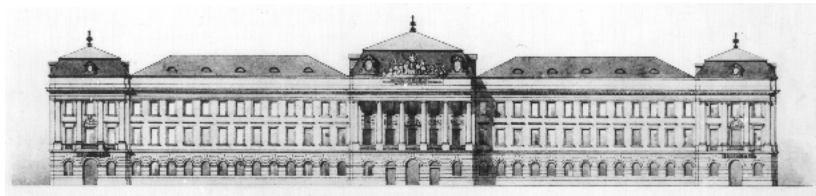
I N F S Y S
RESEARCH
R E P O R T



INSTITUT FÜR INFORMATIONSSYSTEME
ABTEILUNG WISSENSBASIERTE SYSTEME

EDITING GRAPHS TO SATISFY DEGREE CONSTRAINTS: A PARAMETERIZED APPROACH

Luke Mathieson   Stefan Szeider

INFSYS RESEARCH REPORT 1843-12-01
JANUARY 2012

Institut für Informationssysteme
Abtg. Wissensbasierte Systeme
Technische Universität Wien
Favoritenstraße 9-11
A-1040 Wien, Austria
Tel: +43-1-58801-18405
Fax: +43-1-58801-18493
sek@kr.tuwien.ac.at
www.kr.tuwien.ac.at

TECHNISCHE UNIVERSITÄT WIEN

INFSYS RESEARCH REPORT
INFSYS RESEARCH REPORT 1843-12-01, JANUARY 2012

EDITING GRAPHS TO SATISFY DEGREE CONSTRAINTS: A PARAMETERIZED APPROACH

Luke Mathieson[1] Stefan Szeider[2]

Abstract. We study a wide class of graph editing problems that ask whether a given graph can be modified to satisfy certain degree constraints, using a limited number of vertex deletions, edge deletions, or edge additions. The problems generalize several well-studied problems such as the General Factor Problem and the Regular Subgraph Problem. We classify the parameterized complexity of the considered problems taking upper bounds on the number of editing steps and the maximum degree of the resulting graph as parameters.

Keywords: Parameterized Complexity, Computational Complexity, Kernelization, Graph Editing, Regular Subgraph, General Factor.

---

[1] CIBM, Discipline of Computer Science and Software Engineering, The University of Newcastle, Callaghan, NSW 2308, Australia
[2] Institute of Information Systems, Vienna University of Technology, A-1040 Vienna, Austria

Acknowledgements: Research of Szeider was partially supported by the European Research Council (ERC), Grant 239962 (COMPLEX REASON).

Publication Information: This is the author's self-archived copy of a paper that appeared in the *Journal of Computer and Systems Sciences*, vol. 78, no. 1, pp. 179–191, 2012. doi:10.1016/j.jcss.2011.02.001. The final publication is available at www.sciencedirect.com
This paper contains results that were published in preliminary form in [23, 24]

1 Introduction

The problem of finding a regular subgraph in a given graph and related problems have a long history [2, 3, 4, 5, 6, 8, 9, 21, 22, 28, 29, 30, 31, 32, 34, 36, 37]. It is often of relevance how much the regular subgraph differs from the given graph. For instance, if an interconnection network is regular, then one can use identical switches for its nodes to save costs [12]; if it is not regular, then the question arises whether we can make it regular with a small editing cost. Indeed, Moser and Thilikos [25] studied the problem of whether it is possible to obtain from the given graph an r-regular subgraph by deleting at most k vertices. They showed that this problem is tractable if k and r are considered small; more precisely they showed that the problem is *fixed-parameter tractable* for parameter $k + r$ which means that for fixed $k + r$ the problem can be solved in polynomial time where the order of the polynomial is independent of $k + r$ (more background on parameterized complexity is given in Section 2.2). In this paper we extend this study to a more general setting as follows:

- We consider several editing operations including edge addition, so that in some cases the obtained graph is not a subgraph of the given graph. In particular we consider the editing operations *vertex deletion* (denoted v), *edge deletion* (denoted e), and *edge addition* (denoted a).
- Each vertex v of the given graph has assigned a list $\delta(v)$ of numbers; after the editing process the degree of the vertex must belong to its list. For example, by assigning all vertices the list $\{r\}$ we can force the target graph to be r-regular. By allowing arbitrary lists we can express variants of the GENERAL FACTOR problem introduced and studied by Lovász [19, 20].
- Vertices and edges can have positive integer weights, giving each edit operation a certain *cost*.

Now, for each non-empty subset $S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$ we consider the general problem WEIGHTED DEGREE CONSTRAINT EDITING(S), or WDCE(S) for short:

WDCE(S)

Instance: A graph $G = (V, E)$, two integers k and r, a weight function $\rho : V \cup E \to \{1, 2, \dots\}$, and a degree list function $\delta : V \to 2^{\{0,\dots,r\}}$.

Question: Can we obtain from G a graph $G' = (V', E')$ using editing operations from S only, such that for all $v \in V'$ we have $\sum_{uv \in E'} \rho(uv) \in \delta(v)$, with total editing cost at most k?

Our main result is the classification of the parameterized complexity of WDCE(S) for all choices of S and where the parameter is either just the total editing cost k or the combination $k + r$ of the total editing cost and the upper bound on the desired degrees:

Theorem 1.1 (Classification Theorem)**.** *For all non-empty subsets S of $\{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$ the problem* $\mathrm{WDCE}(S)$ *is fixed-parameter tractable for parameter $k + r$, and* W[1]*-hard for parameter k. If $\mathsf{v} \in S$ then* $\mathrm{WDCE}(S)$ *remains* W[1]*-hard for parameter k even when all degree lists are restricted to $\{r\}$ and all vertices and edges have unit weight* 1.

The W[1]-hardness of a parameterized problem provides strong evidence that the problem is not fixed-parameter tractable, similar to NP-hardness providing strong evidence that a problem is not polynomial-time tractable (see Section 2.2).

We write $\mathrm{WDCE}_1(S)$ to indicate that the given graph is unweighted, and we write $\mathrm{WDCE}^*(S)$ if all degree lists are singletons; if all singletons are $\{r\}$ then we write $\mathrm{WDCE}^r(S)$. Furthermore, we write ${}_\infty\mathrm{WDCE}(S)$ to indicate that the editing cost is not restricted (or set to a value that exceeds the sum of weights of all vertices and edges). We omit set braces whenever the context allows, and write, for example, $\mathrm{WDCE}(\mathsf{v})$ instead of $\mathrm{WDCE}(\{\mathsf{v}\})$.

Related Work Next we review some known problems and results and indicate how they map into our general problem $\mathrm{WDCE}(S)$.

The problem CUBIC SUBGRAPH $= {}_\infty\mathrm{WDCE}_1^3(\mathsf{v}, \mathsf{e})$ was shown NP-complete by Chvátal [17]. Stewart has shown that the problem r-REGULAR SUBGRAPH $= {}_\infty\mathrm{WDCE}_1^r(\mathsf{v}, \mathsf{e})$ remains NP-complete for any $r \geq 3$, even for graphs with maximum degree 7, and planar graphs with maximum degree 4 [30, 31]. With $\mathrm{WDCE}_1^r(\mathsf{v})$ we can express the problem of finding an induced r-regular subgraph of largest order which includes several known NP-complete problems: MAXIMUM INDEPENDENT SET $= \mathrm{WDCE}_1^0(\mathsf{v})$ [17], MAXIMUM INDUCED MATCHING $= \mathrm{WDCE}_1^1(\mathsf{v})$ [4, 34], and MAXIMUM INDUCED r-REGULAR SUBGRAPH $= \mathrm{WDCE}_1^r(\mathsf{v})$ [5, 25].

The problem r-FACTOR $= {}_\infty\mathrm{WDCE}_1^r(\mathsf{e})$ asks whether a given graph has a *spanning* r-regular subgraph; this problem is well known to be solvable in polynomial time as it can be reduced to the problem PERFECT MATCHING $= {}_\infty\mathrm{WDCE}_1^1(\mathsf{e})$ as shown by Tutte [35]. In fact Tutte's result holds for the more general problems f-FACTOR $= {}_\infty\mathrm{WDCE}_1^*(\mathsf{e})$ and PERFECT b-MATCHING $= {}_\infty\mathrm{WDCE}^*(\mathsf{e})$ [18]. Lovász [19, 20] considered the generalization of this problem, GENERAL FACTOR $= {}_\infty\mathrm{WDCE}_1(\mathsf{e})$, where each vertex is given a list of possible degrees. By a result of Cornuéjols [9], GENERAL FACTOR is NP-complete, apart from certain trivial cases, if the lists may contain gaps of length > 1, such as in $\{2, 3, 6, 7\}$, but is otherwise polynomially solvable.

Moser and Thilikos [25] have established the fixed-parameter tractability of k-ALMOST r-REGULAR GRAPH $= \mathrm{WDCE}_1^r(\mathsf{v})$ with combined parameter $k + r$ but left the parameterized complexity of the same problem with parameter k as an open question. Our Classification Theorem shows that the latter is W[1]-hard, being a special case of $\mathrm{WDCE}(\mathsf{v})$.

