



The Balanced Satisfactory Partition Problem

Ajinkya Gaikwad, Soumen Maity^(✉), and Shuvam Kant Tripathi

Indian Institute of Science Education and Research, Pune 411008, India
{ajinkya.gaikwad, tripathi.shuvamkant}@students.iiserpune.ac.in,
soumen@iiserpune.ac.in

Abstract. The Satisfactory Partition problem asks whether it is possible to partition the vertex set of a given undirected graph into two parts such that each vertex has at least as many neighbours in its own part as in the other part. The Balanced Satisfactory Partition problem is a variant of the above problem where the two partite sets are required to have the same cardinality. Both problems are known to be NP-complete but its parameterized complexity remains open until now. We enhance our understanding of the problem from the viewpoint of parameterized complexity. The two main results of the paper are the following: (1) The Satisfactory Partition problem and its balanced version are fixed parameter tractable (FPT) when parametrized by neighbourhood diversity, (2) The Balanced Satisfactory Partition problem is W[1]-hard when parametrized by treewidth.

Keywords: Parameterized complexity · FPT · W[1]-hard · Treewidth · Neighbourhood diversity

1 Introduction

Gerber and Kobler [7] introduced the problem of deciding if a given graph has a vertex partition into two non-empty parts such that each vertex has at least as many neighbours in its part as in the other part. A graph satisfying this property is called *partitionable*. For example, complete graphs, star graphs, complete bipartite graphs with at least one part having odd size are not partitionable, where as some graphs are easily partitionable: cycles of length at least 4, trees that are not star graphs [4].

Given a graph $G = (V, E)$ and a subset $S \subseteq V(G)$, we denote by $d_S(v)$ the degree of a vertex $v \in V$ in $G[S]$, the subgraph of G induced by S . For $S = V$, the subscript is omitted, hence $d(v)$ stands for the degree of v in G . In this paper, we study the parameterized complexity of SATISFACTORY PARTITION and BALANCED SATISFACTORY PARTITION problems. We define these problems as follows:

S. Maity—The author’s research was supported in part by the Science and Engineering Research Board (SERB), Govt. of India, under Sanction Order No. MTR/2018/001025.

SATISFACTORY PARTITION

Input: A graph $G = (V, E)$.

Question: Is there a nontrivial partition (V_1, V_2) of V such that for every $v \in V$, if $v \in V_i$ then $d_{V_i}(v) \geq d_{V_{3-i}}(v)$?

A variant of this problem where the two parts have equal size is:

BALANCED SATISFACTORY PARTITION

Input: A graph $G = (V, E)$ on an even number of vertices.

Question: Is there a nontrivial partition (V_1, V_2) of V such that $|V_1| = |V_2|$ and for every $v \in V$, if $v \in V_i$ then $d_{V_i}(v) \geq d_{V_{3-i}}(v)$?

Given a partition (V_1, V_2) , we say that a vertex $v \in V_i$ is *satisfied* if $d_{V_i}(v) \geq d_{V_{3-i}}(v)$, or equivalently if $d_{V_i}(v) \geq \lceil \frac{d(v)}{2} \rceil$. A graph admitting a nontrivial partition where all vertices are satisfied is called *satisfactory partitionable*, and such a partition is called *satisfactory partition*. For the standard concepts in parameterized complexity, see the recent textbook by Cygan et al. [5]. We now review the concept of a tree decomposition, introduced by Robertson and Seymour in [12].

Definition 1. A *tree decomposition* of a graph G is a pair $(T, \{X_t\}_{t \in V(T)})$, where T is a tree and each node t of the tree T is assigned a vertex subset $X_t \subseteq V(G)$, called a bag, such that the following conditions are satisfied:

1. Every vertex of G is in at least one bag.
2. For every edge $uv \in E(G)$, there exists a node $t \in T$ such that bag X_t contains both u and v .
3. For every $u \in V(G)$, the set $\{t \in V(T) \mid u \in X_t\}$ induces a connected subtree of T .

It is important to note that a graph may have several different tree decomposition. Similarly, the same tree decomposition can be valid for several different graphs. Every graph has a trivial tree decomposition for which T has only one vertex including all of V . However, this is not effective for the purpose of solving problems.

Definition 2. The *width* of a tree decomposition is defined as $width(T) = \max_{t \in V(T)} |X_t| - 1$ and the treewidth $tw(G)$ of a graph G is the minimum width among all possible tree decomposition of G .

The reason for subtracting 1 in the above definition for width is so that we can define forests as having treewidth 1.

Our Results: Our main results are the following:

- The SATISFACTORY PARTITION and BALANCED SATISFACTORY PARTITION problems are fixed parameter tractable (FPT) when parameterized by neighbourhood diversity.

- The BALANCED SATISFACTORY PARTITION problem is W[1]-hard when parameterized by treewidth.

Related Work: In the first paper on this topic, Gerber and Kobler [7] considered a generalized version of this problem by introducing weights for the vertices and edges and showed that a general version of the problem is strongly NP-complete. For the unweighted version, they presented some sufficient conditions for the existence of a solution. This problem was further studied in [1, 6, 8]. The SATISFACTORY PARTITION problem is NP-complete and this implies that BALANCED SATISFACTORY PARTITION problem is also NP-complete via a simple reduction in which we add new dummy vertices and dummy edges to the graph [2, 4]. Both problems are solvable in polynomial time for graphs with maximum degree at most 4 [4]. They also studied generalizations and variants of this problem when a partition into $k \geq 3$ nonempty parts is required. Bazgan, Tuza, and Vanderpooten [1, 3] studied an “unweighted” generalization of SATISFACTORY PARTITION, where each vertex v is required to have at least $s(v)$ neighbours in its own part, for a given function s representing the degree of satisfiability. Obviously, when $s = \lceil \frac{d}{2} \rceil$, where d is the degree function, we obtain satisfactory partition. They gave a polynomial-time algorithm for graphs of bounded treewidth which decides if a graph admits a satisfactory partition, and gives such a partition if it exists.

