Manipulating Citation Indices in a Social Context

Chrystalla Pavlou
University of Edinburgh
Edinburgh, United Kingdom
C.Pavlou@sms.ed.ac.uk

Edith Elkind
University of Oxford
Oxford, United Kingdom
elkind@cs.ox.ac.uk

ABSTRACT

The h-index [6] is a popular measure of a researcher's publication activity: a researcher's h-index is the largest number x such that she has at least x papers that have received at least x citations each. It has been observed that one can manipulate her h-index by strategically merging one or more articles, and the complexity of finding a successful/optimal manipulation has been investigated for a variety of models [3, 11]. In this paper, we extend this line of research to two other popular citation indices, namely, the g-index [4] and the i10-index, and show that these indices are somewhat easier to manipulate than the h-index. We then consider settings where the manipulator would like to take into account the impact of her actions on other researchers (she may want to make sure that her manipulation does not harm her friends or that it hurts her competitors) or a group of researchers manipulate their indices simultaneously. We analyze the complexity of these problems, both in the worst-case and in the parameterized framework.

Keywords

h-index; g-index; i10-index; manipulation by merging

1. INTRODUCTION

Measuring a researcher's productivity is a notoriously difficult task: typically, hiring and promotion decisions are made based on a multitude of factors. However, many large organizations employ simple one-dimensional numerical measures of productivity as a primary filter. The perils of using the number of publications in this context are evident, and it has been argued that the total number of citations is not a good measure either, as it can be skewed by surveys and popular textbooks. Motivated by these considerations, Hirsch [6] proposed in 2005 a new citation index, which became known as the h-index: the h-index of a researcher is the largest number x such that she has at least x articles that receive at least x citations each. For instance, if Alice has 9 articles, cited by, respectively, $32, 25, 13, 8, 6, 3, 3, 2, 1$ other articles, then her h-index is 5.

This index has become very well known, and is reported by major bibliometric engines, including Google Scholar. It was covered in popular press [1], is widely discussed on blogs (see, e.g., [2, 8, 9] for a sample), and has been analyzed from an axiomatic perspective [13]. However, as it is becoming popular with administrators at universities and funding agencies, it is now evident that one needs to understand whether this index can be abused. Indeed, one of Hirsch's original motivations for proposing the h-index was that it is less susceptible to manipulation by self-citations than the total citation count: in the example above, if Alice were to write a new article that cites all of her previous articles, this would increase her citation count by 9, but her h-index would remain the same.

However, there exists a tool that is intended for legitimate purposes, yet can be quite efficient in increasing one's h-index: merging articles. Specifically, Google Scholar enables the authors to merge two or more articles into a single article. The reason for this is that Google Scholar identifies one's publications automatically by crawling the web, so if Alice's article has been posted on arXiv, appeared in a couple of workshops, was published in a conference and finally in a journal, Google Scholar would list this article several times. Alice can then merge all these entries of her Google Scholar profile and designate the representative version of her article, so that other researchers who are looking for it can identify the definitive version of her work. The citations of the merged article are computed from the citations received by its component articles (a naive approach is just to add up the citations counts of all versions; more sophisticated approaches discussed below take into account that another article may cite more than one of the component articles, and the component articles themselves may cite each other), and it is possible that the merge increases Alice's h-index. In our running example, if the citation count is updated naively, Alice can merge the two articles with 3 citations each to increase her h-index to 6.

Therefore, Alice may be tempted to merge several unrelated articles in order to increase her h-index. Google Scholar does not prevent her from doing so: she can merge articles with very different titles and groups of authors (provided she is an author on each of them). There is a good reason for this policy: for instance, if two groups of authors write two independent papers on a related topic, and then join forces for a journal version, they may want to merge all three articles on Google Scholar even though the two conference papers appear unrelated (in this case, the authors would have to proceed in two steps). There is also no limit on the number of articles one can merge, and, as argued above, it is not unusual for a paper to have 5 or 6 versions. Thus, there is very little to prevent a dishonest researcher from trying to find an optimal way of partitioning her articles into a few highly cited "super-articles"—except perhaps the sheer complexity of this task.

Indeed, recently de Keijzer and Apt [3] have demonstrated that, in the naive model, there is a polynomial-time algorithm for determining if one can increase their h-index by merging articles, but maximizing one's h-index in this way is NP-hard. In a follow-up paper, van Bevern et al. [11] consider the same questions in a

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

richer and more realistic model. Specifically, they work with a directed graph whose vertices are articles and whose arcs represent citations, and consider three ways of calculating the number of citations to a merged article, which they call sumCite, unionCite, and fusionCite. Briefly, sumCite computes the sum of indegrees of the articles within a merged article (i.e., this is exactly the naive measure discussed above), unionCite counts the number of distinct articles citing the components of the merged article (this is the measure currently used by Google Scholar), and fusionCite is derived by modifying unionCite so as not to double-count the citations by component articles of a merged article (see Section 2 for formal definitions). Van Bevern et al. also model the fact that a researcher may want to hide her manipulation and avoid merging obviously unrelated articles. To this end, they introduce the notion of a *compatibility graph* G, which indicates which articles can be merged, and require that each merged article corresponds to a clique in this graph. They then study the complexity of increasing/maximizing one's h-index under each of their citation measures (i.e., sumCite, unionCite, and fusionCite), both for general compatibility graphs and for the case where G is a clique. In addition to considering the worst-case complexity of these problems, they explore their parameterized complexity with respect to several natural parameters.

1.1 Our Contribution

Against this background, the contribution of our paper is twofold.

First, following the suggestion of van Bevern et al. [11], we extend the study of manipulation by merging to two other popular impact indices: the g-index and the i10-index. The g-index is the largest number x such that the researcher has at least x papers that in total receive at least x^2 citations. This index was proposed by Egghe [4] in 2006; subsequently, Woeginger [12] proposed an axiomatization for this index. The g-index behaves similarly to the h-index, but is more flexible in that it rewards a researcher both for producing a significant number of reasonably well-cited articles (as the h-index does) and for producing a small number of very highly cited articles and then remaining active in research. The i10-index is simply the number of articles with at least 10 citations; it is computed by Google Scholar, and can be seen as a very crude, but simple measure of one's impact.

We show that both of these indices are somewhat easier to manipulate than the h-index: all computational questions that are easy for the h-index remain easy for the g-index and the i10-index, and some problems that were NP-hard for the h-index become easy for the g-index and the i10-index. This provides an argument in favor of the h-index, as resistance to manipulation is a desirable quality.

