Counting Homomorphisms to Square-Free Graphs, Modulo 2

Andreas Göbel, Leslie Ann Goldberg, and David Richerby(✉)

Department of Computer Science, University of Oxford, Oxford, UK
davidr@cs.ox.ac.uk

Abstract. We study the problem $\oplus$HomsToH of counting, modulo 2, the homomorphisms from an input graph to a fixed undirected graph H. A characteristic feature of modular counting is that cancellations make wider classes of instances tractable than is the case for exact (non-modular) counting, so subtle dichotomy theorems can arise. We show the following dichotomy: for any H that contains no 4-cycles, $\oplus$HomsToH is either in polynomial time or is $\oplus$P-complete. This partially confirms a conjecture of Faben and Jerrum that was previously only known to hold for trees and for a restricted class of tree-width-2 graphs called cactus graphs. We confirm the conjecture for a rich class of graphs including graphs of unbounded tree-width. In particular, we focus on square-free graphs, which are graphs without 4-cycles. These graphs arise frequently in combinatorics, for example in connection with the strong perfect graph theorem and in certain graph algorithms. Previous dichotomy theorems required the graph to be tree-like so that tree-like decompositions could be exploited in the proof. We prove the conjecture for a much richer class of graphs by adopting a much more general approach.

1 Introduction

A homomorphism from a graph G to a graph H is a function from $V(G)$ to $V(H)$ that preserves edges, in the sense of mapping every edge of G to an edge of H; non-edges of G may be mapped to edges or non-edges of H. Many structures arising in graph theory can be represented naturally as homomorphisms. For example, the proper q-colourings of a graph G correspond to the homomorphisms from G to a q-clique. For this reason, homomorphisms from G to a graph H are often called "H-colourings" of G. Independent sets of G correspond to the homomorphisms from G to the connected graph with two vertices and one self-loop (vertices of G which are mapped to the self-loop are out of the

Theorem numbering in this extended abstract matches matches the full version: ArXiv, CoRR, abs/1501.07539. The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013) ERC grant agreement no. 334828. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.

M.M. Halldórsson et al. (Eds.): ICALP 2015, Part I, LNCS 9134, pp. 642–653, 2015.
DOI: 10.1007/978-3-662-47672-7_52

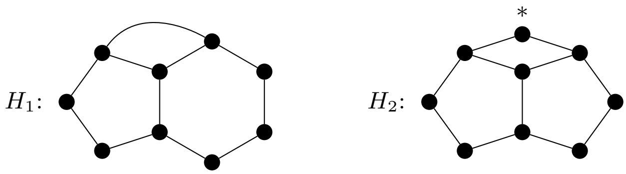

Fig. 1. Theorem 1.2 shows that $\oplus$HomsToH_1 is $\oplus$P-complete, whereas $\oplus$HomsToH_2 is in P. The role of the starred vertex is explained later in this section.

corresponding independent set; vertices which are mapped to the other vertex are in it). Homomorphism problems can also be seen as constraint satisfaction problems (CSPs) in which the constraint language consists of a single symmetric binary relation. Partition functions in statistical physics such as the Ising, Potts and hard-core models arise naturally as weighted sums of homomorphisms [2,8].

In this paper, we study the complexity of counting homomorphisms modulo 2. For graphs G and H, $\mathrm{Hom}(G \to H)$ denotes the set of homomorphisms from G to H. For each fixed H, we study the computational problem $\oplus$HomsToH, which is the problem of computing $|\mathrm{Hom}(G \to H)| \bmod 2$, for an input graph G.

The structure of H strongly influences the complexity of $\oplus$HomsToH. For example, consider the graphs H_1 and H_2 in Figure 1. Our result (Theorem 1.2) shows that $\oplus$HomsToH_1 is $\oplus$P-complete, whereas $\oplus$HomsToH_2 is in P.

The aim of research in this area is to understand for which graphs H the problem $\oplus$HomsToH is in P, for which graphs H the problem is $\oplus$P-complete, and to prove that, for all graphs H, one or the other is true. Note that it isn't obvious, a priori, that there are no graphs H for which $\oplus$HomsToH has intermediate complexity – proving that there are no such graphs H is the main work of a so-called *dichotomy theorem*.