2 FPT Algorithm Parameterized by Neighbourhood Diversity

In this section, we present an FPT algorithm for the SATISFACTORY PARTITION and BALANCED SATISFACTORY PARTITION problems parameterized by neighbourhood diversity. We say two vertices u and v in G have the same type if and only if $N_G(u) \setminus \{v\} = N_G(v) \setminus \{u\}$. The relation of having the same type is an equivalence relation. The idea of neighbourhood diversity is based on this type structure.

Definition 3 [10]. The neighbourhood diversity of a graph $G = (V, E)$, denoted by $\text{nd}(G)$, is the least integer k for which we can partition the set V of vertices into k classes, such that all vertices in each class have the same type.

If neighbourhood diversity of a graph is bounded by an integer k , then there exists a partition $\{C_1, C_2, \dots, C_k\}$ of $V(G)$ into k type classes. It is known that such a minimum partition can be found in linear time using fast modular decomposition algorithms [14]. Notice that each type class could either be a clique or an independent set by definition. For algorithmic purpose it is often useful to consider a *type graph* H of graph G , where each vertex of H is a type class in G , and two vertices C_i and C_j are adjacent iff there is complete bipartite clique between these type classes in G . It is not difficult to see that there will be either a complete bipartite clique or no edges between any two type classes. In this section, we prove the following theorem:

Theorem 1. *The SATISFACTORY PARTITION problem is fixed-parameter tractable when parameterized by the neighbourhood diversity.*

Let G be a connected graph such that $\text{nd}(G) = k$. Let C_1, \dots, C_k be the partition of $V(G)$ into sets of type classes. We assume $k \geq 2$ since otherwise the problem becomes trivial. We define $I_1 = \{C_i \mid C_i \subseteq V_1\}$, $I_2 = \{C_i \mid C_i \subseteq V_2\}$ and $I_3 = \{C_i \mid C_i \cap V_1 \neq \emptyset, C_i \cap V_2 \neq \emptyset\}$ where (V_1, V_2) is a satisfactory partition. We next guess if C_i belongs to I_1, I_2 , or I_3 . There are at most 3^k possibilities as each C_i has three options: either in I_1, I_2 , or I_3 . We reduce the problem of finding a satisfactory partition to an integer linear programming optimization with k variables. Since integer linear programming is fixed parameter tractable when parameterized by the number of variables [11], we conclude that our problem is FPT when parameterized by the neighbourhood diversity.

ILP Formulation: Given I_1, I_2 and I_3 , our goal here is to answer if there exists a satisfactory partition (V_1, V_2) of G with all vertices of C_i are in V_1 if $C_i \in I_1$, all vertices of C_i are in V_2 if $C_i \in I_2$, and vertices of C_i are distributed amongst V_1 and V_2 if $C_i \in I_3$. For each C_i , we associate a variable: x_i that indicates $|V_1 \cap C_i| = x_i$. Because the vertices in C_i have the same neighbourhood, the variables x_i determine (V_1, V_2) uniquely, up to isomorphism. We now characterize a satisfactory partition in terms of x_i . Note that $x_i = n_i = |C_i|$ if $C_i \in I_1$; $x_i = 0$ if $C_i \in I_2$.

Lemma 1. *Let C be a clique type class. Then C is either in I_1 or I_2 .*

Proof. Let C be a clique type class. Let $u, v \in C$. Let us denote $N(u) \setminus \{v\} = N(v) \setminus \{u\}$ by S and let $a = |S \cap V_1|$ and let $b = |S \cap V_2|$. The satisfiability of u implies $a \geq b + 1$ and the satisfiability of v implies $b \geq a + 1$. Clearly, u and v cannot be satisfied simultaneously, as the two inequalities imply $a \geq b + 1 \geq a + 2$, a contradiction. This proves the lemma.

Now we consider the following four cases:

Case 1: Suppose v belongs to a clique type class C_j in I_1 . Then the number of neighbours of v in V_1 , that is,

$$d_{V_1}(v) = \sum_{i:C_i \in N_H[C_j] \cap I_1} n_i + \left(\sum_{i:C_i \in N_H[C_j] \cap I_3} x_i \right) - 1.$$

The number of neighbours of v in V_2 , that is,

$$d_{V_2}(v) = \sum_{i:C_i \in N_H[C_j] \cap I_2} n_i + \sum_{i:C_i \in N_H[C_j] \cap I_3} (n_i - x_i).$$

Therefore, vertex v is satisfied if and only if

$$\sum_{i:C_i \in N_H[C_j] \cap I_1} n_i + \sum_{i:C_i \in N_H[C_j] \cap I_3} 2x_i \geq 1 + \sum_{i:C_i \in N_H[C_j] \cap I_2} n_i + \sum_{i:C_i \in N_H[C_j] \cap I_3} n_i \tag{1}$$

Case 2: Suppose v belongs to a clique type class C_j in I_2 . Then similarly, v is satisfied if and only if

$$\sum_{i:C_i \in N_H(C_j) \cap I_2} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} n_i \geq 1 + \sum_{i:C_i \in N_H(C_j) \cap I_1} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} 2x_i \tag{2}$$

Case 3: Suppose v belongs to an independent type class C_j in V_1 , that is, $C_j \in I_1 \cup I_3$. Then the number of neighbours of v in V_1 , that is,

$$d_{V_1}(v) = \sum_{i:C_i \in N_H(C_j) \cap I_1} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} x_i.$$

Note that if $C_j \in I_3$, then only x_j vertices of C_j are in V_1 and the remaining y_j vertices of C_j are in V_2 . The number of neighbours of v in V_2 , that is,

$$d_{V_2}(v) = \sum_{i:C_i \in N_H(C_j) \cap I_2} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} (n_i - x_i).$$

Therefore, v is satisfied if and only if

$$\sum_{i:C_i \in N_H(C_j) \cap I_1} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} 2x_i \geq \sum_{i:C_i \in N_H(C_j) \cap I_2} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} n_i \tag{3}$$

