Publications of Shaily Verma

This document lists all peer-reviewed publications of Shaily Verma, Chair for Algorithm Engineering, Hasso Plattner Institute, Potsdam, Germany. This listing was automatically generated on May 6, 2024. An up-to-date version is available online at hpi.de/friedrich/docs/publist/verma.pdf.

Journal articles

[1] Jana, S., Saha, S., Sahu, A., Saurabh, S., Verma, S., Partitioning subclasses of chordal graphs with few deletions. In: *Theoretical Computer Science*, pp. 114288, 2024.

In the (Vertex) k-Way Cut problem, input is an undirected graph G, an integer s, and the goal is to find a subset S of edges (vertices) of size at most s, such that $G - S$ has at least k connected components. Downey et al. [Electr. Notes Theor. Comput. Sci. 2003] showed that k-Way Cut is W[1]-hard parameterized by k. However, Kawarabayashi and Throup [FOCS 2011] showed that the problem is fixed-parameter tractable (FPT) in general graphs with respect to the parameter s and provided a $\mathcal{O}(s^{s^{\mathcal{O}(s)}} n^2)$ time algorithm, where n denotes the number of vertices in G. The best-known algorithm for this problem runs in time $s^{\mathcal{O}(s)} n^{\mathcal{O}(1)}$ given by Lokshtanov et al. [ACM Tran. of Algo. 2021]. On the other hand, Vertex k-Way Cut is W[1]-hard with respect to either of the parameters, k or s or $k + s$. These algorithmic results motivate us to look at the problems on special classes of graphs. In this paper, we consider the (Vertex) k-Way Cut problem on subclasses of chordal graphs and obtain the following results. We first give a sub-exponential FPT algorithm for k-Way Cut running in time $2^{\mathcal{O}(\sqrt{s}\log s)} n^{\mathcal{O}(1)}$ on chordal graphs. It is known that Vertex k-Way Cut is W[1]-hard on chordal graphs, in fact on split graphs, parameterized by $k + s$. We complement this hardness result by designing polynomial-time algorithms for Vertex k-Way Cut on interval graphs, circular-arc graphs and permutation graphs.

[2] Paul, S., Pradhan, D., Verma, S., Vertex-edge domination in interval and bipartite permutation graphs. In: *Discuss. Math. Graph Theory*, pp. 947–963, 2023.

Given a graph $G = (V, E)$, a vertex $u \in V$ ve-dominates all edges incident to any vertex of $N_G[u]$. A set $D \subseteq V$ is a vertex-edge dominating set if for any edge $e \in E$, there exists a vertex $u \in D$ such that u ve-dominates e. Given a graph G, our goal is to find a minimum cardinality ve-dominating set of G. In this paper, we designed two linear time algorithms to find a minimum cardinality ve-dominating set for interval and bipartite permutation graphs.

[3] Verma, S., Fu, H.-L., Panda, B. S., Adjacent vertex distinguishing total coloring in split graphs. In: *Discret. Math.*, pp. 113061, 2022.

An adjacent vertex distinguishing (AVD)-total coloring of a graph G is a total coloring such that any two adjacent vertices u and v have the distinct set of colors, that is, $C(u) \neq C(v)$, where $C(v)$ is the set of colors of the edges incident on v and the color of v. The adjacent vertex distinguishing (AVD)-total chromatic number of a graph G, $\chi''_a(G)$ is the minimum integer k such that there exists an AVD-total coloring of G using k colors. It is known that $\chi''_a(G) \geq \Delta + 1$, where Δ is the maximum degree of the graph. The AVD-total coloring conjecture states that for any graph G, $\chi''_a(G) \leq \Delta + 3$. In this paper, we study the AVD-total coloring in split graphs. We prove the AVD-total coloring conjecture for split graphs and classify certain classes of split graphs according to their AVD-total chromatic number.

[4] Derbisz, J., Kanesh, L., Madathil, J., Sahu, A., Saurabh, S., Verma, S., A Polynomial Kernel for Bipartite Permutation Vertex Deletion. In: *Algorithmica*, pp. 3246–3275, 2022.

