, there are at least  $n/3$  even degree vertices. These vertices are then split-off. Our next challenge is to drop edges from the given graph so it remains  $2k$  connected and yet has  $\Omega(n)$  even degree vertices. We provide two algorithms for this. The first algorithm produces an edge-minimal

$2k$  connected graph in time  $\tilde{O}(nk^5 + m)$ ; this improves the previous best  $\tilde{O}(n^2k^2 + m)$  algorithm for this task due to Gabow [?]. The second algorithm speeds this up by a factor of  $k$  by giving up the minimality criterion; it discards edges specifically to produce  $\Omega(n)$  even degree vertices while maintaining connectivity  $2k$  and takes time  $\tilde{O}(nk^4 + m)$ .

## 61 On the Maximum Quadratic Assignment Problem

Quadratic assignment is a basic problem in combinatorial optimization, which generalizes several other problems such as Traveling Salesman, Linear Arrangement, Dense  $k$  Subgraph, and Clustering with given sizes. Quadratic Assignment is defined relative to two  $n$  by  $n$  symmetric non-negative matrices. In this paper, we study the Maximum Quadratic Assignment problem, and give an  $\sqrt{n}$  approximation algorithm, which is the first non-trivial approximation guarantee for this problem. The above guarantee also holds when the input matrices are asymmetric. An indication of the hardness of Maximum Quadratic Assignment is that it contains as a special case, the Dense  $k$  Subgraph problem, for which the best known approximation guarantee is  $n^{1/3}$  (Feige et al. 2001). When one of the matrices satisfies triangle inequality, we give an algorithm achieving approximation ratio 3.16, which improves over the previous best factor of 4 (Arkin et al. 2001). See complete abstract in pdf file.

## 62 Natural Algorithms

We provide further evidence that the study of complex self-organizing systems can benefit from an algorithmic perspective. The subject has been traditionally viewed through the lens of physics and control theory. Using tools associated with computer science, we settle a longstanding open question in theoretical ecology: the convergence of bird flocks in the Reynolds model. We bound the time to reach steady state by a tower-of-twos of height exponential in the number of birds. We prove that, surprisingly, the tower-of-twos growth is intrinsic to the model. This unexpected result demonstrates the merits of approaching biological dynamical systems as "natural algorithms" and applying algorithmic tools to them.

## 63 Comparison-Based, Time-Space Lower Bounds for Selection

We establish the first nontrivial lower bounds on time-space tradeoffs for the selection problem. We prove that any comparison-based, randomized algorithm for finding the median requires  $\Omega(n \log \log_s n)$  expected time in the RAM model (or more generally in the comparison branching program model), if we have  $S$  bits of extra space besides the read-only input array. This bound is tight for all  $S \gg \log n$ , and remains true even if the array is given in a random order. Our result thus answers a 16-year-old question of Munro and Raman, and also complements recent lower bounds that are restricted to sequential access, as in the multi-pass streaming model [Chakrabarti *et al.*, SODA 2008]. We also prove that any comparison-based, deterministic, multi-pass streaming algorithm for finding the median requires  $\Omega(n \log^*(n/s) + n \log_s n)$  worst-case time (in scanning plus comparisons), if we have  $s$  cells of space. This bound is also tight for all  $s \gg \log^2 n$ . We can get similar deterministic lower bounds for I/O-efficient algorithms, or for the two-dimensional linear programming problem. All proofs are "elementary".

## 64 Optimal Halfspace Range Reporting in Three Dimensions

We give the first optimal solution to a standard problem in computational geometry: three-dimensional halfspace range reporting. We show that  $n$  points in 3-d can be stored in a linear-space data structure so that all  $k$  points inside a query halfspace can be reported in  $O(\log n + k)$  time. The data structure can be built in  $O(n \log n)$  expected time. The previous methods with optimal query time requires superlinear ( $O(n \log \log n)$ ) space. We mention consequences to external-memory data structures, and as an aside, we partially answer another open question concerning the crossing number in Matoušek’s *shallow partition theorem* in the 3-d case (a tool used in known halfspace range reporting methods).

## 65 Optimality of Belief Propagation for Random Assignment Problem

The assignment problem concerns finding the minimum cost perfect matching in a complete weighted bipartite graph. The best known algorithm, due to Edmonds and Karp (1972), for this classical question finds a solution in  $O(n^3)$  time for a graph of size  $n$ . Clearly, any algorithm requires  $\Omega(n^2)$  time. For decades, it has not been answered whether optimal computation time is closer to  $n^3$  or  $n^2$ ? In this paper, we provide answer to this question for random instance of assignment problem. Specifically, we establish that Belief Propagation essentially finds solution in  $O(n^2)$  time when edge weights are i.i.d. with light tailed distribution. To establish this result, we use the formalism of local weak convergence introduced by Aldous (1992, 2000). The standard next step requires proving existence of ‘‘correlation decay’’. Interestingly enough, even though we find that there is no correlation decay, we are still able to prove convergence of belief propagation. Understanding such different method of convergence beyond correlation decay for BP should be of interest in its own right.

## 66 Compressed Counting

Counting the  $\alpha$ th frequency moment,  $F_{(\alpha)} = \sum_{i=1}^D A_t[i]^\alpha$ , of a streaming signal  $A_t$  (where  $t$  denotes time), has been an active area of research. When  $0 < \alpha \leq 2$ , one popular algorithm for approximating  $F_{(\alpha)}$  is based on symmetric stable random projections, whose sample complexity is  $O(1/\epsilon^2)$ , even for  $\alpha = 1$ . Compressed Counting (CC) is proposed for efficiently computing the  $\alpha$ th frequency moment, also for  $0 < \alpha \leq 2$ . In the neighborhood of  $\alpha = 1$ , its sample complexity is  $O(1/\epsilon)$ , instead of  $O(1/\epsilon^2)$ . The underlying technique of CC is maximally-skewed stable random projections. Our contributions include: 1) Compressed Counting (CC) is the first proposal of using skewed projections in data stream computations. We also show that maximally-skewed projections can achieve the best performance. 2) CC is the first algorithm which captures the intuition that, when  $\alpha = 1$  a simple counter suffices, and when  $\alpha = 1 \pm \Delta$  with small  $\Delta$ , the sample complexity should be low. We show the sample complexity of a proposed estimator  $= \frac{G}{\epsilon^2} \log(\frac{2}{\delta})$ , where  $G = O(\epsilon)$  as  $\Delta \rightarrow 0$ . In other words, for small  $\Delta$ , CC has the complexity  $= O(1/\epsilon)$ . Previous results only achieved  $O(1/\epsilon^2)$  (even for  $\alpha = 1$ ), which in general is expected and can not be improved, according to the central limit theorem. 3) We show, when  $\alpha = 1 \pm \Delta$ , the asymptotic variance of a proposed estimator is  $\propto \Delta$ . As  $\alpha \rightarrow 1$ , CC achieves ‘‘an infinite improvement’’ over the previous results, in terms of the asymptotic variances. 4) Our results (especially when  $\alpha = 1 \pm \Delta \approx 1$ ) have important applications. In some situation,  $\Delta$  might be interpreted as the ‘‘decay rate’’ or ‘‘interest rate,’’ which is usually small. The method of moments is popular for statistical inference, which requires computing frequency moments. Also, some important summary statistics such as Renyi entropy and Tsallis entropy are functions of frequency moments and hence CC naturally provides an efficient algorithm to compute them. Very importantly, CC may serve the basic building element for developing other algorithms, for example, approximating the

Shannon entropy using Renyi entropy and Tsallis entropy with  $\alpha \approx 1$ . 4) Our results enrich the fundamental theory of skewed stable distributions.

## 67 The Geometry of Binary Search Trees

We present a novel connection between binary search trees (BSTs) and points in the plane satisfying a simple property. Using this correspondence, we achieve the following results: 1. A surprisingly clean restatement in geometric terms of many results and conjectures relating to BSTs and dynamic optimality. 2. A new lower bound for searching in the BST model, which subsumes the previous two known bounds of Wilber [FOCS'86]. 3. The first proposal for dynamic optimality not based on splay trees. A natural greedy but offline algorithm was presented by Lucas [1988], and independently by Munro [2000], and was conjectured to be an (additive) approximation of the best binary search tree. We show that there exists an equal-cost online algorithm, transforming the conjecture of Lucas and Munro into the conjecture that the greedy algorithm is dynamically optimal.

## 68 A Simple Combinatorial Algorithm for Submodular Function Minimization

This paper presents a new simple algorithm for minimizing submodular functions. For integer valued submodular functions, the algorithm runs in  $O(n^6EO \log nM)$  time, where  $n$  is the cardinality of the ground set,  $M$  is the maximum absolute value of the function value, and  $EO$  is the time for function evaluation. The algorithm can be improved to run in  $O((n^4EO + n^5) \log nM)$  time. The strongly polynomial version of this faster algorithm runs in  $O((n^5EO + n^6) \log n)$  time for real valued general submodular functions. These are comparable to the best known running time bounds for submodular function minimization. The algorithm can also be implemented in strongly polynomial time using only additions, subtractions, comparisons, and the oracle calls for function evaluation. This is the first fully combinatorial submodular function minimization algorithm that does not rely on the scaling method.

## 69 Towards Computing the Grothendieck Constant

The *Grothendieck constant*  $K_G$  is the smallest constant such that for every  $n \in \mathbb{Z}$  and every matrix  $A = (a_{ij})$  the following holds,

$$\sup_{\vec{u}_i, \vec{v}_j \in B^{(n)}} \sum_{ij} a_{ij} \langle \vec{u}_i, \vec{v}_j \rangle \leq K_G \cdot \sup_{x_i, y_j \in [-1, 1]} \sum_{ij} a_{ij} x_i y_j,$$

where  $B^{(n)}$  is the unit ball in  $\mathbb{R}^n$ . Despite several efforts [?, ?], the value of the constant  $K_G$  remains unknown. The Grothendieck constant  $K_G$  is precisely the integrality gap of a natural SDP relaxation for the  $K_{N,N}$ -QUADRATICPROGRAMMING problem. The input to this problem is a matrix  $A = (a_{ij})$  and the objective is to maximize the quadratic form  $\sum_{ij} a_{ij} x_i y_j$  over  $x_i, y_j \in [-1, 1]$ .

In this work, we apply techniques from [?] to the BQPproblem. Using some standard but non-trivial modifications, the reduction in [?] yields the following hardness result: Assuming the Unique Games Conjecture, it is NP-hard to approximate the BQPproblem to any factor better than the Grothendieck constant  $K_G$ .

By adapting a “bootstrapping” argument used in a proof of Grothendieck inequality [?], we perform a tighter analysis than [?]. Through this careful analysis, we obtain the following new results:

- An approximation algorithm for BQP that is guaranteed to achieve an approximation ratio arbitrarily close to the Grothendieck constant  $K_G$  (optimal approximation ratio assuming the Unique Games Conjecture).
- We show that the Grothendieck constant  $K_G$  can be computed within an error  $\eta$ , in time depending only on  $\eta$ . Specifically, for each  $\eta$ , we formulate an explicit finite linear program, whose optimum is  $\eta$ -close to the Grothendieck constant.

We also exhibit a simple family of operators on the Gaussian Hilbert space that is guaranteed to contain tight examples for the Grothendieck inequality. To achieve this, we give a direct conversion from dictatorship tests to integrality gap instances bypassing the use of a reduction from UNIQUEGAMES and the Khot-Vishnoi [?] integrality gap instances for UNIQUEGAMES.

## 70 A Generic Top-Down Dynamic-Programming Approach to Prefix-Free Coding

Given a probability distribution over a set of  $n$  words, the Huffman-Coding problem is to find a minimal-cost prefix free code for transmitting/storing those words. Due to its nice structural properties, the basic Huffman-coding problem can be solved in  $O(n \log n)$  time but, because they lack these properties, variations of the problem are more difficult. A small number of such variations have been shown to be solvable using a bottom-up (on the code tree) dynamic programming approach but the most generic solution method utilizes a top-down dynamic-programming one. In this paper we show that the top-down approach can be sped up, permitting an order of magnitude improvement in the running time of many algorithms in the literature for such Huffman-coding variations as mixed radix, reserved length and one-ended coding. These speedups are immediate implications of a general structural property that permits batching together the calculation of many DP entries.

