11

FIXED PARAMETER SET SPLITTING

Fixed Parameter Set Splitting, Linear Kernel and Improved Running Time[1]

Daniel Lokshtanov   Christian Sloper

Abstract

We study the problem k-SET SPLITTING in fixed parameter complexity. We show that the problem can be solved in time $\mathcal{O}^*(2.6494^k)$, improving on the best currently known running time of $\mathcal{O}^*(8^k)$. This is done by showing that a non-trivial instance must have a small minimal SET COVER, and using this to reduce the problem to a series of small instances of MAX SAT.

We also give a linear kernel containing $2k$ elements and $2k$ sets. This is done by reducing the problem to a bipartite graph problem where we use crown decomposition to reduce the graph. We show that this result also gives a good kernel for MAX CUT.

11.1 INTRODUCTION

The problem we study in this short note is MAXIMUM SET SPLITTING. The transformation from MAXIMUM SET SPLITTING to MAX CUT preserves the parameter and thus our kernel applies for this problem as well.

k-SET SPLITTING
INSTANCE: A tuple $(X, \mathcal{F}, k)$ where $\mathcal{F}$ is a collection of subsets of a finite set X, and a positive integer k
PARAMETER: k

[1]This paper appeared at the conference 'Algorithms and Complexity in Durham', 2005 and has later been invited to a special issue of Journal of Discrete Algorithms [LS05].

QUESTION: Is there a subfamily $\mathcal{F}' \subseteq \mathcal{F}, |\mathcal{F}'| \geq k$, and a partition of X into disjoint subsets X_0 and X_1 such that for every $S \in \mathcal{F}'$, we have $S \cap X_0 \neq \emptyset$ and $S \cap X_1 \neq \emptyset$?

SET SPLITTING, or HYPERGRAPH COLORING as it is named in some sources, is a well studied problem. A decision version of the problem appears in [GJ79] as problem [SP4]. It is APX-complete [Pe94] and there have been several approximation algorithms published. The most notable are Anderson and Engebretsen [AE97] with a factor of 0.7240, and Zhang and Ling [ZL01] with a factor of 0.7499.

In the area of parameterized algorithms there have been several results published. The first by Dehne, Fellows, and Rosamond [DFR03] who give a $\mathcal{O}^*(72^k)$ FPT algorithm. Dehne, Fellows, Rosamond, and Shaw [DFRS04] then improved on this result giving a $\mathcal{O}^*(8^k)$ algorithm using a combination of the techniques *greedy localization* and *crown decomposition*.

To improve the running time we show that any non-trivial solution of SET SPLITTING has a SET COVER of size at most k. We can then reduce the problem to 2^k instances of MAX SAT with k clauses each. By using Chen and Kanj's [CK04] exact algorithm with running time $\mathcal{O}^*(1.3247^k)$ on each instance, we get a total running time of $\mathcal{O}^*(2.6494^k)$.

We will also show how we can use crown decomposition to obtain a linear kernel. We do this by reducing the problem to a bipartite graph problem, BIPARTITE COLORFUL NEIGHBORHOOD. We will use crown decomposition to reduce the graph; then show that a simple greedy algorithm decides instances where $k \leq |\mathcal{F}|/2$. Together the two results give a linear kernel with at most $2k$ elements and at most $2k$ sets.

11.2 PRELIMINARIES

We assume that in a SET SPLITTING instance every set contains at least two elements of X. This is a natural assumption as sets of size one cannot be split in any case.

We employ the $\mathcal{O}^*$ notation introduced in [W03], which suppresses the polynomials in the running time and focus on the exponentials. Thus for a $\mathcal{O}^*(2^k)$ algorithm, there exists a constant c such that the running time is $\mathcal{O}(2^k n^c)$.

Throughout the text we will use lower case letters for elements, edges and vertices, capitals for sets, and caligraphy for sets of sets, i.e., $x, X, \mathcal{X}$, respectively.

In graphs, the set of neighbors of a vertex v is denoted $N(v)$, and the neighbors of a set $S \subseteq V$ is denoted $N(S) = \bigcup_{v \in S} N(v) - S$.

11.3 USING SET COVER TO IMPROVE RUNNING TIME

Let a set cover be a subset $S \subseteq X$ such that for every set $P \in \mathcal{F}$, we have $P \cap S \neq \emptyset$. We will prove that an instance either has a set cover of size k or it has a k-SET SPLITTING. As we will show, obtaining a small set cover allows us to reduce the problem to a series of MAX SAT problems.