This line of work was introduced by Faben and Jerrum [6]. They made the following important conjecture (which requires a few definitions to state). An *involution* of a graph is an automorphism of order 2, i.e., an automorphism ρ that is not the identity but for which ρ^2 is the identity. Given a graph H and an involution ρ, H^ρ denotes the subgraph of H induced by the fixed points of ρ. We write $H \Rightarrow H'$ if there is an involution ρ of H such that $H^\rho = H'$ and we write $H \Rightarrow^* H'$ if either H is isomorphic to H' (written $H \cong H'$) or, for some positive integer k, there are graphs $H_1, \dots, H_k$ such that $H \cong H_1$, $H_1 \Rightarrow \cdots \Rightarrow H_k$, and $H_k \cong H'$. Faben and Jerrum showed [6, Theorem 3.7] that for every graph H there is (up to isomorphism) exactly one involution-free graph H^* such that $H \Rightarrow^* H^*$. This graph H^* is called the *involution-free reduction* of H.

Conjecture 1.1. (Faben and Jerrum [6]) Let H be a graph. If its involution-free reduction H^* has at most one vertex, then $\oplus$HomsToH is in P; otherwise, $\oplus$HomsToH is $\oplus$P-complete.

Note that our claim in Figure 1 is consistent with Conjecture 1.1. H_1 is involution-free, so it is its own involution-free reduction, but the involution-free reduction of H_2 is the single vertex marked $*$ in the figure.

Faben and Jerrum [6, Theorem 3.8] proved Conjecture 1.1 for the case in which H is a tree. Subsequently, the present authors [7, Theorem 1.6] proved the conjecture for a well-studied class of tree-width-2 graphs, namely *cactus graphs*, which are graphs in which each edge belongs to at most one cycle.

The main result of this paper is to prove the conjecture for a much richer class of graphs. In particular, we prove the conjecture for every graph H whose involution-free reduction has no 4-cycle. Graphs without 4-cycles are called "square-free" graphs. These graphs arise frequently in combinatorics, for example in connection with the strong perfect graph theorem [4] and certain graph algorithms [1]. Our main theorem is the following.

Theorem 1.2. *Let H be a graph whose involution-free reduction H^* is square-free.* $\oplus$HomsToH *is in* P *if H^* has at most one vertex; otherwise,* $\oplus$HomsToH *is* $\oplus$P*-complete.*

If H is square-free, then so is every induced subgraph, including its involution-free reduction H^*. Thus, we have the following corollary.

Corollary 1.3. *Let H be a square-free graph. If its involution-free reduction H^* has at most one vertex, then* $\oplus$HomsToH *is in* P*; otherwise,* $\oplus$HomsToH *is* $\oplus$P*-complete.*

In Section 1.3 we will discuss the reasons that we require H^* to be square-free in the proof of Theorem 1.2. First, in Section 1.1, we will describe the background to counting modulo 2. In Section 1.2, we will explain why Conjecture 1.1 is so much more difficult to prove for graphs with unbounded tree-width. Very briefly, in order to prove that $\oplus$HomsToH is $\oplus$P-hard without having a bound on the tree-width of H, it is necessary to take a much more abstract approach. Since it is not possible to decompose H using a tree-like decomposition as we did in [7, Theorem 1.6], we have instead come up with an abstract characterisation of graph-theoretic structures in H which lead to $\oplus$P-hardness. As we shall see, the proof that such structures always exist in square-free graphs involves interesting non-constructive elements, leading to a more abstract, and less technical (graph-theoretic) proof than [7], while applying to a substantially richer set of graphs H, including graphs with unbounded tree width.

1.1 Counting Modulo 2

Although counting modulo 2 produces a one-bit answer, the complexity of such problems has a rather different flavour from the complexity of decision problems. The complexity class $\oplus$P was first studied by Papadimitriou and Zachos [13] and by Goldschlager and Parberry [10]. $\oplus$P consists of all problems of the form "compute $f(x)$ mod 2" where computing $f(x)$ is a problem in #P. Toda [15] has shown that there is a randomised polynomial-time reduction from every problem in the polynomial hierarchy to some problem in ⊕P. As such, ⊕P is a large complexity class and ⊕P-completeness seems to represent a high degree of intractability.

The unique flavour of modular counting is exhibited by Valiant's famous restricted version of 3-SAT [16] for which counting solutions is #P-complete [17], counting solutions modulo 7 is in polynomial-time but counting solutions modulo 2 is ⊕P-complete [16]. The seemingly mysterious number 7 was subsequently explained by Cai and Lu [3], who showed that the k-SAT version of Valiant's problem is tractable modulo any prime factor of $2^k - 1$.

