Available online at www.sciencedirect.com
ScienceDirect
Theoretical Computer Science 385 (2007) 60–70
Theoretical Computer Science
www.elsevier.com/locate/tcs

On the complexity of dominating set problems related to the minimum all-ones problem☆

Hajo Broersma[a,b,*], Xueliang Li[b]

[a] *Department of Computer Science, Durham University, Science Labs, South Street, DH1 3LE, Durham, United Kingdom*
[b] *Center for Combinatorics and LPMC, Nankai University, Tianjin 300071, People's Republic of China*

Received 25 July 2006; received in revised form 21 May 2007; accepted 28 May 2007

Communicated by O. Watanabe

Abstract

The minimum all-ones problem and the connected odd dominating set problem were shown to be NP-complete in different papers for general graphs, while they are solvable in linear time (or trivial) for trees, unicyclic graphs, and series-parallel graphs. The complexity of both problems when restricted to bipartite graphs was raised as an open question. Here we solve both problems. For this purpose, we introduce the related decision problem of the existence of an odd dominating set without isolated vertices, and study its complexity. Our main result shows that this new problem is NP-complete, even when restricted to bipartite graphs. We use this result to deduce that the minimum all-ones problem and the connected odd dominating set problem are also NP-complete for bipartite graphs. We show that all three problems are solvable in linear time for graphs with bounded treewidth. We also show that the new problem remains NP-complete when restricted to other graph classes, e.g., planar graphs, graphs with girth at least five, and graphs with a small maximum degree, in particular 3-regular graphs.

Keywords: All-ones problem; Odd dominating set; Computational complexity; Bipartite graph; Planar graph

1. Introduction and related work

In this paper we study the complexity of a graph problem that has been introduced and studied under various names. This problem and its variations have received considerable attention [1–4,6–11,13,14,16–26]. The term all-ones problem was coined by Sutner in [24], where he also discussed applications of this problem in linear cellular automata (we refer to [24] for the details and more motivation and references). He described the all-ones problem for square grids as follows: Suppose each square of an $n \times n$ chessboard is equipped with an indicator light and a button. If the button of a square is pressed, the light of that square will change from off to on, and vice versa; the same happens

☆ Supported by NSFC.

* Corresponding author at: Department of Computer Science, Durham University, Science Labs, South Street, DH1 3LE, Durham, United Kingdom. Tel.: +44 1913341712.

E-mail addresses: hajo.broersma@durham.ac.uk (H. Broersma), lxl@nankai.edu.cn (X. Li).

0304-3975/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2007.05.027

to the lights of all the edge-adjacent squares. Initially all lights are off. Now, consider the following questions: is it possible to press a sequence of buttons in such a way that in the end all lights are on? This is referred to as the *all-ones problem*. If there is such a solution, how can we find it? And finally, how can we find a solution that presses as few buttons as possible? This is referred to as the *minimum all-ones problem*. All of the above questions can be asked for arbitrary graphs. Here and in what follows, we consider connected simple undirected graphs only. One can deal with disconnected graphs component by component. For all terminology and notation not defined here, we refer to [5]; for computational complexity terminology we refer to [15].

Instead of using the term all-ones problem, we prefer to adopt the terminology of Caro et al. [9], since from a graph-theoretical point of view the problem fits into the well-developed area of dominating sets. The all-ones problem is equivalent to the following dominating set problem: Given a graph $G = (V, E)$ with vertex set V and edge set E, one asks for a subset $S \subseteq V$ with the property that every vertex in S has an even number of neighbors in S, while every vertex in $V \setminus S$ has an odd number of neighbors in S. Since this implies that every vertex is dominated by an odd number of vertices in S (including the vertex itself if it belongs to S), S is called an *odd dominating set* (*OD-set* for short). In Sutner [24] it is called an *odd parity cover*. An equivalent version of the all-ones problem was proposed by Peled in [21], where it was called the *lamp lighting problem*.

Although it is not immediately clear from the definition, every graph has an *OD*-set. This has been proved by Sutner [26], using linear algebra. Another proof based on linear algebra is due to Lossers [20]. A short and elegant graph-theoretic proof appeared in [13].

If one asks for a smallest *OD*-set, the problem gets more complicated. Sutner [22] proved that deciding whether a graph has an *OD*-set of cardinality at most k is NP-complete. Here k is not fixed of course, since otherwise the problem is clearly solvable in polynomial time. For trees and unicyclic graphs, there is a linear time algorithm for finding the smallest *OD*-set [10,11], as well as for series-parallel graphs [2]. Other graph classes were studied by Caro et al. [7,8]. The complexity of this problem restricted to bipartite graphs was left as an open problem.

A variation of the problem in which one asks for the existence of a connected *OD*-set was introduced and studied in [9]. This problem is obviously trivial for trees and unicyclic graphs, but is NP-complete for general graphs [9]. Also here, the complexity of the problem restricted to bipartite graphs was left as an open problem.

1.1. Results of this paper

In order to solve the complexity questions for the two problems restricted to bipartite graphs, we introduce and study the complexity of a new variant in which we weaken the connectivity condition to the condition that the *OD*-set contains no isolated vertices (i.e., vertices with no neighbors in the *OD*-set). This problem is also trivial for trees and unicyclic graphs. It is interesting in its own right, but we show that it is a useful intermediate for proving complexity results for the minimum all-ones problem as well as the connected *OD*-set problem. Our main results show that all three problems are NP-complete when restricted to bipartite graphs.

