Information Processing Letters 115 (2015) 134–138

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Circulant graphs and GCD and LCM of subsets

Joachim von zur Gathen [a], Igor E. Shparlinski [b,*]

[a] B-IT, Universität Bonn, 53113 Bonn, Germany
[b] Department of Pure Mathematics, University of New South Wales, Sydney, NSW 2052, Australia

ARTICLE INFO

Article history:
Received 21 February 2014
Received in revised form 14 June 2014
Accepted 25 July 2014
Available online 8 August 2014
Communicated by Ł. Kowalik

Keywords:
Analysis of algorithms
Computational complexity
Graph algorithms
Circulant graphs
Minimum cover

ABSTRACT

Given two sets A and B of integers, we consider the problem of finding a set $S \subseteq A$ of the smallest possible cardinality such the greatest common divisor of the elements of $S \cup B$ equals that of those of $A \cup B$. The particular cases of $B = \emptyset$ and $\#B = 1$ are of special interest and have some links with graph theory. We also consider the corresponding question for the least common multiple of the elements. We establish NP-completeness and approximation results for these problems by relating them to the Set Cover Problem.

1. Introduction

1.1. Description of the problem and motivation

For a nonempty set A of integers, $\gcd(A)$ and $\operatorname{lcm}(A)$ denote the greatest common divisor and the least common multiple of the elements of A, respectively. We consider some questions of how gcd and lcm behave on various subsets S of the original set A.

We are interested in both designing algorithms to construct such sets S with prescribed properties of $\gcd(S)$ and $\operatorname{lcm}(S)$ and also in upper and lower bounds on what one can possibly achieve.

We consider the question of finding a subset $S \subseteq A$ of the smallest possible cardinality with minimal gcd, namely, $\gcd(S) = \gcd(A)$, or with maximal lcm, namely, $\operatorname{lcm}(S) = \operatorname{lcm}(A)$. We also consider a modification of this question where we impose that a specific set B of integers be contained in S. This B may contain elements of A. This question arises in the theory of *circulant graphs* and is a special case of graph *editing problems*, see Damaschke & Molokov [4], Golovachy [6], Mathieson [8], and Mathieson & Szeider [9] for the background and further references.

To explain this connection we recall that an (undirected) *circulant graph* $G(A, m)$ on m nodes, labelled $0, 1, \dots, m-1$, is defined by a set A of integers called *links*, where the nodes i and j are connected if and only if $|i - j| \equiv a \bmod m$ for some $a \in A$. Clearly, $G(A, m)$ is connected if and only if $\gcd(A \cup \{m\}) = 1$. Thus it is natural to ask how many links can at most be removed from A so that the new circulant graph is still connected. This leads to the above question with $B = \{m\}$.

The above can be generalized as follows:

Question 1. Given two sets A and B of positive integers, find a subset $S \subseteq A$ of the smallest possible size with $\gcd(S \cup B) = \gcd(A \cup B)$.

Similarly, we also ask:

[*] Corresponding author.
E-mail addresses: gathen@bit.uni-bonn.de (J. von zur Gathen), igor.shparlinski@unsw.edu.au (I.E. Shparlinski).

http://dx.doi.org/10.1016/j.ipl.2014.07.014
0020-0190/© 2014 Elsevier B.V. All rights reserved.

Question 2. Given two sets A and B of positive integers, find a subset $S \subseteq A$ of the smallest possible size with $\operatorname{lcm}(S \cup B) = \operatorname{lcm}(A \cup B)$.

We first formalize these questions as decision problems.

Problem 3. *Minimum subset with minimal* gcd, MinGcd

Input Sets A and B of positive integers, positive integer k.

Question Does A contain a subset S with $\#S \le k$ and $\gcd(S \cup B) = \gcd(A \cup B)$?

Problem 4. *Minimum subset with maximal* lcm, MaxLcm

Input Sets A and B of positive integers, positive integer k.

Question Does A contain a subset S with $\#S \le k$ and $\operatorname{lcm}(S \cup B) = \operatorname{lcm}(A \cup B)$?

Throughout the paper, we assume that $A \subseteq \mathbb{Z}$ is nonempty when we write $\gcd(A)$ or $\operatorname{lcm}(A)$.