Second, we investigate the settings where the manipulating researcher has to take into account the impact of her actions on other researchers. We observe that when Alice merges some of her articles, this may change Bob's citation index, either because Bob has written joint articles with Alice, or because his work is cited by Alice. (Our model assumes that when Alice merges two articles that she co-authored with Bob, these articles are automatically merged in Bob's profile. While this is currently not the case in Google Scholar, we believe that for non-strategic researchers, who probably constitute the overwhelming majority of authors, this would be a desirable feature: when Alice merges two articles, she thereby indicates that they are closely related, and this valuable information should be propagated through the system). The index of Bob may go up or down. If Bob is a friend of Alice, then Alice may decide not to merge her articles if the impact on Bob is negative: for instance, if Bob is a PhD student of Alice who is about to graduate, Alice may choose to postpone a legitimate merge, so as not to hurt Bob's chances in the job market. In contrast, if Alice and Bob are competing for the same position or award, Alice may try to maliciously merge her papers so as to lower Bob's index as much as possible (without lowering her own index). If an entire group of researchers is evaluated (as in, for instance, UK's Research Excellence Framework or Italy's Research Assessment Program), the researchers in the group may attempt to manipulate their indices simultaneously, with every researcher trying to achieve a value that is appropriate for his or her career stage. Thus, we consider the complexity of manipulating citation indices in a *social context*. Our results indicate that having to account for friends or enemies makes the manipulator's task harder: we get NP-hardness results even for very simple settings. However, some of our polynomial-time and FPT algorithms can be extended to this model.

2. PRELIMINARIES

We start by formally defining the citation indices that will be discussed in this paper. For readability, the definitions below are formulated in terms of articles and citations; however, they treat each article simply as a number, so all three indices can be defined for any multiset of non-negative integers.

DEFINITION 1. *A researcher's* h-index *is the largest natural number x such that she has at least x articles that were cited by at least x articles each. Her* g-index *is the largest natural number x such that she has x articles that in total receive at least x^2 citations. Her* i10-*index is simply the number of her articles that are cited at least* 10 *times.*

The value 10 in the definition of the i10-index can be replaced by any other natural number θ; we will refer to the resulting index as the iθ-*index*. For $I \in \{h, g, i10\}$, we denote the I-index of a researcher with the set of articles W by $I(W)$.

In what follows, we will mostly use the model of van Bevern et al. [11]. In this model, citations are represented by a directed acyclic graph $D = (V, A)$, where the vertices are articles and there is an arc (u, v) if article u cites article v. For each $v \in V$ and a subset of vertices $S \subseteq V$, we set $N_S^{\text{in}}(v) = \{u \mid u \in S, (u, v) \in A\}$ and $\deg_S^{\text{in}}(v) = |N_S^{\text{in}}(v)|$. The articles of each researcher correspond to a subset W of V, and the researcher can manipulate her citation indices by merging arbitrary subsets of articles in W. The result of a merge operation is a partition $\mathcal{P}$ of W, where a *merged article* is a part with cardinality at least 2. We will sometimes refer to articles in V as *atomic articles*, and, given a merged article $P \in \mathcal{P}$, we will refer to elements of P as *component articles* of P. Van Bevern et al. [11] consider three methods to compute the number of citations to a merged article P in a partition $\mathcal{P}$ of W:

- *sumCite*: the number of citations to a merged article P is set to be equal to the sum of citation counts of its component articles:

$$\mathsf{sCite}(P) = \sum_{v \in P} \deg_D^{\text{in}}(v).$$

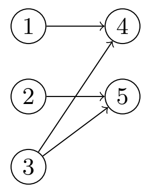
- *unionCite*: the number of citations to a merged article P is set to be equal to the number of distinct articles that cite at least one of its component articles:

$$\mathsf{uCite}(P) = \left| \bigcup_{v \in P} N_D^{in}(v) \right|.$$

- *fusionCite*: the number of citations to a merged article P is set to be equal to the sum of two quantities: (1) the number of distinct articles in $V \setminus W$ that cite at least one component article of P and (2) the number of merged articles in $\mathcal{P}$ whose component articles cite P. Formally, for each $P' \in \mathcal{P} \setminus \{P\}$ we set $\delta(P', P) = 1$ if there exist some $v \in P'$, $w \in P$ such that $(v, w) \in A$ and $\delta(P', P) = 0$ otherwise, and let

$$\mathsf{fCite}(P) = |\bigcup_{v \in P} N^{in}_{V \setminus W}(v)| + \sum_{P' \in \mathcal{P}, P' \neq P} \delta(P', P).$$

Intuitively, these three citation measures correspond to different ways of treating parallel edges in the citation graph that arises after the merge, with sCite being the most generous and fCite being the stingiest. This is illustrated by the following example.



EXAMPLE 1. Consider the citation graph in the figure on the right. Suppose that a researcher decides to merge article 1 with article 2, and article 4 with article 5. It can be verified that for the merged article $P = \{4, 5\}$ we have $\mathsf{sCite}(P) = 4$, $\mathsf{uCite}(P) = 3$, $\mathsf{fCite}(P) = 2$. By scaling up this example appropriately, we can make the gap between sCite and uCite and between uCite and fCite arbitrarily large.

We note that, while the formal definition of fCite may appear to be complex, this is arguably the most intuitive citation measure: effectively, it recomputes the citation graph treating each merged article as a singleton.

To compute an I-index, $I \in \{h, g, i10\}$, of a partition $\mathcal{P}$ with respect to a measure $\mu \in \{\mathsf{sCite}, \mathsf{uCite}, \mathsf{fCite}\}$, we apply the usual procedure for computing this index to the multiset of non-negative integers $\{\mu(P) \mid P \in \mathcal{P}\}$; we denote the result by $I(\mathcal{P}, \mu)$.

In practice, a researcher who wants her manipulative actions to remain undetected may avoid merging obviously unrelated articles. To capture this, van Bevern et al. [11] introduce another graph on V, which they call the *compatibility graph*. This is an undirected graph $G = (V, E)$; an edge $\{u, v\} \in E$ indicates that u and v can be merged. This graph constrains the space of possible partitions of W: a partition $\mathcal{P}$ *complies with* G if each merged article $P \in \mathcal{P}$ is a clique in G. The case where arbitrary merges are allowed corresponds to G being a complete graph.

When considering an arbitrary graph H, we write $V(H)$ to denote the set of vertices of H and $E(H)$ to denote the set of edges of H.

3. COMPUTATIONAL PROBLEMS

We will now state the computational problems that will be studied in this paper. In what follows, for brevity we identify a researcher with her set of articles, i.e., instead of saying "the citation index of a researcher with the set of articles W", we say "the citation index of the set of articles W".

The most basic question one can ask in this framework is whether a researcher can improve her citation index at all; more ambitiously, one can ask whether a certain value of the index can be achieved. These questions can be asked for each $I \in \{h, g, i10\}$ and each $\mu \in \{\mathsf{sCite}, \mathsf{uCite}, \mathsf{fCite}\}$, and are captured by the following two families of problems.