For graphs with bounded treewidth, however, all three problems are shown to be solvable in linear time, by using monadic second-order logic (MSOL). The use of MSOL in this context may look a bit surprising since one cannot express parity problems in MSOL, but we can get around it by using the paradigm of the lamp lighting problem.

Finally, we show that the problem related to *OD*-sets without isolated vertices is NP-complete when restricted to several other graph classes, like planar graphs, graphs with girth at least five, and graphs with a small maximum degree, in particular 3-regular graphs.

The paper is organised as follows. In the next section we introduce the necessary terminology and notation. In Section 3 we prove NP-completeness of the new variant restricted to bipartite graphs, while in Sections 4 and 5 we use this result to prove NP-completeness of the original two problems restricted to bipartite graphs. In Section 6 we show that all three problems can be solved in linear time when restricted to graphs with bounded treewidth, thereby generalizing the known results on trees and series-parallel graphs. In Section 7, we show that the problem related to *OD*-sets without isolated vertices is NP-complete when restricted to several other graph classes.

2. Preliminaries

Before we present our main results, we introduce some additional terminology and notation. Let $G = (V, E)$ be a graph. If $S \subseteq V$ and $S \neq \emptyset$, then $G[S]$ denotes the subgraph of G induced by S, i.e., $G[S]$ has vertex set S and

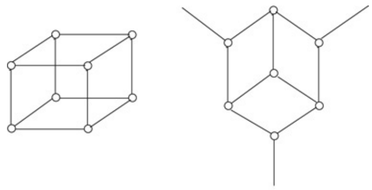

Fig. 1. The cube Q and the oQ-gadget.

its edge set contains all the edges of G with both end vertices in S. A vertex $s \in S$ is called an *isolate* (or isolated vertex) of S if it has no neighbors in S. The set of neighbors of a vertex $v \in V$ is denoted by $N(v)$, and the degree of v by $d(v) = |N(v)|$. The closed neighborhood of v is $N[v] = N(v) \cup \{v\}$. An *OD-set* of G is a set $D \subseteq V$ such that $|N[v] \cap D| \equiv 1 \bmod 2$ for every vertex $v \in V$; this implies that all vertices of $G[D]$ have an even degree, a fact that we will use frequently in the sequel. We use MOD-set for an *OD*-set of minimum cardinality, *COD*-set for an *OD*-set D such that $G[D]$ is connected, and $\neg 0$*OD*-set for an *OD*-set without isolated vertices.

The *Cube* Q is the graph illustrated in Fig. 1(a). For an arbitrary vertex $v \in V(Q)$, the graph $Q - v$ is illustrated in Fig. 1(b), together with three *half-edges* that will be used to attach $Q - v$ to some other graph. We call this an *open Cube* or *oQ-gadget*.

A *k-lollipop* consists of a k-cycle, i.e., a cycle C_k on $k \geq 3$ vertices, together with one half-edge, the *stick* of the lollipop, incident with one vertex of the k-cycle.

3. Odd dominating sets without isolates

We consider the decision problem $\neg 0$*ODS* defined as follows:

<u>$\neg 0$*ODS*</u>
INSTANCE: Graph $G = (V, E)$.
QUESTION: Is there a $\neg 0$*OD*-set $S \subseteq V$?

We first prove that $\neg 0$*ODS* is NP-complete for general graphs. We use a reduction from 1-in-3 3SAT with no negated literals ([15], Problem LO4; see the comments). We use 3SAT* to denote this problem and recall the definition for convenience:

<u>3SAT*</u>
INSTANCE: Set U of variables, collection C of clauses over U such that each clause $c \in C$ has $|c| = 3$ and contains no negated literal.
QUESTION: Is there a truth assignment for U such that each $c \in C$ has exactly one true literal?

Theorem 1. *$\neg 0$ODS is NP-complete.*

Proof. $\neg 0$*ODS* is obviously in NP. We complete the proof by showing that for any instance I of 3SAT* we can construct a graph G_I of polynomial size in terms of the size of the instance I such that I has a satisfying truth assignment if and only if G_I has a $\neg 0$*OD*-set.

Let I be an instance of 3SAT* with clause set $C = \{c_1, c_2, \ldots, c_p\}$ and variable set $U = \{u_1, u_2, \ldots, u_q\}$. We construct a graph G_I as follows, as indicated in Fig. 2.

For each clause we introduce an oQ-gadget in which the half-edges are incident with the three variables in the clause; these vertices are called the *variable vertices*. If two clauses have a variable in common, we add a 4-lollipop and join its stick to a vertex which we also join to the two corresponding variable vertices; we call this vertex the *stick vertex*. We add disjoint 4-lollipops for each of the common variable pairs. We add a new vertex for each clause and join it to the vertex of the corresponding oQ-gadget that is a neighbor of none of the variable vertices; we call these vertices of degree 1 the *clause vertices*. We make a number of simple observations, each followed by a short proof.

(1) None of the stick vertices are in any $\neg 0$*OD*-set. This is clear, as every vertex of a 4-lollipop is dominated by a $\neg 0$*OD*-set and therefore all vertices of the 4-cycle are contained in the $\neg 0$*OD*-set. As every vertex of a $\neg 0$*OD*-set has an even number of neighbors in the $\neg 0$*OD*-set, the stick vertices are not contained in any $\neg 0$*OD*-set.

