Notre Dame Journal of Formal Logic
Volume 58, Number 2, 2017

On Polynomial-Time Relation Reducibility

Su Gao and Caleb Ziegler

Abstract We study the notion of polynomial-time relation reducibility among computable equivalence relations. We identify some benchmark equivalence relations and show that the reducibility hierarchy has a rich structure. Specifically, we embed the partial order of all polynomial-time computable sets into the polynomial-time relation reducibility hierarchy between two benchmark equivalence relations E_λ and id. In addition, we consider equivalence relations with finitely many nontrivial equivalence classes and those whose equivalence classes are all finite.

1 Introduction

It is well known that the relative complexity of binary relations can be studied by two different notions of reducibility. First, a binary relation R with domain X is a subset of $X \times X$. Thus, reducibility between relations is a special case of reducibility between sets on product spaces. In this sense, if R is a binary relation on X and S is a binary relation on Y, then a *reduction function* f from R to S is a function from $X \times X$ to $Y \times Y$ such that, for all $x_1, x_2 \in X$,

$$\langle x_1, x_2 \rangle \in R \iff f(x_1, x_2) \in S.$$

A stronger notion of reducibility can also be defined. For the above-mentioned R and S, a *strong reduction function* f is a function from X to Y such that, for all $x_1, x_2 \in X$,

$$\langle x_1, x_2 \rangle \in R \iff \big\langle f(x_1), f(x_2) \big\rangle \in S.$$

To distinguish between the two notions of reducibility, we call the latter notion *relation reducibility*, since it only applies to relations.

Relation reducibility has been studied intensively in at least two different contexts. The best known example is the Borel reducibility between analytic quasiorders and

Received February 10, 2014; accepted September 29, 2014
First published online March 3, 2017
2010 Mathematics Subject Classification: Primary 03D15, 68Q15; Secondary 68Q17
Keywords: polynomial-time relation reducibility, strong reduction function, finitary equivalence relations, finite equivalence relations
 10.1215/00294527-3867118

analytic equivalence relations on Polish spaces. Another instance is the computable reducibility between computably enumerable equivalence relations on natural numbers. In this article, we study relation reducibility for equivalence relations, where the domains are subsets of Σ^*, which is the set of all finite *sequences* over a certain finite alphabet Σ, and the strong reduction function is polynomial-time computable. We denote this reducibility notion by $\leq_R$.

The study of equivalence relations in the context of descriptive complexity theory originated in Blass and Gurevich [1], [2]. The earlier research concentrated on issues such as transversals and selection functions. The notion of relation reducibility has only recently been studied by various authors. The literature has not been extensive.

This reducibility was defined by Fortnow and Grochow [5] and was called *kernel reduction*. The more recent preprint by Finkelstein and Hescott [4] adopted this terminology and studied NP-equivalence relations and the issue of completeness. The same notion was called *strong equivalence reduction* by Buss, Chen, Flum, Friedman, and Müller [3], who considered this reducibility primarily restricted to classes of finite structures under the isomorphism relation (which they called *strong isomorphism reduction*). Furthermore, [3] was the first paper that contained structural results on the hierarchy of relation reducibility. It was shown in [3] that the partial order $\leq_R$, particularly restricted to isomorphism relations that are polynomial-time computable, has a rich structure. Specifically, they showed that the partial order on the countable atomless Boolean algebra is embeddable into the partial order $\leq_R$ on isomorphism relations that are in P.

In this article, we obtain some more structural results for $\leq_R$. We also focus on computable equivalence relations, even P-equivalence relations and NP-equivalence relations, although some of our results are general. Among P-equivalence relations, we consider some natural benchmark equivalence relations id, E_λ, and E_σ (defined in Section 2). It turns out that the embeddability result from [3] is about isomorphism relations below E_λ. We show in this article that the partial order $\leq_R$ between id and E_λ also has a rich structure. Our main embeddability result is the following.

Theorem 1.1 *There is an assignment $X \mapsto \mathsf{E}_X$ from the collection of all subsets of natural numbers $X \in \mathrm{P}$ to equivalence relations between E_λ and id such that, for any subsets of natural numbers $X, Y \in \mathrm{P}$,*

$$X \subseteq Y \iff \mathsf{E}_X \leq_R \mathsf{E}_Y .$$

Some reducibility or nonreducibility results can be obtained under assumptions on the relationships among complexity classes. We will identify several other interesting equivalence relations and investigate their mutual reducibility with assumptions such as $\mathrm{P} = \mathrm{NP}$ or $\mathrm{P} \neq \mathrm{NP}$. We also consider equivalence relations with finitely many nontrivial equivalence classes and those whose equivalence classes are all finite.

Some of our stated results are easy to prove and can be found in the literature (especially [3], [4]), but we include them for the completeness and coherence of our exposition.

2 Preliminaries

Throughout the article, we fix a finite alphabet Σ with at least two elements 0 and 1. Let Σ^* be the set of all finite sequences over the alphabet Σ. For $x \in \Sigma^*$, we denote by x the length of x. All equivalence relations we consider will have some subsets of Σ as their domains.

We use P to denote the class of all subsets of Σ^* that are polynomial-time computable by a deterministic Turing machine, NP to denote the class of all subsets of Σ^* that are polynomial-time acceptable by a nondeterministic Turing machine, and PF to denote the class of all polynomial-time computable functions from Σ^* to Σ^*.

In addition, coNP denotes the class of all subsets of Σ^* whose complement is in NP, and DP denotes the class of all sets of the form $L_1 \cap L_2$ where $L_1 \in$ NP and $L_2 \in$ coNP. Obviously,

$$\mathrm{P} \subseteq \mathrm{NP} \cap \mathrm{coNP} \subseteq \mathrm{NP} \cup \mathrm{coNP} \subseteq \mathrm{DP}.$$

We use $\mathbb{N}$ to denote the set of all natural numbers $\{0, 1, 2, \dots\}$. Via a fixed coding that we will not specify, we view $\mathbb{N}$ as a subset of Σ^*. In particular, we will speak of polynomial-time computable functions from $\mathbb{N}$ to $\mathbb{N}$ and subsets of $\mathbb{N}$ that are in various complexity classes. In some cases, we may only specify the action of a polynomial-time computable function on a limited domain. In this situation, we impose no conditions on the function acting outside of the specified domain.

We define again the main notion of reducibility under investigation.

Definition 2.1 For binary relations R and S, we say that R is *polynomial-time relation reducible* to S, denoted $R \leq_R S$, if there is a function $f : \mathrm{dom}(R) \to \mathrm{dom}(S)$ such that $f \in$ PF and, for all $x, y \in \mathrm{dom}(R)$, $\langle x, y\rangle \in R \iff \langle f(x), f(y)\rangle \in S$.