Lemma 11.3.1 *Any instance $(X, \mathcal{F}, k)$ of Set Splitting that has a* minimal *set cover S, has a partitioning of X into disjoint subsets X_0 and X_1 such that at least $|S|$ sets are split.*

Proof. Let $S = \{s_1, s_2, s_3, \ldots, s_n\}$ be a minimal set cover in $(X, \mathcal{F}, k)$. By minimality of S, we have that for all $s_i \in S$ there is a set $P_i \in \mathcal{F}$ such that $S \cap P_i = \{s_i\}$. Since every set is of size at least two we can obtain a split of each of these sets P_i by partitioning $X_0 = S$ and $X_1 = X - S$. □

We will now show that we can solve the problem of set splitting by creating at most 2^k small instances (at most k clauses) of MAX SAT.

> MAX SAT
> INSTANCE: A collection $\mathcal{C}$ of clauses over a set of variables X
> QUESTION: What is the truth assignment that satisfies the maximum number of clauses?

A recent paper by Chen and Kanj [CK04] gives a $\mathcal{O}^*(1.3247^m)$ algorithm for MAX SAT where m is the number of clauses in the formula. We will use this algorithm to solve our MAX SAT instances.

Theorem 11.3.1 *Set Splitting can be solved in time $\mathcal{O}^*(2.6494^k)$*

Proof. We obtain a minimal set cover S by greedily selecting vertices to cover all sets. By Lemma 11.3.1 we know that S has size less than k, otherwise we can immediatly answer 'Yes'. Let $\mathcal{P} = \{P \mid P \in \mathcal{F}, P \nsubseteq S\}$. It is clear that $|\mathcal{P}| < k$, otherwise the partition $(S, X \setminus S)$ splits at least k sets. The remaining sets are only affected by how we partition S.

Observe that if S was already partitioned into disjoint subsets X_0', X_1' every set in $\mathcal{P}$ has at least one member in X_0' or in X_1'.

Assume we have a partitioning (X_0', X_1') of S. For each set $R \in \mathcal{P}$, where R is not split by X_0', and X_1', create a clause C_R. If R contains an element in X_0' add literals x_i for each element $x_i \in R - S$ to C_R. If R contains an element in X'_1, then add literals $\overline{x_i}$, for each element $x_i \in R - S$ to C_R.

Adding an element x to X'_0 now corresponds to setting variable x false, and vice versa. Observe that a set $R \in \mathcal{P}$ is split if and only if its clause C_R is satisfied. We can now employ Chen and Kanj's exact algorithm for MAX SAT. There are 2^k different partitions of the set cover S, for each we construct an instance of MAX SAT with at most k clauses. Thus we get a total running time of $\mathcal{O}^*(2^k \cdot 1.3247^k) = \mathcal{O}^*(2.6494^k)$. □

11.4 REDUCING TO A GRAPH PROBLEM

The running time of the algorithm in the previous section is multiplicative, i.e., of the form $\mathcal{O}(f(k) \cdot n^c)$. It is often advantagous to have the exponential function as an additive term of the form $\mathcal{O}(f(k) + n^c)$. We can achieve this by reducing, in polynomial time, the problem to a kernel. A *kernel* is a smaller instance of the same problem where the size of the instance is bounded by a function $g(k)$. If $g(k)$ is a linear function we call the kernel a *linear kernel*. Having a linear kernel is often advantagous when designing brute force algorithms for a problem. In this section we show how a linear kernel can be achieved using *crown decompositon*.

Recently the fixed parameter kernels for many problems have been improved using crown decompositions. It is a common technique [FHRST04, PS04] to create an auxiliary graph model from the problem instance and then show that a reduction (using crown decomposition) in the graph model leads to reduction of the problem instance. This technique would apply to this problem, but we will instead reduce our problem to a problem on bipartite graphs.

We reformulate the problem as a problem on bipartite graphs. Let $G(V_{\mathcal{F}}, V_X, E)$ be a bipartite graph, where $V_{\mathcal{F}}$ is a set of vertices with a vertex v_M for each set $M \in \mathcal{F}$, and V_X is a set of vertices with a vertex v_x for each element $x \in X$ and let $(v_x, v_M) \in E$ be an edge if $x \in M$.

The problem is now reduced to color the set V_X black and white such that at least k vertices of $V_{\mathcal{F}}$ have a *colorful neighborhood*, i.e., at least one neighbor of each color. It is easy to see that this problem is equivalent to k-SET SPLITTING.