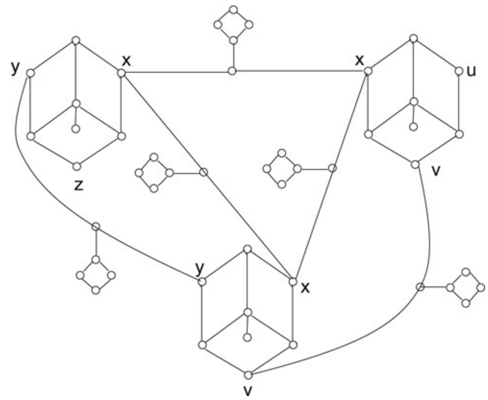

Fig. 2. G_I for $I = \{\{x, y, z\}, \{x, u, v\}, \{x, y, v\}\}$.

(2) All vertices of the 4-lollipops are in any $\neg 0OD$-set. This is clear.

(3) If a variable is shared by several clauses, then the corresponding variable vertices are either all in the $\neg 0OD$-set or all not in the $\neg 0OD$-set. This follows from (1) and (2) and the fact that the stick vertices have an odd number of neighbors in any $\neg 0OD$-set.

(4) The clause vertices are in no $\neg 0OD$-set. This is obvious; otherwise we cannot avoid isolates in the $\neg 0OD$-set.

(5) The neighbors of the clause vertices are in every $\neg 0OD$-set. This follows immediately from (4).

(6) Exactly one C_4 of every oQ-gadget is in any $\neg 0OD$-set. By (5) and the fact that there are no isolates in any $\neg 0OD$-set, two neighbors of exactly one of the variable vertices of every oQ-gadget are in any $\neg 0OD$-set. The observation follows easily.

If I has a satisfying truth assignment, then in G_I we define a $\neg 0OD$-set D as follows. We let all vertices of all C_4's corresponding to the true variables and the 4-lollipops belong to D. Then D is clearly a set without isolates, it is consistent with variables appearing in more than one clause, and it is easy to check that D is an OD-set.

Conversely, suppose D is a $\neg 0OD$-set of G_I. Then by observations (1)–(6), each oQ-gadget intersects D in exactly one C_4, with C_4's of the corresponding variable vertices that are shared by several clauses all appearing or all not appearing in D. If we set a variable true if and only if the corresponding C_4's appear in D, we get a satisfying truth assignment for I. This completes the proof. ∎

Since the graphs G_I that appear in the above proof are clearly bipartite, we obtain the following consequence.

Corollary 2. *$\neg 0ODS$ is NP-complete for bipartite graphs.*

We will use this result in the next two sections to prove that the minimum all-ones problem as well as the connected OD-set problem remain NP-complete when restricted to bipartite graphs.

Note that, for ease of presentation, in the above proof we joined up all variable vertices that occur in more than one clause as a complete graph (with 4-lollipops in between). We could have joined them up as a path (with 4-lollipops in between) just as well. We will use this observation later to restrict the problem to graphs with a small maximum degree and to 3-regular graphs.

4. Minimum all-ones problem for bipartite graphs

In this section we use the results of Section 3 to prove that the following problem remains NP-complete for bipartite graphs.

MODS
INSTANCE: Graph $G = (V, E)$ and integer $k \leq |V|$.
QUESTION: Is there an OD-set $S \subseteq V$ with $|S| \leq k$?

As we remarked before, *MODS* is known to be NP-complete for general graphs [22], while it is solvable in linear time for trees [10]. It is natural to ask for the complexity of *MODS* when restricted to bipartite graphs.

Theorem 3. *MODS is NP-complete for bipartite graphs.*

Proof. *MODS* is obviously in NP. Let G be an instance graph for $\neg 0ODS$ from the proof of Theorem 1. Let N denote the number of stick vertices (the number of lollipops) and O denote the number of oQ-gadgets (the number of clause vertices) of G. Let A, B be the bipartition classes of G, chosen such that all variable vertices are in B. We add two new vertices for each A-vertex in each oQ-gadget (but not for the clause vertex) and join them to the corresponding A-vertex. We also add $4N + 6O + 2$ new vertices for each A-vertex in the C_4 of every 4-lollipop and join them to the corresponding A-vertex. Let this new graph be G^*. Since all of the newly added vertices have degree 1, they will be in an OD-set of G^* if and only if their neighbor is not in the OD-set. This enables us to prove the following equivalence: G has a $\neg 0OD$-set if and only if G^* has an OD-set of cardinality at most $4N + 6O$.

If G has a $\neg 0OD$-set D, then as in the proof of Theorem 1, a 4-cycle of each oQ-gadget and 4-lollipop is in D. This can be extended to an OD-set D^* of G^* by adding the two degree 1 vertices for each of the O A-vertices of the oQ-gadgets that are not in D. In total this gives $2O + 4O + 4N = 6O + 4N$ vertices in D^*.

For the converse, we assume that G^* has an OD-set D^* with $|D^*| \leq 6O + 4N$. This immediately implies that all A-vertices of the 4-lollipops (minus the sticks) are in D^*, hence that their 4-cycles are in $G^*[D^*]$. We complete the proof by showing that in D^* precisely one 4-cycle through one variable vertex of each of the oQ-gadgets belongs to D^*. This immediately translates to a $\neg 0OD$-set in G.