The input size of an instance (A, B) for both MinGcd and MaxLcm is naturally defined as

$$\operatorname{size}(A, B) = \sum_{a \in A \cup B} \lceil \log(a+1) \rceil$$

where $\log z$ denotes the binary logarithm of $z \ge 1$.

We also write

$$\operatorname{size}(A) = \operatorname{size}(A, \emptyset).$$

For each of these (and other similar) problems X, we denote as Opt-X the corresponding optimization problem, where one has to find subsets as described with minimal k.

1.2. Main results

We can now formulate our main results.

Theorem 5. MinGcd *and* MaxLcm *are* NP-*complete.*

Furthermore, a combination of the classical greedy approximation algorithm of Theorem 4 of Johnson [7] and known inapproximability results, see, for example, Theorem 7 of Alon, Moshkovit & Safra [1], yield the following.

Theorem 6. Opt-MinGcd *and* Opt-MaxLcm *can be approximated in polynomial time within a factor* $O(\log \operatorname{size}(A, B))$, *but not within a factor* $o(\log \operatorname{size}(A, B))$ *if* $P \ne NP$.

2. Various reductions

In this section, we produce reductions between various problems that conserve optimal solutions. This allows us to transfer (in)approximability results between these problems, and also provides standard polynomial-time Cook-reductions as used in the theory of NP-completeness.

2.1. Reduction to $B = \emptyset$

We start by constructing from $A, B \subseteq \mathbb{Z}$ a set $A_B \subseteq \mathbb{Z}$ so that

$$\text{Opt-MinGcd}(A, B) = \text{Opt-MinGcd}(A_B, \emptyset).$$

Lemma 7. *For any sets $A, B \subseteq \mathbb{Z}$ one can construct, in polynomial time, a set $A_B \subseteq \mathbb{Z}$ such that $\operatorname{size}(A_B) \le \operatorname{size}(A, B)$ and*

$$\gcd(A \cup B) = \gcd(A_B),$$

and for subset $S \subseteq A$ and $T \subseteq A_B$ of the smallest possible sizes with $\gcd(S \cup B) = \gcd(A \cup B)$ and $\gcd(T) = \gcd(A_B)$, respectively, we have

$$\#S = \#T.$$

Proof. For any integer a, we define the nonnegative integer

$$a_B = \gcd\bigl(\{a\} \cup B\bigr),$$

and apply this element-wise to any $S \subseteq \mathbb{Z}$:

$$S_B = \{a_B : a \in S\}.$$

We claim that for any $S \subseteq A$ we have

$$\gcd(S \cup B) = \gcd(S_B). \tag{1}$$

For any $c \in \mathbb{Z}$, we have

$$\begin{aligned} c \mid \gcd(S \cup B) &\Longleftrightarrow \forall a \in S\ \forall b \in B \quad c \mid a \quad \text{and} \quad c \mid b \\ &\Longleftrightarrow (\forall a \in S\ c \mid a) \quad \text{and} \quad c \mid \gcd(B) \\ &\Longleftrightarrow \forall a \in S \quad c \mid a_B \\ &\Longleftrightarrow c \mid \gcd(S_B). \end{aligned}$$

In particular, we have $\gcd(A \cup B) = \gcd(A_B)$.

Distinct $a \in A$ may yield the same a_B. However, if $S \subseteq A$ has minimal size with $\gcd(S \cup B) = \gcd(A \cup B)$, then $a \mapsto a_B$ is injective on S, and $\#S = \#S_B$. Thus

$$\text{Opt-MinGcd}(A, B) \ge \text{Opt-MinGcd}(A_B, \emptyset).$$

For the reverse direction, we take an "inverse" map $\sigma : A_B \to A$ of $a \mapsto a_B$ on A, namely $\sigma(b) = \min\{a \in A : a_B = b\}$. That is, for every $b \in A_B$, the image $\sigma(b)$ is an element $a \in A$ with $a_B = \gcd(\{a\} \cup B) = b$. For $T \subseteq A_B$ we also define

$$\sigma(T) = \bigcup_{b \in T} \sigma(b).$$

Then $(\sigma(T))_B = T$, since $(\sigma(T))_B \subseteq T$ and for $b \in T$ and $a = \sigma(b)$ we have $b = a_B \in (\sigma(T))_B$. Thus σ is a bijection between T and $\sigma(T)$. For any $T \subseteq A_B$ of minimal size with $\gcd(T) = \gcd(A_B)$, we claim that