Case 4: Suppose v belongs to an independent type class C_j in V_2 , that is, $C_j \in I_2 \cup I_3$. Similarly, vertex v is satisfied if and only if

$$\sum_{i:C_i \in N_H(C_j) \cap I_2} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} n_i \geq \sum_{i:C_i \in N_H(C_j) \cap I_1} n_i + \sum_{i:C_i \in N_H(C_j) \cap I_3} 2x_i \tag{4}$$

We now formulate ILP formulation of satisfactory partition, for given I_1, I_2 and I_3 . The question is whether there exist x_j under the conditions $x_j = n_j$ if $C_j \in I_1$, $x_j = 0$ if $C_j \in I_2$, $x_j \in \{1, 2, \dots, n_j - 1\}$ if $C_j \in I_3$ and the additional conditions described below:

- Inequality 1 for all clique type classes $C_j \in I_1$
- Inequality 2 for all clique type classes $C_j \in I_2$
- Inequality 3 for all independent type classes $C_j \in I_1$
- Inequality 4 for all independent type classes $C_j \in I_2$

$$\sum_{C_i \in N_H(C_j) \cap I_2} n_i + \sum_{C_i \in N_H(C_j) \cap I_3} n_i = \sum_{C_i \in N_H(C_j) \cap I_1} n_i + \sum_{C_i \in N_H(C_j) \cap I_3} 2x_i$$

for all independent type classes $C_j \in I_3$.

For BALANCED SATISFACTORY PARTITION problem, we additionally ask that

$$\sum_{i:C_i \in I_1} n_i + \sum_{i:C_i \in I_3} x_i = \sum_{i:C_i \in I_3} (n_i - x_i) + \sum_{i:C_i \in I_2} n_i.$$

Solving the ILP: Lenstra [11] showed that the feasibility version of p -ILP is FPT with running time doubly exponential in p , where p is the number of variables. Later, Kannan [9] designed an algorithm for p -ILP running in time $p^{O(p)}$.

p -VARIABLE INTEGER LINEAR PROGRAMMING FEASIBILITY (p -ILP): Let matrices $A \in \mathbb{Z}^{m \times p}$ and $b \in \mathbb{Z}^{p \times 1}$ be given. The question is whether there exists a vector $x \in \mathbb{Z}^{p \times 1}$ satisfying the m inequalities, that is, $A \cdot x \leq b$. We use the following result:

Lemma 2 [9, 11]. p -ILP can be solved using $O(p^{2.5p+o(p)} \cdot L)$ arithmetic operations and space polynomial in L . Here L is the number of bits in the input.

In the formulation for SATISFACTORY PARTITION problem, we have at most k variables. The value of any variable in the integer linear programming is bounded by n , the number of vertices in the input graph. The constraints can be represented using $O(k^2 \log n)$ bits. Lemma 2 implies that we can solve the problem with the given guess I_1, I_2 and I_3 in FPT time. There are at most 3^k choices for (I_1, I_2, I_3) , and the ILP formula for a guess can be solved in FPT time. Thus Theorem 1 holds.

3 Hardness of Balanced Satisfactory Partition Parameterized by Treewidth

In this section, we prove the following theorem:

Theorem 2. The BALANCED SATISFACTORY PARTITION problem is W[1]-hard when parameterized by the treewidth of the graph.

We introduce several variants of BALANCED SATISFACTORY PARTITION that we require in our proofs. The problem BALANCED SATISFACTORY PARTITION^S generalizes BALANCED SATISFACTORY PARTITION where some vertices are forced to be in the second part V_2 . This variant can be formalized as follows:

BALANCED SATISFACTORY PARTITION^S
Input: A graph $G = (V, E)$ on an even number of vertices, and a set $V_\square \subseteq V(G)$.
Question: Is there a balanced satisfactory partition (V_1, V_2) of V such that $V_\square \subseteq V_2$.

BALANCED SATISFACTORY PARTITION^{FS} is a further generalization where some vertices are forced to be in the first part V_1 and some other vertices are forced to be in the second part V_2 . This variant can be formalized as follows:

BALANCED SATISFACTORY PARTITION^{FS}

Input: A graph $G = (V, E)$ on an even number of vertices, a set $V_\Delta \subseteq V(G)$, and a set $V_\square \subseteq V(G)$.

Question: Is there a balanced satisfactory partition (V_1, V_2) of V such that (i) $V_\Delta \subseteq V_1$ (ii) $V_\square \subseteq V_2$.

Finally, we introduce the generalization **BALANCED SATISFACTORY PARTITION^{FSC}** in which we are also given a subset of “complementary pairs” of vertices and feasible solutions are only those for which neither V_1 nor V_2 contains a complementary pair.

BALANCED SATISFACTORY PARTITION^{FSC}

Input: A graph $G = (V, E)$ on an even number of vertices, a set $V_\Delta \subseteq V(G)$, a set $V_\square \subseteq V(G)$, and a set $C \subseteq V(G) \times V(G)$.

Question: Is there a balanced satisfactory partition (V_1, V_2) of V such that (i) $V_\Delta \subseteq V_1$ (ii) $V_\square \subseteq V_2$, and (iii) for all $(a, b) \in C$, V_1 contains either a or b but not both?

Let $G = (V, E)$ be an undirected and edge weighted graph, where V , E , and w denote the set of nodes, the set of edges and a positive integral weight function $w : E \rightarrow Z^+$, respectively. An orientation Λ of G is an assignment of a direction to each edge $\{u, v\} \in E(G)$, that is, either (u, v) or (v, u) is contained in Λ . The weighted outdegree of u on Λ is $w_{\text{out}}^u = \sum_{(u,v) \in \Lambda} w(\{u, v\})$. We define **MINIMUM MAXIMUM OUTDEGREE** problem as follows:

MINIMUM MAXIMUM OUTDEGREE

Input: A graph G , an edge weighting w of G given in unary, and a positive integer r .