In a permutation graph, vertices represent the elements of a permutation, and edges represent pairs of elements that are reversed by the permutation. In the sc Permutation Vertex Deletion problem, given an undirected graph G and an integer k, the objective is to test whether there exists a vertex subset $S \subseteq V(G)$ such that $|S| \leq k$ and $G - S$ is a permutation graph. The parameterized complexity of Permutation Vertex Deletion is a well-known open problem. Bożyk et al. [IPEC 2020] initiated a study on this problem by requiring that $G - S$ be a bipartite permutation graph (a permutation graph that is bipartite). They called this the Bipartite Permutation Vertex Deletion (BPVD) problem. They showed that the problem admits a factor 9-approximation algorithm as well as a fixed parameter tractable (FPT) algorithm running in time $\mathcal{O}(9^k |V(G)|^9)$. Moreover, they posed the question whether BPVD admits a polynomial kernel. We resolve this question in the affirmative by designing a polynomial kernel for BPVD. In particular, we obtain the following: Given an instance (G, k) of BPVD, in polynomial time we obtain an equivalent instance (G', k') of BPVD such that $k' \leq k$, and $|V(G')| + |E(G')| \leq k^{\mathcal{O}(1)}$.

Conference papers

[5] Jana, S., Saha, S., Sahu, A., Saurabh, S., Verma, S., Partitioning Subclasses of Chordal Graphs with Few Deletions. In: *Conference on Algorithms and Complexity (CIAC)*, pp. 293–307, 2023.

In the (Vertex) k-Way Cut problem, input is an undirected graph G, an integer s, and the goal is to find a subset S of edges (vertices) of size at most s, such that $G - S$ has at least k connected components. Downey et al. [Electr. Notes Theor. Comput. Sci. 2003] showed that k-Way Cut is W[1]-hard parameterized by k. However, Kawarabayashi and Throup [FOCS 2011] showed that the problem is fixed-parameter tractable (FPT) in general graphs with respect to the parameter s and provided a $\mathcal{O}(s^{s^{\mathcal{O}(s)}} n^2)$ time algorithm, where n denotes the number of vertices in G. The best-known algorithm for this problem runs in time $s^{\mathcal{O}(s)} n^{\mathcal{O}(1)}$ given by Lokshtanov et al. [ACM Tran. of Algo. 2021]. On the other hand, Vertex k-Way Cut is W[1]-hard with respect to either of the parameters, k or s or $k + s$. These algorithmic results motivate us to look at the problems on special classes of graphs. In this paper, we consider the (Vertex) k-Way Cut problem on subclasses of chordal graphs and obtain the following results. (i) We first give a sub-exponential FPT algorithm for k-Way Cut running in time $2^{\mathcal{O}(\sqrt{s}\log s)}n^{\mathcal{O}(1)}$ on chordal graphs. (ii) It is known that Vertex k-Way Cut is W[1]-hard on chordal graphs, in fact on split graphs, parameterized by $k+s$. We complement this hardness result by designing polynomial-time algorithms for Vertex k-Way Cut on interval graphs, circular-arc graphs and permutation graphs.

[6] Bhyravarapu, S., Jana, S., Kanesh, L., Saurabh, S., Verma, S., Parameterized Algorithms for Eccentricity Shortest Path Problem. In: *International Workshop on Combinatorial Algorithms (IWOCA)*, pp. 74–86, 2023.

Given an undirected graph $G=(V,E)$ and an integer ℓ, the Eccentricity Shortest Path Problem (ESP) problem asks to check if there exists a shortest path P such that for every vertex $v \in V(G)$, there is a vertex $w \in P$ such that $d_G(v,w) \leq \ell$, where $d_G(v,w)$ represents the distance between v and w in G. Dragan and Leitert [Theor. Comput. Sci. 2017] studied the optimization version of this problem which asks to find the minimum ℓ for ESP and showed that it is sf NP-hard even on planar bipartite graphs with maximum degree 3. They also showed that ESP is sf W[2]-hard when parameterized by ℓ. On the positive side, Kucera and Suchy [IWOCA 2021] showed that ESP is fixed-parameter tractable (FPT) when parameterized by modular width, cluster vertex deletion set, maximum leaf number, or the combined parameters disjoint paths deletion set and ℓ. It was asked as an open question in the same paper, if ESP is FPT parameterized by disjoint paths deletion set or feedback vertex set. We answer these questions and obtain the following results. We prove that ESP is FPT when parameterized by disjoint paths deletion set, split vertex deletion set, or the combined parameters feedback vertex set and ℓ. Moreover, we design a $(1+\epsilon)$-factor FPT approximation algorithm when parameterized by the feedback vertex set number. We also show that ESP is W[2]-hard when parameterized by the chordal vertex deletion set.