## 71 Additive Approximation Algorithms for List-Coloring Minor-Closed Class of Graphs

It is known that computing the list-chromatic number is much harder than the chromatic number (assuming that the complexity classes NP and coNP are different.). In fact, the problem of deciding if a given graph is  $f$ -list-colorable for a function  $f : V \rightarrow \{k-1, k\}$  for  $k \geq 3$  is  $\Pi_2^P$ -complete. In general, it is believed that approximating list-coloring is hard for dense graphs. In this paper, we are interested in sparse graphs. More specifically, we deal with minor-closed class of graphs, i.e., graphs without  $K_k$ -minors. We develop the seminal structure theorem of Robertson and Seymour, and then give an additive approximation for list-coloring within  $k-2$  of the list-chromatic number. This improves the  $O(k)$ -multiplicative-approximation algorithm by Kawarabayashi and Mohar (STOC'06). Clearly our result gives rise to an additive approximation algorithm for graph-coloring of minor-closed class of graphs too. This result is comparable to the 2-approximation algorithm of graph-coloring by Demaine, Hajiaghayi and Kawarabayashi (FOCS'05), since our additive approximation algorithm may give a better graph-coloring. This result also answers a question posed by Robin Thomas (who conjectured that there is an additive approximation algorithm for graph-coloring of minor-closed class of graphs). Our structure theorem is of independent interest in the sense that it gives rise to a new insight on well-connected  $H$ -minor-free graphs. In particular, this class of graphs can be easily decomposed into two parts so that one part has bounded tree-width and the other part is disjoint union of bounded genus graphs. Moreover, we can control the number of edges between two parts. Also, the proof method itself tells us how knowledge of a local structure can be used to

gain a global structure. This may be of interest, since this gives a new insight how to decompose a graph, with help of a local structure information.

## 72 A Nearly Linear Time Algorithm For The Half Integral Parity Disjoint Paths Packing Problem

We consider the following problem, which is called the *half integral parity disjoint paths packing problem*. **Input:** A graph  $G$ ,  $k$  pair of vertices  $(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)$  in  $G$  (which are sometimes called *terminals*), and a parity  $l_i$  for each  $i$  with  $1 \leq i \leq k$ , where  $l_i = 0$  or  $1$ . **Output :** Paths  $P_1, \dots, P_k$

in  $G$  such that  $P_i$  joins  $s_i$  and  $t_i$  for  $i = 1, 2, \dots, k$  and parity of length of the path  $P_i$  is  $l_i$ , i.e, if  $l_i = 0$ , then length of  $P_i$  is even, and if  $l_i = 1$ , then length of  $P_i$  is odd for  $i = 1, 2, \dots, k$ . In addition, each vertex is on at most two of these paths. We present an  $O(m\alpha(m, n) \log n)$  algorithm for fixed  $k$ , where

$n, m$  are the number of vertices and the number of edges, respectively, and the function  $\alpha(m, n)$  is the inverse of the Ackermann function. This is the first polynomial time algorithm for this problem, and generalizes polynomial time algorithms by Kleinberg (STOC'98) and Kawarabayashi and Reed (SODA'08), respectively, for the half integral disjoint paths packing problem, i.e., without the parity requirement. As with the Robertson-Seymour algorithm to solve the  $k$  disjoint paths problem, in each iteration, we would like to either use a huge clique minor as a "crossbar", or exploit the structure of graphs in which we cannot find such a minor. Here, however, we must maintain the parity of the paths and can only use an "odd clique minor". We must also describe the structure of those graphs in which we cannot find such a minor and discuss how to exploit it. In fact, we also have algorithms running in  $O(m^{1+\epsilon})$  time for any  $\epsilon > 0$  for this problem, if  $k$  is up to  $o(\log \log \log n)$  for general graphs, up to  $o(\log \log n)$  for planar graphs, and up to  $o(\log \log n/g)$  for graphs on the surface, where  $g$  is Euler genus. Furthermore, if  $k$  is fixed, then we have linear time algorithms for the planar case and for the bounded genus case.

## 73 Approximating Submodular Functions Everywhere

Submodular functions are a key concept in combinatorial optimization. Algorithms that involve submodular functions usually assume that they are given by a (value) oracle. Using only polynomially many queries to the oracle, one can minimize a submodular function or approximate its maximum within a constant. In this paper, we consider the problem of approximating a monotone submodular function  $f$  on a ground set of size  $n$  *everywhere*, after only polynomially many oracle queries. Our main result is a deterministic algorithm that make polynomially many oracle queries to derive a function  $\hat{f}$  of the form  $\hat{f}(S) = \sqrt{\sum_{i \in S} c_i}$  such that, for *every* set  $S$ ,  $\hat{f}(S)$  approximates  $f(S)$  within a factor  $\alpha(n)$ , where  $\alpha(n) = \sqrt{n+1}$  for rank functions of matroids and  $\alpha(n) = O(\sqrt{n} \log n)$  for general monotone submodular functions. Our result is based on approximately finding a maximum volume inscribed ellipsoid in a symmetrized polymatroid, and the analysis involves various properties of submodular functions and polymatroids. We should point out that the general algorithm of Grötschel, Lovász and Schrijver for finding a large inscribed ellipsoid in a centrally symmetric convex body given by a separation oracle would only give us an approximation factor of  $O(n)$ . Our algorithm is tight up to logarithmic factors. Indeed, we show that no algorithm can achieve a factor better than  $\Omega(\sqrt{n}/\log(n))$ , even for rank functions of a matroid.

## 74 Stepwise Randomized Combinatorial Auctions Achieve Revenue Monotonicity

In combinatorial auctions that use VCG, a seller can sometimes increase revenue by dropping bidders. In our previous work (AAAI'07), we showed that such failures of “revenue monotonicity” occur under an extremely broad range of deterministic strategyproof combinatorial auction mechanisms, even when bidders have “known single-minded” valuations. In this work we consider the question of whether revenue monotonic, strategyproof mechanisms for such bidders can be found in the broader class of randomized mechanisms. We demonstrate that—surprisingly—such mechanisms do exist, show how they can be constructed, and consider algorithmic techniques for implementing them in polynomial time. More formally, we characterize a class of randomized mechanisms defined for known single-minded bidders that are strategyproof and revenue monotonic, and furthermore satisfy some other desirable properties, namely participation, consumer sovereignty and maximality, representing the mechanism as a solution to a quadratically constrained linear program. We prove that the QCLP is always feasible (i.e., for all bidder valuations) and give its solution analytically. Furthermore, we give an algorithm for running such a mechanism in time quadratic in the number of bidders; this is interesting because constructing an instance of such mechanisms from our QCLP formulation in a naive way can be of exponential time.

**Ilia Binder and Mark Braverman. The complexity of simulating Brownian Motion** We analyze the complexity of the Walk on Spheres algorithm for simulating Brownian Motion in a domain  $\Omega$  in  $\mathbb{R}^d$ . The algorithm, which was first proposed in the 1950s, produces samples from the hitting probability distribution of the Brownian Motion process on boundary of  $\Omega$  within an error of  $\epsilon$ . The algorithm is used as a building block for solving a variety of differential equations, including the Dirichlet Problem. The WoS algorithm simulates a BM starting at a point  $X_0 = x$  in a given bounded domain  $\Omega$  until it gets  $\epsilon$ -close to the boundary of  $\Omega$ . At every step, the algorithm measures the distance  $d_k$  from its current position  $X_k$  to the boundary of  $\Omega$  and jumps a distance of  $d_k/2$  in a uniformly random direction from  $X_k$  to obtain  $X_{k+1}$ . The algorithm terminates when it reaches  $X_n$  that is  $\epsilon$ -close to the boundary of  $\Omega$ . It is not hard to see that the algorithm requires at least  $\Omega(\log 1/\epsilon)$  steps to converge. Only partial results with respect to upper bounds existed. In 1959 M. Motoo established an  $O(\log 1/\epsilon)$  bound on the running time for convex domains. The results were later generalized for a wider, but still very restricted, class of planar and 3-dimensional domains by G.A. Mikhailov (1979). In our earlier work (2007), we established an upper bound of  $O(\log^2 1/\epsilon)$  on the rate of convergence of WoS for arbitrary planar domains. In this paper we introduce subharmonic energy functions to obtain very general upper bounds on the convergence of the algorithm. Special instances of the upper bounds yield the following results for bounded domains  $\Omega$ : \* if  $\Omega$  is a planar domain with connected exterior, the WoS converges in  $O(\log 1/\epsilon)$  steps; \* if  $\Omega$  is a domain in  $\mathbb{R}^3$  with connected exterior, the WoS converges in  $O(\log^2 1/\epsilon)$  steps; \* for  $d > 2$ , if  $\Omega$  is a domain in  $\mathbb{R}^d$ , the WoS converges in  $O((1/\epsilon)^{2-4/d})$  steps; \* for  $d > 3$  if  $\Omega$  is a domain in  $\mathbb{R}^d$  with connected exterior, the WoS converges in  $O((1/\epsilon)^{2-4/(d-1)})$  steps; \* for any  $d$  if  $\Omega$  is a domain in  $\mathbb{R}^d$  bounded by a smooth surface, the WoS converges in  $O(\log 1/\epsilon)$  steps. We also demonstrate that the bounds are tight, i.e. we construct a domain from each class for which the upper bound is exact. Our results give the optimal upper bound of  $O(\log 1/\epsilon)$  in many cases for which only a bound polynomial in  $1/\epsilon$  was previously known. Prosenjit Bose, Eric Chen, Meng He, Anil Maheshwari and Pat Morin. **Succinct Geometric Indexes Supporting Point Location Queries** We propose to design data structures called succinct geometric indexes of negligible space (more precisely,  $o(n)$  bits) that, by taking advantage of the  $n$  points in the data set permuted and stored elsewhere as a sequence, to support geometric queries in optimal time. Our first and main result is a succinct geometric index that can answer point location queries, a fundamental problem in computational geometry, on planar triangulations in  $O(\lg n)$  time. We also design three variants of this index. The first

supports point location using  $\lg n + 2\sqrt{\lg n} + O(\lg^{1/4} n)$  point-line comparisons. The second supports point location in  $o(\lg n)$  time when the coordinates are integers bounded by  $U$ . The last variant can answer point location queries in  $O(H+1)$  expected time, where  $H$  is the entropy of the query distribution. These results match the query efficiency of previous point location structures that use  $O(n)$  words or  $O(n \lg n)$  bits, while saving drastic amounts of space. We generalize our succinct geometric index to planar subdivisions, and design indexes for other types of queries. Finally, we apply our techniques to design the first implicit data structures that support point location in  $O(\lg^2 n)$  time.

## 75 Monotone Minimal Perfect Hashing: Searching a Sorted Table with $O(1)$ Accesses

A minimal perfect hash function maps a set  $S$  of  $n$  keys into the set  $0, 1, \dots, n-1$  bijectively. Classical results state that minimal perfect hashing is possible in constant time using a structure occupying a number of bits close to the lower bound of  $\log e$  bits per element. Here we consider the problem of monotone minimal perfect hashing, in which the bijection is required to preserve the lexicographical ordering of the keys. A monotone minimal perfect hash function can be seen as a very weak form of index that provides just ranking on the set  $S$  (and answers randomly outside of  $S$ ). Our goal is to minimize the description size of the hash function: we show that, for a set  $S$  of  $n$  elements out of a universe of  $u$  elements,  $O(n \lg \lg \lg u)$  bits are sufficient to hash monotonically in time  $O(\lg \lg u)$ . Alternatively, we can get space  $O(n \lg \lg u)$  bits with  $O(1)$  query time. Both of these improve a straightforward construction with  $O(n \lg \lg u)$  space and  $O(\lg \lg u)$  query time. As a consequence, it is possible to search a sorted table with just one access to the table (using additional  $O(n \lg \lg \lg u)$  bits). Our results are based on a structure (of independent interest) that represents very compactly a trie, but allows errors. As a further application of the same structure, we show how to compute the predecessor of any element of a sorted table, using  $O(1)$  accesses in expectation and an index of  $O(n \lg \lg u)$  bits, improving the trivial result of  $O(n \lg u)$  bits. This implies an efficient index for searching a blocked memory.