Question: Is there an orientation Λ of G such that $w_{\text{out}}^u \leq r$ for each $u \in V(G)$?

It is known that **MINIMUM MAXIMUM OUTDEGREE** is $W[1]$ -hard when parameterized by the treewidth of the input graph [13]. To prove Theorem 2, we give a 4-step reduction. In the first step of the reduction, we reduce **MINIMUM MAXIMUM OUTDEGREE** to **BALANCED SATISFACTORY PARTITION^{FSC}**. In the second step of the reduction we reduce the **BALANCED SATISFACTORY PARTITION^{FSC}** to **BALANCED SATISFACTORY PARTITION^{FS}**. In the third step of the reduction we reduce the **BALANCED SATISFACTORY PARTITION^{FS}** to **BALANCED SATISFACTORY PARTITION^S**. Finally we reduce **BALANCED SATISFACTORY PARTITION^S** to **BALANCED SATISFACTORY PARTITION**. To measure the treewidth of a **BALANCED SATISFACTORY PARTITION^{FSC}** instance, we use the following definition. Let $I = (G, V_\Delta, V_\square, C)$ be a **BALANCED SATISFACTORY PARTITION^{FSC}** instance. The *primal graph* G' of I is defined as follows: $V(G') = V(G)$ and $E(G') = E(G) \cup C$.

Lemma 3. *The BALANCED SATISFACTORY PARTITION^{FSC} is $W[1]$ -hard when parameterized by the treewidth of the primal graph.*

Proof. Let $G = (V, E, w)$ and a positive integer r be an instance of MINIMUM MAXIMUM OUTDEGREE. We construct an instance of BALANCED SATISFACTORY PARTITION^{FSC} as follows. An example is given in Figure 1. For each vertex $v \in V(G)$, we introduce a set of new vertices $H_v = \{h_1^{v\Delta}, \dots, h_{2r}^{v\Delta}\}$. For each edge $(u, v) \in E(G)$, we introduce the set of new vertices $V_{uv} = \{u_1^v, \dots, u_{w(u,v)}^v\}$, $V'_{uv} = \{u_1^{v'v}, \dots, u_{w(u,v)}^{v'v}\}$, $V_{vu} = \{v_1^u, \dots, v_{w(u,v)}^u\}$, $V'_{vu} = \{v_1^{u'v}, \dots, v_{w(u,v)}^{u'v}\}$, $V_{uv}^\square = \{u_1^{v\square}, \dots, u_{w(u,v)}^{v\square}\}$, $V'_{uv}^\square = \{u_1^{v'\square}, \dots, u_{w(u,v)}^{v'\square}\}$, $V_{vu}^\square = \{v_1^{u\square}, \dots, v_{w(u,v)}^{u\square}\}$, $V'_{vu}^\square = \{v_1^{u'\square}, \dots, v_{w(u,v)}^{u'\square}\}$. Let $\omega = \sum_{(u,v) \in E} w(u, v)$. Finally we add a set V_0 of $8\omega + |V|(2r + 1) - 4$ isolated vertices. We now define the graph G' with

$$V(G') = V(G) \bigcup_{v \in V(G)} H_v \bigcup_{(u,v) \in E(G)} (V_{uv} \cup V_{uv}^\square \cup V_{vu} \cup V_{vu}^\square) \bigcup_{(u,v) \in E(G)} (V'_{uv} \cup V'_{uv}^\square \cup V'_{vu} \cup V'_{vu}^\square) \bigcup V_0$$

and

$$E(G') = \{(v, h) \mid v \in V(G), h \in H_v\} \bigcup \{(u, x) \mid (u, v) \in E(G), x \in V_{uv} \cup V_{uv}^\square\} \bigcup \{(x, v) \mid (u, v) \in E(G), x \in V_{vu} \cup V_{vu}^\square\} \bigcup \{(u_i^v, u_i^{v'}), (u_i^{v\square}, u_i^{v'\square}), (v_i^u, v_i^{u'}), (v_i^{u\square}, v_i^{u'\square}) \mid (u, v) \in E(G), 1 \leq i \leq w(u, v)\}.$$

The number of vertices in $V(G') \setminus V_0$ is $8\omega + |V|(2r + 1)$. We define the complementary vertex pairs

$$C = \{(u_i^v, v_i^{u'}), (u_{i+1}^v, v_{i+1}^{u'}), (u_i^v, v_i^{u\square}), (u_i^{v\square}, v_i^{u'}) \mid (u, v) \in E(G), 1 \leq i \leq w(u, v)\}$$

Complementary vertex pairs are shown in dashed lines in Fig. 1. Finally we define $V_\Delta = V(G) \bigcup_{v \in V(G)} H_v$ and $V_\square = \bigcup_{(u,v) \in E(G)} (V_{uv}^\square \cup V'_{uv}^\square \cup V_{vu}^\square \cup V'_{vu}^\square)$. We use I to denote $(G', V_\Delta, V_\square, C)$ which is an instance of BALANCED SATISFACTORY PARTITION^{FSC}.

Clearly, it takes polynomial time to compute I . We now prove that the treewidth of the primal graph G' of I is bounded by a function of the treewidth of G . We do so by modifying an optimal tree decomposition τ of G as follows:

- For every edge (u, v) of G , there is a node in τ whose bag B contains both u and v ; add to this node a chain of nodes $1, 2, \dots, w(u, v) - 1$ where the bag of node i is $B \cup \{u_i^v, u_i^{v'}, v_i^{u'}, v_i^u, u_{i+1}^v, u_{i+1}^{v'}, v_{i+1}^{u'}, v_{i+1}^u\}$.
- For every edge (u, v) of G , there is a node in τ whose bag B contains u ; add to this node a chain of nodes $1, 2, \dots, w(u, v)$ where the bag of node i is $B \cup \{u_i^{v\square}, u_i^{v'\square}\}$.