Counting modulo 2 closely resembles ordinary, non-modular counting, but is still very different. Clearly, if a counting problem can be solved in polynomial time, the corresponding decision and parity problems are also tractable, but the converse does not necessarily hold. A characteristic feature of modular counting is cancellations, which can make the modular versions of hard counting problems tractable. For example, consider not-all-equal SAT, the problem of assigning values to Boolean variables such that each of a given set of clauses contains both true and false literals. The number of solutions is always even, since solutions can be paired up by negating every variable in one solution to obtain a second solution. This makes counting modulo 2 trivial, while determining the exact number of solutions is #P-complete [9] and even deciding whether a solution exists is NP-complete [14].

We use cancellations extensively in this paper. For example, if we wish to compute the size of a set S modulo 2 then, for any even-cardinality subset $X \subseteq S$, we have $|S| \equiv |S \setminus X| \bmod 2$. This means that we can ignore the elements of X. It is also helpful to partition the set S into disjoint subsets $S_1, \ldots, S_\ell$ exploiting the fact that $|S|$ is congruent modulo 2 to the number of odd-cardinality S_i. We use this idea frequently.

1.2 Going Beyond Bounded Tree-Width

Trees. All known hardness results for counting homomorphisms modulo 2 start with the following basic "pinning" approach. Let p be a function from $V(G)$ to $2^{V(H)}$. A homomorphism $f \in \mathrm{Hom}(G \to H)$ *respects* the pinning function p if, for every $v \in V(G)$, $f(v)$ is in the set $p(v)$. Let $\mathrm{PinHom}(G, H, p)$ be the set of homomorphisms from G to H that respect the pinning function p and let ⊕PinnedHomsToH be the problem of counting, modulo 2, the number of homomorphisms in $\mathrm{PinHom}(G, H, p)$, given an input graph G and a pinning function p.

Faben and Jerrum [6, Corollary 4.18] give a polynomial-time Turing reduction from the problem ⊕PinnedHomsToH to the problem ⊕HomsToH for the special case in which the pinning function pins only two vertices of G, and these are both pinned to entire orbits of the automorphism group of H. The reduction relies on a result of Lovász [12].

In order to use the reduction, it is necessary to show that the special case of the problem ⊕PinnedHomsToH is itself ⊕P-hard. Faben and Jerrum restrict their attention to the case in which H is a tree, and this is helpful.

Every involution-free tree is asymmetric (so the orbit of every vertex is trivial), so the pinning function p is actually able to pin two vertices of G to any two *particular* vertices of H. The reduction that they used to prove hardness of $\oplus$PinnedHomsToH is from $\oplus$IS, the problem of counting independent sets modulo 2, which was shown to be $\oplus$P-complete by Valiant [16].

We first give an informal description of a general reduction from $\oplus$IS to the problem $\oplus$PinnedHomsToH. (The general description is actually based on our current approach in this paper, but we can also present past approaches in this context.) The vertices and edges of an input G of $\oplus$IS are replaced by gadgets to give a graph J. In J, the gadget corresponding to the vertex v of G has a vertex y^v. We also choose an appropriate vertex i in H. Any homomorphism σ from J to the target graph H defines a set $I(\sigma) = \{v \in V(G) \mid \sigma(y^v) = i\}$ (mnemonic: "i" means "in" because $\sigma(y^v)$ is i exactly when v is in $I(\sigma)$). The configuration of the gadgets ensures that a set $I \subseteq V(G)$ has an odd number of homomorphisms σ with $I(\sigma) = I$ if and only if I is an independent set of G. Next, the homomorphisms $\sigma \in \mathrm{Hom}(J \to H)$ can be partitioned according to the value of $I(\sigma)$. By the partitioning argument mentioned at the end of Section 1.1, the number of independent sets in G is equivalent to $|\mathrm{Hom}(J \to H)|$, modulo 2.

The gadgets are chosen according to the structure and properties of H. Since Faben and Jerrum were working with trees, they were able to use gadgets with very simple structure: their gadgets are essentially paths and they exploit the fact that any non-trivial involution-free tree has at least two even-degree vertices and, of course, these have a unique path between them (which turns out to be useful).

Cactus Graphs. The situation for cactus graphs is much more complicated. Non-trivial involution-free cactus graphs still contain even-degree vertices but the presence of cycles means that paths, even shortest paths, are no longer guaranteed to be unique. Our solution in [7] was to use more complicated gadgets. They are still (loosely) based on paths, since they are defined in terms of numbers of walks between vertices of H. However, rather than requiring appropriate even-degree vertices (which might not exist), we used a second, and more complicated, gadget to "select" an even-cardinality subset of a vertex's neighbours. To find such gadgets in H, we used tree-like decompositions. Given a decomposition that breaks H into independent fragments, we inductively found gadgets (or, sometimes, partial gadgets) in the fragments, carefully putting them together across the join of the decomposition. All of this led to a very technical, very graph-theoretic solution, and also to a solution that does not generalise to graphs without tree-like decompositions.