## 76 Approximate Nearest Neighbors for Affine Subspaces Queries

We consider the problem of approximate nearest neighbors in high dimensions, when the queries are lines. In this problem, given  $n$  points in  $\mathbb{R}^d$ , we want to construct a data structure to support efficiently the following queries: given a line  $L$ , report the point  $p$  closest to  $L$ . This problem generalizes the more familiar nearest neighbor problem. From a practical perspective, lines, and low-dimensional flats in general, may model data under linear variation, such as physical objects under different lighting. For approximation  $1 + \epsilon$ , we achieve a query time of  $n^{0.5+t}$ , for arbitrary small  $t > 0$ , with a polynomial space. To the best of our knowledge, this is the first algorithm for this problem with polynomial space and sub-linear query time.

## 77 Collecting Weighted Items from a Dynamic Queue

We consider the problem of collecting weighted items from a dynamic queue  $S$ . Before each step, some items at the front of  $S$  can be deleted and some other items can be added to  $S$  at any place. An item, once deleted, cannot be re-inserted – in other words, it “expires”. We are allowed to collect one item from  $S$  per step. Each item can be collected only once. The objective is to maximize the total weight of the collected items. We study the online version of the dynamic queue problem. It is quite easy to see that the greedy algorithm that always collects the maximum-value item is 2-competitive,

and that no deterministic online algorithm can be better than 1.618-competitive. We improve both bounds: We give a 1.89-competitive algorithm for general dynamic queues and we show a lower bound of 1.632 on the competitive ratio. We also provide other upper and lower bounds for restricted versions of this problem. The dynamic queue problem is a generalization of the well-studied buffer management problem, and it is an abstraction of the buffer management problem for network links with intermittent access.

## 78 The Uniform Hardcore Lemma via Approximate Bregman Projections

We give a simple, more efficient and uniform proof of the hard-core lemma, a fundamental result in complexity theory with applications in machine learning and cryptography. Our result follows from the connection between boosting algorithms and hard-core set constructions discovered by Klivans and Servedio [KS03]. Informally stated, our result is the following: suppose we fix a family of boolean functions. Assume there is an efficient algorithm which for every input length and every smooth distribution (i.e. one that doesn't assign too much weight to any single input) over the inputs produces a circuit such that the circuit computes the boolean function noticeably better than random. Then, there is an efficient algorithm which for every input length produces a circuit that computes the function correctly on almost all inputs. Our algorithm significantly simplifies previous proofs of the uniform and the non-uniform hard-core lemma, while matching or improving the previously best known parameters. The algorithm uses a generalized multiplicative update rule combined with a natural notion of approximate Bregman projection. Bregman projections are widely used in convex optimization and machine learning. We present an algorithm which efficiently approximates the Bregman projection onto the set of high density measures when the Kullback-Leibler divergence is used as a distance function. Our algorithm has a logarithmic runtime over any domain provided that we can efficiently sample from this domain. High density measures correspond to smooth distributions which arise naturally, for instance, in the context of online learning. Hence, our technique may be of independent interest.

## 79 Transitive-Closure Spanners

We define the notion of a transitive-closure spanner of a directed graph. Given a directed graph  $G = (V; E)$  and an integer  $k \geq 0$ , a  $k$ -transitive-closure-spanner ( $k$ -TC-spanner) of  $G$  is a directed graph  $H = (V; E')$  that has (1) the same transitive-closure as  $G$  and (2) diameter at most  $k$ . These spanners were studied implicitly in access control, property testing, and data structures, and properties of these spanners have been rediscovered over the span of 20 years. We bring these areas under the unifying framework of TC-spanners. We abstract the common task implicitly tackled in these diverse applications as the problem of constructing sparse TC-spanners. We study the approximability of the size of the sparsest  $k$ -TC-spanner for a given digraph. Our technical contributions fall into three categories: algorithms for general digraphs, inapproximability results, and structural bounds for a specific graph family which imply an efficient algorithm with a good approximation ratio for that family. [Algorithms.] We present two efficient deterministic algorithms that find  $k$ -TC-spanners of size approximating the optimum. The first algorithm gives an  $O(\tilde{n}^{1-1/k})$ -approximation for  $k \geq 2$ . Our method, based on a combination of convex programming and sampling, yields the first sub-linear approximation ratios for (1) DIRECTED  $k$ -SPANNER, a well-studied generalization of  $k$ -TC-SPANNER, and (2) its variants CLIENT/SERVER DIRECTED  $k$ -SPANNER, and the  $k$ -DIAMETER SPANNING SUBGRAPH. This resolves the main open question of Elkin and Peleg (IPCO, 2001). The second algorithm, specific to the  $k$ -TC-spanner problem, gives an  $O(\tilde{n}/(k^2))$ -approximation. It shows that for  $k = \Omega(\sqrt{n})$ , our problem has a provably better approximation ratio than DIRECTED

k-SPANNER and its variants. This algorithm also resolves an open question of Hesse (SODA, 2003). [Inapproximability.] Our main technical contribution is a pair of strong inapproximability results. We resolve the approximability of 2-TC-spanners, showing that it is  $\Theta(\log n)$  unless  $P=NP$ . For constant  $k > 2$ , we prove that the size of the sparsest  $k$ -TC-spanner is hard to approximate within  $2^{\log^{1-\epsilon} n}$ , for any  $\epsilon > 0$ , under standard complexity assumptions. Our hardness result helps explain the difficulty in designing general efficient solutions for the applications above, and it cannot be improved without resolving a long-standing open question in complexity theory. It uses an involved application of generalized butterfly and broom graphs, as well as noise-resilient transformations of hard problems, which may be of independent interest. [Structural bounds.] Finally, we study the size of the sparsest TC-spanner for  $H$ -minor-free digraphs, which include planar, bounded genus, and bounded tree-width graphs, explicitly investigated in applications above. We show that every  $H$ -minor-free digraph has an efficiently constructable  $k$ -TC-spanner of size  $O(\tilde{n})$ , which implies an  $O(\tilde{1})$ -approximation algorithm for this family. Furthermore, using our insight that 2-TC-spanners yield property testers, we obtain a monotonicity tester with  $O(\log^2 n/\epsilon)$  queries for any poset whose transitive reduction is an  $H$ -minor free digraph. This improves and generalizes the previous  $\Theta(\sqrt{n} \log n/\epsilon)$ -query tester of Fischer et al (STOC, 2002).

## 80 On the Complexity of Nash Equilibria of Action-Graph Games

In light of much recent interest in finding a model of multi-player multi-action games that allows for efficient computation of Nash equilibria yet remains as expressive as possible, we investigate the computational complexity of Nash equilibria in the recently proposed model of Action-Graph Games (AGGs). AGGs, introduced by Bhat and Leyton-Brown, are succinct representations of games that encapsulate both local dependencies as in graphical games, and partial indifference to other agents' identities as in anonymous games, which occur in many natural settings such as financial markets. This is achieved by specifying a graph on the set of actions, so that the payoff of an agent for selecting a strategy depends only on the number of agents playing each of the neighboring strategies in the action graph. We present a simple Polynomial Time Approximation Scheme for computing mixed Nash equilibria of AGGs with constant treewidth and a constant number of agent types (but an arbitrary number of strategies). However, the main results of this paper are negative, showing that when either of these conditions is relaxed the problem becomes intractable. In particular, we show that even if the action graph is a tree but the number of agent-types is unconstrained, it is NP-complete to decide the existence of a pure-strategy Nash equilibrium and PPAD-complete to compute a mixed Nash equilibrium (even an approximate one). Analogously, for AGGs where all agents are of a single type (symmetric AGGs), if the treewidth is unconstrained then deciding the existence of a pure-strategy Nash equilibrium is NP-complete and computing a mixed Nash is PPAD-complete. These hardness results suggest that, in some sense, our PTAS is as strong a positive result as one can expect. In the broader context of trying to pin down the boundary where the equilibria of multi-player games can be computed efficiently, these results complement recent hardness results for graphical games and algorithmic results for anonymous games.

## 81 Overcoming the $L_1$ Non-Embeddability Barrier: Algorithms for Product Metrics

A common approach for solving computational problems over a difficult metric space is to embed the "hard" metric into  $L_1$ , which admits efficient algorithms and is thus considered an "easy" metric. This approach has proved successful or partially successful for important spaces such as the edit distance, but it also has inherent limitations: it is provably impossible to go below certain approximation for

some metrics. We propose a new approach, of embedding the difficult space into richer host spaces, namely iterated products of standard spaces like  $L_1$  and  $L_\infty$ . We show that this class is /rich/ since it contains useful metric spaces with only a constant distortion, and, at the same time, it is /tractable/ and admits efficient algorithms. Using this approach, we obtain for example the first nearest neighbor data structure with  $O(\log \log d)$  approximation for edit distance in non-repetitive strings (the Ulam metric). This approximation is exponentially better than the *lower bound* for embedding into  $L_1$ . Furthermore, we give constant factor approximation for two other computational problems. Along the way, we answer positively a question left open in [Ajtai-Jayram-Kumar-Sivakumar, STOC'02]. One of our algorithms has already found applications for smoothed edit distance over 0-1 strings [Andoni-Krauthgamer, ICALP'08].

## 82 Finding repeats in a data-stream

Given a string of length  $n$  over the alphabet  $[m]$  where  $n > m$ , we consider the problem of finding a repeated element in a single pass. We give a randomized algorithm for this problem that uses  $O((\log m)^4)$  space. In contrast, we show that any deterministic single-pass algorithm for finding a repeated element needs space at least  $m$ , even if  $n$  is arbitrarily large compared to  $m$ . However, we show that when we have access to a clock that records the number of input symbols read so far, the space required drops to  $\log m + O(1)$  if  $n \geq 2^m$ ; thus for this problem non-uniformity adds considerable power to deterministic algorithms. For non-uniform deterministic algorithms we show that finding a repeated element in a stream of length  $n$  requires space at least  $m - \log n$ ; we show that this lower bound is close to optimal for a range of values of  $n$ .

## 83 Approximate Shared-Memory Counting Despite a Strong Adversary

A new randomized asynchronous shared-memory data structure is given for implementing an approximate counter that can be incremented up to  $n$  times. For any fixed  $\epsilon$ , the counter achieves a relative error of  $\delta$  with high probability, at the cost of  $O(((1/\delta) \log n)^{O(1/\epsilon)})$  register operations per increment and  $O(n^{4/5+\epsilon}((1/\delta) \log n)^{O(1/\epsilon)})$  register operations per read. The counter combines randomized sampling for estimating large values with an expander for estimating small values. This is the first sublinear solution to this problem that works despite a strong adversary scheduler that can observe internal states of processes. An application of the improved counter is an improved protocol for solving randomized shared-memory consensus, which reduces the best previously known individual work complexity from  $O(n \log n)$  to an optimal  $O(n)$ , resolving one of the last remaining open problems concerning consensus in this model.

## 84 On Low Dimensional Local Embeddings

We study the problem of embedding metric spaces into low dimensional  $L_p$  spaces while faithfully preserving distances from each point to its  $k$  nearest neighbors. We show that any metric space can be embedded into  $L_p^{O(e^p \log^2 k)}$  with  $k$ -local distortion of  $O((\log k)/p)$ . We also show that any ultrametric can be embedded into  $L_p^{O(\log k)/\epsilon^3}$  with  $k$ -local distortion  $1 + \epsilon$ . Our embedding results have immediate applications to local Distance Oracles. We show how to preprocess a graph in polynomial time to obtain a data structure of  $O(nk^{1/t} \log^2 k)$  bits, such that distance queries from any node to its  $k$  nearest neighbors can be answered with stretch  $t$ .