- For every edge (u, v) of G , there is a node in τ whose bag B contains v and add to this node a chain of nodes $1, 2, \dots, w(u, v)$ where the bag of node i is $B \cup \{v_i^{u\Box}, v_i'^{u\Box}\}$.
- For every vertex v of G , there is a node in τ whose bag B contains v and add to this node a chain of nodes $1, 2, \dots, 2r$ where the bag of node i is $B \cup \{h_i^{v\Delta}\}$.

Clearly, the modified tree decomposition is a valid tree decomposition of the primal graph of I and its width is at most the treewidth of G plus eight.

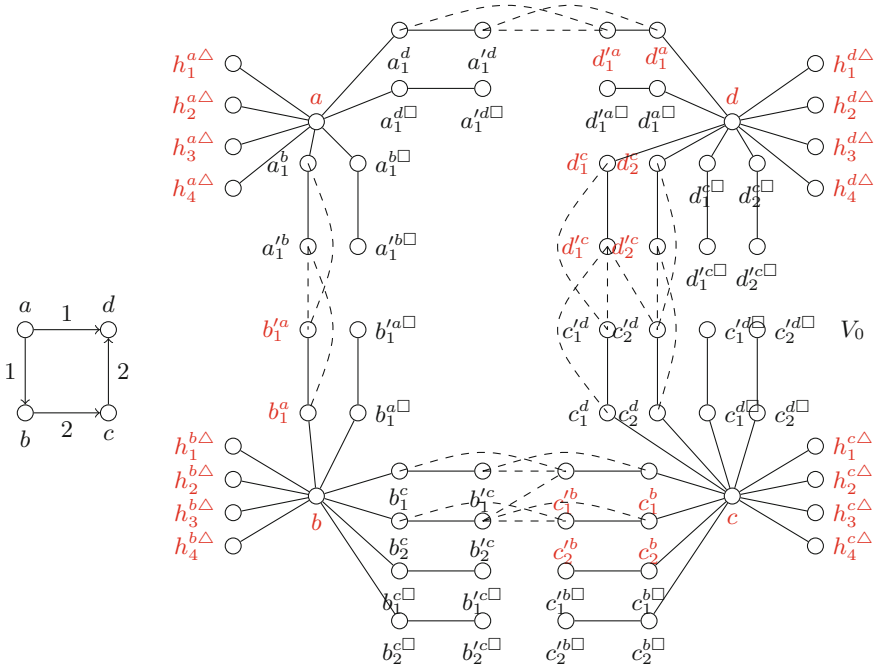


Fig. 1. Result of our reduction on a MINIMUM MAXIMUM OUTDEGREE instance G with $r = 2$. The graph G long with its orientation is shown at the left; and G' is shown at the right. Complementary vertex pairs are shown using dashed lines. The vertices in the first part of satisfactory partition (V_1, V_2) of G' are shown in red for the given orientation of G . Here $\omega = 6$ and V_0 contains 64 isolated vertices. The vertices of V_0 are distributed among V_1 and V_2 so that (V_1, V_2) becomes balanced satisfactory partition.

Let D be the directed graph obtained by an orientation of the edges of G such that for each vertex the sum of the weights of outgoing edges is at most r . Consider the partition of $G' - V_0$

$$V_1 = V_\Delta \bigcup_{(u,v) \in E(D)} (V_{vu} \cup V'_{vu}) = V(G) \bigcup_{v \in V(G)} H_v \bigcup_{(u,v) \in E(D)} (V_{vu} \cup V'_{vu})$$

and

$$V_2 = \bigcup_{(u,v) \in E(D)} (V_{uv} \cup V'_{uv} \cup V_{uv}^\square \cup V'_{uv}{}^\square) \bigcup_{(u,v) \in E(D)} (V_{vu}^\square \cup V'_{vu}{}^\square).$$

To prove that (V_1, V_2) is a satisfactory partition, first we prove that $d_{V_1}(x) \geq d_{V_2}(x)$ for all $x \in V_1$. If x is a vertex in H_v or $V_{vu} \cup V'_{vu}$, then clearly all neighbours of x are in V_1 , hence x is satisfied. Suppose $x \in V(G)$. Let w_{out}^x and w_{in}^x denote the sum of the weights of outgoing and incoming edges of vertex x , respectively. Hence $d_{V_1}(x) = 2r + w_{\text{in}}^x$ and $d_{V_2}(x) = 2w_{\text{out}}^x + w_{\text{in}}^x$ in G' . This shows that x is satisfied as $w_{\text{out}}^x \leq r$. Now we prove that $d_{V_2}(x) \geq d_{V_1}(x)$ for all $x \in V_2$. If x is a vertex in $V_{uv} \cup V_{uv}^\square \cup V'_{vu} \cup V'_{vu}^\square$ then x has one neighbour in V_1 and one neighbour in V_2 . If $x \in V'_{uv} \cup V'_{uv}^\square \cup V_{vu} \cup V_{vu}^\square$ then x has one neighbour in V_2 and no neighbours in V_1 . Thus the vertices in V_2 are satisfied. The isolated vertices of V_0 are distributed among V_1 and V_2 so that it becomes balanced satisfactory partition for G' .

Conversely, suppose (V_1, V_2) is a balanced satisfactory partition of G' . That is $|V_1| = |V_2| = 8\omega + (2r + 1)|V| - 2$. Then $V'_1 = V_1 \setminus V_0$ and $V'_2 = V_2 \setminus V_0$ form a satisfactory partition of $G' - V_0$. For every $(u, v) \in E(G)$, either $V_{uv} \cup V'_{uv} \in V'_1$ or $V_{vu} \cup V'_{vu} \in V'_1$ due to the complementary vertex pairs. We define a directed graph D by $V(D) = V(G)$ and

$$E(D) = \left\{ (u, v) \mid V_{vu} \cup V'_{vu} \in V'_1 \right\} \cup \left\{ (v, u) \mid V_{uv} \cup V'_{uv} \in V'_1 \right\}.$$

Suppose there is a vertex x in D for which $w_{\text{out}}^x > r$. Clearly $x \in V'_1$. We know $d_{V'_1}(x) = 2r + w_{\text{in}}^x$ and $d_{V'_2}(x) = 2w_{\text{out}}^x + w_{\text{in}}^x$. Then $d_{V'_2}(x) > d_{V'_1}(x)$, as by assumption $w_{\text{out}}^x > r$, a contradiction to the fact that (V'_1, V'_2) is a satisfactory partition of $G' - V_0$. Hence $w_{\text{out}}^x \leq r$ for all $x \in V(D)$.