$$\gcd\bigl(\sigma(T) \cup B\bigr) = \gcd(A \cup B).$$

This follows from (1), since $(\sigma(T))_B = T$ and

$$\gcd(A \cup B) = \gcd(A_B) = \gcd(T) = \gcd\bigl(\sigma(T) \cup B\bigr).$$

Since $\#\sigma(T) = \#T$, we have

$$\text{Opt-MinGcd}(A, B) \le \text{Opt-MinGcd}(A_B, \emptyset).$$

Overall, it follows that the minimal solution sizes for (A, B) and A_B are equal, and that the solution sets are related by the above correspondence. Clearly the set A_B can be constructed in time polynomial in $\text{size}(A, B)$. □

An almost identical argument implies an analog of Lemma 7 for $\text{lcm}(A, B)$.

Lemma 8. *For any sets $A, B \subseteq \mathbb{Z}$ one can construct in polynomial time a set $A_B \subseteq \mathbb{Z}$ such that $\text{size}(A_B) \le \text{size}(A, B)$ and*

$$\text{lcm}(A \cup B) = \text{lcm}(A_B),$$

and for subset $S \subseteq A$ and $T \subseteq A_B$ of the smallest possible sizes with $\text{lcm}(S \cup B) = \text{lcm}(A \cup B)$ and $\text{lcm}(T) = \text{lcm}(A_B)$, respectively, we have

$$\#S = \#T.$$

Thus both the decision and the optimization versions of the general and the special cases are polynomial-time equivalent.

Lemmas 7 and 8 reduce the general case to the special situation where $B = \emptyset$. Moreover, given a minimum solution set S for one of the two problems, one can easily find a solution for the other one.

So from now on we assume that the input consists of one set A and use $\text{MinGcd}(A)$ and $\text{MaxLcm}(A)$.

2.2. Equivalence of MaxLcm and MinGcd

Here we show that both the decision and the approximation versions of MaxLcm and MinGcd are equivalent. We start with a reduction from MaxLcm to MinGcd.

Lemma 9. *For a set $A = \{a_1, \dots, a_n\}$ of positive integers and $L = \text{lcm}(A)$, we define the set*

$$B = \{L/a_j \colon j = 1 \le j \le n\}.$$

Then for any set $J \subseteq \{1, \dots, n\}$ we have

$$\text{lcm}(\{a_i \colon j \in J\}) = L \Longleftrightarrow \text{gcd}(\{b_i \colon j \in J\}) = 1.$$

Proof. For a prime p and $1 \le j \le n$ let α_j and β_j be the p-adic orders of a_j and b_j, respectively. Then

$$\alpha = \max_{1 \le j \le n} \alpha_j$$

is the p-adic order of L, and $\beta_j = \alpha - \alpha_j$ for all j. For any $J \subseteq \{1, \dots, n\}$ we have

$$\begin{aligned}\max_{j \in J} \alpha_j = \alpha &\Longleftrightarrow \max_{j \in J}\{\alpha_j - \alpha\} = 0\\ &\Longleftrightarrow \min_{j \in J}\{\alpha - \alpha_j\} = 0\\ &\Longleftrightarrow \min_{j \in J} \beta_j = 0.\end{aligned}$$

Denoting the dependence on p by a superscript, we have

$$\begin{aligned}\text{lcm}(\{a_i \colon j \in J\}) = \text{lcm}(A) &\Longleftrightarrow \forall p \quad \max_{j \in J} \alpha_j^{(p)} = \alpha^{(p)}\\ &\Longleftrightarrow \forall p \quad \min_{j \in J} \beta_j^{(p)} = 0\\ &\Longleftrightarrow \text{gcd}(\{b_i \colon j \in J\}) = 1,\end{aligned}$$

which concludes the proof. □

The reverse reduction from MinGcd to MaxLcm goes as follows.

Lemma 10. *For a set $A = \{a_1, \dots, a_n\}$ of positive integers and $d = \text{gcd}(A)$, we define the set*

$$B = \{\text{lcm}(A)/a_j \colon j = 1 \le j \le n\}.$$

Then for any set $J \subseteq \{1, \dots, n\}$ we have

$$\text{gcd}(\{a_i \colon j \in J\}) = d \Longleftrightarrow \text{lcm}(\{b_i \colon j \in J\}) = \text{lcm}(A)/d.$$

Proof. Without loss of generality we may assume that $d = 1$. Then $\text{lcm}(B) = \text{lcm}(A)$.