> k-BIPARTITE COLORFUL NEIGHBORHOOD (k-BCN)
> INSTANCE: A bipartite graph $G = (V_{\mathcal{F}}, V_X, E)$, and a positive integer k
> PARAMETER: k
> QUESTION: Is there a two-coloring of V_X such that there exists a set $S \subseteq V_{\mathcal{F}}$ of size at least k where each element of S has a colorful neighborhood?

As mentioned we will use crown decomposition to reduce the problem. Crown decomposition is particularly well suited for use in bipartite graphs, as Lemma 11.4.1 ensures us the existence of a crown decomposition in any bipartite graph.

Definition 11.4.1 *A* crown decomposition (H, C, R) *in a graph* $G = (V, E)$ *is a partitioning of the vertices of the graph into three sets* H*,* C*, and* R *where* H *and* C *are nonempty such that they have the following properties:*

1. H (the head) *is a vertex separator in* G*, such that there are no edges in* G *between vertices belonging to* C *and vertices belonging to* R*.*
2. $C = C_u \cup C_m$ (the crown) *is an independent set in* G*.*
3. *There is a bijective mapping* $f : H \to C_m$*, where* $f(v) = u \Rightarrow (u, v) \in E$ *(i.e., a perfect matching).*

We can find the following lemma in [CFJ04].

Lemma 11.4.1 *If a graph* $G = (V, E)$ *has an independent set* $I \subseteq V(G)$ *such that* $|N(I)| < |I|$ *then a crown decomposition* (H, C, R) *with* $C \subseteq I$ *for* G *can be found in time* $\mathcal{O}(|V| + |E|)$*.*

Our main reduction rule is the following lemma that states that any crown decomposition can be transformed to a crown decomposition where the head and crown can be removed from the graph.

Lemma 11.4.2 *Given a bipartite graph* $G = (V_{\mathcal{F}}, V_X, E)$ *where* $|V_{\mathcal{F}}| < |V_X|$*, there exists a nontrivial crown decomposition* (H, C, R) *such that* G *is a 'Yes'-instance for* k*-BCN* $\iff$ $G' = (V_{\mathcal{F}} \setminus H, V_X - C, E)$ *is a 'Yes'-instance for* $(k - |H|)$*-BCN*

Proof. Since $|V_{\mathcal{F}}| < |V_X|$ there exists a component $V'_{\mathcal{F}} \subseteq V_{\mathcal{F}}, V'_X \subseteq V_X$ where $|V'_{\mathcal{F}}| < |V_X|'$. By Lemma 11.4.1 we know that this component has a crown decomposition (H', C', R') where $H' \subseteq V'_{\mathcal{F}}$. We now use this crown to identify another crown (H, C, R) with the desired properties.

We assume $R \neq \emptyset$, if this is not the case we can move a vertex from C_u to R. If $C_u \cup R = \emptyset$ then $|V'_{\mathcal{F}}| = |V'_X|$, contradicting $|V'_{\mathcal{F}}| < |V'_X|$.

We iteratively compute this new crown in the following manner. Let $H_0 \subseteq H'$ be the set of vertices of H' that have a neighbor in $V_X - C$. The set H_0 is nonempty since $R \neq \emptyset$ and H' is a vertex separator. Let C_0 be the vertices of C that are matched to

H_0. Let $H_{i+1} = N(C_i)$ and C_{i+1} be the vertices matched to H_{i+1}. Run iteratively until $H_{i+1} = H_i$ then let $H = H_i, C = \{v \mid v \in V_X, N(v) \subseteq H\}$ and R be the remainder.

From the construction of (H, C, R) it is clear that this is a crown decomposition. We proceed to show that G is a Yes-instance for k-BCN if and only if $G' = (V_{\mathcal{F}} - H, V_X - C, E)$ is a YES instance for $(k - |H|)$-BCN.

In one direction assume on the contrary that G is a Yes- instance for k-BCN, but that $G' = (V_{\mathcal{F}} - H, V_X - C, E)$ is a No instance for $(k - |H|)$-BCN. Then the removed elements C must have participated in a colorful neighborhood for more than $|H|$ vertices in $V_{\mathcal{F}}$. This is clearly impossible as $N(C) \subseteq H$.

