Publications of Davis Issac

This document lists all peer-reviewed publications of Davis Issac, Chair for Algorithm Engineering, Hasso Plattner Institute, Potsdam, Germany. This listing was automatically generated on February 6, 2023. An up-to-date version is available online at hpi.de/friedrich/docs/publist/issac.pdf.

Journal articles

[1] Issac, D., Chandran, L. S., Zhou, S., Hadwiger's conjecture for squares of 2-trees. In: 76, pp. 159–174, 2019.

Hadwiger's conjecture asserts that any graph contains a clique minor with order no less than the chromatic number of the graph. We prove that this well-known conjecture is true for all graphs if and only if it is true for squares of split graphs. This observation implies that Hadwiger's conjecture for squares of chordal graphs is as difficult as the general case, since chordal graphs are a superclass of split graphs. Then we consider 2-trees which are a subclass of each of planar graphs, 2-degenerate graphs and chordal graphs. We prove that Hadwiger's conjecture is true for squares of 2-trees. We achieve this by proving the following stronger result: If G is the square of a 2 tree, then G has a clique minor of size $\chi(G)$, where each branch set is a path.

[2] Issac, D., Bhattacharya, A., Kumar, A., Jaiswal, R., Sampling in space restricted settings. In: 80, pp. 1439–1458, 2018.

Space efficient algorithms play an important role in dealing with large amount of data. In such settings, one would like to analyze the large data using small amount of "working space". One of the key steps in many algorithms for analyzing large data is to maintain a (or a small number) random sample from the data points. In this paper, we consider two space restricted settings-(i) the streaming model, where data arrives over time and one can use only a small amount of storage, and (ii) the query model, where we can structure the data in low space and answer sampling queries. In this paper, we prove the following results in the above two settings:

- In the streaming setting, we would like to maintain a random sample from the elements seen so far. We prove that one can maintain a random sample using $\mathcal{O}(\log n)$ random bits and $\mathcal{O}(\log n)$ bits of space, where n is the number of elements seen so far. We can extend this to the case when elements have weights as well.
- In the query model, there are n elements with weights $w_1, \ldots, w_n$ (which are w-bit integers) and one would like to sample a random element with probability proportional to its weight. Bringmann and Larsen (STOC 2013) showed how to sample such an element using $n\omega + 1$ bits of space (whereas, the information theoretic lower bound is $n\omega$). We consider the approximate sampling problem, where we are given an error parameter ε, and the sampling probability of an element can be off by an ε factor. We give matching upper and lower bounds for this problem.

Conference papers

[3] Friedrich, T., Issac, D., Kumar, N., Mallek, N., Zeif, Z., A Primal-Dual Algorithm for Multicommodity Flows and Multicuts in Treewidth-2 Graphs. In: *Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pp. 55:1–55:18, 2022.

We study the problem of multicommodity flow and multicut in treewidth-2 graphs and prove bounds on the multiflow-multicut gap. In particular, we give a primal-dual algorithm for computing multicommodity flow and multicut in treewidth-2 graphs and prove the following approximate max-flow min-cut theorem: given a treewidth-2 graph, there exists a multicommodity flow of value f with congestion 4, and a multicut of capacity c such that $c \leq 20f$. This implies a multiflow-multicut gap of 80 and improves upon the previous best known bounds for such graphs. Our algorithm runs in polynomial time when all the edges have capacity one. Our algorithm is completely combinatorial and builds upon the primal-dual algorithm of Garg, Vazirani and Yannakakis for multicut in trees and the augmenting paths framework of Ford and Fulkerson.

[4] Cooley, M., Greene, C., Issac, D., Pividori, M., Sullivan, B., Parameterized Algorithms for Identifying Gene Co-Expression Modules via Weighted Clique Decomposition. In: *Applied and Computational Discrete Algorithms (ACDA)*, pp. 111–122, 2021.

We present a new combinatorial model for identifying regulatory modules in gene co-expression data using a decomposition into weighted cliques. To capture complex interaction effects, we generalize the previously-studied weighted edge clique partition problem. As a first step, we restrict ourselves to the noise-free setting, and show that the problem is fixed parameter tractable when parameterized by the number of modules (cliques). We present two new algorithms for finding these decompositions, using linear programming and integer partitioning to determine the clique weights. Further, we implement these algorithms in Python and test them on a biologically-inspired synthetic corpus generated using real-world data from transcription factors and a latent variable analysis of co-expression in varying cell types.