## 85 Online Topological Ordering

Let  $G=(V,E)$  be a directed acyclic graph (dag) with  $n = |V|$  and  $m = |E|$ . We say that a bijection  $P$  from  $V$  to the integers  $1,2,\dots,n$  is a "topological ordering" if for every edge  $(u,v)$  in  $E$ , we have  $P(u) < P(v)$ . In this paper, we consider the problem of maintaining a topological ordering subject to dynamic changes to the underlying graph. That is, we begin with an empty graph  $G = (V, \emptyset)$  consisting of  $n$  nodes. The adversary adds  $m$  edges to the graph  $G$ , one edge at a time. Throughout this process, we maintain an online topological ordering of the graph  $G$ . In this paper, we present a new algorithm for online topological ordering that has total cost  $O(n^2 \log^2 n)$  for all edge additions. At the heart of our algorithm is a new approach for maintaining a topological ordering that differs from previous algorithms. Instead of attempting to maintain the nodes in an ordered list, we assign each node a label that preserves the appropriate ordering, and yet can be updated efficiently as edges are inserted. When the graph is dense, our algorithm is more efficient than all existing algorithms. By way of contrast, the best known prior algorithms achieve only  $O(\min(m^{1.5}, n^{2.5}))$  cost.

## 86 The Johnson-Lindenstrauss lemma almost characterizes Hilbert space, but not quite

Let  $X$  be a normed spaces that satisfies the Johnson-Lindenstrauss lemma, i.e. for any integer  $n$  and any  $x_1, \dots, x_n \in X$  there exists a linear mapping  $L : X \rightarrow F$ , where  $F \subseteq X$  is a linear subspace of dimension  $O(\log n)$ , such that  $\|x_i - x_j\| \leq \|L(x_i) - L(x_j)\| \leq O(1) \cdot \|x_i - x_j\|$  for all  $i, j \in \{1, \dots, n\}$ . We show that this implies that  $X$  is almost Euclidean in the following sense: Any  $n$ -dimensional subspace of  $X$  embeds into Hilbert space with distortion  $2^{2^{O(\log^* n)}}$ . On the other hand, we show that there exists a normed space  $Y$  which satisfies the Johnson-Lindenstrauss lemma, but for every  $n$  there exists an  $n$ -dimensional subspace  $E_n \subseteq Y$  whose Euclidean distortion is at least  $2^{\Omega(\alpha(n))}$ , where  $\alpha$  is the inverse of the Ackermann function.

## 87 Parameterized approximation scheme for the multiple knapsack problem

The multiple knapsack problem (MKP) is a well-known generalization of the classical knapsack problem. We are given a set  $A$  of  $n$  items and set  $B$  of  $m$  bins (knapsacks) such that each item  $a \in A$  has a size  $\text{size}(a)$  and a profit value  $\text{profit}(a)$ , and each bin  $b \in B$  has a capacity  $c(b)$ . The goal is to find a subset  $U \subseteq A$  of maximum total profit such that  $U$  can be packed into  $B$  without exceeding the capacities. The decision version of MKP is strongly NP-complete, since it is a generalization of the classical knapsack and bin packing problem. Furthermore, MKP does not admit an FPTAS even if the number  $m$  of bins is two. Kellerer gave a PTAS for MKP with identical capacities and Chekuri and Khanna presented a PTAS for MKP with general capacities with running time  $n^{O(\log(1/\epsilon)/\epsilon^8)}$ . In this paper we propose an EPTAS with parameterized running time  $2^{O(\log(1/\epsilon)/\epsilon^5)} \cdot \text{poly}(n) + O(m)$  for MKP.

## 88 Inserting a Vertex into a Planar Graph

We consider the problem of computing a crossing minimum drawing of a given planar graph  $G = (V, E)$  augmented by a star, i.e., an additional vertex  $v$  together with its incident edges  $E_v = \{(v, u) \mid u \in V\}$ , in which all crossings involve  $E_v$ . Alternatively, the problem can be stated as finding a planar embedding of  $G$ , in which the given star can be inserted requiring the minimum number of crossings. This is a generalization of the crossing minimum edge insertion problem [12], and can help to find

improved approximations for the crossing minimization problem. Indeed, in practice, the algorithm for the crossing minimum edge insertion problem turned out to be the key for obtaining the currently strongest approximate solutions for the crossing number of general graphs. The generalization considered here can lead to even better solutions for the crossing minimization problem. Furthermore, it offers new insight into the crossing number problem for almost-planar and apex graphs. It has been an open problem whether the star insertion problem is polynomially solvable. We give an affirmative answer by describing the first efficient algorithm for this problem. This algorithm uses the SPQR-tree data structure to handle the exponential number of possible embeddings, in conjunction with dynamic programming schemes for which we introduce *partitioning cost* subproblems.

## 89 Paging and List Update under Bijective Analysis

It has long been known that for the paging problem in its standard form, competitive analysis cannot adequately distinguish algorithms based on their performance: there exists a vast class of algorithms which achieve the same competitive ratio, ranging from extremely naive and inefficient strategies (such as Flush-When-Full), to strategies of excellent performance in practice (such as Least-Recently-Used and some of its variants). A similar situation arises in the list update problem: in particular, under the cost formulation studied by Martinez and Roura [TCS 2000] and Munro [ESA 2000] every list update algorithm has, asymptotically, the same competitive ratio. Several refinements of competitive analysis, as well as alternative performance measures have been introduced in the literature, with varying degrees of success in narrowing this disconnect between theoretical analysis and empirical evaluation. In this paper we study these two fundamental online problems under the framework of bijective analysis [Angelopoulos, Dorrigiv and Lopez-Ortiz, SODA 2007 and LATIN 2008]. This is an intuitive technique which is based on pairwise comparison of the costs incurred by two algorithms on sets of request sequences of the same size. Coupled with a well-established model of locality of reference due to Albers, Favrholt and Giel [JCSS 2005], we show that Least-Recently-Used and Move-to-Front are the unique optimal algorithms for paging and list update, respectively. Prior to this work, only measures based on average-cost analysis have separated LRU and MTF from all other algorithms. Given that bijective analysis is a fairly stringent measure (and also subsumes average-cost analysis), we prove that in a strong sense LRU and MTF stand out as the best algorithms.

## 90 On stars and Steiner stars II

A *Steiner star* for a set  $P$  of  $n$  points in  $\mathbb{R}^d$  connects an arbitrary center point to all points of  $P$ , while a *star* connects a point  $p \in P$  to the remaining  $n - 1$  points of  $P$ . All connections are realized by straight line segments. Fekete and Meijer showed that the minimum star is at most  $\sqrt{2}$  times longer than the minimum Steiner star for any finite point configuration in  $\mathbb{R}^d$ . The maximum ratio between them, over all finite point configurations in  $\mathbb{R}^d$ , is called the *star Steiner ratio* in  $\mathbb{R}^d$ . It is conjectured that this ratio is  $4/\pi = 1.2732\dots$  in the plane and  $4/3 = 1.3333\dots$  in three dimensions. Here we give upper bounds of 1.3631 in the plane, and 1.3833 in 3-space, thereby substantially improving recent upper bounds of 1.3999, and  $\sqrt{2} - 10^{-4}$ , respectively. Our results also imply improved bounds on the maximum ratios between the minimum star and the maximum matching in two and three dimensions. Our method exploits the connection with the classical problem of estimating the maximum sum of pairwise distances among  $n$  points on the unit sphere, first studied by László Fejes Tóth. It is quite general and yields the first non-trivial estimates below  $\sqrt{2}$  on the star Steiner ratios in arbitrary dimensions. We show, however, that the star Steiner ratio in  $\mathbb{R}^d$  tends to  $\sqrt{2}$ , the upper bound given by Fekete and Meijer, as  $d$  goes to infinity. Our estimates on the star Steiner ratios are therefore

much closer to the conjectured values in higher dimensions! As it turns out, our estimates as well as the conjectured values of the Steiner ratios (in the limit, for  $n$  going to infinity) are related to the classical infinite Wallis product:  $\frac{\pi}{2} = \prod_{n=1}^{\infty} \left( \frac{4n^2}{4n^2-1} \right) = \frac{2}{1} \cdot \frac{2}{3} \cdot \frac{4}{3} \cdot \frac{4}{5} \cdot \frac{6}{5} \cdot \frac{6}{7} \cdot \frac{8}{7} \cdot \frac{8}{9} \cdots$

## 91 Reasoning about Online Algorithms with Weighted Automata

We describe an automata-theoretic approach for the competitive analysis of *online algorithms*. Our approach is based on *weighted automata*, which assign to each input word a cost in  $\mathbb{R}^{\geq 0}$ . By relating the “unbounded look ahead” of optimal offline algorithms with nondeterminism, and relating the “no look ahead” of online algorithms with determinism, we are able to solve problems about the competitive ratio of online algorithms, and the memory they require, by reducing them to questions about *determinization* and *approximated determinization* of weighted automata.

## 92 On smoothed k-CNF formulas and the Walksat algorithm

In this paper we study the model of  $\epsilon$ -smoothed  $k$ -CNF formulas. Starting from an arbitrary instance  $F$  with  $n$  variables and  $m = dn$  clauses, apply the  $\epsilon$ -smoothing operation of flipping the polarity of every literal in every clause independently at random with probability  $\epsilon$ . Keeping  $\epsilon$  and  $k$  fixed, and letting the density  $d = m/n$  grow, it is rather easy to see that when  $d = c\epsilon^{-k}$  (where  $c$  is some sufficiently large constant that depends only on  $k$ ), every  $F$  will whp become unsatisfiable after smoothing. We show that a lower density that behaves roughly like  $\epsilon^{-k+1}$  suffices for this purpose. We also show that our bound on  $d$  is nearly best possible in the sense that there are  $k$ -CNF formulas  $F$  of slightly lower density that whp remain satisfiable after smoothing. Moreover, the well known Walksat algorithm will whp find a satisfying assignments in these smoothed formulas (whenever  $\epsilon \geq 1/k$ ). Our proof (when setting  $\epsilon = 1/2$ ) gives a new lower bound of  $\Omega(2^k/k^2)$  on the density in which for most  $k$ -CNF formulas the Walksat algorithm whp terminates successfully in polynomial time. We are not aware of any previous rigorous analysis that shows that Walksat is successful at densities that are increasing as a function of  $k$ .

## 93 How hard is it to approximate the best Nash equilibrium?

The quest for a PTAS for Nash equilibrium in a two-player game, which emerged as a major open question in Algorithmic Game Theory, seeks to circumvent the PPAD-completeness of an (exact) Nash equilibrium but finding an approximate equilibrium. A related problem of special interest is that of finding an equilibrium maximizing a certain objective, such as the social welfare. This optimization problem was shown to be NP-hard by Gilboa and Zemel [Games and Economic Behavior 1989]. However, this NP-hardness is unlikely to extend to finding an approximate equilibrium, since the latter admits a quasi-polynomial time algorithm, as proved by Lipton, Markakis and Mehta [EC, 2003]. We show that this optimization problem, namely, finding in a two-player game an approximate equilibrium achieving large social welfare is unlikely to have a polynomial time algorithm. One interpretation of our results is that the quest for a PTAS for Nash equilibrium should NOT extend to a PTAS for finding the best Nash equilibrium, which stands in contrast to certain algorithmic techniques used so far (e.g. sampling and enumeration). Technically, our result is a reduction from a notoriously difficult problem in modern combinatorics, of finding a planted (but hidden) clique in a random graph  $G(n, 1/2)$ . Our reduction starts from an instance with planted clique size  $k = O(\log n)$ . For comparison, the currently known algorithms due to Alon, Krivelevich and Sudakov [RSA, 1998], and Krauthgamer and Feige [RSA, 2000], are only effective for a much larger clique size  $k = \Omega(\sqrt{n})$ .