The proof is complicated by the fact that there are involution-free graphs (even involution-free cactus graphs!) that have non-trivial automorphisms, unlike the situation for trees. Thus, the fact that the pinning function pins vertices to entire orbits (rather than to particular vertices) causes complications. The solution in [7, Section 8] relies on special properties of cactus graphs, and it is not clear how it could be generalised.

Unbounded Tree-Width. Since they are based around a tree-like decomposition, the techniques of [7] are not suitable for graphs with unbounded tree-width. To prove Conjecture 1.1 for a richer class of graphs, we adopt a much more abstract approach. Since we do not have tree-like decompositions, we instead mostly use structural properties of the whole graph to find gadgets. The structural properties do not always require technical detail – as we will see below, re-examining a result of Lovász [12] even allows us to demonstrate non-constructively the existence of some of the gadgets that we use.

In order to support our more general approach, we first have to generalise the pinning problem $\oplus$PinnedHomsToH. We use the following **important definitions, which will be used later.** For any graph H, a *partially H-labelled graph* $J = (G, \tau)$ consists of an *underlying graph* G and a *pinning function* τ, which in this paper is a partial function from $V(G)$ to $V(H)$. Thus, every vertex v in the domain of τ is pinned to a *particular* vertex of H and *not* to a subset such as an orbit. A homomorphism from a partially labelled graph $J = (G, \tau)$ to H is a homomorphism $\sigma\colon G \to H$ such that, for all vertices $v \in \mathrm{dom}(\tau)$, $\sigma(v) = \tau(v)$. The intermediate problem that we study then is $\oplus$PartLabHomsToH, the problem of computing $|\mathrm{Hom}(J \to H)| \bmod 2$, given a partially H-labelled graph J. In Section 3, we generalise the application of Lovász's theorem to show (Theorem 3.1) that $\oplus$PartLabHomsTo$H \leq \oplus$HomsToH.

Armed with a stronger pinning technique, we then abstract away most of the complications that arose for graphs with small tree-width by instead using more general gadgets, defined in Section 4. Because they are not based on paths, they do not rely on uniqueness of any path in H. Instead, the gadgets have three main parts. Our new reduction from $\oplus$IS to $\oplus$HomsToH can be seen informally as assigning colours to both the vertices and the edges of G, where each "colour" is a vertex of H. One part of the gadget controls which colours can be assigned to each vertex, one controls which colours can be assigned to each edge and a third part determines how many homomorphisms there are from G to H, given the choice of colours for the vertices and edges. In addition to all of this, we identify two special vertices of H, one of which is the vertex i mentioned above.

The much more general nature of our gadgets compared to those used previously makes them much easier to find and, in some cases, allows us to find the parts of them non-constructively. We no longer need to find unique shortest paths in H or, indeed, any paths at all. In fact, all the gadgets that we construct in this paper use a "caterpillar gadget" (Definition 4.3) which allows us to use *any* specified path in the graph H instead of relying on a unique shortest path. Rather than finding hardness gadgets in components in some decomposition of H, we mostly find gadgets "in situ".

When a graph has two even-degree vertices, we can directly use those vertices and a caterpillar gadget to produce a hardness gadget (see Lemma 5.3). This already provides a self-contained proof of Faben and Jerrum's dichotomy for trees. Next, for graphs with only one even-degree vertex, we show (Corollary 5.5) that deleting an appropriate set of vertices leaves a component with two even-degree vertices and show (Lemma 5.7) how to simulate that vertex deletion with gadgets. This leaves only graphs in which every vertex has odd degree. In such a graph, we are able to use any shortest odd-length cycle to construct a gadget (Lemma 5.13). If there are no odd cycles, the graph is bipartite. In this interesting case (Lemma 5.15) we use our version of Lovász's result to find a gadget non-constructively.

1.3 Squares and Related Work

It is natural to ask why the involution-free reduction H^* in Theorem 1.2 is required to be square-free. We do not believe that the restriction to square-free graphs is fundamental, since our results on pinning apply to all involution-free graphs (Section 3) and neither our definition of hardness gadgets (Definition 4.1) nor our proof that the existence of a hardness gadget for H implies that $\oplus$HomsToH is $\oplus$P-complete (Theorem 4.2) requires H to be square-free. However, all the actual hardness gadgets that we find for graphs do rely on the absence of 4-cycles, as discussed in the full version, and removing this restriction seems technically challenging. We note that dealing with 4-cycles also caused significant difficulties in cactus graphs [7].