If $R \leq_R S$ but $S \not\leq_R R$, then we denote $R <_R S$ and say that R is *strictly polynomial-time relation reducible* to S. If $R \leq_R S$ and $S \leq_R R$, then we write $R \equiv_R S$ and say that R and S are *polynomial-time relation bireducible* to each other.

It is easy to see that $\leq_R$ is a quasiorder (i.e., reflexive and transitive) and $\equiv_R$ is an equivalence relation.

We will be primarily concerned with the structure of $\leq_R$ on the class of all computable equivalence relations. We introduce a few canonical examples of computable equivalence relations, which will serve as benchmarks for the hierarchy under $\leq_R$.

Definition 2.2

(1) We denote by id the *identity relation*: for $x, y \in \Sigma^*$,

$$\langle x, y\rangle \in \mathsf{id} \iff x = y.$$

(2) We denote by E_λ the *equality of length relation*: for $x, y \in \Sigma^*$,

$$\langle x, y\rangle \in \mathsf{E}_\lambda \iff x = y.$$

(3) For $x, y \in \Sigma^*$, define

$$\langle x, y\rangle \in \mathsf{E}_\sigma \iff \text{if } 2^k \leq x < 2^{k+1} \text{ and } 2^m \leq y < 2^{m+1}, \text{ then } k = m.$$

We will show below that $\mathsf{E}_\sigma <_R \mathsf{E}_\lambda <_R \mathsf{id}$. Before doing this, we prove a general lemma that allows us to obtain nonreducibility results from counting. All of our nonreducibility results that do not require assumptions on relationships between complexity classes will be proved by the counting method used in the proof of the lemma. This concept is a restatement of the notion of potential reducibility from [3]. However, because we repeatedly use the result, but do not require the relation for potential reducibility, we include a proof that does not reference potential reducibility.

Lemma 2.3 *Let E, F be equivalence relations on Σ^*, and let $\#E(n), \#F(n)$ denote the number of equivalence classes containing elements of length at most n. If, for any polynomial p, there is n so that $\#E(n) < \#F(p(n))$, then $E \not\leq_R F$.*

Proof Assume $E \leq_R F$ via a strong reduction function $f \in \mathrm{PF}$. Let $p(n)$ be a polynomial bound for the running time of f; that is, for any input $x \in \Sigma^*$ with $|x| = n$, it takes no more than $p(n)$ steps for the computation of $f(x)$ to halt. Assume without loss of generality that $p(n)$ is monotone increasing. It follows that if $x \leq n$, then $|f(x)| \leq p(n)$. Now fix any length n, let $\#E(n) = k$, and let $x_1, \ldots, x_k \in \Sigma^*$ be pairwise non-E-equivalent elements of length at most n. Then $f(x_1), \ldots, f(x_k)$ are pairwise non-F-equivalent since f is a strong reduction function. We conclude that $\#F(p(n)) \geq k$, or $\#E(n) \leq \#F(p(n))$. □

Proposition 2.4 *We have that* $\mathsf{E}_\sigma <_R \mathsf{E}_\lambda <_R \mathsf{id}$.

Proof It is easy to see that $\mathsf{E}_\lambda \leq_R \mathsf{id}$: just define the strong reduction function $f : \Sigma^* \to \Sigma^*$ by $f(x) = 0^{|x|}$. To see that $\mathsf{E}_\sigma \leq_R \mathsf{E}_\lambda$, use the strong reduction function $f : \Sigma^* \to \Sigma^*$ where $f(x) = 0^{\log |x|}$. To prove nonreducibility, we apply Lemma 2.3. It is easy to see that the lemma applies since $\#\mathsf{id}(n) \geq 2^n$, $\#\mathsf{E}_\lambda(n) = n + 1$, and $\#\mathsf{E}_\sigma(n) \leq \log n$. □

In general, one can replace the exponential function used in the definition of E_σ by other growth functions. For instance, if one uses functions with greater growth rates, then the resulting equivalence relation will be $<_R \mathsf{E}_\sigma$. In this fashion one can easily produce an infinite descending chain of equivalence relations below E_σ.

None of the results in this section are new since it has been shown in [3] that the partial order $\leq_R$ below E_λ contains a copy of the countable atomless Boolean algebra.

3 Structural Results between E_λ and id

In this section, we show that the reducibility $\leq_R$ between E_λ and id already has a rich structure. Among other things, we will demonstrate an infinite ascending chain and an infinite antichain between E_λ and id. We will prove the main embeddability result mentioned in Section 1, which in particular implies that any finite partial order can be embedded into the relation $<_R$ among equivalence relations between E_λ and id. To achieve these, we construct a class of equivalence relations as follows.

Definition 3.1

(a) We call a set $A \subseteq \mathbb{N}$ *good* if A is infinite, $A \in \mathrm{P}$, and, by letting $\{a_n : n \geq 1\}$ enumerate the elements of A in strictly increasing order, the function $a_n \mapsto n$ is polynomial-time computable.

(b) For a good set $A \subseteq \mathbb{N}$, let $\{a_n : n \geq 1\}$ enumerate all elements of A in strictly increasing order. Define $L_A : \Sigma^* \to \mathbb{N}$ by

$$L_A(x) = \begin{cases} 0 & \text{if } |x| < a_1, \\ n & \text{if } n \geq 1 \text{ is such that } a_n \leq x < a_{n+1}. \end{cases}$$

Note that L_A is polynomial-time computable (see Lemma 3.2 below).

(c) Let $A \subseteq \mathbb{N}$ be good, and let $\{a_n : n \geq 1\}$ enumerate all elements of A in strictly increasing order. Let $\varphi : \mathbb{N} \to \mathbb{N}$ be a polynomial-time computable function with $\varphi(n) \leq a_n$ for all $n \geq 1$. Define an equivalence relation $\mathsf{E}_{\varphi,A}$ by

$$\langle x, y\rangle \in \mathsf{E}_{\varphi,A} \iff L_A(x) = L_A(y) \text{ and}$$
$$\text{if } L_A(x) > 0, \text{ then } x \text{ and } y \text{ agree in the first } \varphi\big(L_A(x)\big) \text{ bits.}$$

Lemma 3.2 *If $A \subseteq \mathbb{N}$ is good, then L_A is polynomial-time computable.*

Proof Let A be good, and let $\{a_n : n \geq 1\}$ enumerate all elements of A in strictly increasing order. Then there exist a polynomial-time computable function φ recognizing A, that is, $\varphi(x) = 1$ if $x \in A$ and $\varphi(x) = 0$ otherwise, and a polynomial-time computable function $\psi : a_n \mapsto n$. Note that the values that ψ takes on elements not in A are not relevant. We compute L_A as follows. Given a string x, we compute $|x|$. We then run $\varphi(m)$ for $m = |x|, |x| - 1, \ldots, 0$ until we find the largest m with $\varphi(m) = 1$. This process requires at most $|x|$ applications of φ on strings of length at most $|x|$, so it can be done in time that is polynomial in $|x|$. If $\varphi(m) = 1$, then $m = a_n$ for some $n \geq 1$. Since we run this computation in decreasing order, we see that if m is the largest with $\varphi(m) = 1$, then $m = a_n$ where $|x| < a_{n+1}$.