(μ, I)-INDEXIMP
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$ and a set $W \subseteq V$ of articles.
Question: Is there a partition $\mathcal{P}$ of W that complies with G and satisfies $I(\mathcal{P}, \mu) > I(W)$?

(μ, I)-INDEXACH
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$, a set $W \subseteq V$ of articles and a non-negative integer t.
Question: Is there a partition $\mathcal{P}$ of W that complies with G and satisfies $I(\mathcal{P}, \mu) \geq t$?

For the h-index, these problems have been studied in previous work. Specifically, de Keijzer and Apt [3] considered the case $\mu = \mathsf{sCite}$ under the assumption that G is a complete graph, and van Bevern et al. [11] considered all three citation measures and general compatibility graphs. Thus, in what follows, we will only analyze them for $I \in \{g, i10\}$.

REMARK 1. Note that the only information about the citation graph D that is used to compute $I(\mathcal{P}, \mathsf{sCite})$, $I \in \{h, g, i10\}$, is the indegree of the nodes in W. Thus, for $\mu = \mathsf{sCite}$ we can represent the input by omitting the graph D and listing the citation counts of each article in W. When V is exponentially larger than W, this representation (which is used by de Keijzer and Apt [3]) is exponentially more succinct. Thus, a polynomial-time algorithm for the graph-based representation that needs to look at each article citing a given article is only a pseudopolynomial-time algorithm for the list-based representation. Conversely, a weak NP-hardness result for the list-based representation (e.g., one obtained by a reduction from PARTITION [5]) would not imply a hardness result for the graph-based representation. Fortunately, this distinction turns out to be immaterial: the efficient algorithms in our and previous work run in polynomial time given the list-based representation, and the hardness results are strong NP-hardness results for the list-based representation (and hence imply that the respective problems remain NP-hard under the graph-based representation).

The next type of computational problems we consider deals with maximizing one's citation index without lowering the index of another researcher (a friend), or jointly maximizing the indices of a group of friends.

(μ, I)-INDEXIMPFRIEND
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$ and two sets of articles $W_1 \subseteq V, W_2 \subseteq V$.
Question: Is there a partition $\mathcal{P}$ of W_1 that complies with G such that $I(\mathcal{P}, \mu) > I(W_1)$ and merging the articles according to $\mathcal{P}$ does not lower the I-index of W_2 with respect to μ?

(μ, I)-INDEXACHFRIEND
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$, two sets of articles $W_1 \subseteq V, W_2 \subseteq V$ and a nonnegative integer t.
Question: Is there a partition $\mathcal{P}$ of W_1 that complies with G such that $I(\mathcal{P}, \mu) \geq t$ and merging the articles according to $\mathcal{P}$ does not lower the I-index of W_2 with respect to μ?

We note that, in general, the manipulator is allowed to merge an article in $W_1 \cap W_2$ with an article in $W_1 \setminus W_2$; the discussion in the introduction suggests that merges of this form may be legitimate, and if they are not, we expect this information to be captured by G.

To model multiple researchers manipulating their citation indices simultaneously, we need a precise definition of what merges are legal. To illustrate the difficulty, consider the situation where Alice and Bob each have a single-author paper as well as a joint paper. In Google Scholar they are allowed to merge all three papers (indeed, the joint paper may be based on combining two related single-author conference papers into a journal paper): first, Alice merges her single-author paper with the joint paper, and then Bob merges his paper with this combined paper. However, this means that a merge can no longer be described as a partition of one's papers, and in particular a paper where Alice is not a co-author may contribute to her citation index.

To handle this, we propose the following formalism. Given a set of articles V and k subsets of articles $W_1 \subseteq V, \ldots, W_k \subseteq V$

we define the *mergeability graph* $M = (V, L)$, where $\{u, v\} \in L$ if and only if $u, v \in W_i$ for some $i = 1, \ldots, k$. We say that a partition $\mathcal{P}$ of $W_1 \cup \ldots \cup W_k$ is *admissible* if each part of $\mathcal{P}$ is connected in M. For $i = 1, \ldots, k$, let $I(i, \mathcal{P}, \mu)$ be the I-index of the set of non-negative integers $\{\mu(P) \mid P \in \mathcal{P}, P \cap W_i \neq \emptyset\}$. We can now define our computational problem.

(μ, I, k)-INDEXMACH
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$, k sets of articles $W_1 \subseteq V, \ldots, W_k \subseteq V$ and nonnegative integers $t_1, \ldots, t_k$.
Question: Is there an admissible partition $\mathcal{P}$ of $W_1 \cup \ldots \cup W_k$ that complies with G and satisfies $I(i, \mathcal{P}, \mu) \geq t_i$ for $i = 1, \ldots, k$?

Conversely, a researcher may want to manipulate her citation index in the presence of a competitor. Then her goal may be to increase/maximize her index while not increasing the index of another researcher, or, alternatively, to lower the competitor's index without lowering her own.

(μ, I)-INDEXIMPENEMY
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$ and two sets of articles $W_1 \subseteq V, W_2 \subseteq V$.
Question: Is there a partition $\mathcal{P}$ of W_1 that complies with G such that $I(\mathcal{P}, \mu) > I(W_1)$ and merging the articles according to $\mathcal{P}$ does not increase the I-index of W_2 with respect to μ?

(μ, I)-INDEXACHENEMY
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$, two sets of articles $W_1 \subseteq V, W_2 \subseteq V$ and a nonnegative integer t.
Question: Is there a partition $\mathcal{P}$ of W_1 that complies with G such that $I(\mathcal{P}, \mu) \geq t$ and merging the articles according to $\mathcal{P}$ does not increase the I-index of W_2 with respect to μ?

(μ, I)-INDEXHARMENEMY
Input: A citation graph $D = (V, A)$, a compatibility graph $G = (V, E)$ and two sets of articles $W_1 \subseteq V, W_2 \subseteq V$.
Question: Is there a partition $\mathcal{P}$ of W_1 that complies with G such that $I(\mathcal{P}, \mu) \geq I(W_1)$ and merging the articles according to $\mathcal{P}$ lowers the I-index of W_2 with respect to μ?

REMARK 2. Some of the computational problems introduced so far are related to each other. In particular, the improvement problem (with or without the social context) is a special case of the respective achievability problem that corresponds to setting $t = I(W) + 1$. By the same argument, INDEXMACH is at least as hard as INDEXACHFRIEND. Furthermore, it can be checked that all our hardness proofs for $\mu = \mathsf{sCite}$ can be adapted to show hardness for $\mu = \mathsf{uCite}$ and hardness proofs for $\mu = \mathsf{uCite}$ can be adapted to show hardness for $\mu = \mathsf{fCite}$. Also, if a problem is hard for a single researcher, it clearly remains hard in a social context (note, however, that INDEXHARMENEMY is *not* a generalization of any of our single-researcher problems). Therefore, in what follows, once we prove a hardness result for an "easier" problem, we do not discuss the "harder" problems separately. Conversely, once we obtain an algorithm for a "harder" problem, we do not discuss the "easier" problems.