We have already mentioned earlier work on counting graph homomorphisms modulo 2. The problem of counting graph homomorphisms (exactly, rather than modulo a fixed constant) was previously studied by Dyer and Greenhill [5]. They showed the problem of counting homomorphisms to a fixed graph H is solvable in polynomial time if every connected component of H is a complete graph with a self-loop on every vertex or a complete bipartite graph with no self-loops, and is #P-complete, otherwise. Their work builds on an earlier dichotomy by Hell and Nešetřil [11] for the complexity of the graph homomorphism decision problem (the problem of distinguishing between the case where there are no homomorphisms and the case where there is at least one).

Note that much of the notation that we use below has been defined in the introduction. In addition, we write $[n] = \{1, \ldots, n\}$ and, for a set S and an element x, we often write $S - x$ for $S \setminus \{x\}$.

3 Partially Labelled Graphs and Pinning

It is often convenient to regard a graph as having some distinguished vertices $x_1, \ldots, x_r$ and we denote such a graph by $(G, x_1, \ldots, x_r)$. The distinguished vertices need not be distinct. A homomorphism from a graph $(G, x_1, \ldots, x_r)$ to $(H, y_1, \ldots, y_r)$ is a homomorphism σ from G to H with the property that $\sigma(x_i) = y_i$ for each $i \in [r]$. Isomorphisms of these graphs are defined similarly. In the full version, we generalise a result of Lovász [12] to prove the following.

Lemma 3.6. *Let $(H, \bar{y})$ and $(H', \bar{y}')$ be involution-free graphs, each with r distinguished vertices. $(H, \bar{y}) \cong (H', \bar{y}')$ if and only if, for all (not necessarily connected) graphs $(G, \bar{x})$ with r distinguished vertices,* $|\mathrm{Hom}((G, \bar{x}) \to (H, \bar{y}))| \equiv |\mathrm{Hom}((G, \bar{x}) \to (H', \bar{y}'))| \pmod 2$.

Recall that ⊕PartLabHomsToH is the problem of computing $|\mathrm{Hom}(J \to H)| \bmod 2$, given a partially H-labelled graph J. Using Lemma 3.6, and the implementation technique of Faben and Jerrum [6], we prove the following.

Theorem 3.1. ⊕PartLabHomsTo$H \leq$ ⊕HomsToH *for any involution-free graph* H*.*

The difference between Lemma 3.6 and similar previous lemmas is the inclusion of the distinguished vertices. This is necessary both for our more general pinning technique (Theorem 3.1) and because we will use Lemma 3.6 to non-constructively find hardness gadgets in Section 4.

4 Hardness Gadgets

In this section, we define the gadgets that we will use to prove ⊕P-completeness of ⊕HomsToH problems, by reduction from the parity independent set problem ⊕IS, i.e., the problem of computing the number of independent sets in an input graph, modulo 2. ⊕IS was shown to be ⊕P-complete by Valiant [16].

The gadgets that we use are considerably more general than the ones we defined for cactus graphs in [7]. This allows us to quickly prove hardness for large classes of square-free graphs and even to find gadgets non-constructively.

In the discussion that follows, we will choose a set $\Omega_y \subseteq V(H)$ and a vertex $i \in \Omega_y$. Given a graph G whose independent sets we wish to count modulo 2, we will construct a partially H-labelled graph $J = (G(J), \tau(J))$ and consider homomorphisms from J to H. $G(J)$ will contain a copy of $V(G)$ and we will be interested in homomorphisms that map every vertex in this copy to Ω_y. Vertices mapped to i will be in the independent set under consideration; vertices mapped to $\Omega_y - i$ will not be in the independent set.

Given a partially labelled graph $J = (G(J), \tau(J))$ and vertices $x_1, \ldots, x_r$ of $G(J)$ that are not in $\mathrm{dom}(\tau(J))$ and given vertices $y_1, \ldots, y_r$ of H, a homomorphism from $(J, x_1, \ldots, x_r)$ to $(H, y_1, \ldots, y_r)$ is a homomorphism from J to H which maps each x_i to y_i (for $i \in \{1, \ldots, r\}$).