In the other direction we have that $G' = (V_{\mathcal{F}} - H, V_X - C, E)$ is a Yes-instance for $(k - |H|)$-BCN. We can assume that every vertex in $V_X - C$ has been colored. We can now color C such that every vertex in H has a colorful neighborhood. For every vertex $h \in H_0$ we can color the vertex matched to h different from h's neighbor in $V_X - C$. Observe that after coloring C_j, all vertices in $H_{j+1} - H_j$ have a neighbor in C_j. Thus we can obtain a colorful neighborhood for each vertex $h \subseteq H_{j+1} - H_j$ by coloring its matched vertex appropriately. Thus every vertex in H has a colorful neighborhood and G is a YES instance for k-BCN. □

We say that a bipartite graph is *irreducible* if we cannot apply the reduction in Lemma 11.4.2. The following corollary follows directly.

Corollary 11.4.1 *In an irreducible bipartite graph $G = (|V_F|, |V_X|, E)$, we always have $|V_X| \leq |V_{\mathcal{F}}|$.*

We have obtained the inequality $|V_X| \leq |V_{\mathcal{F}}|$. We now show that we can obtain a similar relationship between $|V_{\mathcal{F}}|$ and k by analyzing the effectiveness of a simple greedy algorithm for the problem.

Greedy algorithms for SET SPLITTING seem to do quite well, and it is indeed possible to prove that there is a polynomial time algorithm that splits at least half of the sets. For our graph problem this is the equivalent of proving that it is always possible to two-color V_X such that at least half of $V_{\mathcal{F}}$ has a colorful neighborhood.

Lemma 11.4.3 *It is always possible to find a partitioning (B, W) of V_X such that at least half of the vertices in $V_{\mathcal{F}}$ have a colorful neighborhood.*

Proof. For a subset $V'_X \subseteq V_X$ we define $M(V'_X) = \{v_M \mid v_M \in V_{\mathcal{F}}, N(v_M) \subseteq V'_X\}$. We proceed by induction on the size of V'_X.

Base case: If $|V'_X| = 1$, then $M(V'_X) = \emptyset$. Thus the statement is trivially true.

Inductive Hypothesis: We assume that for all sets $V'_X \subseteq V_X$ of size n_0 we can find a partitioning B', W' of V'_X such that at least half of the vertices in $M(V'_X)$ has a colorful neighborhood.

Inductive Step: Assume any set $V''_X \subseteq V_X$ where $|V''_X| = n_0 + 1$. Let $v_x \in V''_X$ be an arbitrary vertex in V''_X, and let $M' = M(V''_X - v_x)$. By the inductive hypothesis we can find a partitioning B', W' such that half of the vertices in M' have a colorful neighborhood. Since every vertex in $V_{\mathcal{F}}$ has degree at least 2, every vertex in $M(V''_X) - M'$ has at least one neighbor in $B' \cup W'$. We can assume without loss of generality that half of the vertices of $M(X'') - M'$ have a neighbor in B'. Hence the partitioning $B', W' \cup \{v_x\}$ ensures that at least half of the vertices in $M(V''_X)$ have a colorful neighborhood.

□

The following corollary follows directly from the above lemma. It is easy to design a greedy algorithm that mimic the inductive procedure in the proof and produces the necessary partitioning.

Corollary 11.4.2 *All instances where $k \leq |V_{\mathcal{F}}|/2$ are trivially 'Yes'-instances.*

Theorem 11.4.1 *k-BCN has a linear kernel where $|V_X| \leq |V_{\mathcal{F}}| < 2k$.*

Proof. By Corollary 11.4.2 we have that for a nontrivial instance $(G, k), k > |V_{\mathcal{F}}|/2$. By Corollary 11.4.1 we have that $|V_X| \leq |V_{\mathcal{F}}|$ after reducing the graph. Thus the inequality $|V_X| \leq |V_{\mathcal{F}}| < 2k$ holds for the kernel. □

The following corollary then follows by a transformation of the kernel back to k-SET SPLITTING.

Corollary 11.4.3 k-SET SPLITTING *has a linear kernel of $2k$ sets and $2k$ elements.*

11.5 AN APPLICATION TO MAX CUT

In this section we mention that our kernelization result also applies to the more known MAX CUT, which can be encoded using SET SPLITTING.

MAX CUT
INSTANCE: A graph $G = (V, E)$, and a positive integer k
PARAMETER: k
QUESTION: Is there a partitioning of V into two sets V', V'' such that the number of edges between V' and V'' is at least k?

Let the set of elements $X = V$ and for every edge $(v, u) \in E$ create a set $\{v, u\}$. A splitting of a set vu now corresponds to placing u and v in different partitions in MAX CUT. The results on SET SPLITTING thus apply to MAX CUT.