All of the problems listed above are in NP, so, in what follows, to prove that any of them is NP-complete, it is sufficient to show that it is NP-hard. To prove our hardness results, we will use reductions from the following well-known NP-hard problems.

CLIQUE [5]
Input: An n-vertex graph H and a positive integer k with $k \leq n$.
Question: Does H contain a clique of size at least k?

INDEPENDENT SET [5]
Input: An n-vertex graph H and a positive integer $k \leq n$.
Question: Does H have an independent set of size at least k?

3D MATCHING [5]
Input: A tuple (X, Y, Z, M), where X, Y and Z are disjoint sets of size n each and $M \subset X \times Y \times Z$.
Question: Does M contain a subset Q of size n such that for every pair of triples $(x, y, z), (x', y', z') \in Q$ it holds that $x \neq x'$ and $y \neq y'$ and $z \neq z'$?

SPECIAL 3-PARTITION [5, 3]
Input: Positive integers m and b, where b is bounded by a polynomial in m, $p(m)$, and a multiset S of $3m$ positive integers in $(\frac{b}{4}, \frac{b}{2})$ with $\sum_{x \in S} x = mb$.
Question: Is there a partition of S into m submultisets of cardinality 3 each such that the numbers in each submultiset sum up to exactly b?

4. MANIPULATION OF THE G-INDEX: ONE RESEARCHER

Interestingly, the g-index turns out to be more susceptible to manipulation than the h-index: we can show that (sCite, g)-INDEXACH is polynomial-time solvable as long as G is a clique; in contrast, (sCite, h)-INDEXACH is known to be NP-hard in this case [3].

PROPOSITION 1. (sCite, g)-INDEXACH *is solvable in polynomial time if the compatibility graph G is a clique.*

PROOF. Fix an instance (D, G, W, t) of (sCite, g)-INDEXACH. Suppose that $|W| = w$ and the total number of citations to articles in W is s. If $w < t$ or $s < t^2$ then clearly it is not possible to improve the researcher's g-index to t. Otherwise, pick some $w - t + 1$ articles in W, and merge them into a single article. After the merge, the researcher has exactly t articles, and his total number of citations is $s \geq t^2$. □

However, we obtain hardness results even for INDEXIMP if G is not a clique or if we use more sophisticated citation measures.

THEOREM 2. (sCite, g)-INDEXIMP *is* NP*-complete.*

PROOF. We give a reduction from CLIQUE. Given an instance (H, k) of CLIQUE, we build an instance (D, G, W) of (sCite, g)-INDEXIMP as follows. We set $W_1 = V(H)$, let W_2 consist of $k-1$ new articles, and set $W = W_1 \cup W_2$. To create the citation graph D, for each article $w \in W_2$ we add k articles that cite w and no other article, and for each article $w \in W_1$ we add one article that cites w and no other article. The compatibility graph G is created from H by adding each article in W_2 as an isolated vertex. We now show that (D, G, W) is a yes-instance of (sCite, g)-INDEXIMP if and only if (H, k) is a yes-instance of CLIQUE.

Suppose that (H, k) is a yes-instance of CLIQUE. Then H contains a clique T of size at least k. Merging the articles in $T \subseteq W_1$ complies with G, and hence we obtain a merged article with k citations. Together, with the $k - 1$ articles in W_2, this results in k articles with k citations, i.e., a partition with g-index k.

On the other hand, let (D, G, W) be a yes-instance of (sCite, g)-INDEXIMP. If $\mathcal{P}$ is an improving partition of W, it has at least one merged article P with at least k citations. As only articles in W_1 can be merged and the merge has to comply with G, it follows that $G[P] = H[P]$ is a clique. Since each vertex in W_1 has only one citation, we have $|P| \geq k$, and the proof is complete. □

THEOREM 3. (uCite, g)-INDEXIMP *is* NP*-complete even if the compatibility graph G is a clique.*

PROOF. We give a reduction from INDEPENDENT SET. Given an instance (H, k) of INDEPENDENT SET, we assume without loss of generality that $|V(H)| < |E(H)|$ and let $m = |E(H)|$, and construct an instance (D, G, W) of (uCite, g)-INDEXIMP as follows. We set $W = W_1 \cup W_2 \cup W_3$, where $W_1 = V(H)$, W_2 consists of $m - k$ (without loss of generality we assume $m > k$) new articles and W_3 consists of k new articles. To create the citation graph D, we add a new article w that cites all articles in W_1 and W_2. Then, for each edge $\{u, v\}$ in $E(H)$, we add a new article $e_{\{u,v\}}$ that cites u and v as well as all articles in W_2 and W_3. As a result, in W_2 we have $m - k$ articles with the same $m + 1$ citations, and in W_3 we have k articles with the same m citations; the articles in W_1 have at most m citations each. Finally, we set G to be a clique. We observe that the g-index of W is m, so the aim is to improve it to at least $m + 1$. We now show that (D, G, W) is a yes-instance of (uCite, g)-INDEXIMP if and only if (H, k) is a yes-instance of INDEPENDENT SET.

Suppose that (H, k) is a yes-instance of INDEPENDENT SET. Then H contains an independent set T of size k. As T is an independent set, $W_1 \setminus T$ is a vertex cover of H. Thus, by merging the articles in $W_1 \setminus T$, we get a merged article with $m + 1$ citations. Then, by pairing up articles in T and W_3 and merging the articles in each pair, we get k articles with $m + 1$ citations each. Together with the $m - k$ articles in W_2, we have $m + 1$ articles with $m + 1$ citations each. Hence, we have a partition with g-index $m + 1$.

On the other hand, suppose that (D, G, W) is a yes-instance of (uCite, g)-INDEXIMP. That is, we have $m+1$ articles with at least $(m + 1)^2$ total citations in an improving partition $\mathcal{P}$. Since there are $m + 1$ distinct articles citing articles in W, no merged article can obtain more than $m + 1$ citations. Consequently, in $\mathcal{P}$ there are $m + 1$ articles that get exactly $m + 1$ citations each. Since each article in W_2 gets $m + 1$ citations, we can assume that every such article is a singleton in $\mathcal{P}$. Further, since $|W_2| + |W_3| = m$, there has to be at least one merged article P with $P \subseteq W_1$. For P to get $m + 1$ citations, it must correspond to a vertex cover of H, so its complement $W_1 \setminus P$ is an independent set. Now, suppose that $|W_1 \setminus P| < k$, and hence $|W_3| + |W_1 \setminus P| < 2k$. But then any partition of $W_3 \cup (W_1 \setminus P)$ into at least k parts would contain fewer than k non-singleton parts, and every singleton part of this partition receives at most m citations. Thus, altogether we would have at most $|W_2|+1+(k-1) < m+1$ parts with $m+1$ citations, a contradiction. Hence, $W_1 \setminus P$ is an independent set of size at least k, and the proof is complete. □

5. MANIPULATION OF THE I10-INDEX: ONE RESEARCHER

Both the h-index and the g-index are hard to manipulate when there are constraints on which articles can be merged (i.e., G is not a clique). In contrast, for the i10-index we can obtain some easiness results even for this case.