Definition 4.1. *A* hardness gadget $(i, s, (J_1, y), (J_2, z), (J_3, y, z))$ *for a graph* H *consists of vertices* i *and* s *of* H *together with three connected, partially* H*-labelled graphs with distinguished vertices that satisfy the following properties. Let*

$$\Omega_y = \{a \in V(H) \mid |\mathrm{Hom}((J_1, y) \to (H, a))| \text{ is odd}\}$$
$$\Omega_z = \{b \in V(H) \mid |\mathrm{Hom}((J_2, z) \to (H, b))| \text{ is odd}\}$$
$$\Sigma_{a,b} = \mathrm{Hom}((J_3, y, z) \to (H, a, b))\,.$$

The properties that we require are that, for each $o \in \Omega_y - i$ *and each* $x \in \Omega_z - s$*, (1)* $|\Omega_y|$ *is even and* $i \in \Omega_y$*, (2)* $|\Omega_z|$ *is even and* $s \in \Omega_z$*, (3)* $|\Sigma_{o,x}|$ *is even, and (4)* $|\Sigma_{o,s}|$*,* $|\Sigma_{i,x}|$ *and* $|\Sigma_{i,s}|$ *are odd.*

The following theorem shows that the presence of a hardness gadget implies that ⊕HomsToH is ⊕P-complete.

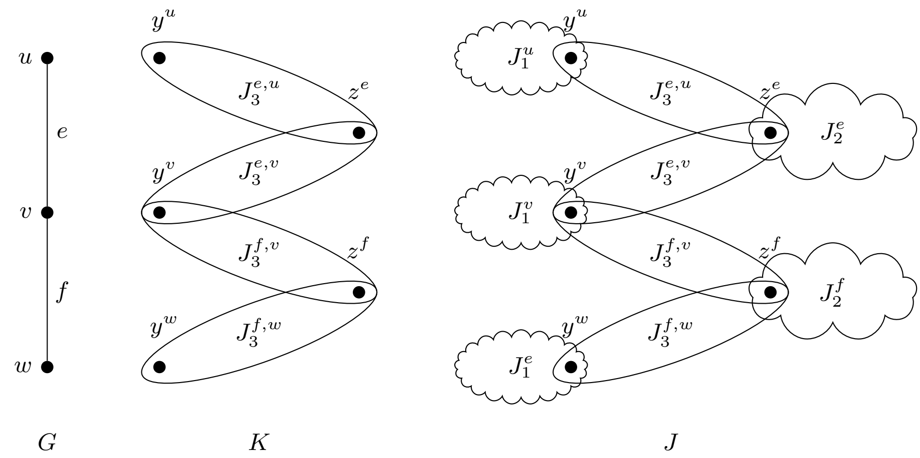

Fig. 2. The construction of the partially labelled graphs K and J from an example graph G, as in the proof of Theorem 4.2

Theorem 4.2. *$\oplus$*HomsTo*H is $\oplus$*P*-complete for any involution-free graph H that has a hardness gadget.*

The proof of Theorem 4.2 consists of a reduction from the $\oplus$P-complete problem $\oplus$IS to $\oplus$PartLabHomsToH together with Theorem 3.1. The reduction from $\oplus$IS to $\oplus$PartLabHomsToH is illustrated in Figure 2.

Given an input graph G to $\oplus$IS, we first construct the partially H-labelled graph K from G by replacing every edge of G with two disjoint copies of J_3, as shown in the figure. To construct the partially H-labelled graph J, we then take K and add a disjoint copy of J_1 for every vertex $v \in G$ and a disjoint copy of J_2 for every edge $e \in G$ as shown in the figure. In the full version, we calculate the number of homomorphisms from J to H and show that $|\mathrm{Hom}(J \to H)|$ is equivalent modulo 2 to the number of independent sets in G. Intuitively, the role of J_1^u is to cancel all homomorphisms, apart from those in which the vertex y^u is mapped to a vertex in Ω_y. Similarly, J_2^e cancels all homomorphisms, apart from those in which the vertex z^e is mapped to Ω_z. Then the four properties in the definition of hardness gadget and the connections using J_3 cancel all homomorphisms apart from those in which the set of vertices y^u that are mapped to the special vertex "i" form an independent set of G.

In the paper, we use a particular gadget called a "caterpillar gadget" as the partially H-labelled graph J_3.