Next we prove the following result which eliminates complementary pairs.

Lemma 4. *The BALANCED SATISFACTORY PARTITION^{FS} problem, parameterized by the treewidth of the graph, is $W[1]$ -hard.*

Proof. Let $I = (G, V_\square, V_\Delta, C)$ be an instance of BALANCED SATISFACTORY PARTITION^{FSC}. Consider the primal graph of I , that is the graph G^p where $V(G^p) = V(G)$ and $E(G^p) = E(G) \cup C$. From this we construct an instance $I' = (G', V'_\square, V'_\Delta)$ of BALANCED SATISFACTORY PARTITION^{FS} problem. For each $(a, b) \in C$ in the primal graph G^p , we introduce two new vertices Δ^{ab} and \square^{ab} and four new edges in G' . We now define the G' with $V(G') = V(G) \cup_{(a,b) \in C} \{\Delta^{ab}, \square^{ab}\}$ and $E(G') = E(G) \cup_{(a,b) \in C} \{(a, \Delta^{ab}), (a, \square^{ab}), (b, \Delta^{ab}), (b, \square^{ab})\}$. Finally, we define the sets $V'_\Delta = V_\Delta \cup_{(a,b) \in C} \{\Delta^{ab}\}$ and $V'_\square = V_\square \cup_{(a,b) \in C} \{\square^{ab}\}$. We illustrate our construction in Fig. 2. It is easy to see that we can compute I' in polynomial time and its treewidth is linear in the treewidth of I . The following holds for every solution (V'_1, V'_2) of I' : V'_1 contains Δ^{ab} for every $(a, b) \in C$, so it must also contain a or b . It cannot contain both a and b for any $(a, b) \in C$, because $\square^{ab} \in V'_2$.

Restricting (V'_1, V'_2) to the original vertices thus is a solution to I . Conversely, for every solution (V_1, V_2) of I , the partition (V'_1, V'_2) where $V'_1 = V_1 \cup_{(a,b) \in C} \{\Delta^{ab}\}$ and $V'_2 = V_2 \cup_{(a,b) \in C} \{\square^{ab}\}$, is a solution of I' .

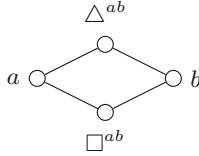


Fig. 2. Gadget for a pair of complementary vertices (a, b) in the reduction from BALANCED SATISFACTORY PARTITION^{FSC} to BALANCED SATISFACTORY PARTITION^{FS}.

Lemma 5. *The BALANCED SATISFACTORY PARTITION^S is $W[1]$ -hard when parameterized by the treewidth of the graph.*

Proof. Let $I = (G, V_\Delta, V_\square)$ be a BALANCED SATISFACTORY PARTITION^{FS} instance; let n denote the number of vertices in G . First, we fix a vertex $v \in V_\square$. For every pair (u, v) of vertices where $u \in V_\Delta$, we introduce two sets of new vertices $X_{uv} = \{x_1^{uv}, x_2^{uv}, \dots, x_n^{uv}\}$ and $Y_{uv}^\square = \{y_1^{uv\square}, y_2^{uv\square}, \dots, y_n^{uv\square}\}$. Next, we define the BALANCED SATISFACTORY PARTITION^S instance $I' = (G', V'_\square)$ where $V'_\square = V_\square \cup_{u \in V_\Delta} Y_{uv}^\square$ and G' is the graph defined by

$$V(G') = V(G) \cup_{u \in V_\Delta} X_{uv} \cup_{u \in V_\Delta} Y_{uv}^\square$$

and

$$\begin{aligned} E(G') = & E(G) \cup_{u \in V_\Delta} \left\{ (u, x_i^{uv}), (u, y_i^{uv\square}), (x_i^{uv}, v), (y_i^{uv\square}, v) \mid 1 \leq i \leq n \right\} \\ & \cup_{u \in V_\Delta} \left\{ (x_i^{uv}, y_i^{uv\square}), (x_i^{uv}, y_{i+1}^{uv\square}) \mid 1 \leq i \leq n - 1 \right\} \cup \left\{ (x_n^{uv}, y_n^{uv\square}), (x_n^{uv}, y_1^{uv\square}) \right\} \\ & \cup_{u \in V_\Delta} \left\{ (x_i^{uv}, x_{i+1}^{uv}) \mid 1 \leq i \leq n - 1 \right\} \cup \left\{ (x_n^{uv}, x_1^{uv}) \right\} \\ & \cup_{u \in V_\Delta} \left\{ (y_i^{uv\square}, y_{i+1}^{uv\square}) \mid 1 \leq i \leq n - 1 \right\} \cup \left\{ (y_n^{uv\square}, y_1^{uv\square}) \right\} \end{aligned}$$

An example is given in Fig. 3. The treewidth of G' is equal to the treewidth of G plus 5. We now claim that I is a yes-instance if and only if I' is a yes-instance. Assume first that there exists a balanced satisfactory partition (V_1, V_2) of I such

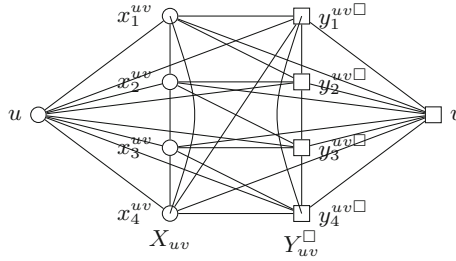


Fig. 3. Let $n = 4$. Gadget for a pair of vertices (u, v) where $u \in V_{\Delta}$ and v is a fixed vertex in V_{\square} in the reduction from BALANCED SATISFACTORY PARTITION^{FS} to BALANCED SATISFACTORY PARTITION^S.