Organization, Approach, and Further Results After giving further preliminaries and background information in Section 2, we establish in Section 3 the W[1]-hardness part of the Classification Theorem. We proceed by reducing the parameterized CLIQUE problem. In particular, we use a special variant of CLIQUE which asks whether a given graph, vertex colored with k colors, contains a properly colored k-clique. This W[1]-complete problem has recently proven very useful for several W[1]-hardness results.

In Section 4 we establish the fixed-parameter tractability part of the Classification Theorem. We take a logic approach and apply (after preprocessing) the algorithmic meta-theorem of Frick and Grohe [16]. The meta-theorem allows a quick and elegant classification of the parameterized complexity of problems but does not provide practical algorithms since the hidden constants are large. Therefore we develop more practical algorithms for some of the fixed-parameter tractable cases.

In Section 5 we show that the problems $\mathrm{WDCE}^*(S)$ and $\mathrm{WDCE}_1^*(S)$ for $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$ are in fact solvable in polynomial time. Ultimately we make use of Edmond's minimum weight perfect matching algorithm [13, 22].

In Section 6 we use *kernelization*, an algorithmic technique that transforms in polynomial time a problem instance into an equivalent instance of size bounded by a function of the parameter (the reduced instance is called a problem kernel). In particular, we develop efficient kernelizations for the problems $\mathrm{WDCE}^*(\mathsf{v}, \mathsf{e})$ and $\mathrm{WDCE}^*(\mathsf{v})$ which also apply to $\mathrm{WDCE}_1^r(\mathsf{v})$, obtaining a kernel of size $O(kr(k + r))$, improving upon the $O(kr(k + r)^2)$ kernel of Moser and Thilikos [25]. An interesting feature of our approach is that we can achieve smaller kernels for the unweighted versions of the editing problems by considering them as special cases of the weighted version. This was initially our motivation for considering weighted versions of the editing problems. Further discussion of weights (or annotations) with respect to kernelization can be found in Abu-Khzam and Fernau's paper [1].

It appears that our kernelization approach cannot be extended to all fixed-parameter tractable cases of the Classification Theorem, in particular not to cases that include both vertex deletion and edge addition: we show that our approach applied to $\mathrm{WDCE}^*(\mathsf{v}, \mathsf{a})$ involves an NP-hard task.

2 Preliminaries

2.1 Graph theoretic terminology

All graphs considered are finite, simple, and undirected. We denote a graph G with vertex set V and edge set E by $G = (V, E)$ and write $V(G) = V$ and $E(G) = E$. The edge between two vertices u and v is denoted uv or equivalently vu. The degree of a vertex u in a graph G is denoted $d_G(u)$; when the context allows, we omit the subscript. A graph $H = (V', E')$ is a *subgraph* of a graph $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$. The subgraph H is *induced* by V' if every edge $uv \in E$ with $u, v \in V'$ also belongs to E'. We consider graph editing operations that alter a graph $G = (V, E)$ into a new graph $G' = (V', E')$.

Applying the operation of *edge deletion* e to an edge uv removes that edge from the graph (i.e., $E' = E \setminus \{uv\}$, and $V' = V$). Applying the operation of *edge addition* a to a pair of vertices u, v inserts an edge between u and v (i.e., $E' = E \cup \{uv\}$, $V' = V$). In the unweighted case, edge addition is only possible if the edge uv is not already present in E. Applying the operation of *vertex deletion* v to a vertex u removes u and all edges incident with u (i.e., $V' = V \setminus \{u\}$, $E' = E \setminus \{\, uv \mid v \in V \,\}$); we also write $G' = G - v$. Each of these operations has unit *cost* 1.

We also consider *weighted graphs* where each vertex and edge x is given with an integral positive weight $\rho(x)$. The weights have two purposes: firstly, the weight of an edge contributes to the weighted degree of the incident vertices, and secondly, the weight of a vertex or edge determines the cost of deleting that vertex or edge; more precise definitions follow.

The *weighted degree* of a vertex $v \in V$ is given by $d_G^\rho(v) = \sum_{uv \in E} \rho(uv)$. Thus an edge uv of weight j can be thought as representing j "parallel edges" between u and v.

Applying the operation e to a weighted edge e either reduces the weight of e by 1 if $\rho(e) > 1$, or removes e if $\rho(e) = 1$. Similarly, applying the operation a to a pair of distinct vertices u, v either increases the weight of the edge $e = uv$ by 1 if e is already present, or adds the edge e with weight 1 if the edge is not present; thus, for edge addition we need to consider weighted and unweighted graphs separately, as the latter is not just a special case of the former. Applying the operation v to a vertex u either removes the vertex (and all incident edges) if $\rho(u) = 1$ or decreases $\rho(u)$ by 1 if $\rho(u) > 1$. Again, each of these operations has unit cost 1. Hence the total cost of deleting a vertex or edge of a weighted graph is the weight of that vertex or edge.

2.2 Basic Parameterized Complexity

Here we introduce the relevant concepts of parameterized complexity theory. For a more in-depth coverage we refer to the books of Downey and Fellows [11], Flum and Grohe [15] and Niedermeier [26].

When considering the complexity of a problem in a traditional P vs. NP setting, the only measure available is n, the instance size (or some function thereof). Parameterized complexity adds a second measure, that of a parameter k, which is given as a special part of the input. If a (decision) problem has an algorithm that runs in time $O(f(k)p(n))$, where p is a polynomial and f is a computable function of k, then the problem is *fixed-parameter tractable*, or in the class FPT. Conversely, the demonstration of hardness for the class W[t] for some $t \geq 1$ provides strong evidence that the problem is unlikely to be fixed-parameter tractable. This is analogous to a problem being NP-hard in the traditional framework.

Reduction to a problem kernel, or *kernelization*, is one of the fundamental techniques for demonstrating fixed-parameter tractability. Kernelization is a polynomial-time transformation that replaces an instance (I, k) of a parameterized problem (where I is the input and k the parameter) by an instance (I', k') where $|I'| \leq g(k')$ and $k' \leq h(k)$ for computable functions g and h, such that (I, k) is a *yes*-instance if and only if (I', k') is a *yes*-instance. It is known that a parameterized problem admits a kernelization if and only if it is fixed-parameter tractable [10].

To demonstrate W[t]-hardness we use *FPT reductions*. Given two parameterized problems Π_1 and Π_2, an FPT reduction from Π_1 to Π_2 maps an instance (I, k) of Π_1 to an instance (I', k') of Π_2 such that

1. $k' = h(k)$ for some computable function h,
2. (I, k) is a *yes*-instance of Π_1 if and only if (I', k') is a *yes*-instance of Π_2, and
3. the mapping can be computed in time $O(f(k)p(n))$, where f is some computable function of the parameter k and p is a polynomial of the instance size n.

An FPT reduction is a *polynomial-time* FPT reduction if the function f is a polynomial.

The classes W[t], $t = 1, 2, \ldots$, are defined as the classes of parameterized problems that can be reduced to certain weighted satisfiability problems via FPT reductions (see [11, 15]). The classes form a chain FPT $\subseteq$ W[1] $\subseteq$ W[2] $\subseteq \cdots$ where all inclusions are believed to be strict. The class M[1] sits between FPT and W[1], and is believed to at least be distinct from FPT under a parameterized version of the Exponential Time Hypothesis.

Assume there is an FPT reduction from Π_1 to Π_2. If Π_2 is in FPT then so is Π_1, and if Π_1 is W[t]-hard then so is Π_2. If the reduction is a polynomial-time FPT reduction and Π_1 is NP-hard, then clearly Π_2 is NP-hard as well.

2.3 A Useful Construction: The Fixing Gadget

Here we introduce a construction that we will later use in several proofs. For any $r \geq 2$, this construction produces an almost r-regular graph G, where all vertices have degree r except two with degree $r - 1$. We will refer to an instance of G_r as a *fixing gadget*. The vertex set of G_r consists of the vertices x^j, y_i^j, z_i^j for $1 \leq j \leq 2$ and $1 \leq i \leq r$. The edge set consists of all edges $x^j y_i^j$ and $y_i^j z_{i'}^j$ for $1 \leq j \leq 2$ and $1 \leq i, i' \leq r$, $i \neq i'$, and all edges $z_i^1 z_i^2$ for $1 \leq i \leq r-1$.

Thus each vertex has degree r, except z_r^1 and z_r^2 which have degree $r - 1$. The two vertices of degree $r - 1$ will be used as *attachment points*.

3 W[1]-Hardness of WDCE for Parameter k

We now turn to the hardness part of the Classification Theorem.

Ultimately, all our reductions are from the CLIQUE problem, parameterized by the number of vertices that form the clique, a fundamental W[1]-complete problem [11].

> CLIQUE
>
> *Instance:* A graph $G = (V, E)$ and an integer k.
>
> *Question:* Does G contain a k-clique (i.e., a complete subgraph on k vertices)?

Without parameterization, CLIQUE is NP-complete [17], and all our reductions are in fact polynomial-time FPT reductions. Therefore, all problems that we show to be W[1]-hard are NP-hard if considered without parameterization. This holds in particular for the problem considered in the next lemma; we shall use its NP-hardness part explicitly at the end of Section 6.

First we show that REGULAR CLIQUE (the problem CLIQUE restricted to regular graphs) is W[1]-complete.