Definition 4.3. *Given a path $P = v_0 \dots v_k$ in H of length at least 1, define the* caterpillar gadget *$J_P = (G, \tau)$ with distinguished vertices y and z as follows. $V(G) = \{u_1, \dots, u_{k-1}, w_1, \dots, w_{k-1}, y, z\}$ and G is the path $yu_1 \dots u_{k-1}z$ together with edges (u_j, w_j) for $1 \le j \le k-1$. $\tau = \{w_1 \mapsto v_1, \dots, w_{k-1} \mapsto v_{k-1}\}$. (See Figure 3).*

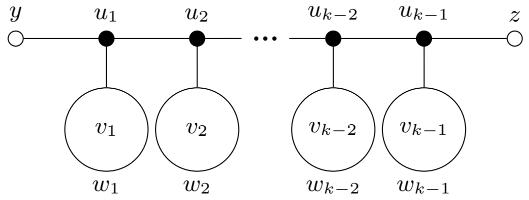

Fig. 3. The caterpillar gadget corresponding to a path $v_0 \dots v_k$. The vertices $w_1, \dots, w_{k-1}$ in the gadget are pinned to vertices $v_1, \dots, v_{k-1}$ in H, respectively. A label next to a vertex indicates its identity; a label inside a white circle indicates what that vertex is pinned to.

The following lemma explains why we use caterpillar gadgets as the J_3 gadgets that appear in hardness gadgets. The point is that the properties guaranteed here coincide with the ones required in the definition of hardness gadgets (Definition 4.2). We write $\Gamma_H(v)$ for the neighbourhood of a vertex v in a graph H.

Lemma 4.5. *Let H be a square-free graph. Let $k > 0$ and let $P = v_0 \dots v_k$ be a path in H with $\deg_H(v_j)$ odd for all $j \in \{1, \dots, k-1\}$. Let $\Omega_y \subseteq \Gamma_H(v_0)$ and $\Omega_z \subseteq \Gamma_H(v_k)$, with $i = v_1 \in \Omega_y$ and $s = v_{k-1} \in \Omega_z$. For each $o \in \Omega_y - i$ and each $x \in \Omega_z - s$ the following properties hold. (1) $|\mathrm{Hom}((J_P, y, z) \to (H, o, x))| = 0$, (2) $|\mathrm{Hom}((J_P, y, z) \to (H, o, s))| = 1$, (3) $|\mathrm{Hom}((J_P, y, z) \to (H, i, x))| = 1$, and (4) $|\mathrm{Hom}((J_P, y, z) \to (H, i, s))|$ is odd.*

The proof of Lemma 4.5 relies on the fact that H is square-free. It can be found in the full version. The point is that, even if the proof is a little bit technical — it is sufficiently general that it applies to every square-free graph H. As long as H is square-free, *any* caterpillar gadget has the desired properties, so J_3 can always be taken to be a caterpillar, without requiring a detailed structural analysis of H.

5 Finding Hardness Gadgets

In this section we show how to identify hardness gadgets in different graphs. If H has two or more even-degree vertices, we can directly use them to construct a hardness gadget. In this case, the partially H-labelled graphs J_1 and J_2 will just be edges. In each of these, exactly one vertex is pinned, and it is pinned to an even-degree vertex of H. This is captured in the following lemma, which already provides a self-contained proof of Faben and Jerrum's dichotomy for trees.

Lemma 5.3. *Let H be a connected, square-free graph with at least two even-degree vertices. Then H has a hardness gadget.*

If H has exactly one even-degree vertex, we first show that deleting an appropriate set of vertices leaves a component with two even-degree vertices.

Corollary 5.5. *Let H be an involution-free graph that has exactly one vertex v of positive, even degree. For some r, the graph formed from H by deleting the ball at distance r around v has an involution-free component H^* that does not contain v but does contain at least two even-degree vertices.*

Corollary 5.5 allows us to construct a hardness gadget for H by attaching a path to the gadget already constructed in Lemma 5.3. We prove in the full version that the additional path essentially simulates the vertex deletion from Corollary 5.5. After calculations we are able to prove the following.

Lemma 5.7. *Any involution-free, square-free graph H that has exactly one vertex v of positive, even degree has a hardness gadget.*

This leaves only graphs H in which every vertex has odd degree. If such a graph has an odd-length cycle then we can use it to construct an appropriate hardness gadget. In the full version, we prove the following lemma.

Lemma 5.13. *Let H be a square-free graph in which every vertex has odd degree. If H contains an odd cycle, then it has a hardness gadget.*

The most interesting case, and the only one left, is the case in which H is a bipartite graph in which every vertex has odd degree. We use the following definition.