## 94 The Ratio Index for Budgeted Learning, with Applications

In the budgeted learning problem, we are allowed to experiment on a set of alternatives (given a fixed experimentation budget) with the goal of picking a single alternative with the largest possible expected payoff. Approximation algorithms for this problem were developed by Guha and Munagala by rounding a linear program that couples the various alternatives together. In this paper we present an index for this problem, which we call the ratio index, which also guarantees a constant factor approximation. Index based policies have the advantage that a single number (i.e. the index) can be computed for each alternative irrespective of all other alternatives, and the alternative with the highest index is experimented upon. This is analogous to the famous Gittins index for the discounted multi-armed bandit problem. The ratio index has several interesting structural properties. First, we show that it can be computed in strongly polynomial time; the techniques we develop may be useful for deriving strongly polynomial algorithms for other Markov Decision Problems. Second, we show that with the appropriate discount factor, the Gittins index and our ratio index are constant factor approximations of each other, and hence the Gittins index also gives a constant factor approximation to the budgeted learning problem. Finally, we show that the ratio index can be used to create an index based policy that achieves an  $O(1)$ -approximation for the finite horizon version of the multi-armed bandit problem. Moreover, the policy does not require any knowledge of the horizon (whereas we compare its performance against an optimal strategy that is aware of the horizon). This yields the following surprising result: there is an index based policy that achieves an  $O(1)$ -approximation for the multi-armed bandit problem, oblivious to the underlying discount factor.

## 95 A universally fastest algorithm for Max 2-Sat, Max 2-CSP, and everything in between

In this paper we consider formulas consisting of "simple clauses", namely conjunctions and disjunctions of pairs of variables, and general integer-weighted 2-variable clauses, which can be any functions of pairs of variables. We present an exact (exponential-time) algorithm which is the fastest polynomial-space algorithm for Max 2-Sat, tied for fastest for Max 2-CSP, and fastest for everything in between. That is, parametrizing an instance by the fraction  $p$  of non-simple clauses, we give a single algorithm which is the fastest known for  $p=0$  (which includes the well-studied Max 2-Sat problem but also instances with arbitrary mixtures of AND and OR clauses); the only efficient algorithm for mixtures of AND, OR, and general integer-weighted clauses; and tied for fastest for general Max 2-CSP ( $p=1$ ). Since a pure 2-Sat input instance may be transformed to a general instance in the course of being solved, the algorithm's efficiency and generality go hand in hand. Our novel analysis results in a family of running-time bounds, each optimized for a particular value of  $p$ , but valid for all  $p$ . The algorithm uses some new reductions introduced here, as well as recent reductions such as "clause-learning" and "2-reductions" adapted to our setting's mixture of simple and general clauses. Each reduction imposes constraints on various parameters, the running-time bound is an "objective function" of these parameters and  $p$ , and the optimal running-time bound is obtained by solving a convex nonlinear program, which can be done efficiently and with a certificate of optimality.

## 96 Decomposition of Multiple Coverings into More Parts

We prove that for every centrally symmetric convex polygon  $Q$ , there exists a constant  $\alpha$  such that any  $\alpha \cdot k$ -fold covering of the plane by translates of  $Q$  can be decomposed into  $k$  coverings. This improves on a quadratic upper bound proved by Pach and Toth (SoCG'07). The question is motivated

by a sensor network problem, in which a region has to be monitored by sensors with limited battery lifetime.

## 97 Computing the nucleolus of weighted voting games

Weighted voting games (WVG) are coalitional games in which an agent's contribution to a coalition is given by his *weight*, and a coalition wins if its total weight meets or exceeds a given quota. These games model decision-making in political bodies as well as collaboration and surplus division in multiagent domains. The computational complexity of various solution concepts for weighted voting games received a lot of attention in recent years. In particular, Elkind et al.(2007) studied the complexity of stability-related solution concepts in WVGs, namely, of the core, the least core, and the nucleolus. While they have completely characterized the algorithmic complexity of the core and the least core, for the nucleolus they have only provided an NP-hardness result. In this paper, we solve an open problem posed by Elkind et al. by showing that the nucleolus of WVGs, and, more generally,  $k$ -vector weighted voting games with fixed  $k$ , can be computed in pseudopolynomial time, i.e., there exists an algorithm that correctly computes the nucleolus and runs in time polynomial in the number of players  $n$  and the maximum weight  $W$ . In doing so, we propose a general framework for computing the nucleolus, which may be applicable to a wider of class of games.

## 98 Maximum Biconnected Subgraphs of Random Planar Graphs

Let  $CP_n$  be the class of simple labeled planar graphs with  $n$  vertices, and denote by  $SP_n$  a graph drawn uniformly at random from this set. Basic properties of  $SP_n$  were first investigated by Denise, Vasconcellos, and Welsh [dvw96]. Since then, the random planar graph has attracted considerable attention, and is nowadays an important and challenging model for evaluating methods that are developed to study properties of random graphs from classes with structural side constraints. In this paper we study the structure of  $SP_n$ . More precisely, let  $b(\ell; SP_n)$  be the number of blocks (i.e. maximal biconnected subgraphs) of  $SP_n$  that contain exactly  $\ell$  vertices, and let  $lb(SP_n)$  be the number of vertices in the largest block of  $SP_n$ . We show that with high probability  $SP_n$  contains a *giant* block that includes up to lower order terms  $cn$  vertices, where  $c \approx 0.959$  is an analytically given constant. Moreover, we show that the *second* largest block contains only  $n^{2/3} \text{polylog}(n)$  vertices, and prove sharp concentration results for  $b(\ell; SP_n)$ , for all  $2 \leq \ell \leq n^{2/3}$ . In fact, we obtain this result as a consequence of a much more general result that we prove in this paper. Let  $CG$  be a class of labeled connected graphs, and let  $SG_n$  be a graph drawn uniformly at random from graphs in  $CG$  that contain exactly  $n$  vertices. Under certain assumptions on  $CG$ , and depending on the behavior of the singularity of the generating function enumerating the elements of  $CG$ ,  $SG_n$  belongs with high probability to one of the following three categories, which differ vastly in complexity.  $SG_n$  either

- (1) behaves like a random planar graph, i.e.  $lb(SG_n) \sim cn$ , for some analytically given  $c = c(CG)$ , and the second largest block is of order  $n^\alpha$ , where  $1 > \alpha = \alpha(CG)$ , or
- (2)  $lb(SG_n) = \mathcal{O}(\log n)$ , i.e., all blocks contain at most logarithmically many vertices, or
- (3)  $lb(SG_n) = \Theta(n^\alpha)$ , for some  $\alpha = \alpha(CG)$ .

Planar graphs belong to category (1). In contrast to that, outerplanar and series-parallel graphs belong to category (2), and differ thus substantially from planar graphs. Our proofs exploit recent progress in the development of Boltzmann samplers by Duchon, Flajolet, Louchard and Schaeffer [dfls04] and basic tools from modern probability theory.

## 99 List-Color-Critical Graphs on a Fixed Surface

We prove the following conjecture posed by Thomassen in 1994: There are only finitely many list-color-critical graphs with all lists of cardinality 5 on a fixed surface. This generalizes the well-known result of Thomassen on the usual graph coloring case. We use this theorem to address the complexity status of the following question on list-coloring graphs on a fixed surface  $S$ . **Input:** A graph  $G$  in the surface  $S$ . **Output:** Is  $G$   $k$ -list-colorable? If not, provide a certificate (a list-color-critical subgraph) for this. If yes, given any lists of cardinality  $k$ , then give a  $k$ -list-coloring. In fact, together with our recent algorithmic result, we give a linear time algorithm for the problem when  $k \geq 5$ . The cases  $k = 3, 4$  are known to be NP-hard, and the cases  $k = 1, 2$  are easy. This generalizes the well-known linear time algorithms for planar graphs by Nishizeki and Chiba (for 5-coloring), and Thomassen (for 5-list-coloring). We also give a polynomial time algorithm to address the following algorithmic question: **Input:** A graph  $G$  in the surface  $S$ , and lists of cardinality  $k \geq 5$ . **Output:** Does  $G$  have a valid coloring from the lists? If not, provide a certificate for this. If yes, then give a  $k$ -list-coloring. This is different from the first algorithmic one in the sense that some list-color-critical graph  $H$  may have a valid coloring from some lists of  $H$ . Therefore, the second one is strictly stronger than the first one. We give a polynomial time algorithm for this problem too. We also use our main theorem to prove the following conjecture, made recently by Thomassen: For every fixed surface  $S$ , there exists a positive constant  $c$  such that every 5-list-colorable graph with  $n$  vertices embedded on  $S$ , and any lists of cardinality 5, has at least  $c \times 2^n$  distinct 5-list-colorings. Thomassen himself proved that this conjecture holds for the usual 5-coloring. In addition to these results, we also made partial progress toward a conjecture of Albertson concerning coloring extension.

## 100 Maximizing Submodular Set Functions Subject to Multiple Linear Constraints

The concept of submodularity plays a vital role in combinatorial optimization. In particular, many important optimization problems can be cast as submodular maximization problems, including maximum coverage, maximum facility location and Max Cut in directed/undirected graphs. In this paper we present the first known approximation algorithms for the problem of maximizing a non-decreasing submodular set function subject to multiple linear constraints. Given an oracle for a non-decreasing submodular set function  $f$  over a universe  $\mathcal{U}$ , where each element  $e \in \mathcal{U}$  is associated with a  $d$ -dimensional cost vector and a  $d$ -dimensional budget vector  $b_L$ , for some  $d \geq 1$ , we seek a subset of elements  $\mathbf{S} \subseteq \mathcal{U}$  whose total cost is at most  $b_L$ , such that  $f(\mathbf{S})$  is maximized. We develop a framework for maximizing submodular functions subject to  $d$  linear constraints that yields a  $(1 - \epsilon)(1 - e^{-1})$ -approximation to the optimum for any  $\epsilon > 0$ , where  $d > 1$  is some constant. Our study is motivated from a variant of the classical maximum coverage problem that we call *maximum coverage with multiple packing constraints*. We use our framework to obtain the same approximation ratio for this problem. To the best of our knowledge, this is the first time the theoretical bound of  $1 - e^{-1}$  is (almost) matched for both of these problems.

## 101 Coresets and Approximate Clustering for Bregman Divergences

We study the generalized  $k$ -median problem with respect to a Bregman divergence  $D_\phi$ . Given a finite set  $P \subset \mathbb{R}^d$  of size  $n$ , our goal is to find a set  $C$  of size  $k$  such that the sum of errors

$\text{cost}(P, C) = \sum_{p \in P} \min_{c \in C} D_\phi(p, c)$  is minimized. The Bregman  $k$ -median problem plays an important role in many applications, e.g. information theory, statistics, text classification, and speech processing. We give the first coresets construction for this problem for a large subclass of Bregman divergences, including important dissimilarity measures such as the Kullback-Leibler divergence and the Itakura-Saito divergence. Using these coresets, we give a  $(1 + \epsilon)$ -approximation algorithm for the Bregman  $k$ -median problem with running time  $O(dkn + d^{2(k/\epsilon)^{O(1)}} \log^{k+2} n)$ . This result improves over the previously fastest known  $(1 + \epsilon)$ -approximation algorithm from [ABS08]. Unlike the analysis of most coresets constructions our analysis does not rely on the construction of  $\epsilon$ -nets. Instead, we prove our results by purely combinatorial means. References [ABS08] M. R. Ackermann, J. Blömer, and C. Sohler. Clustering for metric and non-metric distance measures. In Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '08), pages 799–808, 2008.