In the notation of the proof of Lemma 9, we have $\beta_j = \alpha - \alpha_j$ for $1 \le j \le n$. Then,

$$\begin{aligned}&\text{gcd}(\{a_j \colon j \in J\}) = 1\\ &\quad\Longleftrightarrow \forall p \quad \min_{j \in J} \alpha_j^{(p)} = 0\\ &\quad\Longleftrightarrow \forall p \quad \max_{j \in J}\{(\alpha^{(p)} - \alpha_j^{(p)})\} = \alpha^{(p)}\\ &\quad\Longleftrightarrow \forall^p \quad \max_{j \in J} \beta_j^{(p)} = \alpha^{(p)}\\ &\quad\Longleftrightarrow \text{lcm}(\{b_j \colon j \in J\}) = \text{lcm}(B) = \text{lcm}(A),\end{aligned}$$

which concludes the proof. □

Since $\#A = \#B$ in Lemmas 9 and 10, it is enough to prove Theorems 5 and 6 only for MaxLcm and Opt-MaxLcm, respectively.

2.3. The set cover problem

We present polynomial time reductions between MinGcd, MaxLcm and the following problem, SetCover, which is well studied in complexity theory.

Problem 11. SetCover

Input List $\mathcal{C}$ of subsets of a finite set X, positive integer k.

Question Does $\mathcal{C}$ contain a cover for X of size k or less, that is, a subset $\mathcal{D} \subseteq \mathcal{C}$ with $\#\mathcal{D} \le k$ such that every element of X belongs to at least one member of $\mathcal{D}$?

Let n be the input size, which is in $O(\#\mathcal{C} \cdot \#X \cdot \log t)$ if X consists of positive integers less than t. Furthermore Opt-SetCover takes just $(\mathcal{C}, X)$ as input and returns the smallest possible value of k. This task can be approximated in polynomial time within a factor of $O(\log n)$, but no smaller factor (unless $P = NP$), see Alon, Moshkovitz & Safra [1].

It is well known that SetCover is NP-complete, see, for example, Problem SP5 in Section A.3.1 of Garey & Johnson [5]. In the next subsections, we present various reductions between SetCover and MaxLcm. The latter is trivially in NP, and its reduction to SetCover transfers approximation algorithms for the latter to approximation algorithms for MaxLcm. On the other hand, the reduction from SetCover to MaxLcm shows that the latter cannot be approximated too well.

2.4. Coprime bases

The basic tool in this reduction is to compute a *coprime basis* (B, e) of some $A \subseteq \mathbb{Z}$, where $B \subseteq \mathbb{Z}$ is a set of pairwise coprime integers $b \geq 2$ and $e: A \times B \longrightarrow \mathbb{N}$ is such that $a = \prod_{b \in B} b^{e(a,b)}$ for all $a \in A$. By dropping the unneeded elements b where $e(a, b) = 0$ for all $a \in A$ from B, we may assume that

$$\forall b \in B \; \exists a \in A: e(a, b) \geq 1. \qquad (2)$$

Bach & Shallit [2] discuss in their Section 4.8 coprime bases (under the designation of *gcd-free basis*).

2.5. Reduction from MaxLcm to SetCover

```
Input   instance $A \subseteq \mathbb{Z}$ of Opt-MaxLcm.
Output  instance $(\mathcal{C}, B)$ of Opt-SetCover.
1.      Compute a coprime basis $(B, e)$ of $A$ satisfying (2).
2.      For $b \in B$, let $d(b) = \max\{e(a, b): a \in A\}$.
3.      For $a \in A$, let
            $\mathcal{C}_a = \{b \in B: e(a, b) = d(b)\} \subseteq B.$
4.      For $a \in A$, let
            $a^* = \min\{c \in A: \mathcal{C}_a = \mathcal{C}_c\} \in A.$
5.      Set $A^* = \{a^*: a \in A\} \subseteq A$ and $\mathcal{C} = \{\mathcal{C}_c: c \in A^*\}$.
6.      Return $(\mathcal{C}, B)$.
```

Lemma 12. *The above reduction works in time polynomial in* size(A), *and we have*

- Opt-MaxLcm(A) = Opt-SetCover $(\mathcal{C}, B)$;
- *for any $k \geq 1$, transforming (A, k) into $(\mathcal{C}, B, k)$ is a polynomial-time Cook-reduction from* MacLcm *to* SetCover.