[5] Borndörfer, R., Casel, K., Issac, D., Niklanovits, A., Schwartz, S., Zeif, Z., Connected k-Partition of k-Connected Graphs and c-Claw-Free Graphs. In: *Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pp. 27:1–27:14, 2021.

A connected partition is a partition of the vertices of a graph into sets that induce connected subgraphs. Such partitions naturally occur in many application areas such as road networks, and image processing. In these settings, it is often desirable to partition into a fixed number of parts of roughly of the same size or weight. The resulting computational problem is called Balanced Connected Partition (BCP). The two classical objectives for BCP are to maximize the weight of the smallest, or minimize the weight of the largest component. We study BCP on c-claw-free graphs, the class of graphs that do not have $K_{1,c}$ as an induced subgraph, and present efficient $(c-1)$-approximation algorithms for both objectives. In particular, for 3-claw-free graphs, also simply known as claw-free graphs, we obtain a 2-approximation. Due to the claw-freeness of line graphs, this also implies a 2-approximation for the edge-partition version of BCP in general graphs. A harder connected partition problem arises from demanding a connected partition into k parts that have (possibly) heterogeneous target weights $w_1, \ldots, w_k$. In the 1970s Győri and Lovász showed that if G is k-connected and the target weights sum to the total size of G, such a partition exists. However, to this day no polynomial algorithm to compute such partitions exists for $k > 4$. Towards finding such a partition $T_1, \ldots, T_k$ in k-connected graphs for general k, we show how to efficiently compute connected partitions that at least approximately meet the target weights, subject to the mild assumption that each w_i is greater than the weight of the heaviest vertex. In particular, we give a 3-approximation for both the lower and the upper bounded version i.e. we guarantee that each T_i has weight at least $\frac{w_i}{3}$ or that each T_i has weight most $3w_i$, respectively. Also, we present a both-side bounded version that produces a connected partition where each T_i has size at least $\frac{w_i}{3}$ and at most $\max(\{r, 3\})w_i$, where $r \geq 1$ is the ratio between the largest and smallest value in $w_1, \ldots, w_k$. In particular for the balanced version, i.e. $w_1 = w_2 =, \ldots, = wk$, this gives a partition with $\frac{w_i}{3} \leq w(T_i) \leq 3w_i$.

[6] Casel, K., Friedrich, T., Issac, D., Niklanovits, A., Zeif, Z., Balanced Crown Decomposition for Connectivity Constraints. In: *European Symposium on Algorithms (ESA)*, pp. 26:1–26:15, 2021.

We introduce the balanced crown decomposition that captures the structure imposed on graphs by their connected induced subgraphs of a given size. Such subgraphs are a popular modeling tool in various application areas, where the non-local nature of the connectivity condition usually results in very challenging algorithmic tasks. The balanced crown decomposition is a combination of a crown decomposition and a balanced partition which makes it applicable to graph editing as well as graph packing and partitioning problems. We illustrate this by deriving improved approximation algorithms and kernelization for a variety of such problems. In particular, through this structure, we obtain the first constant-factor approximation for the Balanced Connected Partition (BCP) problem, where the task is to partition a vertex-weighted graph into k connected components of approximately equal weight. We derive a 3-approximation for the two most commonly used objectives of maximizing the weight of the lightest component or minimizing the weight of the heaviest component.

[7] Casel, K., Friedrich, T., Issac, D., Klodt, N., Seifert, L., Zahn, A., A Color-blind 3-Approximation for Chromatic Correlation Clustering and Improved Heuristics. In: *Knowledge Discovery and Data Mining (KDD)*, pp. 882–891, 2021.