PROPOSITION 4. $(\mu, i10)$-INDEXIMP *is solvable in polynomial time for* $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$.

PROOF. We can safely ignore all articles with 0 citations, and there is no reason to touch articles that already receive 10 or more citations. Now, let W' be the set of all articles in W that have between 1 and 9 citations. We can enumerate all subsets of W' of size at most 10 in time polynomial in $|W|$. For each subset P, we check whether it corresponds to a clique in G and whether $\mu(P) \geq 10$. If so, we can merge the articles in P to improve the researcher's i10-index. The correctness of this algorithm follows from the fact that there is no need to use more than 10 articles to get 10 citations (to see why this is the case for uCite, consider starting with a set $S \subseteq W$ with $\mu(S) \geq 10$, $|S| > 10$, moving articles from S to P one by one, and discarding an article if it does not contribute to P's citation count). □

The algorithm described in Proposition 4 does not work for fCite: under this measure, merging "small" articles may lower the citation count of "large" articles. Indeed, the following theorem shows that $(\mathsf{fCite}, i10)$-INDEXIMP is NP-hard.

THEOREM 5. $(\mathsf{fCite}, i10)$-INDEXIMP *is* NP-*complete.*

PROOF. We give a reduction from 3D MATCHING. Given an instance (X, Y, Z, M) of 3D MATCHING with $|X| = |Y| = |Z| = n$, we build an instance (D, G, W) of $(\mathsf{fCite}, i10)$-INDEXIMP as follows. Let $W_1 = X \cup Y \cup Z$, let W_2 consist of four new articles, let W_3 consist of n new articles, and set $W = W_1 \cup W_2 \cup W_3$. To create the citation graph D, for each article $w \in W_1$ we add one article that cites w and no other article, and for each article $w \in W_2$ we add three articles that cite w and no other article. For each article w in W_3 we introduce six articles that cite w and no other article, and we also add citations from each article in W_2 to w. Thus, each article in W_3 has 10 citations, so the i10-index of W is n. The compatibility graph G contains edges $(x, y), (y, z)$ and (x, z) for each triple $(x, y, z) \in M$, as well as the edges between all pairs of articles in W_2 and the edges $\{x, y \mid x \in W_3, y \in W_1\}$. We now show that (D, G, W) is a yes-instance of $(\mathsf{fCite}, i10)$-INDEXIMP if and only if (X, Y, Z, M) is a yes-instance of 3D MATCHING.

Suppose that (X, Y, Z, M) is a yes-instance of 3D MATCHING. Let Q be a certificate for that, so Q is a set of n disjoint triples. Then, merging the articles in each triple complies with G and creates n merged articles with three citations each. Next, we merge the four articles in W_2 to get a new merged article with 12 citations. However, this reduces the citations of each article in W_3 to 7. To fix this, we merge each article in W_3 with one merged article from W_1. These merges also comply with G. Thus, we obtain $n + 1$ articles with at least 10 citations each.

On the other hand, suppose that (D, G, W) is a yes-instance of $(\mathsf{fCite}, i10)$-INDEXIMP. Let $\mathcal{P}$ be a partition of W with i10-index $n+1$. Since $G[W_1]$ is a tripartite graph, each article in W_1 has only one citation, and articles in W_1 and W_2 are incompatible, the only way to create a new article with at least 10 citations is to merge the articles in W_2. Thus, we have $W_2 \in \mathcal{P}$. However, as argued above, merging the articles in W_2 reduces the citations of the articles in W_3. Since $\mathcal{P}$ contains $n+1$ articles with at least 10 citations each, it has to contain n merged articles each containing one of the articles in W_3. Since $\mathcal{P}$ has to comply with G, any of these merged articles consists of an article in W_3 and some articles in W_1. As each article in W_1 contributes just one citation and $|W_1| = 3n$, each article in $\mathcal{P} \setminus W_2$ contains exactly three atomic articles from W_1. Now, since these articles have to form a triangle in G, they also form a triple in M, so we obtain n disjoint triples. □

We now move to the study of INDEXACH. Our first result is a polynomial-time algorithm for $(\mathsf{sCite}, i10)$-INDEXACH when there are no restrictions on available merges.

THEOREM 6. $(\mathsf{sCite}, i10)$-INDEXACH *is solvable in polynomial time if G is a clique.*

PROOF. We will show that this problem can be captured by an integer linear program with a constant number of variables; by the classic result of Lenstra [7] optimal solutions of such programs can be found in polynomial time. For compactness, we present our argument for i3 rather than i10; extending it to i10 is straightforward.

Similarly to the proof of Proposition 4, it suffices to focus on articles with 1 or 2 citations. Suppose we have n_1 articles with one citation each and n_2 articles with 2 citations each. There are three ways to merge such articles to get an article with at least 3 citations: we can take (a) three articles with one citation each, (b) one article with one citation and one article with two citations, or (c) two articles with two citations each. Let x_{111}, x_{12}, and x_{22} denote the number of merged articles of type (a), (b), and (c) respectively. Let n_3 be the number of papers with three of more citations; then our task is captured by the following integer linear constraints:

$$\begin{aligned} x_{111} + x_{12} + x_{22} &\geq t - n_3 \\ 3x_{111} + x_{12} &\leq n_1 \\ x_{12} + 2x_{22} &\leq n_2 \\ x_{111}, x_{12}, x_{22} &\geq 0 \end{aligned}$$

For i10 the number of variables is bounded by the total number of partitions of numbers $10, \ldots, 18$, which is 1500[1]. □

In contrast, if G can be arbitrary, $(\mathsf{sCite}, i10)$-INDEXACH becomes NP-hard. The proof is by reduction from PARTITION INTO TRIANGLES [5]; we omit it due to space constraints.

THEOREM 7. *$(\mathsf{sCite}, i10)$-INDEXACH is NP-complete.*

To summarize, we have fully characterized the worst-case complexity of $(\mu, i10)$-INDEXIMP and $(\mu, i10)$-INDEXACH, for $\mu \in \{\mathsf{sCite}, \mathsf{uCite}, \mathsf{fCite}\}$. When all merges are allowed, we have some positive results, but the complexity of $(\mathsf{uCite}, i10)$-INDEXACH, $(\mathsf{fCite}, i10)$-INDEXIMP and $(\mathsf{fCite}, i10)$-INDEXACH remains unknown. We can show that the first of these problems becomes easy if we consider i2 instead of i10: $(\mathsf{uCite}, i2)$-INDEXACH reduces to finding a maximum matching in a certain graph. However, it is not clear how to generalize this result even to i3.