Since the 4-cycles of all 4-lollipops are in $G^*[D^*]$, as in the proof of Theorem 1 all variable vertices of one variable are either all in D^* or all not in D^*. If an oQ-gadget has none of its variable vertices in D^*, then not all of these variable vertices can be dominated by an odd number of neighbors. This is easy to check. So each oQ-gadget has at least one of its variable vertices v in D^*. If v has degree 2 in $G^*[D^*]$, we either have a 4-cycle or a 6-cycle of this oQ-gadget in $G^*[D^*]$. One easily checks that the 6-cycle leads to a contradiction for the middle vertex or its neighboring clause vertex. So we obtain that D^* intersects the oQ-gadget in a 4-cycle and that D^* also contains the two pendant vertices that are not adjacent to this 4-cycle. Thus we get a contribution of 6 vertices to D^* for such an oQ-gadget. If none of the variable vertices has degree 2 in $G^*[D^*]$, then the set D^* intersects this oQ-gadget in an independent set. First suppose that at least one A-vertex v of this gadget is in D^*. Then $N(v) \cap D^* = \emptyset$ and the clause vertex is in D^*. Considering the middle vertex of the gadget, we get that exactly one of the other A-vertices, say w, is in D^*. But then the variable vertex in $N(v) \cap N(w)$ has two neighbors in D^*, a contradiction. We conclude that none of the A-vertices of this gadget is in D^*. But then all 6 added pendant vertices are in D^* together with all the variable vertices and the middle vertex of this gadget. Since all oQ-gadgets contribute at least 6 vertices to D^* and all of the 4-lollipops contribute 4 vertices to D^*, we get a contradiction with $|D^*| \leq 6O + 4N$. Hence this case does not occur.

So each oQ-gadget contributes a 4-cycle and two degree 1 vertices to D^*, so the overall cardinality of D^* is $4N+6O$. As mentioned before, the 4-cycles in $G^*[D^*]$ correspond to a $\neg 0OD$-set of G. This completes the proof. ∎

5. Connected odd dominating set for bipartite graphs

The concept of a *COD*-set has been introduced recently by Caro et al. [9]. They proved that the related decision problem is NP-complete for general graphs. It is clearly trivial for trees. In their concluding remarks they mentioned the natural open problem of resolving the complexity for bipartite graphs. We use the results of Section 3 to solve this problem by proving that the following problem remains NP-complete for bipartite graphs.

<u>*CODS*</u>
INSTANCE: Graph $G = (V, E)$.
QUESTION: Is there a *COD*-set $S \subseteq V$?

Theorem 4. *CODS is NP-complete for bipartite graphs.*

Proof. *CODS* is obviously in NP. To complete the proof we use a reduction from $\neg 0ODS$ for bipartite graphs. From a bipartite instance graph G of $\neg 0ODS$ we construct a bipartite graph G^* which is polynomial in the size of G such that G has a $\neg 0OD$-set if and only if G^* has a *COD*-set. Our construction resembles the construction known as Mycielski's construction ([5], page 129) for obtaining triangle-free graphs of arbitrarily high chromatic number.

Let G be an instance graph for $\neg 0ODS$, with bipartition classes A and B. For each vertex $v \in A \cup B$ we create two *buddies* v' and v'' and we join v' and v'' to all the neighbors of v in G. We also add four new vertices x_A, x'_A, x_B and

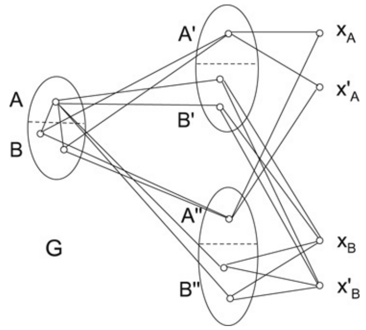

Fig. 3. The construction in the proof of Theorem 4.

x'_B and join x_A, x'_A to all vertices in $\{v'|v \in A\} \cup \{v''|v \in A\}$, and x_B, x'_B to all vertices in $\{v'|v \in B\} \cup \{v''|v \in B\}$. Let the new graph be G^*. The construction is illustrated in Fig. 3.

Note that in Mycielski's construction, only one buddy is created for each vertex of G, and an additional vertex is joined to all buddy vertices, creating a nonbipartite graph.

G^* is clearly bipartite. It remains to prove that G has a $\neg 0OD$-set if and only if G^* has a COD-set. Firstly, let D be a $\neg 0OD$-set of G. Then we can extend D to an OD-set D^* of G^* in the following way: for each $v \in D$ put v, v', v'' in D^*. Furthermore put x_A, x'_A, x_B, x'_B in D^*. As v' and v'' have the same neighbors in G as v, and are both neighbors of either x_A and x'_A or x_B and x'_B, clearly all degrees in $G^*[D^*]$ are even. By similar arguments, each vertex in $V(G^*) \setminus D^*$ has an odd number of neighbors in D^*, so D^* is an OD-set. Since D is a $\neg 0OD$-set of G, each $v \in D$ has at least two neighbors in D. Hence each $v \in D$ is connected to a $u' \in D^*$ for some $u \in D$. Clearly this implies that all $v \in D^*$ are connected through the vertices x_A, x'_A, x_B, x'_B. Hence D^* is a COD-set of G^*.