Lemma 3.1. *The problem* REGULAR CLIQUE *is* NP-*complete and* W[1]-*complete for parameter* k.

Proof. Membership in NP of the non-parameterized version and membership in W[1] for the parameterized version of the problem follow immediately as the problem is a special case of CLIQUE.

To prove hardness we devise a polynomial-time FPT reduction from CLIQUE. Let (G, k) be an instance of CLIQUE. We construct an instance (G', k) of REGULAR CLIQUE by modifying G as follows. Let Δ be the maximum degree of G, and let $r = \Delta + (\Delta \mod 2)$. We will now demonstrate how to make the graph r-regular. We use fixing gadgets (see Section 2.3) to increase the degree of each vertex by attaching as many fixing gadgets as necessary by the two attachment vertices. This attachment is made between a vertex v and an instance of the fixing gadget by adding the edges between each attachment vertex and v (or perhaps only one of these edges, as below). If the degree of v is initially even, then we use an integral number of fixing gadgets; if the degree of v is initially odd, then will reach degree $r-1$ by this method, and we will have to take another degree $r - 1$ vertex and attach one fixing gadget attachment vertex to the first, and the other attachment vertex to the second. There is always some pairing of such vertices as necessary as every graph contains an even number of vertices of odd degree, and so there is an even number of vertices requiring an odd increase of degree.

The fixing gadgets added to create G' contain no non-trivial cliques and do not introduce any new non-trivial cliques since the two attachment vertices in a fixing gadget are not adjacent. Thus G and G' have exactly the same non-trivial cliques. Clearly G' can be constructed from G in polynomial time, and since the parameter remains the same we have a polynomial-time FPT reduction. □

The reduction for several of our hardness results will be from the following variant of REGULAR CLIQUE.

> STRONGLY REGULAR MULTI-COLORED CLIQUE (or SRMCC)
>
> *Instance:* A graph $G = (V, E)$, vertex-colored with k colors, where each vertex has exactly d neighbors in each of the k color classes (thus G is kd-regular and each color class has the same size).
>
> *Question:* Does G contain a properly colored k-clique (i.e., a k-clique whose vertices have all different colors)?

The proof of the next lemma follows closely the W[1]-completeness proof of the problems MULTI-COLORED CLIQUE and PARTITIONED CLIQUE in [14, 27], respectively.

Lemma 3.2. *The problem* SRMCC *is* W[1]*-complete for parameter* k.

Proof. Again W[1] membership follows as the problem is a special case of CLIQUE. We reduce from REGULAR CLIQUE. Given an instance (G, k) of REGULAR CLIQUE where G is d-regular, we let G' be the union of k vertex disjoint copies $G_1, \ldots, G_k$ of G, assigning the vertices of G_i the color i. Then for every pair of vertices u, v in G, if uv is an edge, we add the edges $u_i v_j$, for all i, j, where a_i denotes the vertex in G_i which corresponds to vertex a in G. Clearly every vertex of G' has exactly d neighbors in each color class, and there is a k-clique in G if and only if there is a properly colored k-clique in G'. □

We note that the reductions used for Lemmas 3.1 and 3.2 are computable in polynomial time and the parameter dependence is linear (in fact in both cases the parameter remains unchanged) and hence are "*serf*-reductions" [15]. This observation allows us to obtain the following lower bound: The problems REGULAR CLIQUE and SRMCC have no algorithm running in time $n^{o(k)}$ (omitting polynomial factors) unless FPT = M[1]. Here n is the number of vertices of the input graph and k is size of the clique sought. The lower bound follows from a similar lower bound for CLIQUE [7] and the equivalence of a parameterized version of the Exponential Time Hypothesis and the assumption FPT $\neq$ M[1] [15]. We cannot obtain a similar lower bound for the problems considered in Theorem 3.3 as the parameter transformation in the proof is not linear.

Theorem 3.3. *The problem* $\mathrm{WDCE}_1^r(S)$ *is* W[1]*-hard for parameter* k *and* $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$*,*

Proof. Consider an instance (G, k) of SRMCC. Let $G = (V, E)$ be kd-regular, thus each vertex has exactly d neighbors in each color class. We denote the set of vertices of color i by V_i ($1 \leq i \leq k$). Then $V = \bigcup_{i=1}^{k} V_i$ forms a partition of V. Let $|V_i| = s$ for all $1 \leq i \leq k$.

We construct an instance (G', k') of $\mathrm{WDCE}_1^r(S)$, $G' = (V', E')$, by first defining k sets V_i' ($1 \leq i \leq k$) such that for each vertex $v \in V_i$ we add a vertex v' to V_i'. We add all possible edges between pairs of vertices in the same set V_i'. We call each of these k subgraphs induced by V_i' a (color) *class gadget*.

For each edge uv in G where $u \in V_i$ and $v \in V_j$ with $i \neq j$, we add to G' two vertices $u'_{v'}$ and $v'_{u'}$, with the edges $u'u'_{v'}$, $u'_{v'}v'_{u'}$ and $v'_{u'}v'$. For each pair V_i' and V_j' (where $i \neq j$) of class gadgets, we denote the set of these new vertices and edges as P_{ij}. We denote by P_{ij}^i the set of all vertices $u'_{v'} \in P_{ij}$ where $u' \in V_i'$. Furthermore, for each pair of vertices u_v and $u'_{v'}$ in the same P_{ij}^i we add the edge $u_v u'_{v'}$ to P_{ij}^i if u and u' belong to the same class gadget and $u \neq u'$. We call each such P_{ij} a *connection gadget*, and each P_{ij}^i a *side* of the connection gadget. There are $\binom{k}{2}$ connection gadgets in total. Fig. 1 gives a sketch of the structure of a connection gadget.

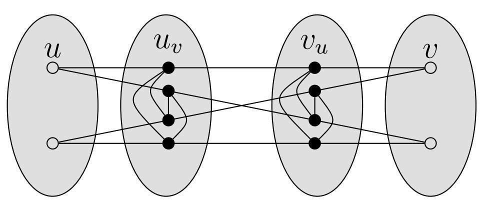


Figure 1: An illustration of the gadget construction in the proof of Theorem 3.3.

At this point we have k class gadgets corresponding to the k color classes in the original graph, each with s vertices of degree $(s-1) + d(k-1)$, and $\binom{k}{2}$ connection gadgets corresponding to the "inter-color-class" edges, each with $2sd$ vertices of degree $2 + (s-1)d$ (sd vertices in each half). Now we choose r for the instance such that $r \geq \max((s-1) + d(k-1), 2 + (s-1)d)$, and $r \equiv s + 1$ modulo 2 (i.e., r is of opposite parity to s). In particular we may choose the smallest r such that this is true.

Now we add for each class gadget V_i' a gadget V_i'' that contains $r+1-((s-1)+d(k-1))$ vertices with s edges per vertex, such that each vertex in V_i'' is adjacent to every vertex in the class gadget V_i'. We refer to V_i'' as a *degree gadget*. We then add a further set of fixing gadgets as before to complete the degree of each vertex in the degree gadget to $r+1$. Note that by choosing r to have opposite parity to s, we guarantee that this is possible (if s is odd, r will be even and each vertex will require $r+1-s$ additional edges, which is even, and thus achievable; if s is even, r will be odd, then $r+1-s$ is again even, and we can complete the construction). Thus each vertex in each class gadget and degree gadget has degree one too many, but the vertices in the fixing gadgets attached to each degree gadget have the correct degree.

We similarly adjust the connection gadgets by adding two degree gadgets, each with $r+1-2+(s-1)d$ vertices, one for each side of the connection gadget. Every vertex in the degree gadget is connected to every vertex in its associated side of the connection gadget. Again we complete the degree of vertices in the degree gadgets to $r+1$ by adding fixing gadgets, and as before, by the choice of r we can guarantee that this can be done (if s is even, r is odd and $r+1-sd$ is even, if s is odd, r is even and $r+1-sd$ is even). Thus each vertex in the connection gadgets has degree $r+1$, as does each vertex in the degree gadgets. Each vertex in each fixing gadget has degree r.

We can construct G' from G in polynomial time, as we are adding only $(4r+3)(2r+2s-s-sd-dk+1)$ vertices, where $r,s,d \leq n$. Thus, by the following claim we have a polynomial-time FPT reduction from SRMCC to $\text{WDCE}_1^r(S)$, and the result follows.

We set $k' = k + 2\binom{k}{2}$.

Claim 3.4. The following statements are equivalent:

1. (G,k) is a *yes*-instance of SRMCC.
2. (G',k') is a *yes*-instance of $\text{WDCE}_1^r(\mathsf{v})$.
3. (G',k') is a *yes*-instance of $\text{WDCE}_1^r(\mathsf{v},\mathsf{e})$.
4. (G',k') is a *yes*-instance of $\text{WDCE}_1^r(\mathsf{v},\mathsf{e},\mathsf{a})$.