6. COOPERATIVE FRAMEWORK

We now consider the problem of manipulating citation indices in the presence of friends. For each citation index we will limit ourselves to scenarios that are easy in the absence of social context (e.g., for the h-index this means considering sCite and assuming that the compatibility graph is a clique).

6.1 h-Index

For the h-index, taking care of a friend complicates the manipulator's task considerably: even for the least computationally demanding citation measure sCite and even if the compatibility graph is a clique, checking if a manipulator can improve her h-index by 1 without harming her friend becomes computationally hard.

THEOREM 8. *(sCite, h)-INDEXIMPFRIEND is NP-complete, even if the compatibility graph G is a clique.*

PROOF. Following Remark 1, we assume that the input is given by three multisets of numbers: S_{12} for the articles co-authored by both researchers, S_1 for the articles of the first researcher not co-authored by the second researcher, and S_2 for the articles of the second researcher not co-authored by the first researcher.

We provide a reduction from SPECIAL 3-PARTITION; importantly, this problem is strongly NP-hard [5, 3]. Given an instance (S, m, b) of SPECIAL 3-PARTITION, we construct an instance of (S_{12}, S_1, S_2) of (sCite, h)-INDEXIMPFRIEND as follows. First, we obtain S' from S by adding m to each number in S. We observe that (S, m, b) is a yes-instance of SPECIAL 3-PARTITION if and only if (S', m, k), where $k = b + 3m$, is a yes-instance of SPECIAL 3-PARTITION. Note also that $k - m = b + 2m > 0$. We let S_{12} contain $m - 1$ copies of $k - 1$. Then S_1 is obtained from S' by adding $k - m$ copies of k, and S_2 contains $k - m$ copies of $k - 1$. We will show that (S_{12}, S_1, S_2) is a yes-instance of (sCite, h)-INDEXIMPFRIEND if and only if (S', m, k) is a yes-instance of SPECIAL 3-PARTITION.

Suppose first that (S', m, k) is a yes-instance of SPECIAL 3-PARTITION, and let T be a certificate for that. Then T is a partition of S' into m parts such that the sum of the numbers in each part is k. Now, if we merge articles in S_1 according to T, we get m merged articles with k citations each, $k - m$ singleton articles with k citations each, and further $m - 1$ articles with $k - 1$ citations each. This partition improves the h-index of the first researcher to k, while the articles in S_{12} remain in singletons, so the second researcher is not affected.

On the other hand, suppose that (S_{12}, S_1, S_2) is a yes-instance of (sCite, h)-INDEXIMPFRIEND, and let $\mathcal{P}$ be a certificate for that. We can assume without loss of generality that all articles with k citations appear as singletons in $\mathcal{P}$: indeed, if a merged article contains a component article with k citations, the latter can be split off without lowering the h-index. Further, since the second researcher has $k - 1$ articles, and each of them gets $k - 1$ citations, none of his articles can be merged. Thus, $\mathcal{P}$ contains at least m merged articles whose component articles correspond to elements of S', and each of these merged articles receives at least k citations. Since $\sum_{x \in S'} x = mk$, each of these merged articles gets exactly k citations, so we have obtained a desired partition of S'.

Since our reduction is from a strongly NP-hard problem, our problem remains hard for the graph-based representation. □

6.2 g-Index

We have seen (Section 4) that for a single researcher the g-index is more susceptible to manipulation than the h-index. This is also the case in the cooperative setting: the next theorem describes a polynomial-time algorithm for (sCite, g)-INDEXACHFRIEND, under the assumption that arbitrary merges are allowed.

THEOREM 9. *(sCite, g)-INDEXACHFRIEND can be solved in polynomial time if the compatibility graph G is a clique.*

PROOF. We use the same input representation as in Theorem 8. Set $S_1' = S_1 \cup S_{12}$, $S_2' = S_2 \cup S_{12}$.

Let g_1 and g_2 be the g-indices of S_1' and S_2', respectively. Suppose that $|S_2'| = g_2 + x$, where x is a non-negative integer. As researcher 2 has to have at least g_2 articles after the merge, the top t parts of an improving partition of S_1' may contain at most $x + t$ articles from S_{12}. We consider the following cases:

$|S_{12}| \geq x + t$: Let S^* be the set of $x + t$ most cited articles in S_{12}. Then, if $\sum_{z \in S_1 \cup S^*} z \geq t^2$, we create a single merged article that contains $x + 1$ articles from S^* as well as all articles in S_1; the remaining articles in S_{12} remain singletons. The merged article and the remaining $t - 1$ articles in S^* contribute to the g-index of the first researcher, so his g-index becomes at least t; the second researcher is not affected. If $\sum_{z \in S_1 \cup S^*} z < t^2$, we have a no-instance of our problem: if we were to use more articles from S_{12}, we would decrease the g-index of S_2'.

$|S_{12}| = y < t$: If $|S_1| < t - y$ or $\sum_{z \in S_1'} z < t^2$, then there is no improving partition. Otherwise, we partition articles in S_1 into $t - y$ groups and merge the articles in each group; this improves the g-index of the first researcher to t without touching the articles of the second researcher.

[1]see the On-Line Encyclopedia of Integer Sequences at `https://oeis.org/A000041`

$\boldsymbol{|S_{12}| = y, t \le y < x + t}$: If $\sum_{x \in S'_1} x < t^2$, there is no improving partition. Otherwise, we create a merged article that contains some $y - t + 1$ articles of S_{12} and all articles of S_1; the remaining $t - 1$ articles in S_{12} remain singletons. This partition of S'_1 has t parts and receives at least t^2 citations; on the other hand, the second researcher still has at least g_2 papers. $\square$

However, this is the only easy case: the results of Section 4 imply that (μ, g)-INDEXACHFRIEND is NP-hard if $\mu = \mathsf{sCite}$, but G can be arbitrary, or if $\mu \in \{\mathsf{uCite}, \mathsf{fCite}\}$ (even if G is a clique).

6.3 i10-Index

For the i10-index, we observe that when citations to merged articles are computed according to sCite or uCite, a manipulator need not worry about her friends: as along as she is not merging articles that already have 10 or more citations (and she has no reason to do so), she is not harming other researchers. Moreover, the integer linear program from the proof of Theorem 6 can be modified to capture the setting where we want to increase the i10-indices of several researchers simultaneously (note that the number of researchers k is treated as a constant in our model; the number of variables in our ILP is exponential in k, as we need to keep track of mergeable articles). We obtain the following results.