For the converse, consider the sets $A' = \{v'|v \in A\}$ and $A'' = \{v''|v \in A\}$. Since the corresponding vertices in A' and A'' have the same neighborhoods in G^*, they are both either in an OD-set or not in an OD-set. So any OD-set contains corresponding subsets of A' and A''. This also implies that both x_A and x'_A are in any OD-set of G^*. Similar arguments apply to the analogously defined sets B', B'' and x_B, x'_B. Now, except for x_A, x'_A, a vertex of A also has the same neighbors as its buddies in A', A''. So also for A, A' and A'' corresponding subsets are in any OD-set of G^*, and the same holds for B, B', B''. Now let D^* be a COD-set of G^*. Then D^* contains no isolated vertex. Hence $D^* \cap (A \cup B)$ contains no isolated vertex. So the restriction of D^* to G is a $\neg 0OD$-set of G. This completes the proof. ■

6. Bounded treewidth

In this section we use MSOL; that is, that fragment of second-order logic where quantified relation symbols must have arity 1. For example, the following sentence, which expresses that a graph (whose edges are given by the binary relation E) can be 3-coloured, is a sentence of monadic second-order logic:

$$\exists R \exists W \exists B \{\forall x\, ((R(x) \vee W(x) \vee B(x)) \;\wedge \neg(R(x) \wedge W(x))$$
$$\wedge \neg(R(x) \wedge B(x)) \wedge \neg(W(x) \wedge B(x))\;) \wedge \forall x \forall y\; (\; E(x, y) \Rightarrow$$
$$(\neg(R(x) \wedge R(y)) \wedge \neg(W(x) \wedge W(y)) \wedge \neg(B(x) \wedge B(y)))\;)\}$$

(the quantified unary relation symbols are R, W and B, and should be read as sets of 'red', 'white' and 'blue' vertices, respectively). Thus, in particular, there exist NP-complete problems that can be defined in MSOL.

A seminal result of Courcelle [12] is that on any class of graphs of bounded treewidth, every problem definable in MSOL can be solved in time linear in the number of vertices of the graph. Moreover, Courcelle's result holds not just when graphs are given in terms of their edge relation, as in the example above, but also when the domain of a structure encoding a graph G consists of the disjoint union of the set of vertices and the set of edges, as well as unary relations V and E to distinguish the vertices and the edges, respectively, and also a binary incidence relation I which denotes when a particular vertex is incident with a particular edge (thus, $I \subseteq V \times E$). The reader is referred to [12] for more details as regards MSOL on graphs and also for the definition of treewidth which is not required here. For the proof of our claim that all three problems are solvable in linear time for graphs with bounded treewidth, it is sufficient to show the following.

Proposition 5. *MODS, ¬0ODS and CODS can be defined in MSOL.*

Proof. We first recall the paradigm of the lamp lighting problem. Suppose initially that all vertices of a graph G are in state 0 (no lamp is lighted), and in each step of a *lighting scheme* for one vertex v the vertices of $N[v]$ change state from 0 to 1 or from 1 to 0. Then G has an *OD*-set of cardinality at most k if and only if after k steps of a lighting scheme all vertices of G are in state 1. So a lighting scheme for a graph $G = (V, E)$ is a sequence of graphs

$$G \rightsquigarrow G_1 \rightsquigarrow G_2 \rightsquigarrow \cdots \rightsquigarrow G_k,$$

where each G_i is isomorphic to G, but the states of the vertices (can) differ. Let $W_0 = V$ and, for $1 \leq i \leq k$, let W_i be the set of vertices of G_i that are in state 0. Let $V(v)$ denote that $v \in V$, and let $E(u, v)$ denote that $uv \in E$. (To be precise, instead of $uv \in E$, we should write $\exists e : e \in E \wedge (u, e) \in I \wedge (w, e) \in I$.)

If we can write a formula $\Phi(W_i, W_{i+1})$ of MSOL that says

there exists a vertex v_i in G_i such that starting with the set W_i of vertices in state 0 in G_i, changing the state of the vertices in $N[v_i]$ yields the set W_{i+1} of vertices in state 0 in G_{i+1},

then we could prove the proposition for *MODS* with the following sentence Ω_k which is satisfied if and only if G has an *OD*-set of cardinality at most k:

$$\exists W_0 \exists W_1 \cdots \exists W_k (\forall v (W_0(v) \Leftrightarrow V(v))) \wedge \Phi(W_0, W_1) \wedge \Phi(W_1, W_2) \wedge \cdots \wedge \Phi(W_{k-1}, W_k)$$
$$\wedge (\forall v (\neg W_k(v) \Leftrightarrow V(v))).$$

(Here and elsewhere we have presupposed that each W_i is a set of vertices; we could easily include additional clauses to check this explicitly.)

Since the change of states only affects vertices in $N[v_i]$, it is not difficult to write $\Phi(W_i, W_{i+1})$ (here u plays the role of v_i):

$$\exists u (V(u) \wedge (W_i(u) \Leftrightarrow \neg W_{i+1}(u)) \wedge \forall w ((V(w) \wedge E(u, w) \Rightarrow (W_i(w) \Leftrightarrow \neg W_{i+1}(w))))).$$

Checking for isolated vertices in the *OD*-set or checking whether the *OD*-set induces a connected subgraph can be incorporated in a rather straightforward way. We omit the details. ■

7. OD-sets without isolates revisited

In this section we study ¬0*ODS* restricted to other graph classes. We will show that ¬0*ODS* remains NP-complete for a number of graph classes, including planar graphs, graphs with girth at least 5, and graphs in which the maximum degree is bounded by a small constant, in particular also 3-regular graphs. We start by recalling the observation that in the proof of Theorem 1 we could have connected the variable vertices that occur in more than one clause by a path instead of a complete graph (with 4-lollipops added in the same way as in the proof of Theorem 1). This immediately yields the following result.