Definition 5.14. *An* even gadget *for a bipartite graph H is a connected bipartite graph G with a distinguished edge (w, x) such that $|\mathrm{Hom}((G, w, x) \to (H, a, b))|$ is even, for some edge (a, b) in H.*

Using our extended version of Lovász's result (Lemma 3.6) we are able to prove the following. The key point is that every bipartite (G, w, x) has exactly one homomorphism to the single edge (a, b). Since H is not a single edge, Lemma 3.6 says there is a (G, w, x) with an even number of homomorphisms to (H, a, b). This is not necessarily an even gadget but it allows us to construct one.

Lemma 5.15. *Every connected, bipartite graph except K_2 has an even gadget.*

An even gadget turns out to be useful for the following reason. If G and H are bipartite, then there is always at least one homomorphism from (G, w, x) to (H, a, b), since the whole of G can be mapped to the edge (a, b). Thus, the definition of even gadget implies that $|\mathrm{Hom}((G, w, x) \to (H, a, b))|$ is even and positive. Using this fact, and the additional fact that H is square-free, we are able to apply some additional pinning to the even gadget that is guaranteed to exist, in order to obtain a hardness gadget, so we obtain the following.

Lemma 5.16. *Let H be a connected, bipartite, square-free graph in which every vertex has odd degree. H has a hardness gadget.*

6 Main Theorem

Theorem 1.2 follows rather directly from Lemma 5.3, Lemma 5.7, Lemma 5.13 and Lemma 5.16. A technical issue arises concerning the connectivity of the involution-free reduction. This is dealt with in the full version.

References

1. Arends, F., Ouaknine, J., Wampler, C.W.: On searching for small Kochen-Specker vector systems. In: Kolman, P., Kratochvíl, J. (eds.) WG 2011. LNCS, vol. 6986, pp. 23–34. Springer, Heidelberg (2011)
2. Bulatov, A.A., Grohe, M.: The complexity of partition functions. Theor. Comput. Sci. **348**(2–3), 148–186 (2005)
3. Cai, J.-Y., Lu, P.: Holographic algorithms: From art to science. J. Comput. Syst. Sci. **77**(1), 41–61 (2011)
4. Conforti, M., Cornuéjols, G., Vušković, K.: Square-free perfect graphs. J. Combin. Theory Ser. B **90**(2), 257–307 (2004)
5. Dyer, M.E., Greenhill, C.S.: The complexity of counting graph homomorphisms. Random Struct. Algorithms **17**(3–4), 260–289 (2000)
6. Faben, J., Jerrum, M.: The complexity of parity graph homomorphism: an initial investigation. Theor. Comput. **11**, 35–57 (2015)
7. Göbel, A., Goldberg, L.A., Richerby, D.: The complexity of counting homomorphisms to cactus graphs modulo 2. ACM T. Comput. Theory, **6**(4), article 17 (2014)
8. Goldberg, L.A., Grohe, M., Jerrum, M., Thurley, M.: A complexity dichotomy for partition functions with mixed signs. SIAM J. Comput. **39**(7), 3336–3402 (2010)
9. Goldberg, L.A., Gysel, R., Lapinskas, J.: Approximately counting locally-optimal structures. CoRR, abs/1411.6829 (2014)
10. Goldschlager, L.M., Parberry, I.: On the construction of parallel computers from various bases of Boolean functions. Theor. Comput. Sci. **43**, 43–58 (1986)
11. Hell, P., Nešetřil, J.: On the complexity of H-coloring. J. Comb. Theory, Ser. B **48**(1), 92–110 (1990)
12. Lovász, L.: Operations with structures. Acta Math. Acad. Sci. Hungar. **18**(3–4), 321–328 (1967)
13. Papadimitriou, C.H., Zachos, S.: Two remarks on the power of counting. In: Cremers, A.B., Kriegel, H.-P. (eds.) Theoretical Computer Science. LNCS, vol. 145, pp. 269–275. Springer, Heidelberg (1982)
14. Schaefer, T.J.: The complexity of satisfiability problems. In: Proc. 10th Annual ACM Symposium on Theory of Computing (STOC 1978), pp. 216–226. ACM Press (1978)
15. Toda, S.: PP is as hard as the polynomial-time hierarchy. SIAM J. Comput. **20**(5), 865–877 (1991)
16. Valiant, L.G.: Accidental algorithms. In: Proc. 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), pp. 509–517. IEEE (2006)
17. Xia, M., Zhang, P., Zhao, W.: Computational complexity of counting problems on 3-regular planar graphs. Theoret. Comput. Sci. **384**(1), 111–125 (2007)