So suppose m is the result of this computation. Then, $L_A(x) = \psi(m)$ for this m. If this computation does not return any m with $\varphi(m) = 1$, then $|x| < a_1$, and so in this case, $L_A(x) = 0$. $\square$

It follows easily from the definition that $\mathsf{E}_{\varphi,A} \leq_R \mathsf{id}$. As the following theorem demonstrates, the relative complexity of $\mathsf{E}_{\varphi,A}$ is closely related to the growth rates of the functions φ and $n \mapsto a_n$, which enumerates elements of the set A.

Theorem 3.3 *Let $A \subseteq \mathbb{N}$ be good, and let $\{a_n : n \geq 1\}$ enumerate all elements of A in strictly increasing order. Let φ and ψ be polynomial-time computable functions from $\mathbb{N}$ to $\mathbb{N}$ with $\varphi(n), \psi(n) \leq a_n$ for all $n \geq 1$. Then the following hold.*

(i) If $\varphi(n) \leq \psi(n)$ for all n, then $\mathsf{E}_{\varphi,A} \leq_R \mathsf{E}_{\psi,A}$.

(ii) If φ is increasing, $\varphi(n) = \Omega(\log a_n)$, and for any polynomial $p(n)$

$$\psi(n) \neq O\big(\varphi\big(p(a_n)\big)\big),$$

then $\mathsf{E}_{\psi,A} \not\leq_R \mathsf{E}_{\varphi,A}$.

(iii) Let $B \subseteq \mathbb{N}$ be good, and let $\{b_n : n \geq 1\}$ enumerate all elements of B in strictly increasing order. If $a_n \leq b_n$ for all $n \geq 1$, then $\mathsf{E}_{\varphi,B} \leq_R \mathsf{E}_{\varphi,A}$.

Proof For (i), suppose $\varphi(n) \leq \psi(n)$ for all n. We define a strong reduction function $f : \Sigma^* \to \Sigma^*$ from $\mathsf{E}_{\varphi,A}$ to $\mathsf{E}_{\psi,A}$. Let $x \in \Sigma^*$. If $|x| < a_1$, then $L_A(x) = 0$, and we define $f(x)$ to be the empty sequence. Otherwise, $L_A(x) > 0$ and $\varphi(L_A(x)) \leq a_{L_A(x)} \leq |x|$. In this case, define $f(x)$ to be the sequence of the same length as x which agrees with x on the first $\varphi(L_A(x))$ bits and is 0 for all remaining bits. We check that f works. If $x, y \in \Sigma^*$ are $\mathsf{E}_{\varphi,A}$-equivalent, then we actually have $f(x) = f(y)$ and therefore $f(x)$ and $f(y)$ are $\mathsf{E}_{\psi,A}$-equivalent. If x and y are not $\mathsf{E}_{\varphi,A}$-equivalent, then either $L_A(x) \neq L_A(y)$ or else $L_A(x) = L_A(y) > 0$ but x and y disagree in the first $\varphi(L_A(x))$ bits. If $L_A(x) \neq L_A(y)$, then $L_A(f(x)) \neq L_A(f(y))$ and so $f(x)$ and $f(y)$ are not $\mathsf{E}_{\psi,A}$-equivalent. If $L_A(x) = L_A(y) > 0$ and x and y disagree in the first $\varphi(L_A(x))$ bits, then $f(x)$ and $f(y)$ also disagree in the first $\varphi(L_A(f(x))) = \varphi(L_A(x))$ bits, which implies that $f(x)$ and $f(y)$ are not $\mathsf{E}_{\psi,A}$-equivalent.

For (ii), we give a direct proof instead of applying Lemma 2.3; this turns out to be notationally simpler but we use the same counting technique. Let φ and ψ be such that φ is increasing, $\varphi(n) = \Theta(\log a_n)$, and for any polynomial $p(n)$ $\psi(n) \neq O(\varphi(p(a_n)))$. Assume $E_{\psi,A} \leq_R \mathsf{E}_{\varphi,A}$, and let f be a strong reduction function from $\mathsf{E}_{\psi,A}$ to $\mathsf{E}_{\varphi,A}$. Let $n \geq 1$ be arbitrarily fixed. Let $N_\varphi(n)$ be the number of $\mathsf{E}_{\varphi,A}$-equivalence classes which consist of strings x with $L_A(x) = n$. It is easily seen by direct counting that $N_\varphi(n) = 2^{\varphi(n)}$. Similarly, define $N_\psi(n) = 2^{\psi(n)}$. Now f is a one-to-one function on the equivalence classes of $\mathsf{E}_{\psi,A}$; thus, the $N_\psi(n)$ many distinct $\mathsf{E}_{\psi,A}$-equivalence classes are mapped via f to $N_\psi(n)$ many distinct $\mathsf{E}_{\varphi,A}$-equivalence classes. Let $S(n)$ be the set of all $x \in \Sigma^*$ with $|x| = a_n$. Then for each $x \in S(n)$, $L_A(x) = n$, and for any $y \in \Sigma^*$ with $L_A(y) = n$, there is $x \in S(n)$ which is $\mathsf{E}_{\psi,A}$-equivalent to y (since $\psi(n) \leq a_n$). In other words, each of the $N_\psi(n)$ many distinct $\mathsf{E}_{\psi,A}$-equivalence classes contains an element of $S(n)$.

Let $p(n)$ be a polynomial bound for the running time of f; that is, the computation of f on an input string of length n takes no more than $p(n)$ many steps to halt. Then we must have $|f(x)| \leq p(|x|)$ for any $x \in \Sigma^*$. Without loss of generality, assume that $p(n) \geq n$ and is monotone increasing. Considering the $\mathsf{E}_{\varphi,A}$-equivalence classes of the elements of $\{f(x) : x \in S(n)\}$, we obtain

$$N_\psi(n) \leq \sum_{k \leq p(a_n)} N_\varphi(k) \leq p(a_n) N_\varphi\big(p(a_n)\big).$$

The second inequality follows from the hypothesis that φ is increasing.