$(1 \Rightarrow 2)$ Assume that (G,k) is a *yes*-instance of SRMCC. Then there exist k vertices $v_1,\dots,v_k$, one from each color class, that form a properly colored clique. Assume without loss of generality that $v_i \in V_i$. Then we can delete from G' the corresponding vertices v_i' from V_i', and the pairs of vertices $(v_i')_{v_j'}$ and $(v_j')_{v_i'}$ from P_{ij} that correspond to the edges in the clique. Then each remaining vertex in each class gadget has had precisely one incident edge removed from it, as have the vertices in each degree gadget associated with the class gadget. So the components corresponding to the color classes and their immediate extension are now r-regular. Similarly each vertex in every connection gadget and their associated connection gadgets has had exactly one incident edge removed, either by the vertex removed from the connection gadget, or from the parent vertex in the class gadget (but never both). Now each vertex in these gadgets has degree precisely r. We have chosen one vertex from each V_i', and two vertices from each P_{ij}, giving a total of $k' = k + 2\binom{k}{2}$ vertices, thus (G',k') is also a *yes*-instance of $\text{WDCE}_1^r(\mathsf{v})$.

$(2 \Rightarrow 3, 3 \Rightarrow 4)$ These implications are trivial.

($4 \Rightarrow 1$) Assume that (G', k') is a *yes*-instance of $\text{WDCE}^r_1(\mathsf{v}, \mathsf{e}, \mathsf{a})$. Then there are $k' = k+2\binom{k}{2}$ deletions that can be made to make G' r-regular. Obviously we cannot delete any vertices from the fixing gadgets in the graph. Further we cannot delete any vertices from the degree gadgets, as this would reduce the degree of their attached fixing gadgets. Thus the deleted vertices must come from class and connection gadgets. However, there must be precisely one vertex from each such component: if no vertex is deleted then the degree of some vertex in that component will remain $r + 1$; if more than one vertex is deleted, then the degree of some vertices in the component will drop below r. Also note that for each vertex u_v deleted from one side of a connection gadget, the vertex deleted in the other side must be the vertex v_u. If it were not, then at least one vertex in each side would have degree $r - 1$. Also, the vertex deleted from each side of each connection gadget must be attached to the vertex deleted from the adjacent class gadget, otherwise the vertices attached to vertices deleted from the class gadget will have degree at most $r - 1$. Thus we can see that if (G', k') is a *yes*-instance, the set of vertices to be deleted is very precise and restricted. In fact, if we are to use only the allotted budget of $k + 2\binom{k}{2}$, we must choose precisely one vertex from each class gadget, and two vertices from each connection gadget, where the vertices from the connection gadget component are connected to the vertices deleted from the two class gadgets it is associated with. An edge deletion can only reduce the degree of two vertices, leaving us with too many edges to delete, or vertices of degree less than r. Similarly, the tight budget implies that no edge addition can be used either. Thus (G', k') is a *yes*-instance of $\text{WDCE}^r_1(\mathsf{v})$. Furthermore, the vertices $v_1, \ldots, v_k$ corresponding to the k vertices deleted from the class gadgets induce a properly colored clique in G; the edges of the clique correspond to the $2\binom{k}{2}$ vertices deleted from the connection gadgets. Hence (G, k) is a *yes*-instance of SRMCC. □

Corollary 3.5. *The problems* $\text{WDCE}^*_1(S)$ *and* $\text{WDCE}_1(S)$ *are* W[1]*-hard for parameter* k *and* $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$.

As the proof of Theorem 3.3 ensures that no edge addition is used in the reduction, the W[1]-hardness results hold also for the weighted versions of the problems.

Corollary 3.6. *The problems* $\text{WDCE}^r(S)$*,* $\text{WDCE}^*(S)$*, and* $\text{WDCE}(S)$ *are* W[1]*-hard for parameter* k *and* $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$.

Let $\text{WDCE}^=(S)$ denote the variant of $\text{WDCE}^r(S)$ where we ask for a regular graph of unspecified regularity (i.e., when r is not given).

Corollary 3.7. *The problems* $\text{WDCE}^=_1(S)$ *and* $\text{WDCE}^=(S)$ *are* W[1]*-hard for parameter* k *and* $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$.

Proof. We reduce from $\text{WDCE}^r_1(S)$ and $\text{WDCE}^r(S)$, respectively. Given an instance (G, k) of one of these problems, we construct a new instance (G', k) by adding one r-regular connected component with more than k vertices. This can be done by taking, for example, k fixing gadgets and connecting them in a ring. We clearly cannot alter this component within the budget k, thus the only possible solution is the same as that for (G, k). Thus if (G', k) is a *yes*-instance of $\text{WDCE}^=_1(S)$ or $\text{WDCE}^=(S)$, then it is also a *yes*-instance of $\text{WDCE}^r_1(S)$ or $\text{WDCE}^r(S)$, respectively. The converse direction holds trivially. □

The next theorem covers the remaining W[1]-hard cases of the Classification Theorem.

Theorem 3.8. *The problem* $\mathrm{WDCE}_1(S)$ *is* W[1]-*hard for parameter* k *and* $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$.

Proof. Let (G, k) be an instance of REGULAR CLIQUE where G is r-regular and has n vertices. We construct a graph G' by adding to G additional vertices $v_1, \dots, v_k$ and all edges uv_i for $u \in V(G)$ and $1 \leq i \leq k$. We set $\delta(v_i) = \{n - k\}$, $1 \leq i \leq k$, and $\delta(u) = \{r - k + 1, r + k\}$ for all $u \in V(G)$. We set $k' = \binom{k}{2} + k^2$. Furthermore let $\overline{G}$ be the complement graph of G (i.e., $V(\overline{G}) = V(G)$ and $E(\overline{G}) = \{\, uv \mid u, v \in V(G),\, u \neq v, \text{ and } uv \notin E(G) \,\}$) and label the vertices v of $\overline{G}$ with $\overline{\delta}(v) = \{\, |V(G)| - 1 - d \mid d \in \delta(v) \,\}$.

The theorem follows from Lemma 3.1 and the following claim.

Claim 3.9. The following statements are equivalent.

1. (G, k) is a *yes*-instance of REGULAR CLIQUE.
2. (G', k') is a *yes*-instance of $\mathrm{WDCE}_1(\mathsf{e})$.
3. (G', k') is a *yes*-instance of $\mathrm{WDCE}_1(\mathsf{e}, \mathsf{a})$.
4. $(\overline{G}, k)$ is a *yes*-instance of $\mathrm{WDCE}_1(\mathsf{a})$.

$(1 \Rightarrow 2)$ Assume there is a k-clique in G. Let the set of vertices of the clique be $C = \{u_1, \dots, u_k\}$. Then we can satisfy the degree requirement of G' by deleting the edges u_iu_j, $1 \leq i, j \leq k$, and the edges v_iu_j, $1 \leq i, j \leq k$. The number of edges deleted is exactly k'.

$(2 \Rightarrow 3)$ This implication is trivial.

$(3 \Rightarrow 1)$ Assume that (G', k') is a *yes*-instance of $\mathrm{WDCE}_1(\mathsf{e}, \mathsf{a})$ and fix a solution of total cost at most k'. For $i \in \{r - k + 1, r + k\}$ let $V_i \subseteq V(G)$ be the set of vertices that have degree i after editing. Let D be the set of edges that are deleted in the solution and are incident with a vertex v_i for $1 \leq i \leq k$. Clearly $|D| \geq k^2$. Let $D_i \subseteq D$ denote the subset of deleted edges that have an end point in V_i, $i \in \{r - k + 1, r + k\}$. Each deleted edge $e \in D$ causes the degree of a vertex $v \in V(G)$ to be decreased by one, and therefore causes additional editing operations to repair the degree of v: either by edge additions (if $v \in V_{r+k}$) or by edge deletions (if $v \in V_{r-k+1}$). At least one edge addition is required to repair the degree change caused by 2 edges from D_{r+k}. At least $\binom{k}{2} = k(k-1)/2$ edge deletions are required to repair the degree change caused by k^2 edges from D_{r-k+1}; this is exactly the case if G contains a clique on k vertices $u_1, \dots, u_k$, and $D_{r+1} = \{\, u_iv_j \mid 1 \leq i, j \leq k \,\}$. Thus the average cost caused by an edge in D_{r+k} is at least $1/2$, whereas the average cost caused by an edge in D_{r-k+1} is at least $k(k-1)/(2k^2) = (k-1)/(2k)$, which is smaller than $1/2$. Consequently, the editing cost is smallest if $D_{r+k} = \emptyset$ and $D_{r-k+1} = \{\, u_iv_j \mid 1 \leq i, j \leq k \,\}$ for vertices $u_1, \dots, u_k$ that form a clique in G. However, in this case the total editing cost is exactly $|D_{r-k+1}| + |D_{r-k+1}|(k-1)(2k) = k^2 + \binom{k}{2} = k'$, hence this case is the only one possible.

$(2 \Leftrightarrow 4)$ This equivalence follows immediately from the definition of $\overline{G}$. □

Since the main reduction in the proof of Theorem 3.8 does not use edge addition it also applies to the weighted version of the problems:

Corollary 3.10. *The problem* $\mathrm{WDCE}(S)$ *is* W[1]-*hard for parameter* k *and* $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$.

4 Fixed-Parameter Tractability of WDCE for Parameter $k + r$

In this section we establish the fixed-parameter tractability part of the Classification Theorem:

Theorem 4.1. *The problems* $\mathrm{WDCE}(S)$ *and* $\mathrm{WDCE}_1(S)$ *are fixed-parameter tractable for parameter* $k + r$ *and* $\emptyset \neq S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$.