## 102 Online Scheduling to Minimize the Maximum Delay Factor

In this paper two forms of information dissemination or scheduling are addressed. First is the standard model (unicast) where requests (or jobs) are independent. The other is the broadcast model where broadcasting a page can satisfy multiple outstanding requests for that page. We consider online scheduling of requests when they have deadlines. Unlike previous models which mainly consider the objective of maximizing throughput while respecting deadlines, here we focus on scheduling all jobs with the goal of minimizing the maximum *delay factor*. The delay factor of a schedule,  $\alpha$ , is defined to be the minimum  $\alpha \geq 1$  such that each job  $i$  is completed by time  $a_i + \alpha(d_i - a_i)$  where  $d_i$  is the deadline of request  $i$  and  $a_i$  is the arrival time of request  $i$ . Delay factor generalizes the previously defined measure of maximum stretch which is based only the processing times of requests. We prove strong lower bounds on the achievable competitive ratios for delay factor scheduling even with unit-time requests. Motivated by this, we consider resource augmentation analysis and prove the following positive results. For the unicast model we give algorithms that are  $(1 + \epsilon)$ -speed  $O(\frac{1}{\epsilon})$ -competitive in both the single machine and multiple machine settings. In the broadcast model we give an algorithm for similar-sized pages that is  $(1 + \epsilon)$ -speed  $O(\frac{1}{\epsilon^2})$ -competitive. Online algorithms have been difficult to design for the broadcast setting and ours is the first to give a  $(1 + \epsilon)$ -speed analysis for a natural measure of performance.

## 103 Approximate Clustering without the Approximation

Approximation algorithms for clustering points in metric spaces is a flourishing area of research, with much research effort spent on getting a better understanding of the approximation guarantees possible for many objective functions such as  $k$ -median,  $k$ -means, and min-sum clustering. This quest for better approximation algorithms is further fueled by the implicit hope that these better approximations also give us better clusterings. E.g., for many problems such as clustering proteins by function, or clustering images by subject, there is some unknown “correct” target clustering—one that clusters proteins by function, or that clusters images by content—and there is an implicit hope is that approximately optimizing these objective functions will in fact produce a clustering that is close (in symmetric difference) to the truth. In this paper, we show that if we make this implicit assumption explicit—that is, if we assume that any  $c$ -approximation to the given clustering objective  $F$  is  $\epsilon$ -close to the target—then we can produce clusterings that are  $O(\epsilon)$ -close to the target, *even for values  $c$  for which obtaining a  $c$ -approximation is NP-hard*. In particular, for  $k$ -median and  $k$ -means objectives, we show that we can achieve this guarantee for any constant  $c > 1$ , and for min-sum objective we can do this for any constant  $c > 2$ . Our results also highlight a somewhat surprising conceptual difference between assuming that the *optimal* solution to, say, the  $k$ -median objective is

$\epsilon$ -close to the target, and assuming that any *approximately optimal* solution is  $\epsilon$ -close to the target, even for approximation factor say  $c = 1.01$ . In the former case, the problem of finding a solution that is  $O(\epsilon)$ -close to the target remains computationally hard, and yet for the latter we have an efficient algorithm.

## 104 Secretary Problems: Weights and Discounts

The classical secretary problem studies the problem of selecting online an element (a “secretary”) with maximum value in a randomly ordered sequence. The difficulty lies in the fact that an element must be either selected or discarded upon its arrival, and this decision is irrevocable. It is well-known how to design constant-competitive algorithms for the classical secretary problems (see, e.g., the survey of Freeman (Internat. Statist. Rev. 1983)) and several variants. We study the following two extensions of the secretary problem: 1. In the discounted secretary problem, there is a time-dependent “discount” factor  $d(t)$ , and the benefit derived from selecting an element/secretary  $e$  at time  $t$  is  $d(t) \cdot v(e)$ . For this problem with arbitrary (not necessarily decreasing) functions  $d(t)$ , we show a constant-competitive algorithm when the expected optimum is known in advance. With no prior knowledge, we exhibit a lower bound of  $\Omega(\frac{\log n}{\log \log n})$ , and give a nearly-matching  $O(\log n)$ -competitive algorithm. 2. In the weighted secretary problem, up to  $K$  secretaries can be selected; when a secretary is selected (s)he must be irrevocably assigned to one of  $K$  positions, with position  $k$  having weight  $w(k)$ , and assigning object/secretary  $e$  to position  $k$  has benefit  $w(k) \cdot v(e)$ . The goal is to select secretaries and assign them to positions to maximize  $\sum_{e,k} w(k) \cdot v(e) \cdot x_{ek}$  where  $x_{ek}$  is an indicator variable that secretary  $e$  is assigned position  $k$ . We give constant-competitive algorithms for this problem. Most of these results can also be extended to the *matroid secretary* case (Babai et al., *SODA 2007*) for a large family of matroids with a constant-factor loss, and an  $O(\log \text{rank})$  loss for general matroids. These results are based on a reduction from various matroids to partition matroids which present a unified approach to many of the upper bounds of Babai et al. This problem is closely connected with online mechanism design (see, e.g., Hajiaghayi et al., *Elec. Comm. 2004*). All our algorithms are monotone, and hence lead to truthful mechanisms for the corresponding online auction problem.

## 105 Improved Equilibria via Public Service Advertising

Many natural games have both bad and good Nash equilibria: their Price of Anarchy is high and yet their Price of Stability is low. In such cases, one could hope to “move” behavior from a high cost equilibrium to a low cost one by a “public service advertising campaign” encouraging players to follow the low-cost equilibrium, and if every player follows the advice then we are done. This has motivated much of the research on Price of Stability. However, the assumption that everyone follows instructions is unrealistic. A more natural assumption is that some players will follow them, while other players will not. In this paper we consider the question of to what extent can such an advertising campaign cause behavior to switch from a bad equilibrium to a good one even if only a fraction of people actually follow the given advice, and do so only temporarily. Note that unlike in the “price of altruism” model we assume everyone will ultimately act in their own interest. We consider several important and widely studied classes of games including network design with fair cost sharing, scheduling with unrelated machines, party affiliation games (which include consensus and cut games). We show that for some of these games (such as fair cost sharing), a random  $\alpha$  fraction of the population following the given advice is sufficient to get behavior within an  $O(1/\alpha)$  factor of the price of stability for any  $\alpha > 0$ . However, for some games such as consensus and cut games, there is a strict threshold (in this case,  $\alpha < 1/2$  yields almost no benefit, yet  $\alpha > 1/2$  is enough to reach

near-optimal behavior), and for some games, such as scheduling, no value  $\alpha < 1$  is sufficient. We also consider a “viral marketing” model in which certain players are specifically targeted, and analyze the ability of such targeting to influence behavior using a much smaller number of targeted players.

## 106 Exact Algorithm for Partial Curve Matching via the Frechet Distance

Curve matching is a fundamental problem that occurs in many applications. In this paper, we study the problem of measuring *partial* similarity between curves. More specifically, given two curves, we wish to maximize the total length of subcurves of them that are *close* to each other, where the *closeness* is measured by the Frechet distance, a common distance measure for curves. The resulting maximal length is called the *partial Frechet similarity* between the two input curves. Given two polygonal curves  $P$  and  $Q$  in  $\mathbb{R}^d$  of sizes  $n$  and  $m$ , respectively, we present the first exact algorithm that runs in polynomial time to compute  $d_F(P, Q)$ , the partial Frechet similarity between  $P$  and  $Q$ , under the  $L_1$  and the  $L_\infty$  norms. Specifically, we reduce the problem of computing  $d_F(P, Q)$  to  $O(nm(n + m))$  instances of a simpler problem, each of which can be solved in  $O((n + m) \log(nm))$  time. Hence the partial Frechet similarity between  $P$  and  $Q$  under the  $L_1$  or the  $L_\infty$  norm can be computed in  $O(nm(n + m)^2 \log(nm))$  time. To the best of our knowledge, this is the first paper to study this natural definition of partial curve similarity between curves under the continuous setting (where all points in the curve are considered), and present a polynomial-time exact algorithm for it.

## 107 Approximate Euclidean Shortest Path amid convex obstacles

We study the approximate Euclidean shortest path problem amid a small number  $k$  of convex obstacles of total complexity  $n$ . We obtain algorithms for computing approximate shortest paths between two points whose running time depends linearly in  $n$ , and data structures for answering approximate shortest-path distance queries to a fixed source whose size is independent of  $n$ .

## 108 Sorting and Selection in Posets

Classical problems of sorting and searching assume an underlying linear ordering of the objects being compared. In this paper, we study these problems in the context of partially ordered sets, in which some pairs of objects are incomparable. This generalization is interesting from a combinatorial perspective, and it has immediate applications in ranking scenarios where there is no underlying linear ordering, e.g., conference submissions. It also has applications in certain settings in biology. Our results represent significant progress over previous results from two decades ago by Faigle and Turan. In particular, we present the first algorithm that sorts a width- $w$  poset of size  $n$  with optimal query complexity  $O(n(w + \log n))$ . We also describe a variant of Mergesort with query complexity  $O(wn \log(n/w))$  and total complexity  $O(w^2 n \log(n/w))$ ; an algorithm with the same query complexity was given by Faigle and Turan, but no efficient implementation of that algorithm is known. Both our sorting algorithms can be applied with negligible overhead to the more general problem of reconstructing transitive relations. We also consider two related problems: finding the minimal elements, and its generalization to finding the bottom  $k$  “levels”, called the  $k$ -selection problem. We give efficient deterministic and randomized algorithms for finding the minimal elements with  $O(w n)$  query and total complexity. We provide matching lower bounds for the query complexity up to a factor of 2 and generalize the results to the  $k$ -selection problem. Finally, we present efficient algorithms for computing a linear extension of a poset and computing the heights of all elements.

## 109    **Sorting by Placement and Shift**

In sorting situations where the final destination of each item is known, it is natural to repeatedly choose items and place them where they belong, allowing the intervening items to shift by one to make room. (In fact, a special case of this algorithm is commonly used to hand-sort files.) However, it is not obvious that this algorithm necessarily terminates. We show that in fact the algorithm terminates after at most  $2^{n-1} - 1$  steps in the worst case (confirming a conjecture of L. Larson), and that there are super-exponentially many permutations for which this exact bound can be achieved. The proof involves a curious symmetrical binary representation.

## 110    **3-Bit Dictator Testing: 1 vs 5/8**

In the conclusion of his monumental paper on optimal inapproximability results, Hastad [Has01] suggested that Fourier analysis of Dictator (Long Code) Tests does not seem universally applicable in the study of CSPs. His main open question was to determine if the technique could resolve the approximability of \*satisfiable\* 3-bit constraint satisfaction problems. In particular, he asked if the "Not Two" (NTW) predicate is non-approximable beyond the random assignment threshold of 5/8 on satisfiable instances. Around the same time, Zwick [Zwi98] showed that all satisfiable 3-CSPs are 5/8-approximable, and conjectured that the 5/8 is optimal.

In this work we show that Fourier analysis techniques \*can\* produce a Dictator Test based on NTW with completeness 1 and soundness 5/8. Our test's analysis uses the Bonami-Gross-Beckner hypercontractive inequality. We also show a soundness \*lower bound\* of 5/8 for all 3-query Dictator Tests with perfect completeness. This lower bound for Property Testing is proved in part via a semidefinite programming algorithm of Zwick [Zwi98]. Our work completes the determination of the 3-query "Dictatorship Testing curve". Although this represents progress on Zwick's conjecture, current PCP "outer verifier" technology is insufficient to convert our Dictator Test into an NP-hardness-of-approximation result.