that $V_{\Delta} \in V_1$ and $V_{\square} \in V_2$. In this case, we will get a balanced satisfactory partition (V'_1, V'_2) of I' as follows:

$$V'_1 = V_1 \bigcup_{u \in V_{\Delta}} X_{uv} \quad \text{and} \quad V'_2 = V_2 \bigcup_{u \in V_{\Delta}} Y_{uv}.$$

It is easy to see that (V'_1, V'_2) forms a balanced satisfactory partition of G' as all the vertices in V_1 and V_2 will remain satisfied and also the new vertices in $X_{uv} \cup Y_{uv}$ for all $u \in V_{\Delta}$ are satisfied in their respective part as each vertex has three neighbours in its own part and three neighbors in the other part. Since we are adding equal number of vertices in the balanced partition (V_1, V_2) , we again get a balanced satisfactory partition. This shows that I' is a yes-instance.

Conversely, suppose that there exists a balanced satisfactory partition (V'_1, V'_2) of G' such that $V'_{\square} \in V'_2$. We first show that all the vertices in V_{Δ} must lie in V'_1 . Let us assume that there exists a vertex $u \in V_{\Delta}$ that lies in V'_2 . Then each vertex in X_{uv} has at least 4 neighbors in V'_2 and at most 2 neighbours in V'_1 ; therefore all the vertices in X_{uv} lie in V'_2 . In this case, we cannot get a balanced satisfactory partition as already more than half of the vertices are in V'_2 . This proves that all the vertices in V_{Δ} lie in V'_1 . Next, we show that as $V_{\Delta} \subseteq V'_1$, the vertices in $\bigcup_{u \in V_{\Delta}} X_{uv}$ also lie in V'_1 . Since $u \in V'_1$, it must be

satisfied in V'_1 . As the vertices in Y_{uv} lie in V'_2 , u has at least n neighbors in V'_2 and since u has at most $n - 1$ neighbors in graph G , it implies that at least one vertex from X_{uv} must be in V'_1 . Without loss of generality, we can assume that $x_1^{uv} \in V'_1$. Since $x_1^{uv} \in V'_1$, it must be satisfied in V'_1 and this forces x_n^{uv}, x_2^{uv} to be in V'_1 as well. Repetively applying the above argument we get that all the vertices in set X_{uv} lie in V'_1 . We claim that $(V'_1 \cap V(G), V'_2 \cap V(G))$ forms a balanced satisfactory partition of graph G . As for each vertex in $V'_i \cap V(G)$, $i = 1, 2$, we are removing equal number of neighbors from both the partitions, this implies that all the vertices are satisfied and the partition is balanced. This shows that I is a yes-instance.

Lemma 6. The BALANCED SATISFACTORY PARTITION problem, parameterized by the treewidth of the graph, is $W[1]$ -hard.

Proof. Let $I = (G, V_{\square})$ be a BALANCED SATISFACTORY PARTITION^S instance, where $V_{\square} = \{u_1, u_2, \dots, u_{n'}\}$. For every vertex u_i in the set V_{\square} , we introduce two new sets of vertices $X^{u_i} = \{x_1^{u_i}, x_2^{u_i}, \dots, x_{4n}^{u_i}\}$ and $Y^{u_i} = \{y_1^{u_i}, y_2^{u_i}, \dots, y_{4n}^{u_i}\}$. We also introduce a clique of size 2 containing new vertices $\{s, t\}$ and a set $C = \{c_1, c_2, \dots, c_{8n}\}$ of $8n$ vertices. We add two new vertices $\{s', t'\}$ along with two sets of vertices $S' = \{s'_1, s'_2, \dots, s'_{4n}\}$ and $T' = \{t'_1, t'_2, \dots, t'_{4n}\}$. Now, we define the BALANCED SATISFACTORY PARTITION instance $I' = G'$ where G' is the graph defined by

$$V(G') = V(G) \cup \bigcup_{i=1}^{n'} X^{u_i} \cup \bigcup_{i=1}^{n'} Y^{u_i} \cup \{s, t, s', t'\} \cup S' \cup T' \cup C$$

and

$$E(G') = E(G) \cup \bigcup_{i=1}^{n'} \bigcup_{j=1}^{4n} \{(x_j^{u_i}, u_i), (y_j^{u_i}, u_i), (y_j^{u_i}, s), (y_j^{u_i}, t)\} \cup \{(s, t)\} \\ \bigcup_{j=1}^{8n} \{(c_j, s), (c_j, t)\} \cup \bigcup_{j=1}^{4n} \{(s', s'_j), (t', t'_j)\} \cup \{(s', s)(s', t), (t', s), (t', t)\}.$$

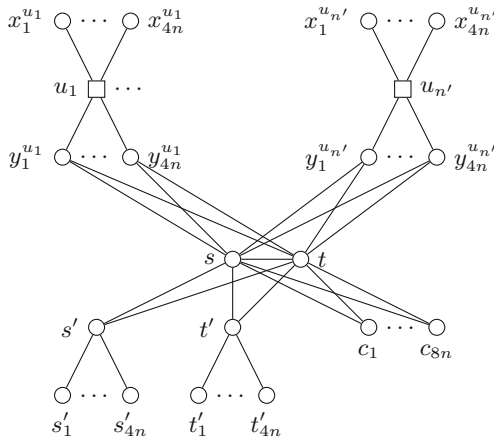


Fig. 4. An illustration of the reduction from BALANCED SATISFACTORY PARTITION^S to BALANCED SATISFACTORY PARTITION.