For establishing Theorem 4.1 we take a logical approach and apply the meta-theorem of Frick and Grohe [16]. They show that, given a sentence ϕ in first-order logic and a relational structure S, the problem of deciding whether S is a model of ϕ is fixed-parameter tractable, parameterized by the length of ϕ, when the structure has "effectively bounded local treewidth." More particularly we use their corollary that the parameterized model checking problem for first-order logic is fixed-parameter tractable for relational structures of *bounded degree*. Stewart [33] pointed out that this result also holds if the degree bound is not global but depends on the parameter. Furthermore he indicated how this can be used to show that r-REGULAR SUBGRAPH with parameter $k+r$ is fixed-parameter tractable. In the following we extend this approach to $\mathrm{WDCE}(S)$ and $\mathrm{WDCE}_1(S)$ for $S \subseteq \{\mathsf{v}, \mathsf{e}, \mathsf{a}\}$.

We must now ensure that the graph is of bounded weighted degree, which is not guaranteed by the problem. However if the graph has a vertex v such that $d^\rho(v) > k + r$ then, if there is a solution of cost at most k, v must be part of the solution, i.e., v must be deleted (this operation will also be part of the kernelization approach in Section 6). Thus we can eliminate all such vertices in polynomial time and decrease the parameter k accordingly. Further, we can assume that all vertices are of weight at most $k+1$ and all edges are of weight at most $r+k+1$ (this or any higher weight renders deletion infeasible). Hence we can assume that the weighted degree and all edge and vertex weights are bounded in terms of the parameters k and r.

We associate with a weighted graph G its *incidence structure* S_G. The universe of S_G consists of the vertices and edges of G. S_G has two unary relations V and E (expressing the property of being a vertex or edge, respectively), a binary relation I (where Ixy expresses that vertex x is incident to edge y), and unary relations W_i, $1 \leq i \leq k + 1$, and D_j, $0 \leq j \leq r$ (where $W_i x$ expresses that vertex x has weight $\rho(x) = i$, and $D_j(x)$ expresses that $j \in \delta(x)$). We represent an edge uv of weight w by w distinct "parallel" elements $x_1, \ldots, x_w$ with Iux_i and Ivx_i, $1 \leq i \leq w$. Note that the maximum degree of the structure S_G (in the sense of [16]) is bounded in terms of k and r.

Thus, for showing Theorem 4.1 it suffices to formulate first-order formulas ϕ_k, $k \geq 0$, such that S_G is a model of ϕ_k if and only if (G, k, r) is a *yes*-instance of the considered editing problem. In the following we write $[n] = \{1, \ldots, n\}$. We define

$$\phi_k = \bigvee_{k',k'',k''' \in [k] \text{ such that } k'+k''+k''' \leq k} \exists u_1, \ldots, u_{k'}, e_1, \ldots, e_{k''}, a_1, \ldots, a_{k'''}, b_1, \ldots, b_{k'''} \; (\phi'_k \wedge \forall v \; \phi''_k)$$

where ϕ'_k and ϕ''_k are given below. The subformula ϕ'_k is the conjunction of the clauses (1)–(3) and ensures that $u_1, \ldots, u_{k'}$ represent deleted vertices, $e_1, \ldots e_{k''}$ represent deleted edges, a_i, b_i, $1 \leq i \leq k'''$, represent end points of added edges, and the total editing cost is at most k. Note that since added edges are not present in the given structure we need to express them in terms vertex pairs. For the unweighted case we must also include subformulas (4) and (5) to ensure that the addition of edges does not produce parallel edges. By restricting k', k'' or k''' to zero as appropriate we can express which editing operations are available.

(1) $\bigwedge_{i\in[k']} Vu_i \wedge \bigwedge_{i\in[k'']} Ee_i$ "u_i is a vertex, e_i is an edge;"

(2) $\bigwedge_{i\in[k''']} Va_i \wedge Vb_i \wedge a_i \neq b_i \wedge \bigwedge_{j\in[k']}(u_j \neq a_i \wedge u_j \neq b_i)$ "a_i and b_i are distinct vertices and not deleted;"

(3) $\bigvee_{w_1,\ldots,w_{k'}\in[k'] \text{ such that } \sum_{i\in[k']} w_i+k''+k'''\leq k} \bigwedge_{i\in[k']} W_{w_i}u_i$ "the weight of deleted vertices is correct;"

(4) $\bigwedge_{1\leq i<j\leq k''}(a_i \neq b_j \vee a_j \neq b_i) \wedge (a_i \neq a_j \vee b_i \neq b_j)$ "the pairs of vertices are mutually distinct;"

(5) $\bigwedge_{i\in[k''']} \forall y(\neg Ia_iy \vee \neg Ib_iy)$ "a_i and b_i are not adjacent."

The subformula ϕ''_k ensures that after editing each vertex v has degree $l \in \delta(v)$.

$$\phi''_k = (Vv \wedge \bigwedge_{i\in[k']} v \neq u_i) \rightarrow \bigvee_{l\in[r]} D_lv \wedge \bigvee_{\substack{l',l''\in[l]\\ l'+l''=l}} \exists x_1,\ldots,x_{l'},y_1,\ldots,y_{l''}\ \phi'''_k,$$

where ϕ'''_k is the conjunction of the clauses (6)–(12).

(6) $\bigwedge_{i\in[l']} Ivx_i$ "v is incident with l' edges;"

(7) $\bigwedge_{1\leq i<j\leq l'} x_i \neq x_j$ "the edges are all different;"

(8) $\bigwedge_{i\in[l'],j\in[k'']} x_i \neq e_j$ "the edges have not been deleted;"

(9) $\bigwedge_{i\in[l'],j\in[k']} \neg Iu_jx_i$ "the ends of the edges have not been deleted;"

(10) $\forall x(Ivx \rightarrow \bigvee_{i\in[l']} x = x_i \vee \bigvee_{i\in[k'']} x = e_i \vee \bigvee_i Ixu_i)$ "v is not incident with any further edges except deleted edges;"

(11) $\bigwedge_{i\in[l'']}\bigvee_j(y_i = a_j \wedge v = b_j) \vee (y_i = b_j \wedge v = a_j)$ "v is incident with at least l'' added edges;"

(12) $\bigwedge_{j\in[l'']}(v = a_j \rightarrow \bigvee_i y_i = b_j) \wedge (v = b_j \rightarrow \bigvee_{j\in[l'']} y_i = a_j)$ "v is incident with at most l'' added edges."

The combination of the W[1]-hardness results of Section 3 and the fixed-parameter tractability results of this section establishes the Classification Theorem.

5 Polynomial Cases

In this section we show that the problems $\mathrm{WDCE}^*(S)$ and $\mathrm{WDCE}^*_1(S)$ can be solved in polynomial time for $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$. If $S = \{\mathsf{e}\}$ then it is not difficult to apply standard methods of matching theory [22]. The case $S = \{\mathsf{a}\}$ can be reduced to the case $S = \{\mathsf{e}\}$ by means of the complement graph construction as in the proof of Theorem 3.8. However it is not immediately apparent that matching techniques can be directly applied to the case where we allow both edge addition and edge deletion. Hence we give a general construction for solving all variants of the problem with $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$.

Theorem 5.1. *The problems* $\mathrm{WDCE}^*(S)$ *and* $\mathrm{WDCE}^*_1(S)$ *can be solved in polynomial time for* $\emptyset \neq S \subseteq \{\mathsf{e}, \mathsf{a}\}$.

Proof. First we consider the problem $\mathrm{WDCE}^*(\mathsf{a}, \mathsf{e})$; we will explain later how the approach can be modified for the other versions of the problem. To simplify notation we write $\delta(v) = d$ instead of $\delta(v) = \{d\}$.

Let (G, k) be an instance of $\mathrm{WDCE}^*(\mathsf{e}, \mathsf{a})$ with $G = (V, E)$. For the scope of this proof it is convenient to allow edges of weight 0, so we may assume that G is a complete graph. Solving the problem is clearly equivalent to finding an edge weight function $\rho' : E(G) \to \{0, 1, 2, \dots\}$ of G such that for each $v \in V(G)$ we have $\sum_{vv' \in E(G)} \rho'(vv') = \delta(v)$ and the cost of ρ', $\sum_{vv' \in E(G)} |\rho(vv') - \rho'(vv')|$, is at most k.

We construct a graph H with edge-weight function η as follows: For each vertex v of G we introduce in H a set $V(v)$ of $\delta(v)$ vertices. For each edge $vv' \in E(G)$ we add the following vertices and edges to H.

1. We add two sets $V_{\text{del}}(v, v')$ and $V_{\text{del}}(v', v)$ of vertices, each of size $\rho(vv')$.
2. We add two sets $V_{\text{add}}(v, v')$ and $V_{\text{add}}(v', v)$ of vertices, each of size $\min(\delta(v), \delta(v'))$.
3. We add all edges uw for $u \in V(v)$ and $w \in V_{\text{del}}(v, v') \cup V_{\text{add}}(v, v')$, and all edges uw for $u \in V(v')$ and $w \in V_{\text{del}}(v', v) \cup V_{\text{add}}(v', v)$.
4. We add edges that form a matching $M_{vv'}$ between the sets $V_{\text{del}}(v, v')$ and $V_{\text{del}}(v', v)$. We will refer to these edges as *deletion edges*.
5. We add edges that form a matching $M'_{vv'}$ between the sets $V_{\text{add}}(v, v')$ and $V_{\text{add}}(v', v)$ and subdivide the edges of $M'_{vv'}$ twice; that is, we replace $xy \in M'_{vv'}$ by a path x, x_y, y_x, y where x_y and y_x are new vertices. We will refer to the edges of the form $x_y y_x$ as *addition edges*.