Taking the logarithm on all sides of the inequality, we obtain

$$\psi(n) \leq \varphi\big(p(a_n)\big) + O(\log a_n).$$

Since $\varphi(n) = \Omega(\log a_n)$, we have $\psi(n) = O(\varphi(p(a_n)))$, which is a contradiction.

For (iii), suppose $a_n \leq b_n$ for all $n \geq 1$. Note that, for any $x \in \Sigma^*$, $L_B(x) \leq L_A(x)$, and if $L_B(x) > 0$, then $a_{L_B(x)} \leq a_{L_A(x)} \leq |x|$. Define

$$f(x) = \begin{cases} \text{the empty sequence} & \text{if } L_B(x) = 0, \\ x \restriction a_{L_B(x)} & \text{otherwise.} \end{cases}$$

Then f is polynomial-time computable. In fact, given x, the computations of $L_A(x)$ and $L_B(x)$ take time that is polynomial in x; then we obtain $x \restriction a_{L_B(x)}$ by checking on all the candidates, which are successive truncations of x. (Note that we do not need to assume that the function $n \mapsto a_n$ is polynomial-time computable.)

We verify that f is a strong reduction. First, suppose x and y are $\mathsf{E}_{\varphi,B}$-equivalent. Then we have $L_B(x) = L_B(y)$. If they are zero, then $f(x) = f(y)$ and is the empty sequence, and in particular, $f(x)$ and $f(y)$ are $\mathsf{E}_{\varphi,A}$-equivalent. Suppose that $L_B(x) = L_B(y) > 0$. Then x and y must agree on the first $\varphi(L_B(x))$ bits. By definition, $f(x)$ and $f(y)$ are truncations to the length $a_{L_B(x)}$ of x and y, respectively. We have $L_A(f(x)) = L_B(x)$ and $L_A(f(y)) = L_B(y)$. Therefore, $L_A(f(x)) = L_A(f(y))$, and we have that $f(x)$ and $f(y)$ still agree on the first $\varphi(L_A(f(x))) = \varphi(L_B(x))$ bits, so $f(x)$ and $f(y)$ are $\mathsf{E}_{\varphi,A}$-equivalent.

For the other direction, suppose that x and y are not $\mathsf{E}_{\varphi,B}$-equivalent. Then either $L_B(x) \neq L_B(y)$ or else $L_B(x) = L_B(y) > 0$ but x and y do not agree on the first $\varphi(L_B(x))$ bits. Since $L_A(f(x)) = L_B(x)$ and $L_A(f(y)) = L_B(y)$, we have $L_A(f(x)) \neq L_A(f(y))$ if $L_B(x) \neq L_B(y)$. If $L_B(x) = L_B(y)$ but x and y do not agree on the first $\varphi(L_B(x))$ bits, then $L_A(f(x)) = L_A(f(y)) > 0$ but $f(x)$ and

$f(y)$ do not agree on the first $\varphi(L_A(f(x)))$ bits, since $f(x)$ and $f(y)$ are just truncations of x and y to the length $a_{L_B(x)}$, and $\varphi(L_A(f(x))) = \varphi(L_B(x)) \le a_{L_B(x)}$. In either case, we see that $f(x)$ and $f(y)$ are not $\mathsf{E}_{\varphi,A}$-equivalent. This shows that f is a strong reduction. $\square$

In the remainder of this section we give some applications of Theorem 3.3. In fact, for these applications we will fix the good set $A = \mathbb{N} \setminus \{0\}$. Thus, $a_n = n$ in the enumeration of elements of A, and we have $L_A(x) = |x|$. For notational simplicity, let E_φ denote $\mathsf{E}_{\varphi,A}$ for this particular A.

In summary, let $\varphi : \mathbb{N} \to \mathbb{N}$ be an increasing, polynomial-time computable function with $\varphi(n) \le n$ for all $n \in \mathbb{N}$, and let E_φ be the equivalence relation defined by

$$\langle x, y\rangle \in \mathsf{E}_\varphi \iff |x| = |y| \text{ and } x \text{ and } y \text{ agree in the first } \varphi\big(|x|\big) \text{ bits.}$$

Note that $\mathsf{E}_\lambda \le_R \mathsf{E}_\varphi \le_R \mathsf{id}$. The first reduction is witnessed by the same strong reduction function $f : \Sigma^* \to \Sigma^*$, where $f(x) = 0^{|x|}$, that we used in the proof of Proposition 2.4.

We are now ready to construct an infinite ascending chain and an infinite antichain between E_λ and id.

Corollary 3.4 *There is an infinite sequence of functions $\varphi_m \in$ PF such that $\mathsf{E}_{\varphi_m} <_R \mathsf{E}_{\varphi_{m+1}}$ for all m.*

Proof For each $m \ge 1$, define $\varphi_m(n) = \min\{n, (\log n)^m\}$. Then each φ_m is polynomial-time computable and $\log n \le \varphi_m(n) \le n$ for all n. Since $\varphi_m(n) \le \varphi_{m+1}(n)$ for all n, we have $\mathsf{E}_{\varphi_m} \le_R \mathsf{E}_{\varphi_{m+1}}$ by Theorem 3.3(i). On the other hand, for any polynomial $p(n)$, $\varphi_m(p(n)) = \Theta((\log n)^m)$. Since $(\log n)^{m+1} \ne O((\log n)^m)$, we have that $\varphi_{m+1}(n) \ne O(\varphi(p(n)))$. Thus, $\mathsf{E}_{\varphi_{m+1}} \not\le_R \mathsf{E}_{\varphi_m}$ by Theorem 3.3(ii). We thus have $\mathsf{E}_{\varphi_m} <_R \mathsf{E}_{\varphi_{m+1}}$ for all m. $\square$

Again, it is easily seen that the construction in the above proof can be modified to obtain longer chains. For instance, if we use functions such as $\log n(\log\log n)^k$, $\log n(\log\log n)(\log\log\log n)^k$, and so on, one can obtain more infinite ascending sequences in-between the ones obtained in the above proof. As a consequence, one can embed the linear order on the ordinal ω^k for any finite k into the relation $<_R$ between E_λ and id.

Corollary 3.5 *There is an infinite sequence of functions $\varphi_m \in$ PF such that $\mathsf{E}_{\varphi_m} \not\le_R \mathsf{E}_{\varphi_{m'}}$ for any $m \ne m'$.*

Proof Let $\langle\cdot,\cdot\rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be a polynomial-time computable pairing function such that the decoding functions $(\cdot)_1$ and $(\cdot)_2$ are both polynomial-time computable. For instance, the usual Cantor diagonalization function is itself a polynomial in both variables and therefore is such a function.