Now we claim that I is a yes-instance if and only if I' is a yes-instance. Assume first that there exists a balanced satisfactory partition (V_1, V_2) in the

graph G such that $V_{\square} \subseteq V_2$. In this case, a balanced satisfactory partition (V'_1, V'_2) for G' is defined as follows:

$$V'_1 = V_1 \cup \bigcup_{i=1}^{n'} Y^{u_i} \cup \{s, t\} \cup C \text{ and } V'_2 = V_2 \cup \bigcup_{i=1}^{n'} X^{u_i} \cup \{s', t'\} \cup S' \cup T'.$$

Clearly, all the vertices are satisfied. Since we are adding equal number of vertices in both the parts, (V'_1, V'_2) is a balanced satisfactory partition of G' . This proves that if I is a yes-instance then I' is a yes-instance.

Conversely, suppose that there exists a balanced satisfactory partition (V'_1, V'_2) of G' . We first prove that all the vertices of V_{\square} are in the same part. Since $N_{G'}[s] = N_{G'}[t]$, both s and t would be in the same part; without loss of generality suppose they lie in V'_1 . For $1 \leq i \leq n'$, each vertex $y_j^{u_i}$ is adjacent to 3 vertices $\{u_i, s, t\}$ and since $\{s, t\}$ belong to V'_1 , it forces $y_j^{u_i}$ to be in V'_1 for $1 \leq j \leq 4n$. Similarly, as $s, t \in V'_1$, each c_i would also be in V'_1 for $1 \leq i \leq 4n$. For the sake of contradiction, suppose the vertices of V_{\square} are distributed among V'_1 and V'_2 , that is, r many vertices of V_{\square} are in V'_1 and remaining $n' - r$ vertices of V_{\square} are in V'_2 . This implies that V'_1 contains at least $4n(n' + r + 2) + r + 2$ vertices and V'_2 contains at most $4n(n' - r + 2) + 2 + (n - r)$. It implies that $|V'_1| > |V'_2|$, a contradiction to our assumption that (V'_1, V'_2) is a balanced satisfactory partition. This shows that all the vertices of V_{\square} must go to V'_2 . Therefore, for every balanced satisfactory partition of G' , we have

$$\bigcup_{i=1}^{n'} \bigcup_{j=1}^{4n} \{y_j^{u_i}\} \cup \{s, t\} \cup C \subseteq V'_1 \text{ and } \bigcup_{i=1}^{n'} \bigcup_{j=1}^{4n} \{x_j^{u_i}\} \cup \{s', t'\} \cup S' \cup T' \cup V_{\square} \subseteq V'_2.$$

We now claim that $(V'_1 \cap V(G), V'_2 \cap V(G))$ forms a balanced satisfactory partition of G . From the above observation, we have $V_{\square} \subseteq V'_2 \cap V(G)$. All the vertices are satisfied in the new partition $(V'_1 \cap V(G), V'_2 \cap V(G))$ and it is a balanced partition because we are removing equal number of vertices from both parts. This shows that if I' is a yes-instance then I is also a yes-instance.

This proves Theorem 2.

4 Conclusion

In this work we proved that the SATISFACTORY PARTITION and BALANCED SATISFACTORY PARTITION problems are FPT when parameterized by neighbourhood diversity; the BALANCED SATISFACTORY PARTITION problem is W[1]-hard when parameterized by treewidth. The parameterized complexity of the SATISFACTORY PARTITION problem remains unsettled when parameterized by other important structural graph parameters like clique-width and modular width.

Acknowledgement. The first author gratefully acknowledges support from the Ministry of Human Resource Development, Government of India, under Prime Minister's Research Fellowship Scheme (No. MRF-192002-211).

References

1. Bazgan, C., Tuza, Z., Vanderpooten, D.: On the existence and determination of satisfactory partitions in a graph. In: Ibaraki, T., Katoh, N., Ono, H. (eds.) ISAAC 2003. LNCS, vol. 2906, pp. 444–453. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24587-2_46
2. Bazgan, C., Tuza, Z., Vanderpooten, D.: Complexity and approximation of satisfactory partition problems. In: Wang, L. (ed.) COCOON 2005. LNCS, vol. 3595, pp. 829–838. Springer, Heidelberg (2005). https://doi.org/10.1007/11533719_84
3. Bazgan, C., Tuza, Z., Vanderpooten, D.: Degree-Constrained decompositions of graphs: bounded treewidth and planarity. *Theor. Comput. Sci.* **355**(3), 389–395 (2006)
4. Bazgan, C., Tuza, Z., Vanderpooten, D.: The satisfactory partition problem. *Discr. Appl. Math.* **154**(8), 1236–1245 (2006)
5. Cygan, M., et al.: *Parameterized Algorithms*. Springer, Cham (2015). <https://doi.org/10.1007/978-3-319-21275-3>
6. Gerber, M.U., Kobler, D.: Classes of graphs that can be partitioned to satisfy all their vertices. *Australas. J. Combin.* **29**, 201–214 (2004)
7. Gerber, M.U., Kobler, D.: Algorithmic approach to the satisfactory graph partitioning problem. *Eur. J. Oper. Res.* **125**(2), 283–291 (2000)
8. Gerber, M.U., Kobler, D.: Algorithms for vertex-partitioning problems on graphs with fixed clique-width. *Theoret. Comput. Sci.* **299**(1), 719–734 (2003)
9. Kannan, R.: Minkowski’s convex body theorem and integer programming. *Math. Oper. Res.* **12**(3), 415–440 (1987)
10. Lampis, M.: Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica* **64**, 19–37 (2012)
11. Lenstra, H.W.: Integer programming with a fixed number of variables. *Math. Oper. Res.* **8**(4), 538–548 (1983)
12. N. Robertson and P. Seymour. Graph minors. iii. planar tree-width. *J. Combinatorial Theory, Series B*, **36**(1), 49–64 (1984)
13. Szeider, S.: Not so easy problems for tree decomposable graphs. CoRR, abs/1107.1177 (2011)
14. Tedder, M., Corneil, D., Habib, M., Paul, C.: Simpler linear-time modular decomposition via recursive factorizing permutations. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfssdóttir, A., Walukiewicz, I. (eds.) ICALP 2008. LNCS, vol. 5125, pp. 634–645. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70575-8_52