For all addition and deletion edges e in $E(H)$ we set $\eta(e) = 1$. All other edges e' have $\eta(e') = 0$.

Claim 5.2. G has a solution ρ' of cost at most k if and only if H has a perfect matching of weight at most k.

($\Rightarrow$) Let ρ' be a solution of cost $k' \leq k$. We define a perfect matching M of H as follows. Consider an edge $e = vv' \in E(G)$.

First we consider the case $\rho(e) \leq \rho'(e)$. Let $d = \rho'(e) - \rho(e)$. We match the vertices in $V_{\text{del}}(v, v')$ to $\rho(e)$ vertices in $V(v)$, and similarly, we match the vertices in $V_{\text{del}}(v', v)$ to $\rho(e)$ vertices in $V(v')$. Now we choose d vertices $x_1, \ldots, x_d$ in $V_{\text{add}}(v, v')$; let $y_1, \ldots, y_d$ be the corresponding vertices in $V_{\text{add}}(v', v)$. We match the vertices x_i to vertices in $V(v)$, the vertices y_i to vertices in $V(v')$, and the vertices $(x_i)_{(y_i)}$ and $(y_i)_{(x_i)}$ to each other, $1 \leq i \leq d$. We match the remaining vertices $x \in V_{\text{add}}(v, v') \setminus \{x_1, \ldots, x_d\}$ to x_y and the remaining vertices $y \in V_{\text{add}}(v', v) \setminus \{y_1, \ldots, y_d\}$ to y_x. Observe that the defined matching has weight d (all edges used for the matching have weight 0 except for d addition edges).

Second we consider the case $\rho'(e) \leq \rho(e)$. Let $d = \rho(e) - \rho'(e)$. We choose d vertices $x_1, \ldots, x_d$ from $V_{\text{del}}(v, v')$ and match them to the corresponding vertices $y_1, \ldots, y_d$ in $V(v', v)$. We match the remaining vertices in $V_{\text{del}}(v, v') \setminus \{x_1, \ldots, x_d\}$ to vertices in $V(v)$, and the remaining vertices in $V_{\text{del}}(v', v) \setminus \{y_1, \ldots, y_d\}$ to vertices in $V(v')$. We match all vertices $x \in V_{\text{add}}(v, v')$ to the corresponding vertices x_y, and symmetrically, all vertices $y \in V_{\text{add}}(v', v)$ to the corresponding vertices y_x. Observe that the defined matching has weight d as the only edges of weight 1 are d deletion edges.

In both cases we have defined a matching that covers exactly $\rho'(e)$ vertices of $V(v)$ and $\rho'(e)$ vertices of $V(v')$. By assumption $\sum_{vv' \in E(G)} \rho'(vv') = \delta(v)$, hence we can proceed in this way for all edges $vv' \in E(G)$, and we end up with a perfect matching M of H of weight $k' \leq k$.

($\Leftarrow$) Conversely, let M be a perfect matching of H of minimum weight $k' \leq k$.

First, we make the observation that for the sets of vertices corresponding to an edge vv', the matching M must be symmetric. That is, if a vertex x from $V_{\text{del}}(v, v')$ is matched to a vertex in $V(v)$, then the adjacent vertex y in $V_{\text{del}}(v', v)$ must be matched to a vertex in $V(v')$, as there is no other possibility if M is a perfect matching. Similarly if a vertex x in $V_{\text{add}}(v, v')$ is matched to a vertex in $V(v)$, then its corresponding vertex y in $V_{\text{add}}(v', v)$ must be matched to a vertex in $V(v')$, and x_y must be matched to y_x, as again there is no other possibility if M is a perfect matching. Thus a perfect matching of H must correspond to some solution for G.

Second, as M is of minimum weight, there can be no edge vv' in G where the matching in H has vertices from $V_{\text{add}}(v, v')$ that are matched to vertices in $V(v)$ and vertices in $V_{\text{del}}(v, v')$ matched to vertices in $V_{\text{del}}(v, v')$. Assume for contradiction that $a \in V_{\text{add}}(v, v')$ is matched to $x \in V(v)$ and $d \in V_{\text{del}}(v, v')$ is matched to $c \in V_{\text{del}}(v', v)$. By symmetry we must have $b \in V_{\text{add}}(v', v)$ matched to $y \in V(v')$, and the vertices a_b and b_a must also be matched to each other. Then we could take the following alternate matching M' where we match d to x, c to y, a to a_b and b to b_a. Then M' is still a perfect matching, but of weight two less than M.

Then we can construct a solution ρ' of weight k. For each addition edge used in M, we increase the weight of the edge between the corresponding vertices in G by one. For each deletion edge used in M, we reduce the weight of the corresponding edge in G. As the vertices in $V(v)$ are matched, we know that $\sum_{vv' \in E(G)} \rho'(vv') = \delta(v)$.

Therefore a minimum weight perfect matching for H corresponds to a minimum weight solution for G. This completes the proof of Claim 5.2.

The graph H can be obtained from G in polynomial time, and we can find a minimum weight perfect matching for H in polynomial time by application of Edmond's minimum weight perfect matching algorithm [13, 22]. Consequently WDCE$^*(\mathsf{a}, \mathsf{e})$ can be solved in polynomial time.

For the unweighted case WDCE$_1^*(\mathsf{e}, \mathsf{a})$ we need to modify the above construction of H to ensure that $\alpha'(e) \leq 1$ holds for all $e \in E(G)$. This, however, can be accomplished by using sets $V_{\text{add}}(v, v')$, $vv' \in E(G)$, of size at most 1 only: $|V_{\text{add}}(v, v')| = 1$ if and only if $\rho(vv') = 0$. Furthermore, the above construction can be modified to deal with the (simpler) case where only edge deletion or edge addition is available: For $S = \{\mathsf{e}\}$ we remove all addition edges from H; for $S = \{\mathsf{a}\}$ we remove all deletion edges from H. It is easy to check that Claim 5.2 also holds for these variants of WDCE$^*(\mathsf{a}, \mathsf{e})$ with H modified as described. □

6 Kernelization

In Section 4 we demonstrated the fixed-parameter tractability of WDCE for parameter $k + r$ using a logic-based approach which, although constructive, does not provide practically feasible algorithms.

In this section we give efficient and practically feasible fixed-parameter algorithms for several of the considered problems by means of kernelization. For WDCE$^*(\mathsf{v})$ and WDCE$^*(\mathsf{v}, \mathsf{e})$ in particular we obtain polynomially sized kernels. At the end of this section we provide theoretical evidence that it is unlikely that our kernelization approach extends to all the remaining fixed-parameter tractable cases covered by Theorem 1.1.

We consider *reduction rules* that replace an instance (G, k, r) of WDCE(S) with a smaller instance (G', k', r'). A reduction rule is *sound* for an editing problem if it replaces a *yes*-instance with a *yes*-instance and a *no*-instance with a *no*-instance. An instance (G, k, r) is *reduced* under a reduction rule if the rule is not applicable to (G, k, r). For all reduction rules considered it is obvious that they can be applied in polynomial time and one can check in polynomial time if they are applicable.

Reduction Rule 1. Let (G, k, r) be an instance of WDCE(S). If there is a vertex v in G such that $d^\rho(v) > k + r$, then replace (G, k, r) with (G', k', r) where $G' = G - v$ and $k' = k - \rho(v)$.

Lemma 6.1. *Reduction Rule 1 is sound for* WDCE(S) *with* $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}\}$.

Proof. Assume there is such a vertex v. At least $k + 1$ vertices or edges must be deleted if we do not delete v, but we may only perform at most k deletions. Thus (G, k, r) is a *yes*-instance of WDCE(S) if and only if (G', k', r) is a *yes*-instance of WDCE(S). □

Consider an instance (G, k, r) of WDCE(S). Extending a notion of Moser and Thilikos [25], we define a *clean region* C of G as a maximal connected subgraph of G where each vertex $v \in V(C)$ has weighted degree $d^\rho_G(v) \in \delta(v)$. The *boundary* $B(C)$ of a clean region C is the set of vertices of G that are adjacent to some vertex in the clean region but do not belong to the clean region.

Let C be a clean region of G with boundary $B(C)$. Let the i-th layer of C be the subset $C_i = \{\, c \in V(C) \mid \min_{b \in B} d_G(c,b) = i \,\}$ where $d_G(c,b)$ denotes is the distance between c and b in G. Note that all the neighbors of a vertex of layer C_i belong to $C_{i-1} \cup C_i \cup C_{i+1}$.