Proof. In step 2, we have $d(b) \geq 1$ by (2), and

$$\operatorname{lcm}(A) = \operatorname{lcm}\left(\prod_{b \in B} b^{e(a,b)}: a \in A\right) = \prod_{b \in B} \operatorname{lcm}\left(b^{e(a,b)}: a \in A\right) = \prod_{b \in B} b^{d(b)};$$

see also Corollary 4.8.2 in Bach & Shallit [2] for a less explicit form of this fact. Since $A^* \subseteq A$, $\operatorname{lcm}(A^*)$ divides $\operatorname{lcm}(A)$. On the other hand, for any of the pairwise coprime factors $b^{d(b)}$ making up $\operatorname{lcm}(A)$, b occurs in some $\mathcal{C}_a$ with $a \in A$ and hence also in $\mathcal{C}_{a^*}$, so that $b^{d(b)}$ divides a^*. It follows that $\operatorname{lcm}(A^*) = \operatorname{lcm}(A)$.

Now we consider the SetCover instance with $X = B$ and $\mathcal{C} = \{\mathcal{C}_a: a \in A^*\}$. For $S \subseteq A^*$, we consider $\mathcal{D} = \{\mathcal{C}_a: a \in S\}$. Then $\#\mathcal{D} = \#S$, and

$$\begin{aligned} \operatorname{lcm}(S) = \operatorname{lcm}(A^*) &\Longleftrightarrow \forall b \in B \quad b^{d(b)} \mid \operatorname{lcm}(S) \\ &\Longleftrightarrow \forall b \in B \; \exists a \in S \quad b^{d(b)} \mid a \\ &\Longleftrightarrow \forall b \in B \; \exists a \in S \quad e(a, b) = d(b) \\ &\Longleftrightarrow \forall b \in B = X \; \exists a \in S \quad b \in \mathcal{C}_a \\ &\Longleftrightarrow \mathcal{D} \text{ covers } X. \end{aligned}$$

Thus a solution S of MaxLcm with $\#S \leq k$ implies one of SetCover with $\#\mathcal{D} \leq k$.

Conversely, given a cover $\mathcal{D} = \{\mathcal{C}_a: a \in S\}$ of X with $\#\mathcal{D} \leq k$ we conclude that $\#S \leq k$, since the sets $\mathcal{C}_a$ for $a \in A^*$ are pairwise distinct.

Thus the smallest size of a set $S \subseteq A$ with $\operatorname{lcm}(S) = \operatorname{lcm}(A)$ and the smallest size of a cover $\mathcal{D}$ of X coincide. This shows that the input and output problems have the same optimal solution, and also establishes the claimed reduction between the decision problems. The claim of polynomial time in step 1 follows from Bach & Shallit [2] who show in their Section 4.8 that it can be computed with $O(\operatorname{size}(A)^2)$ bit operations, where, as before, size(A) is the input size. They use classical arithmetic. According to [3], fast arithmetic yields an algorithm using $\operatorname{size}(A)(\log \operatorname{size}(A))^{O(1)}$ operations. The other steps are simple sorting and selection procedures that can be done in polynomial time. □

By the above, the size of B is polynomial in that of A. We note that the size of B can actually be much smaller than that of A: take the first m primes, all exponent vectors e in $\{1, 2\}^m$, and then all 2^m values $a_e = \prod_{1 \leq i \leq m} p_i^{e_i}$. Then the coprime basis B consists of just these m primes and size(A) is only logarithmic in size(B). That is no worry, since we only use this reduction to derive good approximations for SetCover (which do not exist by the hardness result mentioned above) from good approximations to our problems; hence the latter do not exist either.