Corollary 6. *¬0ODS is NP-complete for bipartite graphs with $\Delta \leq 4$.*

Before we reduce the maximum degree to 3 (losing bipartiteness) in the last subsection, we first introduce another useful gadget in the next subsection that deals with girth restrictions.

7.1. ¬0ODS for graphs with girth at least 5

The *girth* of a graph G is the length of a smallest cycle in G. Since the graphs in the proof of Theorem 1 are bipartite but contain (a lot of) 4-cycles, they have girth 4. In this subsection we show that we can replace the oQ-gadgets by oP-gadgets, that are defined as follows.

Fig. 4(a) shows the Petersen graph P, whereas in Fig. 4(b) we have redrawn $P - v$ for an arbitrary vertex $v \in V(P)$ together with three half-edges. We call this an *open Petersen graph* or *oP-gadget*.

Combined with 5-lollipops this yields the following.

Corollary 7. *¬0ODS is NP-complete for graphs with girth at least* 5.

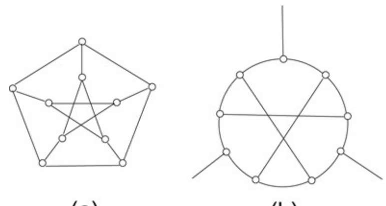

Fig. 4. The Petersen graph and the oP-gadget.

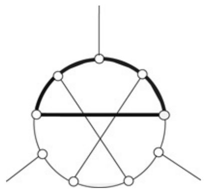

Fig. 5. A $\neg 0OD$-set for the oP-gadget.

Let us explain the construction. Replace each oQ-gadget in the graphs from the proof of Theorem 1 by an oP-gadget, and each 4-lollipop by a 5-lollipop. The only thing that remains is to analyze the possible ways in which a $\neg 0OD$-set can intersect an oP-gadget.

The 5-cycle indicated in Fig. 5 by the thick edges contains precisely one variable vertex, and all vertices of the oP-gadget that are not on this 5-cycle have an odd number of neighbors (one) on this 5-cycle. There is a similar 5-cycle that uses the other two chords of the 9-cycle. These 5-cycles exist for each of the variable vertices. It is easy to check that the larger cycles in the oP-gadget are not induced or dominate a vertex an even number of times (two). Therefore, the 5-cycles have the same properties with respect to $\neg 0OD$-sets as the 4-cycles in the oQ-gadget in the proof of Theorem 1. We omit the details.

We will use the above approach later to prove that $\neg 0ODS$ remains NP-complete for 3-regular graphs, but we will lose the girth restriction.

7.2. $\neg 0ODS$ for planar graphs

In this subsection we will show that $\neg 0ODS$ remains NP-complete for planar graphs. In order to do so we present a gadget for replacing intersecting edges in an embedding of the graphs from the proof of Theorem 1. The hardest part is to show that the proposed gadget has the suitable properties with respect to $\neg 0OD$-sets. First note that the oQ-gadgets are planar and can be put in the plane without intersecting each other. The only thing we have to consider is intersections between connecting paths that join variable vertices corresponding to variables that are shared by several clauses. The lollipops can be neglected as they can always be added to a plane embedding of the graphs in which the lollipops have been contracted to the stick vertex where they have been attached. By the observation that in the proof of Theorem 1 we could have connected the variable vertices that occur in more than one clause by a path instead of a complete graph (with 4-lollipops added in the same way as in the proof of Theorem 1), we may assume that intersections only occur between connecting paths that join pairs associated with different variables. We will use the *grid gadget* that is illustrated in Fig. 6.

As illustrated in Fig. 6, we start with a 4×4-grid and we add a P_3 with a lollipop attached to its middle vertex at two sides (let us say west and south) of the 4×4-grid, while we add a P_3 with nothing attached to it at the other sides (east and north). The situation that is indicated by the 10-cycle with the thick edges in Fig. 6 corresponds to the case that both the x-vertices and y-vertices belong to a $\neg 0OD$-set (since their neighbors are clearly not in the $\neg 0OD$-set, so must have an odd number of neighbors in the $\neg 0OD$-set). If we take a similar 10-cycle containing the north and east corners of the 4×4-grid, this corresponds to the case that both the x-vertices and y-vertices do not belong to a $\neg 0OD$-set. The case that the x-vertices are in a $\neg 0OD$-set that does not contain the y-vertices can be represented by a 10-cycle containing the north and west corner; the final case by a 10-cycle through the east and south corner. It is not difficult to check that there is no $\neg 0OD$-set that could contain another combination with three variable vertices in or

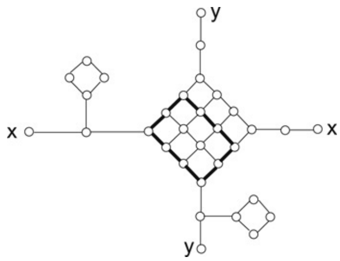

Fig. 6. The grid gadget.

not in the set. In fact, by looking at the degree four vertices of the grid, it is not difficult to show that a ¬0*OD*-set D can only intersect the grid gadget in one of the four 10-cycles as described above:

- If all of them are in D, then one of the corners of the grid is not dominated.
- If three of them are in D, then the other one cannot have three neighbors in D.
- If two of them are in D, the case that they are nonadjacent yields to a contradiction by looking at one of the other degree four vertices: it must have three neighbors in D, but then the other neighbor has two neighbors in D; the case that they are adjacent yields one of the 10-cycles by looking at the number of neighbors the other degree four vertices have in D: if this number is 3 for one of them, then it is 3 for both, and a 10-cycle is immediate; if this number is 1 for both, we get a contradiction in a corner of the grid.
- If one of them is in D, then looking at the nonadjacent degree four vertex it must have one neighbor in D. The way D intersects the grid is then prescribed (up to symmetry) and we get a contradiction in the corner closest to the degree four vertex in D.
- If none of them is in D, we can only have that D intersects the grid in two paths along opposite sides of the grid, a clear contradiction (because of the 4-lollipops in the gadget).

This confirms that the grid gadget has the suitable properties with respect to ¬0*OD*-sets. It is routine to check that we can replace intersections in an embedding one by one, using the grid gadget. Hence we obtain the following result.

Corollary 8. *¬0ODS is NP-complete for planar graphs.*

Using the grid gadget we cannot avoid introducing odd cycles (although triangles can be avoided). It is not unlikely that some other gadget exists to show that ¬0*ODS* remains NP-complete for bipartite planar graphs. However, the above gadget would be useless in the case of *OD*-sets, since then combinations with three variable vertices in an *OD*-set (containing isolates) are possible.

7.3. Further degree restrictions for ¬0ODS

In this subsection we will make some final remarks on restricting the maximum degree of the instance graphs for ¬0*ODS*, and we will show that ¬0*ODS* remains NP-complete for 3-regular graphs.

For the remainder we assume we use the oP-gadgets instead of the oQ-gadgets and we use connecting paths between the variable vertices appearing in more than one clause. By Corollary 6 we only have to take care of removing any degree two and degree four vertices (there are no vertices with degree smaller than two in the graphs in the proof of Theorem 1 if we use oP-gadgets). The only variable vertices with degree 2 or 4 are variable vertices appearing in only one clause or appearing as internal vertices on the connecting paths, respectively; the other variable vertices have degree 3. We will first introduce a new gadget in order to bring the maximum degree down to 3. After that we will work towards 3-regular graphs. In order to do so, we will introduce an alternative for the 4-lollipop to avoid degree 2 vertices in the lollipops, and we will duplicate clause gadgets if at least one of its variable vertices has degree 2, and connect the corresponding variable vertices with paths as before.

In order to get rid of variable vertices that have degree four, we introduce the *3-gadget* that is illustrated in Fig. 7.

We assume that the half-edges of this gadget are attached to vertices that are sticks of lollipops. Then it is obvious that any ¬0*OD*-set intersects the 3-gadget in a number of cycles. By considering the grey vertices of the gadget we are able to make the latter statement more precise.

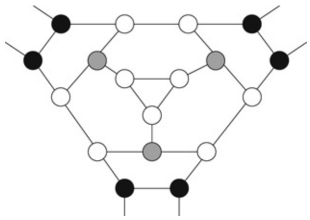

Fig. 7. The 3-gadget.

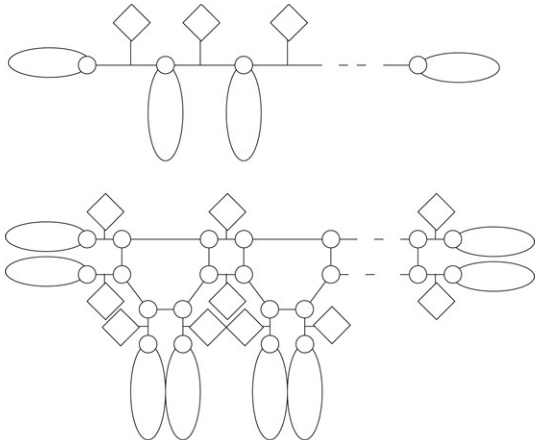

Fig. 8. How to get rid of degree 4 vertices.

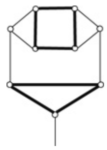

Fig. 9. The extended lollipop.

If all three grey vertices are in a $\neg 0OD$-set D, then it is easy to see that D intersects the gadget in a 9-cycle neither containing any of the black vertices nor any of the triangle vertices. If some grey vertex v is not in D, then the triangle vertices must be in D; otherwise the neighbor of v in the triangle will have 0 or 2 neighbors in D. This implies that in this case none of the grey vertices is in D. Looking at the black vertices, we can conclude that in this case all vertices except the grey ones are in D. So, the set D intersects the gadget in a 12-cycle and a 3-cycle.

In particular, we have that either all or none of the black vertices of a 3-gadget are in a $\neg 0OD$-set.

Suppose now that we have a connecting path between variable vertices that occur in more than two clauses, yielding vertices with degree 4. Then we duplicate all clause gadgets involved in this path, and we use the 3-gadgets to connect them as indicated in Fig. 8. In this figure the ovals indicate the clause gadgets, the diamonds indicate the lollipops, and the 6-cycles indicate the connecting 3-gadgets.

By the property of the 3-gadget, it is clear that this shows that $\neg 0ODS$ remains NP-complete for graphs with $\Delta \leq 3$.