Chromatic Correlation Clustering (CCC) models clustering of objects with categorical pairwise relationships. The model can be viewed as clustering the vertices of a graph with edge-labels (colors). Bonchi et al. [KDD 2012] introduced it as a natural generalization of the well studied problem Correlation Clustering (CC), motivated by real-world applications from data-mining, social networks and bioinformatics. We give theoretical as well as practical contributions to the study of CCC. Our main theoretical contribution is an alternative analysis of the famous Pivot algorithm for CC. We show that, when simply run color-blind, Pivot is also a linear time 3-approximation for CCC. The previous best theoretical results for CCC were a 4-approximation with a high-degree polynomial runtime and a linear time 11-approximation, both by Anava et al. [WWW 2015]. While this theoretical result justifies Pivot as a baseline comparison for other heuristics, its blunt color-blindness performs poorly in practice. We develop a color-sensitive, practical heuristic we call Greedy Expansion that empirically outperforms all heuristics proposed for CCC so far, both on real-world and synthetic instances. Further, we propose a novel generalization of CCC allowing for multi-labelled edges. We argue that it is more suitable for many of the real-world applications and extend our results to this model.

[8] Feldmann, A., Issac, D., Rai, A., Fixed-Parameter Tractability of the Weighted Edge Clique Partition Problem. In: *International Symposium on Parameterized and Exact Computation (IPEC)*, pp. 1–16, 2020.

We develop an FPT algorithm and a bi-kernel for the Weighted Edge Clique Partition (WECP) problem, where a graph with n vertices and integer edge weights is given together with an integer k, and the aim is to find k cliques, such that every edge appears in exactly as many cliques as its weight. The problem has been previously only studied in the unweighted version called Edge Clique Partition (ECP), where the edges need to be partitioned into k cliques. It was shown that ECP admits a kernel with k^2 vertices [Mujuni and Rosamond, 2008], but this kernel does not extend to WECP. The previously fastest algorithm known for ECP has a runtime of $2^{O(k^2)}$ $n^{O(1)}$ [Issac, 2019]. For WECP we develop a bi-kernel with 4^k vertices, and an algorithm with runtime $2^{O(k^{3/2}w^{1/2}\log(k/w))}n^{O(1)}$, where w is the maximum edge weight. The latter in particular improves the runtime for ECP to $2^{O(k^{3/2}\log k)}n^{O(1)}$.

[9] Issac, D., Chandran, L. S., Cheung, Y. K., Spanning tree congestion and computation of gyori lovasz partition. In: *International Colloquium on Automata, Languages, and Programming (ICALP)*, pp. 1–14, 2018.

We study a natural problem in graph sparsification, the Spanning Tree Congestion (STC) problem. Informally, it seeks a spanning tree with no tree-edge routing too many of the original edges. For any general connected graph with n vertices and m edges, we show that its STC is at most $\mathcal{O}(\sqrt{mn})$, which is asymptotically optimal since we also demonstrate graphs with STC at least $\Omega(\sqrt{mn})$. We present a polynomial-time algorithm which computes a spanning tree with congestion $\mathcal{O}(\sqrt{mn})\log n$ $\mathcal{O}(\sqrt{mn}\log n)$. We also present another algorithm for computing a spanning tree with congestion $\mathcal{O}(\sqrt{mn})$; this algorithm runs in sub-exponential time when $m = \omega(n\log^2 n)$. For achieving the above results, an important intermediate theorem is generalized Györi-Lovász theorem. Chen et al. [Jiangzhuo Chen et al., 2007] gave a non-constructive proof. We give the first elementary and constructive proof with a local search algorithm of running time $\mathcal{O}^*(4^n)$. We discuss some consequences of the theorem concerning graph partitioning, which might be of independent interest. We also show that for any graph which satisfies certain expanding properties, its STC is at most $\mathcal{O}^*(n)$, and a corresponding spanning tree can be computed in polynomial time. We then use this to show that a random graph has STC $\Theta(n)$ with high probability.

[10] Issac, D., Leeuwen, E. J., Das, A., Chandran, L. S., Algorithms and bounds for very strong rainbow coloring. In: *Latin American Symposium on Theoretical Informatics Conference (LATIN)*, pp. 625–639, 2018.