Observation 11.5.1 k-MAX CUT *has a linear kernel of* $2k$ *vertices and* $2k$ *edges.*

Using the best known exact algorithm for this problem, an $\mathcal{O}^*(2^{|E|/4})$ algorithm by Fedin and Kulikov [FK02], we get a running time of $\mathcal{O}^*(2^{k/2})$ which is equivalent to Prieto's algorithm in [P04] where she used the *Method of Extremal Structure*, another well known FPT technique, to reach a kernel of k vertices and $2k$ edges. Earlier Mahajan, Raman [MR99] has used yet another technique to reach the same number of edges.

11.6 CONCLUSION

We have improved the current best algorithm for SET SPLITTING of $\mathcal{O}^*(8^k)$ to $\mathcal{O}^*(2.6494^k)$ using an observation about the size and structure of the minimal set covers in any set splitting instance.

We also obtained a linear kernel by using modelled crown decomposition. Our model is different from the one seen in [DFRS04]. This shows how crown decompositions can often be applied in many ways to a single problem, with varying results. This kernel also applies to Max Cut equalling the best known kernels for this problem, but with a different approach.

Having achieved a linear kernel for Set Splitting we believe that it is now possible to improve the running time even further. Applying a variation of the transformation seen in the proof of Theorem 11.3.1 it is possible to transform an instance of SET SPLITTING to an instance of Max Sat. Add two clauses for each set, with one literal for each variable. In one clause all literals are positive and in the other all negative. The set is now split if and only if both clauses are satisfied. With a $2k$ set instance we have at least k sets split if and only if we have at least $3k$ clauses satisfied. With our kernel, this direct approach would be better than the method described in this paper if the Max Sat running time could be improved below $\mathcal{O}(2^{m/3})$, where m is the number of clauses.

We would like to acknowledge Daniel Kral for insightful remarks.

BIBLIOGRAPHY

[AE97] G. Andersson and L. Engebretsen, Better approximation algorithms for set splitting and Not-All-Equal-Sat, *Information Processing Letters*, **65**(1988) 305–311.

[CK04] J. Chen, I. Kanj, Improved Exact Algorithms for Max-Sat, *Discrete Applied Mathematics* **142**(2004), 17–27.

[CFJ04] B. Chor, M. Fellows, and D. Juedes, Linear Kernels in Linear Time, or How to Save k Colors in $\mathcal{O}(n^2)$ steps, in *Proceedings of WG2004, LNCS* (2004).

[DFR03] F. Dehne, M. Fellows, and F. Rosamond, An FPT Algorithm for Set Splitting, in *Proceedings WG2004 - 30th Workshop on Graph Theoretic Concepts in Computer science*,LNCS 2004.

[DFRS04] F. Dehne, M. Fellows, F. Rosamond, and P. Shaw, Greedy Localization, Iterative Compression and Modeled Crown Reductions: New FPT Techniques and Improved Algorithms for Max Set Splitting and Vertex Cover, *Proceedings of IWPEC04* LNCS **3162**(2004), 271–281.

[FK02] S. Fedin and A. Kulikov, A $2^{|E|/4}$-time Algorithm for MAX-CUT. *Zapiski nauchnyh seminarov POMI*, **293** (2002), 129–138. English translation to appear in Journal of Mathematical Sciences.

[FHRST04] M.Fellows, P.Heggernes, F.Rosamond, C. Sloper, J.A.Telle, Finding k disjoint triangles in an arbitrary graph, *To appear WG2004*

[GJ79] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* (W.H. Freeman, San Francisco, 1979).

[MR99] M. Mahajan, V. Raman, Parameterizing above guaranteed values: MaxSat and MaxCut, *Journal of Algorithms* **31**(1999), 335–354.

[P04] E. Prieto, The Method of Extremal Structure on the k-Maximum Cut Problem. *Manuscript, to appear*

[PS04] E.Prieto and C. Sloper, Reducing to Independent Set Structure — the Case of k-INTERNAL SPANNING TREE. *To appear*

[Pe94] E. Petrank, The hardness of approximation: Gap location, *Computational Complexity*, **4**(1994), 133–157.

[W03] G. Woeginger, Exact Algorithms for NP-Hard Problems. A Survey, in *Proceedings of 5th International Workshop on Combinatorial Optimization-Eureka, You Shrink!* Papers dictated to Jack Edmonds, M. Junger, G. Reinelt, and G. Rinaldi (Festschrift Eds.) LNCS **2570** (2003), pp. 184–207.

[ZL01] H. Zhang and C.X.Ling, An improved learning algorithm for augmented naive Bayes, *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, LNCS **2035**(2001), pp.581–586.