Define an increasing sequence of natural numbers N_k by induction:

$$N_0 = 1,$$
$$N_{k+1} = 2^{1+(\log N_k)^2}.$$

We are now ready to define φ_m for any $m \in \mathbb{N}$ as

$$\varphi_m(n) = \begin{cases} \min\{n, (\log n)^2\} & \text{if } N_k \le n < N_{k+1} \text{ and } (k)_1 = m, \\ \min\{n, (\log N_k)^2\} & \text{if } N_k \le n < N_{k+1} \text{ and } (k)_1 \ne m. \end{cases}$$

Each φ_m is polynomial-time computable. It is easy to see that each φ_m is increasing. To see that $\varphi_m(n) = \Omega(\log n)$, it is enough to check that, as functions in k, $\log N_{k+1} = O(\varphi_m(N_k))$. For this, just notice that, asymptotically, $\varphi_m(N_k) = (\log N_k)^2$ whereas $\log N_{k+1} = 1 + (\log N_k)^2$.

We claim that if $m \neq m'$, then $\varphi_m(n) \neq O(\varphi_{m'}(n))$. This would complete the proof by Theorem 3.3(ii).

Suppose this is not so, and assume that $\varphi_m(n) = O(\varphi_{m'}(n))$. Then for some constant $C > 0$ and N, $\varphi_m(n) \leq C\varphi_{m'}(n)$ for all $n \geq N$. Suppose N is large enough that $(\log n)^2 \leq n$ for all $n \geq N$. It follows in particular that, for all k such that $N_k \geq N$,

$$\varphi_m(N_{k+1} - 1) \leq C\varphi_{m'}(N_{k+1} - 1).$$

If in addition $(k)_1 = m$, then by our definition of φ_m we have

$$\varphi_m(N_{k+1} - 1) = \big(\log(N_{k+1} - 1)\big)^2 \leq C(\log N_k)^2.$$

Hence, there is a constant C' such that $\log(N_{k+1} - 1) \leq C' \log N_k$ for the infinitely many k with $(k)_1 = m$. However, when $(k)_1 = m$, we also have

$$\log(N_{k+1} - 1) = \Theta(\log N_{k+1}) = \Theta\big(1 + (\log N_k)^2\big) = \Theta\big((\log N_k)\big)^2$$

as functions of k. This is a contradiction. □

Note that the equivalence relations constructed in the above proofs are all between E_λ and the second equivalence relation constructed in the proof of Corollary 3.4. Again the technique can be easily adapted for other pairs of equivalence relations obtained from the proof of Corollary 3.4.

The functions we constructed in the proof of Corollary 3.5 have a stronger property that we now explore.

Corollary 3.6 *There is an assignment $X \mapsto \mathsf{E}_X$ from the collection of all subsets of natural numbers $X \in \mathrm{P}$ to equivalence relations between E_λ and* id *such that, for any subsets of natural numbers $X, Y \in \mathrm{P}$,*

$$X \subseteq Y \iff \mathsf{E}_X \leq_R \mathsf{E}_Y.$$

Proof Let φ_m be the functions constructed in the proof of Corollary 3.5. For each polynomial-time computable $X \subseteq \mathbb{N}$, define

$$\varphi_X(n) = \max\big\{\varphi_m(n) : m \in X\big\}$$

and $\mathsf{E}_X = \mathsf{E}_{\varphi_X}$. Each φ_X is polynomial-time computable because

$$\varphi_X(n) = \begin{cases} \min\{n, (\log n)^2\} & \text{if } N_k \leq n < N_{k+1} \text{ and } (k)_1 \in X, \\ \min\{n, (\log N_k)^2\} & \text{if } N_k \leq n < N_{k+1} \text{ and } (k)_1 \notin X. \end{cases}$$

Now if $X \subseteq Y \subseteq \mathbb{N}$ and $X, Y \in \mathrm{P}$, then $\varphi_X(n) \leq \varphi_Y(n)$ for all n, and therefore, $\mathsf{E}_X \leq_R \mathsf{E}_Y$ by Theorem 3.3(i). On the other hand, if $X \not\subseteq Y$ and we let $m \in X \setminus Y$, then $\mathsf{E}_{\varphi_m} \not\leq_R \mathsf{E}_Y$ by the same proof of nonreducibility as that of Corollary 3.5. □

Since any finite partial order can be embedded into the Boolean algebra $(\mathrm{P}, \subseteq)$, it follows that any finite partial order is embeddable also into the $\leq_R$ relation among equivalence relations between E_λ and id.

4 More Examples and Nonreducibility Results

An important class of computable equivalence relations is that of isomorphism for finite structures. For each kind of finite structure, such as graphs, groups, and linear orders, there are many reasonable ways to code the structures by sequences in Σ^*. For instance, we may consider only structures whose underlying set is an initial segment of the natural numbers. The isomorphism relation corresponds to an equivalence relation with a computable subset of Σ^* as its domain. We adopt the following notation.

Definition 4.1

(1) We denote by GI the *graph isomorphism* relation: if $x, y \in \Sigma^*$ encode finite graphs, then $\langle x, y\rangle \in \mathsf{GI}$ if and only if there is an isomorphism between the graphs coded, respectively, by x and y.
(2) Similarly, we denote by GROUP the isomorphism for finite groups.
(3) We denote by CLIQ the *clique relation*: if $x, y \in \Sigma^*$ encode finite graphs, then $\langle x, y\rangle \in \mathsf{CLIQ}$ if and only if the maximum size cliques in the graphs coded by x and y are of the same size.

All isomorphism relations for finite structures are NP-equivalence relations. Thus, GI and GROUP are in NP. On the other hand, CLIQ is DP-complete as a set (see Papadimitriou [8]).

In the following, we collect some results from the literature.

Proposition 4.2 *The following are true.*

(i) If E is the isomorphism relation for finite structures of any finite language, then $E \leq_R \mathsf{GI}$.
(ii) If E is the isomorphism relation for any of the following classes of finite structures, then $E \equiv_R \mathsf{id}$:
  (a) finite trees;
  (b) finite planar graphs;
  (c) finite linear orderings with a unary relation.
(iii) If E is the isomorphism relation for any of the following classes of finite structures, then $E \equiv_R \mathsf{E}_\lambda$:
  (a) finite sets;
  (b) finite fields;
  (c) finite abelian groups;
  (d) finite cyclic groups;
  (e) finite linear orderings;
  (f) finite linear orderings with a distinguished point.
(iv) If E is the isomorphism relation for finite Boolean algebras, then $E \equiv_R \mathsf{E}_\sigma$.
(v) If E is the isomorphism relation for any of the following classes of finite structures, then $E \equiv_R \mathsf{GI}$:
  (a) finite regular graphs;
  (b) finite bipartite graphs.