A well-studied coloring problem is to assign colors to the edges of a graph G so that, for every pair of vertices, all edges of at least one shortest path between them receive different colors. The minimum number of colors necessary in such a coloring is the strong rainbow connection number ($\mathbf{src}(G)$) of the graph. When proving upper bounds on $\mathbf{src}(G)$, it is natural to prove that a coloring exists where, for every shortest path between every pair of vertices in the graph, all edges of the path receive different colors. Therefore, we introduce and formally define this more restricted edge coloring number, which we call very strong rainbow connection number ($\mathbf{vsrc}(G)$). In this paper, we give upper bounds on $\mathbf{vsrc}(G)$ for several graph classes, some of which are tight. These immediately imply new upper bounds on $\mathbf{src}(G)$ for these classes, showing that the study of $\mathbf{vsrc}(G)$ enables meaningful progress on bounding $\mathbf{src}(G)$. Then we study the complexity of the problem to compute $\mathbf{vsrc}(G)$, particularly for graphs of bounded treewidth, and show this is an interesting problem in its own right. We prove that $\mathbf{vsrc}(G)$ can be computed in polynomial time on cactus graphs; in contrast, this question is still open for $\mathbf{src}(G)$. We also observe that deciding whether $\mathbf{vsrc}(G) = k$ is fixed-parameter tractable in k and the treewidth of G. Finally, on general graphs, we prove that there is no polynomial-time algorithm to decide whether $\mathbf{vsrc}(G) \leq 3$ nor to approximate $\mathbf{vsrc}(G)$ within a factor $n^{1-\varepsilon}$, unless P = NP.

[11] Issac, D., Leeuwen, E. J., Lauri, J., Lima, P., Heggernes, P., Rainbow Vertex Coloring Bipartite Graphs and Chordal Graphs. In: *Mathematical Foundations of Computer Science (MFCS)*, pp. 1–13, 2018.

Given a graph with colors on its vertices, a path is called a rainbow vertex path if all its internal vertices have distinct colors. We say that the graph is rainbow vertex-connected if there is a rainbow vertex path between every pair of its vertices. We study the problem of deciding whether the vertices of a given graph can be colored with at most k colors so that the graph becomes rainbow vertex-connected. Although edge-colorings have been studied extensively under similar constraints, there are significantly fewer results on the vertex variant that we consider. In particular, its complexity on structured graph classes was explicitly posed as an open question. We show that the problem remains NP-complete even on bipartite apex graphs and on split graphs. The former can be seen as a first step in the direction of studying the complexity of rainbow coloring on sparse graphs, an open problem which has attracted attention but limited progress. We also give hardness of approximation results for both bipartite and split graphs. To complement the negative results, we show that bipartite permutation graphs, interval graphs, and block graphs can be rainbow vertex-connected optimally in polynomial time.

[12] Issac, D., Chandran, L. S., Karrenbauer, A., On the Parameterized Complexity of Biclique Cover and Partition. In: *International Symposium on Parameterized and Exact Computation (IPEC)*, pp. 1–13, 2016.

Given a bipartite graph G, we consider the decision problem called Biclique Cover for a fixed positive integer parameter k where we are asked whether the edges of G can be covered with at most k complete bipartite subgraphs (a.k.a. bicliques). In the Biclique Partition problem, we have the additional constraint that each edge should appear in exactly one of the k bicliques. These problems are both known to be NP-complete but fixed parameter tractable. However, the known FPT algorithms have a running time that is doubly exponential in k, and the best known kernel for both problems is exponential in k. We build on this kernel and improve the running time for Biclique partition to $2^{O(k^2)}$ by exploiting a linear algebraic view on this problem. On the other hand, we show that no such improvement is possible for Biclique Cover unless the Exponential Time Hypothesis (ETH) is false by proving a doubly exponential lower bound on the running time. We achieve this by giving a reduction from 3SAT on n variables to an instance of Biclique Cover with $k = O(\log n)$. As a further consequence of this reduction, we show that there is no subexponential kernel for Biclique Cover unless $P = NP$. Finally, we point out the significance of the exponential kernel mentioned above for the design of polynomial-time approximation algorithms for the optimization versions of both problems. That is, we show that it is possible to obtain approximation factors of $n/\log n$ for both problems, whereas the previous best approximation factor was $n/\sqrt(\log n)$.