2.6. Reduction from SetCover to MaxLcm

```
Input   instance $(\mathcal{C}, X)$ of Opt-SetCover, with $X = \{1, \ldots, m\}$ and
        $\mathcal{C} = \{C_1, \ldots, C_l\}$ with each $C_i \subseteq X$.
Output  instance $A \subseteq \mathbb{Z}$ of Opt-MaxLcm.
1       Let $p_1 < p_2 < \cdots < p_m$ be the first $m$ prime numbers.
2       Set $a = \prod_{j \in X} p_j$ and $a_i = \prod_{j \in C_i} p_j$ for $i \leq l$.
3       Return $A = \{a_1, \ldots, a_l\}$.
```

Lemma 13. *The reduction of Section 2.6 works in time polynomial in* size$(\mathcal{C}, X)$, *and we also have*

- Opt-SetCover$(\mathcal{C}, X)$ = Opt-MaxLcm(A)
- *for any $k \geq 1$, transforming $(\mathcal{C}, B, k)$ into (A, k) is a polynomial-time Cook-reduction from* SetCover *to* MacLcm.

Proof. We may assume that $X = \bigcup_{1 \le i \le l} C_i$. Then $a = \operatorname{lcm}(A)$.

Suppose that $I \subseteq \{1, \ldots, m\}$ is such that $\operatorname{lcm}(S) = \operatorname{lcm}(A)$, where $S = \{a_i : i \in I\}$. Let

$$\mathcal{D} = \{C_i : i \in I\}.$$

Then for any $j \in X$, p_j divides $\operatorname{lcm}(A) = \operatorname{lcm}(S)$ and hence a_i for some $i \in I$. It follows that $j \in C_i \in \mathcal{D}$. Thus $\mathcal{D}$ covers X.

On the other hand, suppose that $I \subseteq \{1, \ldots, m\}$ is such that $\mathcal{D} = \{C_i : i \in I\}$ covers X. Then $S = \{a_i : i \in I\}$ satisfies $\operatorname{lcm}(S) = a = \operatorname{lcm}(A)$.

In both transformations above, the sizes of the sets are conserved.

Since $p_m = (1 + o(1)) m \ln m$ as $m \to \infty$, the bit size of A is in $O(lm \log m)$. The set A can be computed in time polynomial in lm, using the sieve of Eratosthenes for generating the primes. □

Thus both the decision and the optimization versions of MaxLcm are transformed to those of SetCover.

2.7. Concluding the proofs

We start with the upper bounds claimed in Theorems 5 and 6. The fact that MaxLcm is in NP is trivial. Furthermore, Lemmas 7, 8, 10, and 12 show that the known approximation algorithms for SetCover also yield ones for our problem.

Furthermore, our claimed lower bounds (NP-hardness and inapproximability) follow from Lemmas 7, 8, 9, and 13, together with the NP-hardness and inapproximability of SetCover, as cited above.

Acknowledgements

The authors are grateful to Fedor Fomin for some references on the approximability and inapproximability of the set cover problem.

The first author's work was supported by the B-IT Foundation and the Land Nordrhein-Westfalen. The second author's work was supported by the Australian Research Council grants DP110100628 and DP140100118.

References

[1] Noga Alon, Dana Moshkovitz, Shmuel Safra, Algorithmic construction of sets for k-restrictions, ACM Trans. Algorithms 2 (2006) 153–177.

[2] Eric Bach, Jeffrey Shallit, Algorithmic Number Theory, vol. 1: Efficient Algorithms, MIT, Cambridge-MA, 1996.

[3] Daniel J. Bernstein, Factoring into coprimes in essentially linear time, J. Algorithms 54 (2005) 1–30.

[4] Peter Damaschke, Leonid Molokov, Parameterized reductions and algorithms for a graph editing problem that generalizes vertex cover, Theor. Comput. Sci. 452 (2012) 39–46.

[5] Michael R. Garey, David S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Co., San Francisco CA, 1979.

[6] Petr A. Golovachy, Editing to a connected graph of given degrees, preprint, available from http://arxiv.org/abs/1308.1802, 2013.

[7] David S. Johnson, Approximation algorithms for combinatorial problems, J. Comput. Syst. Sci. 9 (1974) 256–278.

[8] Luke Mathieson, The parameterized complexity of editing graphs for bounded degeneracy, Theor. Comput. Sci. 411 (2010) 34–36.

[9] Luke Mathieson, Stefan Szeider, Editing graphs to satisfy degree constraints: a parameterized approach, J. Comput. Syst. Sci. 78 (2012) 179–191.