THEOREM 10. $(\mu, i10)$-INDEXIMPFRIEND *is solvable in polynomial time for* $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$. *Furthermore,* $(\mathsf{sCite}, i10)$-INDEXACHFRIEND *and* $(\mathsf{sCite}, i10, k)$-INDEXMACH *are solvable in polynomial time if G is a clique.*

For most of the remaining problems for the i10-index in the cooperative setting, we obtain NP-completeness results as corollaries of results of Section 5; the three open problems of Section 5 translate into open problems for the cooperative setting.

7. ADVERSARIAL FRAMEWORK

We will now consider the complexity of manipulation in the presence of adversaries. Again, we focus on scenarios where this problem is easy for a single researcher.

7.1 h-Index

For the h-index, the presence of enemies has the same computational cost as the presence of friends: manipulation becomes hard even for $\mu = \mathsf{sCite}$ and even if G is a clique.

THEOREM 11. (sCite, h)-INDEXIMPENEMY *and* (sCite, h)-INDEXHARMENEMY *are* NP-*complete even if G is a clique.*

Just as in the cooperative case, the proofs proceed by reductions from SPECIAL 3-PARTITION, and establish strong NP-hardness for the list-based representation.

7.2 g-Index

Interestingly, for the g-index merging one's papers so as to harm a competitor (without harming oneself) turns out to be easier than maximizing one's own citation index without helping the competitor. Specifically, for $\mu = \mathsf{sCite}$ we obtain the following results.

THEOREM 12. (sCite, g)-INDEXIMPENEMY *is* NP-*complete even when the compatibility graph G is a clique.*

The proof proceeds by reduction from SPECIAL 3-PARTITION and is omitted due to space constraints.

THEOREM 13. (sCite, g)-INDEXHARMENEMY *is solvable in polynomial time when the compatibility graph G is a clique.*

PROOF. We use the same notation as in Theorem 9.

Let g_1 and g_2 be the g-indices of S'_1 and S'_2 respectively. The first researcher can reduce the g-index of S'_2 if and only if $|S_{12}| \ge |S'_2| - g_2 + 2$. Assume that this is the case. Now, if $|S'_1| - |S'_2| + g_2 - 1 \ge g_1$, the first researcher can achieve his goal by merging $|S'_2| - g_2 + 2$ articles in S_{12} into a single article, and not performing any other merges. if this condition does not hold, any partition that harms the second researcher would also harm the first researcher. $\square$

However, (sCite, g)-INDEXHARMENEMY becomes NP-hard if G can be an arbitrary graph; we omit the proof, which is based on a reduction from 3D MATCHING.

THEOREM 14. (sCite, g)-INDEXHARMENEMY *is* NP-*complete.*

Moreover, for uCite this problem is hard even if arbitrary merges are allowed.

THEOREM 15. (uCite, g)-INDEXHARMENEMY *is* NP-*complete even if G is a clique.*

PROOF. We reduce from the restricted version of (uCite, g)-INDEXIMP where the compatibility graph is a clique; this problem is NP-hard by Theorem 3.

Suppose that we are given an instance (D', G', W) of (uCite, g)-INDEXIMP where $D' = (V', A')$ and G' is a clique. Let g be the g-index of W, and let $n = |V'|$. Suppose that the g most cited articles in W receive $s \ge g^2$ citations.

We construct an instance (D, G, W_1, W_2) of (uCite, g)-INDEXHARMENEMY as follows. Let $W_1 = W \cup \{w_1, w_2\}$, $W_2 = \{w_1, w_2, \dots, w_n\}$. To construct D, we start with D', and add the following articles and citations: $n + 6$ articles that cite w_n only, $n + 2$ articles that cite w_i only for each $i = 3, \dots, n - 1$, and $2g + 3$ articles that cite w_1. Now, we pick $x = \max\{0, (g + 2)^2 - (2g + 3) - s\} \le 2g + 1$ among these $2g + 3$ articles, and add a citation from each of these x articles to w_2. The set W_1 contains $g + 2$ articles, which receive at least $s + 2g + 3 + x \ge (g + 2)^2$ citations, so its g-index is $g + 2$; the g-index of W_2 is n. Finally, let G be a clique.

Now, suppose W admits a partition $\mathcal{P}$ with g-index at least $g+1$. Then $\mathcal{P}' = \mathcal{P} \cup \{w_1, w_2\}$ is a partition of W_1 with g-index at least $g + 2$, and the g-index of W_2 drops to $n - 1$.

Conversely, as we have $W_1 \cap W_2 = \{w_1, w_2\}$, the only way for the first researcher to lower the g-index of the second researcher is to merge w_1 and w_2. But then to ensure that her own g-index does not go down, she needs to find a partition of W where the top $g + 1$ articles get at least $(g + 2)^2 - (2g + 3) \ge (g + 1)^2$ citations, i.e., an improving partition of W. This completes the proof. $\square$

7.3 i10-Index

For i10, all easiness results from Section 5 remain true in the presence of enemies.

THEOREM 16. *The problems* $(\mu, i10)$-INDEXIMPENEMY *and* $(\mu, i10)$-INDEXHARMENEMY *are solvable in polynomial time for* $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$. *Also,* $(\mathsf{sCite}, i10)$-INDEXACHENEMY *is solvable in polynomial time if the compatibility graph G is a clique.*

PROOF. If Alice wants to improve her i10-index without helping Bob, she can use the algorithm from Proposition 4 with one modification: when going over subsets of her articles, she should discard the subsets that contain any of her joint articles with Bob. Similarly, to maximize her i10-index without helping Bob, she can discard her joint papers with Bob and then use the ILP from the proof of Theorem 6.

		IndexImp		IndexAch	
		G clique	G any	G clique	G any
h	sCite	P [3]	NPc [10]	NPc [3]	NPc
	uCite	NPc [10]	NPc	NPc	NPc
	fCite	NPc	NPc	NPc	NPc
g	sCite	P (Pr.1)	NPc (Th.2)	P (Pr.1)	NPc
	uCite	NPc (Th.3)	NPc	NPc	NPc
	fCite	NPc	NPc	NPc	NPc
i10	sCite	P	P (Pr.4)	P (Th.6)	NPc
	uCite	P	P (Pr.4)	?	NPc
	fCite	?	NPc (Th.5)	?	NPc

Table 1: Complexity results for citation indices. 'P' stands for 'polynomial-time solvable', 'NPc' stands for 'NP-complete'; '?' indicates that the problem is open. The number of the respective theorem/proposition is given in parentheses; when no reference is given, the result follows by Remark 2.