See [3] for (i), (ii.c), (iii), and (iv). The proof of (ii.a) is folklore and implicit in any proof that a finite tree isomorphism is in P. A proof of (ii.b) is implicit in the proof of the theorem of Hopcroft and Tarjan [6] that the isomorphism for finite planar graphs is in P. The proof of (v.a) can be extracted from Miller [7]. We give a proof of (v.b) below.

Proof of Proposition 4.2(v.b) Let $G = (V, E)$ be an arbitrary finite graph. We define a bipartite graph $G^* = (V^*, E^*)$. The mapping $G \mapsto G^*$ will be a strong reduction from GI to the isomorphism of finite bipartite graphs. Let $V^* = V \cup E$ and $E^* = \{(v, e), (e, v) : v \in V,\ e \in E,\ \text{and } v \text{ is a vertex of } e\}$. G^* is bipartite because $\{V, E\}$ is a partition of its vertices such that every edge in G^* is between a vertex in V and a vertex in E. Suppose G_1 and G_2 are two finite graphs. It is clear that if $G_1 \cong G_2$, then $G_1^* \cong G_2^*$. In fact, any isomorphism between G_1 and G_2 induces an isomorphism of G_1^* and G_2^* by the definition of G^*. For the converse, first note that G is connected if and only if G^* is connected. Without loss of generality, we may assume both G_1 and G_2 are connected. Suppose that π is an isomorphism from G_1^* to G_2^*. Then π must map V_1 to either V_2 or E_2, since the two graphs are bipartite. In the first case, it must happen that $\pi(V_1) = V_2$ and $\pi(E_1) = E_2$, and therefore, π induces an isomorphism between G_1 and G_2. In the second case, $\pi(V_1) = E_2$ and $\pi(E_1) = V_2$. Note that each vertex in E_1 in G_1^* has degree 2. Thus, it must happen that each vertex in V_2 in G_2^* has degree 2, and thus, each vertex in V_1 in G_1^* has degree 2. This means that all vertices in the two graphs have degree 2, and G_1^* must be a simple cycle. The same holds for G_2^*. These imply that both G_1 and G_2 are simple cycles, and they are of the same length. Thus, G_1 and G_2 are isomorphic. □

It was shown in [3, Corollary 18] that

$$\mathsf{E}_\lambda <_R \mathsf{GROUP} <_R \mathsf{GI}$$

and

$$\mathsf{id} \not\leq_R \mathsf{GROUP}.$$

The proofs of the nonreducibility directions are all based on the counting argument used to prove Lemma 2.3.

Note that

$$\mathsf{E}_\lambda \leq_R \mathsf{CLIQ};$$

this is witnessed by the strong reduction function $x \mapsto f(x)$, where $f(x)$ is the code for the complete graph with $|x|$ vertices. Note that $\#\mathsf{CLIQ}(n) = O(n)$. Thus, by Lemma 2.3, we have

$$\mathsf{id} \not\leq_R \mathsf{CLIQ}.$$

Since $\mathsf{id} \leq_R \mathsf{GI}$, it follows that

$$\mathsf{GI} \not\leq_R \mathsf{CLIQ}.$$

We summarize the reducibility and nonreducibility results for the six interesting equivalence relations in a diagram on the following page. In the diagram, a solid arrow represents reducibility $<_R$, an arrow with a cross represents nonreducibility $\not\leq_R$, and a dotted line indicates that the opposite direction is unsettled.

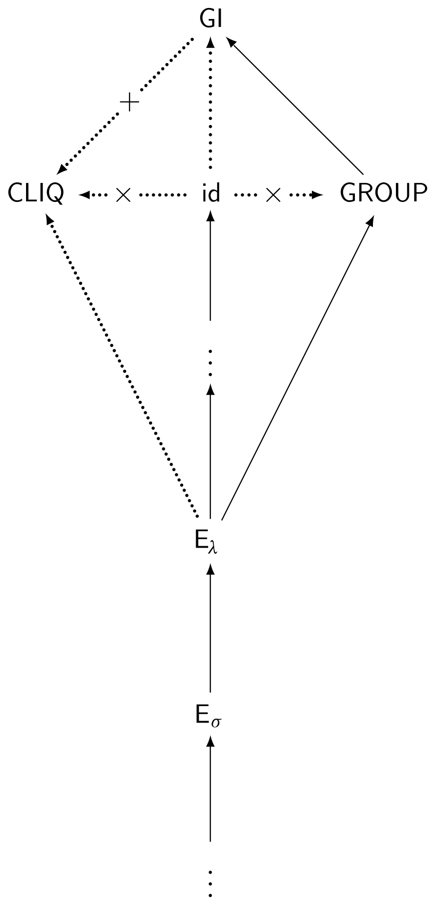
The following propositions clarify the reducibility from CLIQ with other equivalence relations in the diagram.

Proposition 4.3 *The following are equivalent:*

(i) $\mathrm{P} = \mathrm{NP}$*;*
(ii) $\mathsf{CLIQ} \leq_R \mathsf{E}_\lambda$*;*
(iii) $\mathsf{CLIQ} \equiv_R \mathsf{E}_\lambda$*.*

Proof Obviously (ii) and (iii) are equivalent. To see (i)$\Rightarrow$(ii), assume $\mathrm{P} = \mathrm{NP}$. Then there is a polynomial-time algorithm to determine, given a finite graph G and a number k, whether G contains a clique of size k. Let G have n vertices. Then by setting $k = 2, \ldots, n$, the maximal k such that G contains a clique of size k can be computed in polynomial time. If we run this decision algorithm and output a sequence of length this maximal k, then this gives a strong reduction function in P from CLIQ to E_λ. For (ii)$\Rightarrow$(i), note that $\mathsf{NP} \subseteq \mathsf{DP}$ and that CLIQ is DP-complete as a set. If $\mathsf{CLIQ} \leq_R \mathsf{E}_\lambda$, then $\mathsf{CLIQ} \in \mathrm{P}$ and so is every DP set. $\square$

Thus, if $\mathrm{P} = \mathrm{NP}$, then CLIQ is of the same complexity as E_λ.



Proposition 4.4 *If* $\mathrm{P} \neq \mathrm{NP}$*, then* $\mathsf{CLIQ} \not\leq_R \mathsf{id}$.

Proof This is because if $\mathsf{CLIQ} \leq_R \mathsf{id}$, then $\mathsf{CLIQ} \in \mathrm{P}$. $\square$

In other words, if $\mathrm{P} \neq \mathrm{NP}$, then CLIQ is incomparable with id and $\mathsf{E}_\lambda <_R \mathsf{CLIQ}$.

