The Informational Content of Canonical Disjoint NP-Pairs

Christian Glaßer[*] Alan L. Selman[†] Liyu Zhang[‡]

June 17, 2009

Abstract

We investigate the connection between propositional proof systems and their canonical pairs. It is known that simulations between propositional proof systems translate to reductions between their canonical pairs. We focus on the opposite direction and study the following questions.

Q1: For which propositional proof systems f and g does the implication $[can(f) \leq_m^p can(g) \Rightarrow f \leq g]$ hold, and for which does it fail?

Q2: For which propositional proof systems of different strengths are the canonical pairs equivalent?

Q3: What do (non-)equivalent canonical pairs tell about the corresponding propositional proof systems?

Q4: Is every NP-pair (A, B), where A is NP-complete, strongly many-one equivalent to the canonical pair of some propositional proof system?

In short, we show that Q1 and Q2 can be answered with 'almost all', which generalizes previous results by Pudlák and Beyersdorff. Regarding Q3, inequivalent canonical pairs tell that the propositional proof systems are not "very similar," while equivalent, P-inseparable canonical pairs tell that they are not "very different." We can relate Q4 to the open problem in structural complexity that asks whether unions of disjoint NP-complete sets are NP-complete. This demonstrates a new connection between propositional proof systems, disjoint NP-pairs, and unions of disjoint NP-complete sets.

1 Introduction

One reason it is important to study canonical pairs of propositional proof systems (proof systems) is their role in connecting proof systems with disjoint NP-pairs (NP-pairs) [6]. Razborov [15] first defined the canonical pair, $can(f) = (\mathrm{SAT}^*, \mathrm{REF}(f))$, for every proof system f. He showed that if

[*] Lehrstuhl für Informatik IV, Universität Würzburg, Am Hubland, 97074 Würzburg, Germany. Email: glasser@informatik.uni-wuerzburg.de

[†] Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY 14260, USA. Research partially supported by NSF grant CCR-0307077. Email: selman@cse.buffalo.edu

[‡] Department of Computer Science and Computer Information Systems, University of Texas at Brownsville, Brownsville, TX, 78520, USA. Email: Liyu.Zhang@utb.edu

there exists an optimal proof system f, then its canonical pair is a complete pair for DisjNP. Pudlák [14] related canonical NP-pairs of proof systems to the reflection principle and *automatizabilities* of proof systems. In particular, he showed that the canonical NP-pair of a propositional proof system is P-*separable* if and only if the propositional proof system is simulated by an automatizable propositional proof system. In a recent paper [7], we show that every NP-pair is polynomial-time many-one equivalent to the canonical pair of some proof system. So the degree structure of the class of NP-pairs and of all canonical pairs is identical.

Beyersdorff [1] studies proof systems and their canonical pairs from a proof theoretic point of view. He defines the subclasses DNPP(P) of NP-pairs that are representable in some proof system P and shows that the canonical pairs of P are complete for DNPP(P). This interesting result tells us that for certain meaningful subclasses of NP-pairs, complete pairs do exist. Beyersdorff also compares the simulation order of proof systems with the hardness of their canonical pairs, which we will address in this paper too.

Encouraged by the above exciting results on proof systems and their canonical pairs, we continue this line of research and concentrate on the following fundamental correspondence between proof systems and NP-pairs. For proof systems f and g,

$$f \leq g \quad \Rightarrow \quad can(f) \leq_m^p can(g). \tag{1}$$

Pudlák [14] and Beyersdorff [1] give counter examples for the converse implication. This raises the following questions which we investigate in this paper.

Q1: For which proof systems does the following implication hold, and for which proof systems does it fail?

$$can(f) \leq_m^p can(g) \quad \Rightarrow \quad f \leq g \tag{2}$$

Q2: For which proof systems of different strengths are the canonical pairs equivalent?

Q3: What do (non-)equivalent canonical pairs tell about the corresponding proof systems?

Moreover, it is known that every NP-pair is many-one equivalent to the canonical pair of some proof system [7]. Here we investigate the same question for strongly many-one reductions. It is easy to see that this question must be restricted to pairs whose first component is NP-complete.

Q4: Is every NP-pair (A, B), where A is NP-complete, strongly many-one equivalent to the canonical pair of some proof system?

Theorem 3.3 addresses the first part of Q1: The theorem asserts that, for any two disjoint NP-pairs (A, B) and (C, D), there are proof systems f and g such that $can(f) \equiv_m^p (A, B)$, $can(g) \equiv_m^p (C, D)$ and implication (2) holds nontrivially.

Corollary 3.6 addresses the second part of Q1: The following assertion is equivalent to the reasonable assumption that optimal proof systems do not exist. For every proof system f there is a proof system g such that f and g is a counter example to implication (2). More strongly, there is an infinite chain of proof systems $g_0, g_1, \cdots$, such that $f < g_0 < g_1 < \cdots$, but the canonical pairs of all of these proof systems are many-one equivalent. In this way, we address Q2.