Finally, the only way for Alice to lower the i10-index of Bob is to merge two of her joint articles with Bob that receive at least 10 citations each. To ensure that this does not lower her i10-index, she should consider the set of her articles that receive between 1 and 9 citations each, go over all subsets of that set that contain at most 10 articles, and check if any such subset (a) complies with G, (b) receives at least 10 citations in total under μ (for $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$), and (c) does not contain an article by Bob. □

In contrast, for $(\mathsf{fCite}, i10)$-IndexHarmEnemy we get a hardness result by a reduction from $(\mathsf{fCite}, i10)$-IndexImp; we omit the proof.

Theorem 17. *The problem* $(\mathsf{fCite}, i10)$-IndexHarmEnemy *is* NP-*complete.*

8. PARAMETERIZED COMPLEXITY

Many of the computational problems considered so far turned out to be NP-hard. It is therefore natural to ask if they become easier when natural parameters of the input are small. This research agenda was initiated by van Bevern et al. [11], who have considered the following parameters: (i) the size c of the largest connected component of G; (ii) the target citation index t; (iii) the maximum allowed number of merges K. Indeed, it seems plausible that in realistic scenarios these parameters will be fairly small; in particular, a cautious manipulator woud not want to perform more than a few merges so as not to raise suspicions.

Van Bevern et al. obtain a number of fixed-parameter tractability results for these parameters. In particular, they establish that (μ, h)-IndexAch is FPT with respect to c for $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$. Their algorithm proceeds by considering and evaluating all possible partitions of each connected component of G, and then finding an optimal way to combine these partitions by means of dynamic programming. This approach turns out to work for many of the problems we consider. Specifically, we obtain the following results (for compactness, the theorem below is stated so that it subsumes the results of van Bevern et al. and includes some problems that are known to be polynomial-time solvable).

Theorem 18. *For each* $\mu \in \{\mathsf{sCite}, \mathsf{uCite}\}$ *and each* $I \in \{g, h, i10\}$ *the following problems are fixed-parameter tractable with respect to the size of the largest connected component of* G*:*

ImpFriend	$(\mathsf{sCite}, g)^*$ (Th.9), $(\mathsf{sCite}/\mathsf{uCite}, i10)$ (Th.10)
AchFriend	$(\mathsf{sCite}, g)^*$ (Th.9), $(\mathsf{sCite}, i10)^*$ (Th.10)
MAch	$(\mathsf{sCite}, i10)^*$ (Th.10)
ImpEnemy	$(\mathsf{sCite}/\mathsf{uCite}, i10)$ (Th.16)
AchEnemy	$(\mathsf{sCite}, i10)^*$ (Th.16)
HarmEnemy	$(\mathsf{sCite}, g)^*$ (Th.13), $(\mathsf{sCite}/\mathsf{uCite}, i10)$ (Th.16)

Table 2: Easiness results for manipulation in social context. For compactness, we omit the word Index **from the left column. For each problem, we list combinations of citation measures and indices for which it is polynomial-time solvable. * indicates that the result holds only if the compatibility graph is complete.**

- (μ, I)-IndexAch*;*
- (μ, I)-IndexAchFriend *and* (μ, I, k)-IndexMAch*;*
- (μ, I)-IndexAchEnemy *and* (μ, I)-IndexHarmEnemy.

Observe that Theorem 18 does not contain any results for fCite. Indeed, van Bevern et al. [11] show that (fCite, h)-IndexAch is NP-hard even for $c = 2$. We obtain similar hardness results for fCite with $c = 2$ for several other problems, for all three indices.

Theorem 19. *For each* $I \in \{h, g, i10\}$ *the following problems are* NP-*complete even if the size of the largest connected component of* G *is* 2*:*

- (fCite, I)-IndexAch
- (fCite, I)-IndexAchFriend *and* (fCite, I)-IndexMAch
- (fCite, I)-IndexAchEnemy.

A number of other results on the parameterized complexity of the problems considered in this paper can be found in the master's thesis of the first author [10].

9. CONCLUSIONS

We have investigated the complexity of manipulating one's citation indices, both for a researcher who only cares about her own performance and in the setting where one is interested in the performance of her colleagues and competitors. Our results are summarized in Tables 1 and 2. While we do not yet have a complete picture of the complexity of all problems we defined, it emerges that the h-index is harder to manipulate than the g-index and the i10-index.

While we phrased our results in terms of strategic merging of unrelated papers, many of them also apply in the setting where a researcher has to decide whether she can afford to merge several versions of her paper without lowering her (or possibly her students') citation indices; while a reputable researcher is unlikely to merge unrelated papers, *not* merging related papers is a less repugnant transaction.

Finally, we would like to emphasise that we do not want our results to be interpreted as an endorsement of the practice of relying on citation indices (or any other single-dimensional numeric measures) when making funding, hiring and promotion decisions; if anything, our results indicate that such measures can be gamed by dishonest researchers.

Acknowledgements

This work was done while the first author was an MSc student at Oxford, and she gratefully acknowledges financial support from HEFCE. The second author was partially supported by ERC StG 639945 and by Horizon 2020 OpenDreamKit project.

REFERENCES

[1] P. Ball. The h-index, or the academic equivalent of the stag's antlers. *Guardian*, January 6, 2012.

[2] A. Bateman. Why I love the h-index. *PLOS Biologue Community Blog*, 2012. `http://blogs.plos.org/biologue/2012/10/19/why-i-love-the-h-index/`.

[3] B. de Keijzer and K. R. Apt. The h-index can be easily manipulated. *Bulletin of the EATCS*, 110:79–85, 2013.

[4] L. Egghe. Theory and practise of the g-index. *Scientometrics*, 69(1):131–152, 2006.

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

[6] J. E. Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences*, 109(46):16569–16572, 2005.

[7] H. W. Lenstra Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, 1983.

[8] O. Mryglod, R. Kenna, Y. Holovatch, and B. Berche. Predicting the results of the REF using departmental h-index: A look at biology, chemistry, physics, and sociology. *LSE: The impact blog*, 2014. `http://blogs.lse.ac.uk/impactofsocialsciences/2014/12/16/predicting-the-results-of-the-ref/`.

[9] J. Ollerton. How does a scientist's h-index change over time? *Jeff Ollerton's Biodiversity Blog*, 2015. `https://jeffollerton.wordpress.com/2015/05/10/how-does-a-scientists-h-index-change-over-time/`.

[10] C. Pavlou. Manipulation of popular scientific impact indices by merging articles. Master's thesis, University of Oxford, 2015.

[11] R. van Bevern, C. Komusiewicz, R. Niedermeier, M. Sorge, and T. Walsh. H-index manipulation by merging articles: Models, theory, and experiments. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence, IJCAI'15, 2015*, pages 808–814, 2015.

[12] G. J. Woeginger. An axiomatic analysis of Egghe's g-index. *J. Informetrics*, 2(4):364–368, 2008.

[13] G. J. Woeginger. An axiomatic characterization of the Hirsch-index. *Mathematical Social Sciences*, 56(2):224–232, 2008.