Proposition 4.5 *If* $\mathrm{NP} \neq \mathrm{coNP}$*, then* $\mathsf{CLIQ} \not\leq_R \mathsf{GI}$*, and in particular,* $\mathsf{CLIQ} \not\leq_R \mathsf{id}$ *and* $\mathsf{CLIQ} \not\leq_R \mathsf{GROUP}$.

Proof If $\mathsf{CLIQ} \leq_R \mathsf{GI}$, then $\mathsf{CLIQ} \in \mathrm{NP}$. Since $\mathrm{coNP} \subseteq \mathrm{DP}$, it would follow that $\mathrm{NP} = \mathrm{coNP} = \mathrm{DP}$. $\square$

In other words, if NP $\neq$ coNP, then CLIQ is incomparable with either GI or id, and $\mathsf{E}_\lambda <_R \mathsf{CLIQ}$.

The following lemma was essentially proved in [1] (cf. [3, Lemma 32]).

Lemma 4.6 (Blass–Gurevich) *If* P = NP, *then* $E \leq_R \mathsf{id}$ *for every* $E \in$ NP.

Thus, if P = NP, then $\mathsf{GI} \equiv_R \mathsf{id}$ and therefore $\mathsf{GROUP} <_R \mathsf{id}$.

5 Finitary Equivalence Relations

In this section, we note that there is a canonical initial segment of the $\leq_R$ hierarchy for P-equivalence relations. We also show that if P $\neq$ NP, then there are NP-equivalence relations strictly above id. These results are obtained by considering a special class of equivalence relations.

Definition 5.1 An equivalence relation E is called *finitary* if E has only finitely many nontrivial equivalence classes, that is, all but finitely many E-equivalence classes are singletons.

It is easy to see that if E is a finitary equivalence relation on Σ^*, then $E \in$ P if and only if each E-equivalence class is in P if and only if $E \leq_R \mathsf{id}$.

We consider two subclasses of finitary equivalence relations. The first class consists of those with only finitely many equivalence classes. The canonical examples of such equivalence relations are the congruence relations on natural numbers. For each positive $n \in \mathbb{N}$, denote by $\equiv_n$ the congruence relation mod n, that is,

$$x \equiv_n y \iff x \equiv y \bmod n$$

for $x, y \in \mathbb{N}$. Up to $\equiv_R$, the equivalence relations $\equiv_n$ are exactly the same as all P-equivalence relations with finitely many equivalence classes. Moreover, they form an infinite ascending chain that is an initial segment of the $\leq_R$ hierarchy.

Next, we consider equivalence relations induced by a single set.

Definition 5.2 For any subset S of Σ^*, we define an equivalence relation R_S on Σ^* by

$$\langle x, y\rangle \in \mathsf{R}_S \iff \text{either } x = y \text{ or both } x \in S \text{ and } y \in S.$$

Equivalence relations of the form R_S are of course finitary. About their mutual reducibility, we have the following observation.

Lemma 5.3 *Let* $S, T \subseteq \Sigma^*$. *If* $\mathsf{R}_S \leq_R \mathsf{R}_T$, *then either* $\mathsf{R}_S \leq_R \mathsf{id}$ *or* S *is polynomial-time reducible to* T.

Proof Suppose $\mathsf{R}_S \leq_R \mathsf{R}_T$ via a strong reduction function $f \in$ PF. Then for any $x, y \in S$, $\langle f(x), f(y)\rangle \in \mathsf{R}_T$. We have two cases. Case 1: for any $x \in S$, $f(x) \notin T$. In this case, $f(x) = f(y)$ for any $x, y \in S$. Then, f witnesses that $\mathsf{R}_S \leq_R \mathsf{id}$. Case 2: for any $x \in S$, $f(x) \in T$. Note that, for any $x \notin S$, $f(x) \notin T$. Thus, f is a reduction function from S to T. $\square$

Note that id is itself of the form R_S, with $S = \emptyset$. Also, note that there is a coinfinite $S \subseteq \Sigma^*$ with $\mathsf{R}_S <_R \mathsf{id}$. An example is $\mathsf{R}_{\Sigma^* \setminus \{0\}^*} \equiv_R \mathsf{E}_\lambda$.

Proposition 5.4 *Let* $S \subseteq \Sigma^*$ *be nonempty. Then the following hold.*

(i) $S \in$ NP *if and only if* $\mathsf{R}_S \in$ NP.

(ii) *If* S *is* NP-*hard, then* R_S *is* NP-*hard as a set.*

(iii) If $\mathsf{R}_T \leq_R \mathsf{R}_S$ *for all* $T \subseteq \Sigma^*$ *with* $T \in \mathrm{NP}$*, then* S *is* NP*-hard.*
(iv) If $\mathrm{P} \neq \mathrm{NP}$ *and* S *is* NP*-hard, then* $\mathsf{R}_S \not\leq_R \mathsf{id}$.

Proof For (i), the implication ($\Rightarrow$) follows from the definition. For ($\Leftarrow$) of (i) and for (ii), fix an element $a \in S$. Then for any $x \in \Sigma^*$, $x \in S$ if and only if $\langle x, a\rangle \in \mathsf{R}_S$. The function $x \mapsto \langle x, a\rangle$ is a polynomial-time computable function reducing S to R_S as sets.

Now (iv) follows immediately from (ii). To prove (iii), assume $\mathsf{R}_T \leq_R \mathsf{R}_S$ for all NP subsets $T \subseteq \Sigma^*$. In particular, $\mathsf{id} \leq_R \mathsf{R}_S$, and it follows that S is coinfinite. We argue for (iii) in two cases. If $\mathrm{P} = \mathrm{NP}$, then any nonempty, proper subset of Σ^* is NP-hard. Suppose $\mathrm{P} \neq \mathrm{NP}$. Let T be NP-hard. Then $\mathsf{R}_T \not\leq_R \mathsf{id}$ by (iv). By Lemma 5.3 we must have a polynomial-time reduction from T to S, and hence, S is NP-hard. □

Now it is easy to define NP-complete $S \subseteq \Sigma^*$ for which $\mathsf{id} \leq_R \mathsf{R}_S$. If $\mathrm{P} \neq \mathrm{NP}$, then $\mathsf{id} <_R \mathsf{R}_S$ for such an S.

The following notion is in some sense a dual to the notion of finitary equivalence relations.

Definition 5.5 An equivalence relation E is called *finite* if every E-equivalence class is finite.

The equivalence relations $\mathsf{E}_{\varphi,A}$ we defined earlier in this article and the natural isomorphism relations for finite structures are all finite equivalence relations. In particular, GI is finite.

The following proposition shows that the two notions are indeed orthogonal in terms of reducibility.