[7] Ashok, P., Das, S., Kanesh, L., Saurabh, S., Tomar, A., Verma, S., Burn and Win. In: *International Workshop on Combinatorial Algorithms (IWOCA)*, pp. 36–48, 2023.

Given a graph G and an integer k, the graph burning problem asks whether the graph G can be burned in at most k rounds. Graph burning is a model for information spreading in a network, where we study how fast the information spreads in the network through its vertices. In each round, the fire is started at an unburned vertex and the fire spreads to all its neighbors in the next round, burning all of them and so on. The minimum number of rounds required to burn the whole graph G is called the burning number of G. Graph burning problem is NP-hard even for the union of disjoint paths. Moreover, the problem is known to be W[1]-hard when parameterized by the burning number and para-NP-hard when parameterized by treewidth. In this paper, we give a fixed-parameter tractable (FPT) algorithm for graph burning problem when parameterized by treewidth and burning number of the graph. Y. Kobayashi and Y. Otachi [Algorithmica 2022] proved that the problem is FPT parameterized by distance to cographs and gave a double exponential time FPT algorithm when parameterized by distance to split graphs. We improve these results partially and give an FPT algorithm for the problem when parameterized by distance to cographs $\cap$ split graphs (threshold graphs) that runs in $2^{\mathcal{O}(k \ln k)}$. We also design a kernel for the problem in trees. Furthermore, we give an exact algorithm to find the burning number of a graph that runs in time $4^n n^{\mathcal{O}(1)}$, where n is the number of vertices in the input graph.

[8] Ramanujan, M. S., Sahu, A., Saurabh, S., Verma, S., An Exact Algorithm for Knot-Free Vertex Deletion. In: *Mathematical Foundations of Computer Science (MFCS)*, pp. 78:1–78:15, 2022.

The study of the Knot-Free Vertex Deletion problem emerges from its application in the resolution of deadlocks called knots, detected in a classical distributed computation model, that is, the OR-model. A strongly connected subgraph Q of a digraph D with at least two vertices is said to be a knot if there is no arc (u,v) of D with $u \in V(Q)$ and $v \notin V(Q)$ (no-out neighbors of the vertices in Q). Given a directed graph D, the Knot-Free Vertex Deletion problem asks to compute a minimum-size subset $S \subset V(D)$ such that $D[V \setminus S]$ contains no knots. There is no exact algorithm known for the problem in the literature that is faster than the trivial $\mathcal{O}^\star(2^n)$ brute-force algorithm. In this paper, we obtain the first non-trivial upper bound for the problem by designing an exact algorithm running in time $\mathcal{O}^\star(1.576^n)$, where n is the size of the vertex set in D.

[9] Bhyravarapu, S., Jana, S., Panolan, F., Saurabh, S., Verma, S., List Homomorphism: Beyond the Known Boundaries. In: *Theoretical Informatics: Latin American Symposium (LATIN) 2022*, pp. 593–609, 2022.

Given two graphs G and H, and a list $L(u)V(H)$ associated with each $uV(G)$, a list homomorphism from G to H is a mapping $f: V(G)V(H)$ such that (i) for all $uV(G)$, $f(u)L(u)$, and (ii) for all $u, vV(G)$, if $uvE(G)$ then $f(u)f(v)E(H)$. The List Homomorphism problem asks whether there exists a list homomorphism from G to H. Enright, Stewart and Tardos SIAM J. Discret. Math., 2014 showed that the List Homomorphism problem can be solved in $\mathcal{O}(n^{k^2}-3k+4)$ time on graphs where every connected induced subgraph of G admits "a multichain ordering" that includes permutation graphs, biconvex graphs, and interval graphs, where $n=|V(G)|$ and $k=|V(H)|$. We prove that List Homomorphism parameterized by k even when G is a bipartite permutation graph is W1-hard. In fact, our reduction implies that it is not solvable in time $n^o(k)$, unless the Exponential Time Hypothesis (ETH) fails. We complement this result with a matching upper bound and another positive result. We design a $O(n^8 k+3)$ time algorithm for List Homomorphism on bipartite graphs that admit a multichain ordering that includes the class of bipartite permutation graphs and biconvex graphs. Moreover, we show that for bipartite graph G that admits a multichain ordering, List Homomorphism is fixed parameter tractable when parameterized by k and the number of layers in the multichain ordering of G. In addition, we study a variant of List Homomorphism called List Locally Surjective Homomorphism. We prove that List Locally Surjective Homomorphism parameterized by the number of vertices in H is wh, even when G is a chordal graph and H is a split graph.