Reduction Rule 2. Let (G,k,r) be an instance of WDCE(S) and let C be a clean region of G such that $C_{k+2} \neq \emptyset$. Then replace (G,k,r) with (G',k,r) as follows:

1. For each vertex $u \in C_{k+1}$ reduce each entry of $\delta(u)$ by $d^{\rho}_{C_{k+2}}(u)$.
2. Delete all layers C_i for $i \geq k+2$.

Lemma 6.2. *Reduction Rule 2 is sound for* WDCE(S) *with* $\emptyset \neq S \subseteq \{\mathsf{v}, \mathsf{e}\}$.

Proof. Let D be the set of vertices and edges deleted in a minimal solution of (G,k,r); thus $D \subseteq V(G)$ if $S = \{\mathsf{v}\}$ and $D \subseteq V(G) \cup E(G)$ if $S = \{\mathsf{v}, \mathsf{e}\}$. Let $G(D)$ be the subgraph of G induced by all vertices that belong to D or are incident to an edge in D. Each connected component X of $G(D)$ that contains a vertex or edge of a clean region C must also contain a vertex of the boundary $B(C)$ of C, since otherwise we could obtain a solution that is smaller than D by removing those vertices and edges from D that induce X in $G(D)$. Consequently, each vertex $v \in D \cap V(C)$ must be of distance at most $|D|$ from a vertex in the boundary of C, i.e., $v \in C_i$ for $i \leq |D|$. Similarly, each endpoint of an edge $e \in D \cap E(C)$ must belong to some C_i for $i \leq |D|+1$. If $|D| \leq k$ then D is also a solution of (G',k,r) as D does not touch the part deleted from G. On the other hand, each solution D' of (G',k,r) is trivially a solution of (G,k,r). □

Reduction Rule 3. Let (G,k,r) be an instance of WDCE(S), let C be a clean region of G with empty boundary $B(C) = \emptyset$, and let $G' = G - V(C)$. Then replace (G,k,r) with (G',k,r).

Lemma 6.3. *Reduction Rule 3 is sound for* WDCE(S) *with* $\emptyset \neq S \subseteq \{\mathsf{v}, \mathsf{e}\}$.

Proof. As $B(C) = \emptyset$, C is not connected to the rest of the graph and therefore does not affect the solution. □

Next we show our first kernelization lemma.

Lemma 6.4. *Let* $\emptyset \neq S \subseteq \{\mathsf{v}, \mathsf{e}\}$. *Every* yes-*instance* (G,k,r) *of* WDCE(S) *that is reduced under Reduction Rules 1-3 has at most* $k(1 + (k+r)(1 + r^{k+1})) = O(k^2 r^{k+1} + k r^{k+2})$ *vertices.*

Proof. Assume that (G,k,r) is a *yes*-instance of WDCE(S) and is reduced under Reduction Rules 1–3. We define three disjoint sets D, H, and X where $D \subseteq E(G) \cup V(G)$ is the set of vertices and edges whose deletion provides a solution, $H \subseteq V(G)$ is the set of vertices adjacent to vertices in D or incident to edges in D, and $X \subseteq V(G)$ contains the remaining vertices of G that are neither in D nor in H. $|D| \leq k$ by definition, and $D \subseteq V(G)$ if $S = \{\mathsf{v}\}$. Observe that H separates D from X. Furthermore, observe that there are no independent clean regions in G (otherwise the graph would not be reduced under Reduction Rule 3), and $d^{\rho}(x) \in \delta(x)$ for all $x \in X$ (otherwise D would not be a solution).

Claim 6.5. $|H| \leq |D| \cdot (k+r)$.

Since the instance is reduced under Reduction Rule 1, the maximum weighted degree of every vertex is at most $k + r$, and each vertex in H is adjacent or incident to some element of D; hence the claim follows.

Claim 6.6. $|X| \leq |H| \cdot r^{k+1}$.

As all vertices in X are clean, all boundary vertices for all clean regions are contained in $H \cup D$. Then there is no vertex in X of distance greater than $k + 1$ from H, otherwise the graph is not reduced under Reduction Rule 2. Thus X is the disjoint union of the sets $X_1, \ldots, X_{k+1}$ where $X_i = \{ x \in X \mid \min_{h \in H} d_G(x, h) = i \}$. We have $|X_1| \leq |H| \cdot r$ as vertices in H may have all their neighbors in X. For $i > 1$ we have $|X_i| \leq |X_{i-1}|(r-1) \leq |X_1|(r-1)^{i-1}$ as each vertex in X_i has at least one neighbor in X_{i-1} and at most r in neighbors in total. For $r \geq 1$ we assume inductively that $\sum_{i=1}^{m} |X_i| \leq |X_1| \cdot r^{m-1}$. Then $\sum_{i=1}^{m+1} |X_i| \leq |X_1| \cdot r^{m-1} + |X_1| \cdot (r-1)^{m-1}(r-1) \leq |X_1| \cdot r^m$. Therefore with $k + 1$ layers, $|X| \leq |H| \cdot r^{k+1}$. For $r = 0$, H is an independent set after removal of D, therefore $X = \emptyset$, and the claim holds trivially.

Since $|V(G)| \leq |D| + |H| + |X|$ and $|D| \leq k$, the lemma follows from Claims 6.5 and 6.6. □

If we restrict the allowed editing operations to edge deletion alone, then we can obtain a better kernel by observing that if the graph contains any vertex of degree greater than $k + r$, then it is a *no*-instance, as this vertex cannot be fixed with only k edge deletions. Reduction Rules 2 and 3 still apply. Thus WDCE(e) allows an improved kernelization lemma.

Lemma 6.7. *Every* yes-*instance* (G, k, r) *of* WDCE(e) *that is reduced under Reduction Rules 2 and 3 has at most* $2k(1 + r^{k+1}) = O(kr^{k+1})$ *vertices.*

Proof. We define D, H, and X as before. Since D consists only of edges and is of size at most k, we have $|H| \leq 2k$. The rest of the proof follows as for Lemma 6.4. □

Next we consider the case where the degree lists are singletons, i.e., WDCE$^*(S)$ for $S \subseteq \{\mathsf{v}, \mathsf{e}\}$. In this case we can obtain kernels of significantly smaller size.

Reduction Rule 4. Let (G, k, r) be an instance of WDCE$^*(S)$. Let C be a clean region of G containing more than one vertex. Then we replace (G, k, r) with (G', k, r) where G' is obtained from G by contracting C to a single vertex v as follows:

1. We add a new vertex v and delete all vertices that belong to C.
2. For each $b \in B(C)$ we add the edge bv to G' of weight $\rho(bv) = \sum_{bc \in E(G), c \in V(C)} \rho(bc)$.
3. We set $\rho(v) = \min(k + 1, \sum_{u \in V(C)} \rho(u))$ and $\delta(v) = d^\rho(v)$ (i.e., v is clean in G').

Claim 6.8. Reduction Rule 4 is sound for WDCE$^*(S)$ with $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}\}$.

Proof. Let (G, k, r) be an instance of WDCE$^*(S)$ and let C be a clean region of G with boundary B. Let (G', k, r) be the new instance obtained by Reduction Rule 4, contracting C to a new vertex v.

If we delete a vertex $u \in V(C) \cup B(C)$ or an edge incident with a vertex $u \in V(C)$, then any neighbor $u' \in V(C)$ of u will no longer be clean, and subsequently must also be deleted (edge addition is not available to make u' clean again). Clearly this cascades until the entire clean region is removed. Thus any solution for (G, k, r) either deletes all or none of the vertices in $V(C)$. Hence we can represent $V(C)$ by a single vertex v as described in Reduction Rule 4. We can limit the weight of v to $k + 1$ as any weight larger than k prevents deletion. □

Next we state an improved kernelization lemma (we exclude the case WDCE$^*(\mathsf{e})$ as it is solvable in polynomial time and thus admits a kernel of constant size, see Section 5).

Lemma 6.9. *Let $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}\}$. Every* yes-*instance (G, k, r) of* WDCE$^*(S)$ *that is reduced under Reduction Rules 1, 3, and 4 has at most $k(1 + (k + r)(1 + r)) = O(kr(k + r))$ vertices.*

Proof. Assume (G, k, r) is a *yes*-instance of WDCE$^*(S)$ and is reduced under Reduction Rules 1, 3, and 4.

We define sets D, H, and X as in the proof of Lemma 6.4. Clearly Claims 6.5 and 6.6 hold as before.

Every vertex in X belongs to a clean region. However, since we assume that the instance is reduced under Reduction Rule 4, every clean region consists of a single vertex. Therefore each vertex in H can be adjacent to at most r vertices outside H and D, which themselves are adjacent to no vertices outside H. Therefore $|X| \leq |H| \cdot r \leq kr(k + r)$. The Combination of this bound with $|D| \leq k$ and the bounds of Claims 6.5 and 6.6 completes the proof of the lemma. □

We summarize the results of Lemmas 6.4, 6.7, and 6.9:

Theorem 6.10. *For $\{\mathsf{v}\} \subseteq S \subseteq \{\mathsf{v}, \mathsf{e}\}$, the problem* WDCE$(S)$ *admits a kernel with $O(k^2 r^{k+1} + kr^{k+2})$ vertices, and the problem* WDCE$^*(S)$ *admits a kernel with $O(kr(k + r))$ vertices. The problem* WDCE(e) *admits a kernel with $O(kr^{k+1})$ vertices.*

It appears that our kernelization approach does not work under the presence of edge addition. If we apply our approach to a variant of WDCE that includes WDCE$_1^*(\mathsf{v}, \mathsf{a})$ as a subproblem, it becomes necessary to consider deleting a set of vertices from clean regions so that the edges that become available may be used to complete the degree of a vertex of insufficient degree. This gives rise to the following subproblem:

> EDGE REPLACEMENT SET
>
> *Instance:* A graph $G = (V, E)$, two positive integers k and t.
>
> *Question:* Does there exist a set $X \subseteq V$ such that $|X| \leq k$ and there are exactly t edges between vertices in X and vertices in $V \setminus X$?