## 111    **Packing multiway cuts in graphs**

We consider the following "multiway cut packing" problem in undirected graphs: we are given a graph  $G = (V, E)$  and  $k$  commodities, each corresponding to a set of terminals located at different vertices in the graph; our goal is to produce a collection of cuts  $\{C_1, \dots, C_k\}$  such that  $C_i$  is a multiway cut for commodity  $i$  and the maximum load on any edge is minimized. The load on an edge is defined to be the number of cuts in the solution crossing the edge. In the capacitated version of the problem edges have capacities  $c_e$  and the goal is to minimize the maximum *relative* load on any edge – the ratio of the edge's load to its capacity. We present constant factor approximations for this problem in arbitrary undirected graphs. The multiway cut packing problem arises in the context of graph labeling problems where we are given a partial labeling of a set of items and a neighborhood structure over them, and, informally stated, the goal is to complete the labeling in the most consistent way. This problem was introduced by Rabani, Schulman, and Swamy (SODA'08), who developed an  $O(\log n / \log \log n)$  approximation for it in general graphs, as well as an improved  $O(\log^2 k)$  approximation in trees. Here  $n$  is the number of nodes in the graph. We present an LP-based algorithm for the multiway cut packing problem in general graphs that guarantees a maximum edge load of at most  $8\text{OPT} + 4$ . Our rounding approach is based on the observation that every instance of the problem admits a laminar solution (that is, no pair of cuts in the solution crosses) that is near-optimal. For the special case where each commodity has only two terminals and all commodities share a common

sink (the “common sink s-t cut packing” problem) we guarantee a maximum load of  $\text{OPT} + 1$ . Both of these variants are NP-hard; for the common-sink case our result is optimal.

## 112 Analysis of Scalar Fields over Point Cloud Data

Given a real-valued function  $f$  defined over some metric space  $X$ , is it possible to recover some structural information about  $f$  from the sole information of its values at a finite set  $L$  of sample points, whose pairwise distances in  $X$  are given? We provide a positive answer to this question. More precisely, taking advantage of recent advances on the front of stability for persistence diagrams, we introduce a novel algebraic construction, based on a pair of nested families of simplicial complexes built on top of the point cloud  $L$ , from which the persistence diagram of  $f$  can be faithfully approximated. We derive from this construction a series of algorithms for the analysis of scalar fields from point cloud data. These algorithms are simple and easy to implement, they have reasonable complexities, and they come with theoretical guarantees. To illustrate the genericity and practicality of the approach, we also present some experimental results obtained in various applications, ranging from clustering to sensor networks.

## 113 On the hitting times of quantum versus random walks

The *hitting time* of a classical random walk (Markov chain) is the time required to *detect* the presence of (or equivalently, to *find*) a marked state. The hitting time of a quantum walk is subtler to define – in particular, the detection and finding problems are no longer equivalent. We prove relationships among several definitions of classical and quantum hitting times. Then, extending a recent theorem of Tulsi [2007] for the 2D grid, we show that for any state-transitive Markov chain with unique marked state, the quantum hitting time is of the same order for both the detection and finding problems. We conjecture that this remains true for all reversible Markov chains and any number of marked states.

## 114 Better Algorithms for Benign Bandits

The online multi-armed bandit problem and its generalizations are repeated decision making problems, and incur a cost associated with the decision, in such a way that the total cost incurred over all iterations is close to the cost of the best fixed decision in hindsight. The difference in these costs is known as the *regret* of the algorithm. The term *bandit* refers to the setting where one only obtains the cost of the decision used in a given iteration and no other information. Perhaps the most general form of this problem is the non-stochastic bandit linear optimization problem, where the set of decisions is a convex set in some Euclidean space, and the cost functions are linear. Only recently an efficient algorithm attaining  $\tilde{O}(\sqrt{T})$  regret was discovered in this setting. In this paper we propose a new algorithm for the bandit linear optimization problem which obtains a regret bound of  $\tilde{O}(\sqrt{Q})$ , where  $Q$  is the total variation in the cost functions. This regret bound shows that it is possible to incur much less regret in a slowly changing environment, and in fact, matches regret bounds that are attainable in the simpler stochastic setting, when the cost functions are obtained from a probability distribution. This regret bound, previously conjectured to hold in the full information case, is surprisingly attainable even in the bandit setting. Our algorithm is efficient and applies several new ideas to bandit optimization such as reservoir sampling.

## 115 An Online Mechanism for Ad Slot Reservations with Cancellations

Many advertisers (bidders) use Internet systems to buy display advertisements on publishers' web-pages or on traditional media such as radio, TV and newsprint. They seek a simple, online mechanism to reserve ad slots in advance. On the other hand, media publishers (sellers) represent a vast and varying inventory, and they too seek automatic, online mechanisms for pricing and allocating such reservations. Designing mechanisms as effective as repeated Generalized Second Price auctions (that now power spot auctions in sponsored search) is a great challenge. Such a mechanism should not only have certain optimization properties (such as in efficiency or revenue maximization), but also, crucially, have robust game-theoretic properties (such as honest bidding, limited speculations, etc). We propose and study a simple model for auctioning such ad slot reservations in advance. There is one seller with a set of slots to be displayed at some point  $T$  in the future. Until  $T$ , bidders arrive sequentially and place a bid on the slots they are interested in. The seller must decide immediately whether or not to grant a reservation. Our model allows the seller to cancel at any time any reservation made earlier, in which case the holder of the reservation incurs a utility loss amounting to a fraction of its value for the reservation, and may also receive a cancellation fee from the seller. Our main result is an online mechanism for allocation and pricing in this model with many desirable game-theoretic properties. It is individually rational; winners have an incentive to be honest and bidding one's true value dominates any lower bid. Further, it bounds the earnings of speculators who are in the game to obtain the cancellation fees. The mechanism in addition has optimization guarantees. Its revenue is within a constant fraction of the a posteriori revenue of the famed Vickrey-Clarke-Groves (VCG) auction which is known to be truthful (in the offline case). Our mechanism's efficiency is within a constant fraction of the a posteriori optimally efficient solution. If efficiency also takes into account the utility losses of bidders whose reservation was canceled, we show that our mechanism matches (for appropriate values of the parameters) an upper bound on the competitive ratio of any deterministic online algorithm. The technical core of the mechanism is a variant of the online weighted bipartite matching problem where unlike prior variants in which one randomizes edge arrivals or bounds edge weights, we need to consider revoking previously committed edges. On top of an online competitive algorithm for this problem, we impose an appropriate payment structure to obtain the main game-theoretic and optimization guarantees of our mechanism. Our results make no assumptions about the arrival order or value distribution of bidders. They still hold if we replace items with elements of a matroid and matchings with independent sets of the matroid, or if all bidders have linear (additive) value for a set of items.

## 116 The cover time of random geometric graphs

We study the cover time of random geometric graphs. Let  $I(d) = [0, 1]^d$  denote the unit torus in  $d$  dimensions. Let  $D(x, r)$  denote the ball (disc) of radius  $r$ . Let  $\mathcal{U}_d$  be the volume of the unit ball  $D(0, 1)$  in  $d$  dimensions. A random geometric graph  $G = G(d, r, n)$  in  $d$  dimensions is defined as follows: Sample  $n$  points  $V$  independently and uniformly at random from  $I(d)$ . For each point  $x$  draw a ball  $D(x, r)$  of radius  $r$  about  $x$ . The vertex set  $V(G) = V$  and the edge set  $E(G) = \{\{v, w\} : w \neq v, w \in D(v, r)\}$ . Let  $G(d, r, n)$ ,  $d \geq 3$  be a random geometric graph. Let  $c > 1$  be constant, and let  $r = \text{bracc} \log n / (\mathcal{U}_d n)^{1/d}$ . Then w.h.p.

$$C_G \sim c \log \frac{c}{c-1} n \log n.$$

## 117 The extended k-tree algorithm

Consider the following problem: Given  $q = 2^k$  random lists of  $n$ -bit vectors,  $L_1, \dots, L_q$ , each of length  $m$ , find  $x_1 \in L_1, \dots, x_q \in L_q$  such that  $x_1 + \dots + x_q = 0$ , where  $+$  is the XOR operation. This problem has applications in a number of areas, including cryptanalysis, coding and learning theory. The so-called  $k$ -tree algorithm, due to Wagner (CRYPTO 2002) and Blum-Kalai-Wasserman (JACM 2003), solves this problem in  $O(2^{k+n/(k+1)})$  time provided the length  $m$  of the lists is large enough, specifically if  $m \geq 2^{n/(k+1)}$ . In many applications, however, it is necessary to work with lists of smaller length, where the above algorithm breaks down. In this paper we generalize the algorithm to work for significantly smaller values of the list length  $m$ , all the way down to the minimum value for which a solution exists with reasonable probability. Our algorithm exhibits a tradeoff between the value of  $m$  and the running time. We also provide rigorous bounds on the failure probability of both our algorithm and that of Wagner and Blum et al.

## 118 Constructing Laplace operators from Point Clouds

We present an algorithm for approximating the Laplace-Beltrami operator from an arbitrary point cloud obtained from a  $k$ -dimensional submanifold embedded in the  $d$ -dimensional space. We show that this *PCD Laplace (Point-Cloud Data Laplace) operator* converges to the Laplace-Beltrami operator on the underlying manifold as the point cloud becomes denser. Unlike the previous work, we do not assume that the data samples are independent identically distributed from a probability distribution and do not require a global mesh. The resulting algorithm is easy to implement. We present experimental results indicating that even for point sets sampled from a uniform distribution, PCD Laplace converges faster than the weighted graph Laplacian. We also show that certain geometric invariants, such as manifold area, can be estimated directly from the point cloud using our PCD Laplacian with reasonable accuracy. We make the software publically available at the authors' webpages.

## 119 Online story scheduling for web advertising

We study an online job scheduling problem motivated by *storyboarding* in web advertising, where an advertiser derives value from having uninterrupted sequential access to a user surfing the web. The user ceases to browse with probability  $1 - \beta$  at each step, independently. *Stories* (jobs) arrive online; job  $s$  has a length  $\ell_s$  and a per-unit value  $v_s$ . We get a value  $v_s$  for every unit of the job that we schedule consecutively without interruption, discounted for the time at which it is scheduled. Jobs can be preempted, with no further value derived from the residual unscheduled units of the job. We seek an online algorithm whose total reward is competitive against that of the offline scheduler that knows all jobs in advance. We consider two models based on the maximum delay that can be allowed between the arrival and scheduling of a job. In the first, a job can be scheduled anytime after its arrival; in the second a job is lost unless scheduled immediately upon arrival, pre-empting a currently running job if needed. The two settings correspond to two natural models of how long an advertiser retains interest in a relevant user. We show that there is, in fact, a *sharp separation* between what an online scheduler can achieve in these two settings. In the first setting with no deadlines, we give a natural deterministic algorithm with a constant competitive ratio against the offline scheduler. In contrast, we show that in the sharp deadline setting, no (deterministic or randomized) online algorithm can achieve better than a polylogarithmic ratio.

## 120 Generating Random Graphs with Large Girth

We present a simple and efficient algorithm for randomly generating simple graphs with no small cycles. These graphs are used to design high performance Low-Density Parity-Check (LDPC) codes. For any constant  $k$ ,  $\alpha \leq 1/2k(k+3)$  and  $m = O(n^{1+\alpha})$ , our algorithm generates an almost uniform random graph with girth larger than  $k$  in polynomial time. In this range of  $m$  and  $n$ , the fraction of graphs having girth larger than  $k$  is exponentially small in the average degree ( $m/n$ ). Therefore naïve methods such as repeatedly producing graphs from  $G_{n,m}$  till obtaining one with girth larger than  $k$  take exponential amount of time. Our approach is based on *sequential* methods that have been recently successful for efficiently counting and generating random graphs.

## 121 Improved Approximation Algorithms for Scheduling with Fixed Jobs

We study two closely related problems in non-preemptive scheduling of sequential jobs on identical parallel machines. In these two settings there are either fixed jobs or non-availability intervals during which the machines are not available; in either case, the objective is to minimize the makespan. Both of these formulations have different applications, e.g. in turnaround scheduling or overlay computing.

For both problems we contribute approximation algorithms with an improved ratio of  $3/2+\epsilon$ , respectively; these are shown to be basically tight by suitable inapproximability results.

We use dual approximation, creation of a gap structure and job configurations, and a PTAS for the multiple subset sum problem. However, the main feature of our algorithms is a new technique for the assignment of large jobs via flexible rounding. Our new technique is based on an interesting cyclic shifting argument in combination with a network flow model for the assignment of jobs to gaps.