To avoid vertices of degree 2 caused by the 4-lollipops, we can use the *extended lollipop* that is shown in Fig. 9. One easily checks that the 4-cycle and 3-cycle indicated by the thick edges in Fig. 9 are contained in any $\neg 0OD$-set of a graph that contains this extended lollipop attached to some vertex. Using this extended lollipop instead of the 4-lollipops, and duplicating clause gadgets to get rid of degree 2 variable vertices as we discussed before, we obtain the following result.

Corollary 9. $\neg 0ODS$ *is NP-complete for* 3*-regular graphs.*

8. Concluding remarks

In order to solve the complexity questions for *MODS* and *CODS* restricted to bipartite graphs, we introduced and studied the complexity of a new variant ¬0*ODS*. Our main results showed that all three problems are NP-complete when restricted to bipartite graphs. For graphs with bounded treewidth, however, all three problems were shown to be solvable in linear time, by using monadic second order logic. We also studied ¬0*ODS* restricted to other graph classes. By using a collection of different gadgets, we could show that ¬0*ODS* remains NP-complete for a number of graph classes, including planar graphs, graphs with girth at least 5, and graphs in which the maximum degree is bounded by a small constant, in particular also 3-regular graphs. By the nature of the reductions we cannot apply these results to prove complexity results for *MODS* or *CODS* when restricted to planar graphs, graphs with girth at least 5, or graphs with small maximum degree. This implies many open problems with respect to the complexity of *MODS* and *CODS*.

Acknowledgments

We are very grateful to one of the anonymous referees for pointing out several inaccuracies in a previous version of this paper and for supplying several useful observations that simplified some of our proofs.

References

[1] A. Amin, L. Clark, P. Slater, Parity dimension for graphs, Discrete Math. 187 (1998) 1–17.
[2] A. Amin, P. Slater, Neighborhood domination with parity restriction in graphs, Congr. Numer. 91 (1992) 19–30.
[3] A. Amin, P. Slater, All parity realizable trees, J. Combin. Math. Combin. Comput. 20 (1996) 53–63.
[4] R. Barua, S. Ramakrishnan, σ-game, σ^+-game and two-dimensional additive cellular automata, Theoret. Comput. Sci. 154 (1996) 349–366.
[5] J.A. Bondy, U.S.R. Murty, Graph Theory with Applications, Macmillan, London, 1976. Elsevier, New York.
[6] Y. Caro, Simple proofs to three parity theorems, Ars Combin. 42 (1996) 175–180.
[7] Y. Caro, W. Klostermeyer, The odd domination number of a graph, J. Combin. Math. Combin. Comput. 44 (2003) 65–84.
[8] Y. Caro, W. Klostermeyer, J. Goldwasser, Odd and residue domination numbers of a graph, Discuss. Math. Graph Theory 21 (2001) 119–136.
[9] Y. Caro, W. Klostermeyer, R. Yuster, Connected odd dominating sets in graphs, Discuss. Math. Graph Theory 25 (2005) 225–239.
[10] W.Y.C. Chen, X. Li, C. Wang, X. Zhang, The minimum all-ones problem for trees, SIAM J. Comput. 33 (2004) 379–392.
[11] W.Y.C. Chen, X. Li, C. Wang, X. Zhang, Linear time algorithms to the minimum all-ones problem for unicyclic and bicyclic graphs, Electron. Notes Discrete Math. 17 (2004) 93–98.
[12] B. Courcelle, The monadic second-order logic of graphs. I. Recognizable sets of finite graphs, Inform. and Comput. 85 (1990) 12–75.
[13] H. Eriksson, K. Eriksson, J. Sjöstrand, Note on the lamp lighting problem, Adv. Appl. Math. 27 (2001) 357–366.
[14] F. Galvin, Solution to problem 88-8, Math. Intelligencer 11 (1989) 31–32.
[15] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.
[16] J. Goldwasser, W. Klostermeyer, G. Trapp, Characterizing switch-setting problems, Linear Multilinear Algebra 43 (1997) 121–135.
[17] J. Goldwasser, W. Klostermeyer, Maximization versions of "Lights Out" games in grids and graphs, Congr. Numer. 126 (1997) 99–111.
[18] J. Goldwasser, W. Klostermeyer, H. Ware, Fibonacci polynomial and parity domination in grid graphs, Graphs Combin. 18 (2002) 271–283.
[19] M. Halldorson, J. Kratochvíl, J. Telle, Mod-2 independence and domination in graphs, in: Proceedings Workshop on Graph-Theoretic Concepts in Computer Science '99, in: Lecture Notes in Computer Science, 1999, pp. 101–109.
[20] O.P. Lossers, Solution to problem 10197, Amer. Math. Monthly 100 (1993) 806–807.
[21] U. Peled, Problem 10197, Amer. Math. Monthly 99 (1992) 162.
[22] K. Sutner, Additive automata on graphs, Complex Systems 2 (1988) 1–28.
[23] K. Sutner, Problem 88-8, Math. Intelligencer 10 (1988) 101.
[24] K. Sutner, Linear celluar automata and the Garden-of-Eden, Math. Intelligencer 11 (1989) 49–53.
[25] K. Sutner, The σ-game and cellular automata, Amer. Math. Monthly 97 (1990) 24–34.
[26] K. Sutner, σ-automata and Chebyshev-polynomials, Theoret. Comput. Sci. 230 (2000) 49–73.