Unfortunately, EDGE REPLACEMENT SET is NP-complete, thus making the possibility of obtaining a kernel in polynomial time by somehow identifying all relevant sets in the clean regions unlikely. Consider an instance of WDCE$^*(\mathsf{v}, \mathsf{a})$ where we have only one vertex with weight $k(r - k + 2) + 1$ which requires $k(r - k + 1)$ edges and are allowed $k(r - k + 2)$ editing steps, in this case the only possibility is to delete a clique of size k and add the $k(r-k+1)$ edges that are now available. Note that if the graph is unweighted, it is not clear whether such a pathological case exists.

Proposition 6.11. EDGE REPLACEMENT SET *is* NP-*hard and* W[1]-*hard for parameter* k.

Proof. We give a polynomial-time FPT reduction from REGULAR CLIQUE (see Lemma 3.1).

Let (G,k) be an instance of REGULAR CLIQUE where $G=(V,E)$ is r-regular. We may assume that $r>k^2$ since we can use the fixing gadgets (see Section 2.3) to increase the degree of each vertex arbitrarily without introducing non-trivial cliques. For a set $X\subseteq V$ let $d(X)$ denote the number of edges $uv\in E$ with $u\in X$ and $v\in V\setminus X$. If X forms a k-clique in G then $d(X)=k(r-k+1)$. Therefore we put $t=k(r-k+1)$ and consider (G,k,t) as an instance of EDGE REPLACEMENT SET.

Let $X\subseteq V$ with $|X|\leq k$ and $d(X)=t$. We show that X has exactly k elements and forms a clique in G. Assume for the sake of contradiction that $|X|<k$. It follows that $d(X)\leq |X|r\leq r(k-1)<rk-k^2\leq t$, a contradiction; hence $|X|=k$. Each vertex $x\in X$ has at most $k-1$ neighbors in X and at least $r-k+1$ neighbors in $V\setminus X$. Therefore, if at least one $x\in X$ had fewer than $k-1$ neighbors in X, then $d(X)>k(r-k+1)=t$, again a contradiction. Hence X indeed induces a k-clique in G. □

7 Conclusion

We have classified the parameterized complexity of a wide range of graph editing problems; our general finding is that all the problems are fixed-parameter tractable if parameterized by the combination of the total editing cost and the maximum (weighted) degree of the solution graph, but are unlikely to be fixed-parameter tractable if parameterized by the total editing cost alone. For the fixed-parameter tractable problems that exclude the operation of edge addition we could further provide kernelizations: We obtain kernels of polynomial size if the degree lists are singletons, and kernels of singly exponential size if the degree lists are arbitrary.

In our investigations we have only distinguished between singletons and arbitrary degree lists. A further possible line of future research is to make a more refined distinction, aiming at results of similar flavor as Cornuéjols's theorem [9] (see Section 1). For example, one can try to obtain polynomially-sized kernels for the case where all degree lists are intervals and the operations of vertex deletion and edge deletion are both included. However, the most challenging question for further research appears to be whether our kernelization results can be extended to editing problems that include the operations of vertex deletion and edge addition.

References

[1] F. N. Abu-Khzam and H. Fernau. Kernels: Annotated, proper and induced. In *The Second International Workshop on Parameterized and Exact Computation 2006 (IWPEC'06)*, Lecture Notes in Computer Science, pages 264–275. Springer, 2006.

[2] R. P. Anstee. An algorithmic proof of Tutte's f-factor theorem. *J. Algorithms*, 6(1):112–131, 1985.

[3] H. Bodlaender, R. Tan, and J. van Leeuwen. Finding a $\triangle$-regular supergraph of minimum order. *Discrete Appl. Math.*, 131(1):3–9, 2003.

[4] K. Cameron. Induced matchings. *Discrete Appl. Math.*, 24(1–3):97–102, 1989.

[5] D. M. Cardoso, M. Kamiński, and V. Lozin. Maximum k-regular induced subgraphs. *J. Comb. Optim.*, 14(4):455–463, 2007.

[6] F. Cheah and D. G. Corneil. The complexity of regular subgraph recognition. *Discrete Appl. Math.*, 27(1–2):59–68, 1990.

[7] J. Chen, B. Chor, M. Fellows, X. Huang, D.W. Juedes, I.A. Kanj, and G. Xia, lower bounds for certain parameterized NP-hard problems, *Information and Computation*, 201:216–231, 2005.

[8] V. Chvátal, H. Fleischner, J. Sheehan, and C. Thomassen. Three-regular subgraphs of four regular graphs. *J. Graph Theory*, 3(4):371–386, 1979.

[9] G. Cornuéjols. General factors of graphs. *J. Combin. Theory Ser. B*, 45(2):185–198, 1988.

[10] R. Downey, M. R. Fellows, and U. Stege. Parameterized complexity: A framework for systematically confronting computational intractability. In *Contemporary Trends in Discrete Mathematics: From DIMACS and DIMATIA to the Future*, volume 49 of *AMS-DIMACS*, pages 49–99. American Mathematical Society, 1999.

[11] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999.

[12] J. Duato, S. Yalamanchili, and L. Ni. *Interconnection Networks*, Computer Architecture and Design, Morgan Kaufmann, 2002.

[13] J. Edmonds. Paths trees and flowers. *Canad. J. Math.*, 17:449–467, 1965.

[14] M. R. Fellows, D. Hermelin, F. A. Rosamond, and S. Vialette. On the parameterized complexity of multiple-interval graph problems. *Theoret. Comput. Sci.*, 410(1):53–61, 2009.

[15] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[16] M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *J. ACM*, 48(6):1184–1206, 2001.

[17] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[18] B. Korte and J. Vygen. *Combinatorial Optimization*, volume 21 of *Algorithms and Combinatorics*. Springer, Berlin, fourth edition, 2008.

[19] L. Lovász. The factorization of graphs. In *Combinatorial Structures and their Applications (Proc. Calgary Internat. Conf., Calgary, Alta., 1969)*, pages 243–246. Gordon and Breach, New York, 1970.

[20] L. Lovász. The factorization of graphs. II. *Acta Math. Acad. Sci. Hungar.*, 23:223–246, 1972.

[21] L. Lovász. Antifactors of graphs. *Periodica Mathematica Hungarica*, 4(2):121–123, 1973.

[22] L. Lovász and M. D. Plummer. *Matching Theory (North-Holland mathematics studies)*. Elsevier Science Publishers, North-Holland, 1986.

[23] L. Mathieson and S. Szeider, The parameterized complexity of regular subgraph problems and generalizations, in: *Computing: The Australasian Theory Symposium (CATS'08)*, volume 77 of *CRPIT*, *ACS*, 79–86, 2008.

[24] L. Mathieson and S. Szeider, *Combinatorial Optimization and Applications (COCOA'08)*, volume 5165 of Lecture Notes in Computer Science, Springer, 13–22, 2008.

[25] H. Moser and D. Thilikos, Parameterized complexity of finding regular induced subgraphs, *J. Discrete Algorithms*, 7:181–190, 2009.

[26] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[27] K. Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *J. Comput. System Sci.*, 67(4):757–771, 2003.

[28] J. Plesník. A note on the complexity of finding regular subgraphs. *Discrete Math.*, 49(2):161–167, 1984.

[29] A. Sebő. General antifactors of graphs. *J. Combin. Theory Ser. B*, 58(2):174–184, 1993.

[30] I. A. Stewart. Deciding whether a planar graph has a cubic subgraph is NP-complete. *Discrete Math.*, 126(1–3):349–357, 1994.

[31] I. A. Stewart. Finding regular subgraphs in both arbitrary and planar graphs. *Discrete Appl. Math.*, 68(3):223–235, 1996.

[32] I. A. Stewart. On locating cubic subgraphs in bounded-degree connected bipartite graphs. *Discrete Math.*, 163(1–3):319–324, 1997.

[33] I. A. Stewart. On the fixed-parameter tractability of parameterized model-checking problems. *Information Processing Letters*, 106:33–36, 2008.

[34] L. J. Stockmeyer and V. V. Vazirani. NP-completeness of some generalizations of the maximum matching problem. *Information Processing Letters*, 15(1):14–19, 1982.

[35] W. T. Tutte. A short proof of the factor theorem for finite graphs. *Canad. J. Math.*, 6:347–352, 1954.

[36] W. T. Tutte. Spanning subgraphs with specified valencies. *Discrete Math.*, 9(1):97–108, 1974.

[37] W. T. Tutte. Graph factors. *Combinatorica*, 1(1):79–97, 1981.