## 122 Combinatorial Algorithms for Wireless Information Flow

Calculating the capacity of wireless networks is a notoriously hard problem to solve in information theory - even the case of networks consisting of three nodes remains today unsolved. The difficulty comes from the inherently shared nature of the wireless channels. A node transmitting can reach all receiver nodes within a certain range (broadcasting). However, if multiple nodes transmit at the same time, a node can receive the superposition of the transmitted signals (interference). Recently, Avestimehr, Diggavi and Tse proposed a linear binary deterministic model that takes into account the interactions between the signals in a wireless network, i.e., broadcasting and interference, and ignores the effect of noise. The argument is that for high Signal-to-Noise-Ratio, it is these signal interactions that will dominate the performance, and thus the capacity of the deterministic could be very close to that of the noisy network. Thus networks of deterministic channels can be used as approximate models for wireless networks. Constructively taking interference into account allows to achieve, even for a single unicast session, a capacity that can be  $O(N)$  larger than the capacity when only broadcasting and routing are employed, where  $N$  is the number of network nodes, as we show through a developed example. To characterize how much the capacity of a unicast session is, Avestimehr, Diggavi and Tse generalized the min-cut max-flow theorem for graphs to networks of deterministic channels. They showed that the value of the minimum cut is in this case the minimum rank of all the binary adjacency matrices describing source-destination cuts. However, since there exists an exponential number of cuts, identifying the capacity through exhaustive search becomes very fast infeasible. Avestimehr et al. also proved that the value of the minimum cut can be achieved, by using information theoretical tools, where intermediate network nodes have no complexity constraints, and perform random mappings using arbitrarily long blocks of bits. In practical networks, it is not

realistic to envisage such operations; thus the natural question that arises is whether the calculated capacity can be achieved or approximated in practice, using bounded complexity operations at the network nodes, and how well. In this paper, we develop a polynomial time algorithm that allows to achieve the min-cut value in binary linear deterministic networks, for the case of a unicast connection. Our algorithm crucially uses a notion of Linear Independence between edges, and cannot be derived as direct consequence of the Ford-Fulkerson algorithm or other path-augmenting techniques over graphs. Using our algorithm we can calculate the capacity in polynomial time; moreover, we can achieve the capacity by using very simple one-bit processing at the intermediate nodes.

## 123 Fast Algorithms for (Max,Min)-Matrix Multiplication and Bottleneck Shortest Paths

Given a directed graph with a capacity on each edge, the *all-pairs bottleneck paths* (APBP) problem is to determine, for all vertices  $s$  and  $t$ , the maximum flow that can be routed from  $s$  to  $t$ . For dense graphs this problem is equivalent to that of computing the (max, min)-product of two real-valued matrices. In this paper, we give a (max, min)-matrix multiplication algorithm running in time  $O(n^{(3+\omega)/2}) \leq O(n^{2.688})$ , where  $\omega$  is the exponent of binary matrix multiplication. Our algorithm improves on a recent  $O(n^{2+\omega/3}) \leq O(n^{2.792})$ -time algorithm of Vassilevska, Williams, and Yuster. In a similar fashion to Vassilevska et al., we reduce (max, min)-product to computing the *dominance product* of sparse real-valued matrices. However, our reduction is different and we use a completely new sparse dominance product algorithm whose running time is incomparable to that of Vassilevska et al. The running time of our (max, min)-multiplier matches that of the best dominance product algorithm but is slightly slower than the  $O(n^{2.575})$ -time APBP algorithm of Shapira et al. that assumes *vertex-capacitated* graphs. We also give faster algorithms for the *all-pairs bottleneck shortest path* (APBSP) problem on both edge- and vertex-capacitated graphs. The problem here is to find the maximum flow that can be sent from  $s$  to  $t$  along a *shortest* path w.r.t. the unit length function. For edge-capacitated graphs, the running time of our APBSP algorithm is  $O(n^{(3+\omega)/2})$ , matching the time bound of our APBP algorithm. For vertex-capacitated graphs the running time is  $O(n^{2.65})$ , which improves significantly on the  $O(n^{2.86})$  time algorithm of Shapira et al.

## 124 Appointment Scheduling with Discrete Random Durations

We consider the problem of determining optimal appointment times for a given sequence of jobs (medical procedures) on a single processor (operating room, examination facility), to minimize the expected total underage and overage costs when each job has a random duration given by its discrete probability distribution. Simple conditions on the cost rates imply that the objective function is L-convex. Then there exists an optimal appointment schedule which is integer and can be found in polynomial time.

## 125 Expanders via Random Spanning Trees

Motivated by the problem of routing reliably and scalably in a graph, we introduce the notion of a splicer, the union of spanning trees of a graph. We prove that for any bounded-degree  $n$ -vertex graph, the union of two random spanning trees approximates the expansion of every cut of the graph to within a factor of  $O(\log n)$ . For the random graph  $G_{n,p}$ , for  $p > c \log n/n$ , a constant number of spanning trees give an expander. This is suggested by the case of the complete graph, where we prove that a constant number of random spanning trees give an expander. The construction of the

splicer is elementary — each spanning tree is produced independently using an algorithm by Aldous and Broder: a random walk in the graph with edges leading to previously unvisited vertices included in the tree. A second important application of splicers is to graph sparsification where the goal is to approximate every cut (and more generally the quadratic form of the Laplacian) using only a small subgraph of the original graph. Benczur-Karger as well as Spielman-Srivastava have shown sparsifiers with  $O(n \log n / \epsilon^2)$  edges that achieve approximation within factors  $1 + \epsilon$  and  $1 - \epsilon$ . Their methods, based on independent sampling of edges, need  $\Omega(n \log n)$  edges to get any approximation (else the subgraph could be disconnected) and leave open the question of linear-size sparsifiers. Splicers address this question for random graphs by providing sparsifiers of size  $O(n)$  that approximate every cut to within a factor of  $O(\log n)$ .

## 126 Loopless Generation of Multiset Permutations using a Constant Number of Variables by Prefix Shifts

This paper affirmatively answers the following combinatorial question: Can the permutations of any multiset be ordered so that each permutation is a prefix shift of the previous permutation? Previously, the answer was known only for the permutations of any set, and the permutations of any multiset containing at most two distinct elements. As a side benefit, the proof is used to affirmatively answer the following algorithmic question: Can the permutations of any multiset be generated by a loopless algorithm that uses only a constant number of variables? Previously, the best algorithm used a linear number of variables.

## 127 Dual-Failure Distance and Connectivity Oracles

Spontaneous failure is an unavoidable aspect of all networks, particularly those with a physical basis such as communications networks or road networks. Whether due to malicious coordinated attacks or other causes, failures temporarily change the topology of the network and, as a consequence, its connectivity and distance metric. In this paper we look at the problem of efficiently answering connectivity, distance, and shortest route queries in the presence of two node or link failures. Our data structure uses  $\tilde{O}(n^2)$  space and answers queries in  $\tilde{O}(1)$  time, which is within a polylogarithmic factor of optimal and nearly matches the single-failure distance oracles of Demetrescu *et al.* It may yet be possible to find distance/connectivity oracles capable of handling any fixed number of failures. However, the sheer complexity of our algorithm suggests that moving beyond dual-failures will require a fundamentally different approach to the problem.

## 128 Sequential cavity method for computing limits of the log-partition function for

**ABSTRACT:** One of the key computational problems in combinatorics/statistical physics is the problem of computing limits of the log-partition functions for various statistical mechanics models on lattices. In combinatorics this limit corresponds to the exponent of various arrangements on lattices, for example the exponents of the number of independent sets, proper colorings or matchings on a lattice. We propose a new method, sequential cavity, which beats the best known existing methods, such as transfer matrix method, in obtaining sharper bounds on the limits of the log-partition function for two models: independent sets (hard-core) and matchings (monomer-dimer). Our method is based on a surprisingly simple representation of the log-partition function limit in terms of a certain marginal probability of a suitably modified lattice, and using recent deterministic approximation counting

algorithms for these two models. Our method also has a provably better theoretical performance compared with the transfer matrix method.

## 129 Lower Bounds for Succinct Data Structures

In this paper, we consider several data structure problems in the static deterministic cell probe model. We develop a new technique for proving lower bounds for succinct data structures, where the redundancy in the storage can be small compared to the information-theoretic minimum. The lower bounds are truly succinct in the sense that we try to match (up to constant factors) the lower order terms of the existing data structures with the lower order terms provided by our lower bound. Using this technique, we obtain (i) the first lower bound for the problem of searching and retrieval of a substring in text; (ii) a cell probe lower bound for the problem of representing permutation  $\pi$  with queries  $\pi(i)$  and  $\pi^{-1}(i)$  that matches the lower order term of the existing data structures, and (iii) a lower bound for representing binary matrices that is also matches upper bounds for some set of parameters. The nature of all these problems is that we are to implement two operations that are in a reciprocal relation to each other (search and retrieval, computing forward and inverse element, operations on rows and columns of a matrix). As far as we know, this paper is the first to provide an insight to such problems.

## 130 Equilibria of Collusive Flow Games are not Unique

In STOC 2006, Hayrapetyan, Wexler, and Tardos introduced the problem of studying collusion in congestion games. In this work, we show that collusion adds significant complexity to the structure of equilibria in nonatomic routing games, answering an open question posed by Comminetti, Correa, and Steir-Moses (ICALP 2006): Without collusion, it follows from well-known convexity arguments that equilibria are unique (up to induced latencies). The question is, does this uniqueness continue to hold in the presence of collusion? We answer no: We show that if collusion is allowed in nonatomic routing games, there may be multiple equilibria. We demonstrate this multiplicity via 2 specific examples. In addition, we show our examples are topologically minimal by giving a complete characterization of network topologies for which, up to induced latencies, unique equilibria exist.

## 131 Sampling Biased Lattice Configurations using Exponential Metrics

We define a geometric distance function that can be used to show fast mixing of certain natural Markov Chains for biased lattice models, including a reversible tiling model that arises in the context of self-assembly and a coloring model that arises in the context of asynchronous cellular automata. For each of these models, we show that the local Markov chain converges rapidly to the stationary distribution and we believe that couplings based on exponential metrics can be used for many related lattice models.

## 132 Size Complexity of Volume Meshes vs Surface Meshes

Typical volume meshes in three dimensions are designed to conform to an underlying two-dimensional surface mesh, with volume mesh element size growing larger away from the surface. The surface mesh may be uniformly spaced or highly graded, and may have fine resolution due to extrinsic mesh size concerns. When we desire that such a mesh have good aspect ratio, we require that some space-filling scaffold vertices be inserted off the surface. We show that under simple preconditions, the

number of scaffold vertices will only be linear in the number of surface vertices. We analyze the number of scaffold vertices in a setting that encompasses many existing volume meshing algorithms.

### **133 Combinatorial Algorithms for Nearest Neighbors, Near-Duplicates and Small-World Design**

We study the so called combinatorial framework for algorithmic problems in similarity spaces. Namely, the input dataset is represented by a comparison oracle that given three points  $x, y, y'$  answers whether  $y$  or  $y'$  is closer to  $x$ . We assume that the similarity order of the dataset satisfies the four variations of the following disorder inequality: if  $x$  is the  $A$ 'th most similar object to  $y$  and  $y$  is the  $B$ 'th most similar object to  $z$ , then  $x$  is among the  $D(A+B)$  most similar objects to  $z$ . Though the oracle gives much less information compared to the standard general metric space model where distance values are given, one can still design very efficient algorithms for various fundamental computational tasks. For nearest neighbor search we present deterministic and exact algorithm with almost linear time and space complexity of preprocessing, and near-logarithmic time complexity of search. Then, for near-duplicate detection we present the first known deterministic algorithm that requires just near-linear time + time proportional to the size of output. Finally, we show that for any dataset satisfying the disorder inequality a visibility graph can be constructed: all out-degrees are near-logarithmic and greedy routing deterministically converges to the nearest neighbor of a target in logarithmic number of steps. The later result is the first known work-around for Navarro's impossibility of generalizing Delaunay graphs. The technical contribution of the paper consists of handling "false positives" in data structures and an algorithmic technique up-aside-down-filter.