Proposition 5.6 *Let* E *and* F *be equivalence relations on* Σ^**. Suppose* E *is finitary and* F *is finite. Then the following hold.*

(a) If $E \leq_R F$*, then* $E \leq_R \mathsf{id}$*, and in particular,* $E \in \mathrm{P}$.
(b) If $F \leq_R E$*, then* $F \leq_R \mathsf{id}$*, and in particular,* $F \in \mathrm{P}$.

Proof To prove (a), we only need to verify that every E-equivalence class is in P. But any strong reduction function $f \in \mathrm{PF}$ witnesses the reduction of any E-equivalence class to an F-equivalence class, and the latter, being finite, is in P.

For (b), assume that $f \in \mathrm{PF}$ is a strong reduction function from F to E. Let $X = \{x \in \Sigma^* : \text{the } E\text{-equivalence class of } x \text{ is trivial}\}$. Then $\Sigma^* \setminus f^{-1}(X)$ consists of only finitely many F-equivalence classes and, therefore, is finite. Define $g(x) = f(x)$ if $x \in f^{-1}(X)$, and let $g(x)$ be the lexicographically least element of the F-equivalence class of $f(x)$ if $x \notin f^{-1}(X)$. Then $g \in \mathrm{PF}$ and is a strong reduction function from F to id. □

We define a class of finite equivalence relations induced from a single set.

Definition 5.7 For any subset S of Σ^*, we define an equivalence relation D_S on Σ^* by

$$\langle x, y\rangle \in \mathsf{D}_S \iff \text{either } x = y \text{ or } x \upharpoonright \big(|x| - 1\big) = y \upharpoonright \big(|y| - 1\big) \in S.$$

D_S is a finite equivalence relation since Σ is finite. Note that $x \mapsto x^\frown 0$ is a strong reduction function from id to D_S. Thus, $\mathsf{id} \leq_R \mathsf{D}_S$ for all $S \subseteq \Sigma^*$.

Proposition 5.8 *Let* $S \subseteq \Sigma^*$*. The following are equivalent:*

(i) $\mathsf{D}_S \in \mathrm{P}$*;*
(ii) $S \in \mathrm{P}$*;*
(iii) $\mathsf{D}_S \leq_R \mathsf{id}$*;*
(iv) $\mathsf{D}_S \equiv_R \mathsf{id}$.

Proof The implications (ii)⇒(i)⇔(iii) are obvious, and (iii)⇔(iv) follows from the above remark. It suffices to show (i)⇒(ii). Then for any $x \in \Sigma^*$, $x \in S$ if and only if $\langle x^\frown 0, x^\frown 1\rangle \in \mathsf{D}_S$. Thus, if $\mathsf{D}_S \in \mathrm{P}$, then $S \in \mathrm{P}$. □

The above proof also gives the following proposition as an immediate corollary, which we state without proof.

Proposition 5.9 *Let $S \subseteq \Sigma^*$. Then the following hold.*

(i) $S \in \mathrm{NP}$ *if and only if* $\mathsf{D}_S \in \mathrm{NP}$.
(ii) If S is NP*-hard, then* D_S *is* NP*-hard as a set.*
(iii) If $\mathrm{P} \neq \mathrm{NP}$ *and S is* NP*-hard, then* $\mathsf{id} <_R \mathsf{D}_S$.

Thus, if $\mathrm{P} \neq \mathrm{NP}$ and S is NP-complete, then both R_S and D_S are NP-equivalence relations, but they are incomparable in the $\leq_R$ partial order.

6 Open Problems

Results from [3] and this article show that it is possible to prove reducibility and nonreducibility results about $\leq_R$ without assumptions on the relationship between complexity classes. The following open problems might lead to such results.

Problem 6.1 Does $E \leq_R \mathsf{id}$ for all P-equivalence relations E?

Problem 6.2 Does $E \leq_R \mathsf{GI}$ for all *finite* NP-equivalence relations E?

References

[1] A. Blass and Y. Gurevich, "Equivalence relations, invariants, and normal forms," *SIAM Journal on Computing*, vol. 13 (1984), pp. 682–89. Zbl 0545.68035. MR 0764172. DOI 10.1137/0213042. 272, 282

[2] A. Blass and Y. Gurevich, "Equivalence relations, invariants, and normal forms, II," pp. 24–42 in *Logic and Machines: Decision Problems and Complexity (Münster, 1983)*, vol. 171 of *Lecture Notes in Computer Science*, Springer, Berlin, 1984. Zbl 0545.68036. MR 0775154. DOI 10.1007/3-540-13331-3_31. 272

[3] S. Buss, Y. Chen, J. Flum, S.-D. Friedman, and M. Müller, "Strong isomorphism reductions in complexity theory," *Journal of Symbolic Logic*, vol. 76 (2011), pp. 1381–1402. Zbl 1248.03060. MR 2895401. DOI 10.2178/jsl/1318338855. 272, 273, 274, 279, 280, 282, 284

[4] J. Finkelstein and B. Hescott, "Polynomial time kernel reductions," preprint, arxiv:1604.08558v1 [cs.CC]. 272

[5] L. Fortnow and J. A. Grochow, "Complexity classes of equivalence problems revisited," *Information and Computation*, vol. 209 (2011), pp. 748–63. Zbl 1238.68061. MR 2790842. DOI 10.1016/j.ic.2011.01.006. 272

[6] J. E. Hopcroft and R. E. Tarjan, "Isomorphism of planar graphs," pp. 131–152, 187–212 in *Complexity of Computer Computations (Proc. Sympos., IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y., 1972)*, edited by R. E. Miller and J. W. Thatcher, Plenum, New York, 1972. Zbl 0436.05021. MR 0403302. DOI 10.1007/978-1-4684-2001-2_13. 279

[7] G. L. Miller, "Graph isomorphism, general remarks," *Journal of Computer and System Sciences*, vol. 18 (1979), pp. 128–42. Zbl 0403.03029. DOI 10.1016/0022-0000(79)90043-6. 279

[8] C. H. Papadimitriou, *Computational Complexity*, Addison-Wesley, Reading, Mass., 1994. Zbl 0833.68049. MR 1251285. 279

Acknowledgments

We would like to thank Yijia Chen for useful discussions on the topic of this article. We would also like to thank the anonymous referee for many useful comments and suggestions.

Gao's work was supported by National Science Foundation grant DMS-1201290. Ziegler's work was supported by National Science Foundation grant DMS-0943870.

Gao
Department of Mathematics
University of North Texas
Denton, Texas 76203
USA
sgao@unt.edu
http://www.math.unt.edu/~sgao

Ziegler
Department of Mathematics
University of North Texas
Denton, Texas 76203
USA
calebziegler@my.unt.edu