In section 4 we answer Q3 in different ways. Equivalent canonical pairs do not tell much about the mere simulation order of two proof systems (Theorem 4.1). However, inequivalent canonical pairs tell us that the corresponding proof systems do not simulate each other except on a P-subset of TAUT (Proposition 4.2). Hence such systems are not "very similar." In contrast, equivalent, P-inseparable canonical pairs tell us that none of the corresponding proof systems is almost everywhere super-polynomially stronger than the other one (Theorem 4.3). In other words, such proof systems must simulate each other *infinitely often* (Corollary 4.4). So the proof systems are not "very different."

In section 5 we can relate Q4 to the open problem in structural complexity [3, 5] that asks whether unions of disjoint NP-complete sets are NP-complete. We show under the hypothesis NP $\neq$ coNP that if Q4 has an affirmative answer, then unions of disjoint NP-complete sets are NP-complete. This demonstrates a new connection between proof systems, NP-pairs, and problems in structural complexity. Finally, in section 6 we obtain connections between proof systems and the Turing-degrees of their canonical pairs.

2 Preliminaries

A disjoint NP-pair is a pair (A, B) of nonempty sets A and B such that $A, B \in$ NP and $A \cap B = \emptyset$. Let DisjNP denote the class of all disjoint NP-pairs.

Given a disjoint NP-pair (A, B), a *separator* is a set S such that $A \subseteq S$ and $B \subseteq \overline{S}$; we say that S *separates* (A, B). Let $Sep(A, B)$ denote the set of all separators of (A, B). For disjoint NP-pairs (A, B), the fundamental question is whether $Sep(A, B)$ contains a set belonging to P. In that case the pair is P-*separable*; otherwise, the pair is P-*inseparable*. There is evidence [8, 4] that P-inseparable disjoint NP-pairs exist, and this will be our main hypothesis in the paper. The following proposition summarizes known results about P-inseparability.

Proposition 2.1

1. P $\neq$ NP $\cap$ coNP *implies that* DisjNP *contains a* P*-inseparable pair.*
2. P $\neq$ UP *implies that* DisjNP *contains a* P*-inseparable pair [8].*
3. *If* DisjNP *contains* P*-inseparable pairs, then it contains a* P*-inseparable pair whose components are* NP*-complete [8].*

Item 1 in the above proposition is an easy fact since for any $L \in \text{NP} \cap \text{coNP} - P$, $(L, \overline{L})$ is a P-inseparable disjoint NP-pair. While it is probably the case that DisjNP contains P-inseparable pairs, there is an oracle relative to which P $\neq$ NP and P-inseparable pairs in DisjNP do not exist [9]. So P $\neq$ NP probably is not a sufficiently strong hypothesis to show the existence of P-inseparable pairs in DisjNP. On the other hand, if there exist secure public-key cryptosystems (for example, if RSA cannot be cracked in polynomial time), then there exist P-inseparable disjoint NP-pairs [8].

All reducibilities in the paper are polynomial-time reducibilities. We review the natural notions of reducibilities between disjoint pairs. The original notions are nonuniform [8]. Here we state only the known equivalent uniform versions [8, 4].

Definition 2.1 *Let (A, B) and (C, D) be disjoint pairs.*

1. *(A, B) is* many-one reducible in polynomial time *to (C, D), $(A, B) \leq_m^p (C, D)$, if there exists a polynomial-time computable function $f : \Sigma^* \rightarrow \Sigma^*$ such that $f(A) \subseteq C$ and $f(B) \subseteq D$.*
2. *(A, B) is* Turing reducible in polynomial time *to (C, D), $(A, B) \leq_T^p (C, D)$, if there exists a polynomial-time oracle Turing machine M such that for every separator S of (C, D), $L(M, S)$ is a separator of (A, B), where $L(M, S)$ denotes the language accepted by M with oracle S.*

Köbler, Messner, and Torán [10] define the following stronger version of many-one reductions between disjoint NP-pairs:

Definition 2.2 *Let (A, B) and (C, D) be disjoint pairs. (A, B) is* strongly many-one reducible in polynomial time *to (C, D), $(A, B) \leq_{sm}^{\mathrm{p}} (C, D)$, if there exists a polynomial-time computable function f such that $f(A) \subseteq C$, $f(B) \subseteq D$, and $f(\overline{A \cup B}) \subseteq \overline{C \cup D}$.*

Note that if $(A, B) \leq_{sm}^{\mathrm{p}} (C, D)$, then A reduces to C and B reduces D via the same polynomial-time many-one reduction.

Let SAT denote the set of satisfiable propositional formulas and let UNSAT $\overset{df}{=} \overline{\mathrm{SAT}}$. Moreover, let TAUT denote the set of tautologies. It is well known that SAT is many-one complete for NP and both UNSAT and UNSAT are many-one complete for coNP.

Definition 2.3 *A disjoint pair (A,B) is $\leq_m^p$-hard for* NP *if for every separator L of (A, B),* SAT$\leq_m^p$ *L.*

Definition 2.4 *For any disjoint pair (A, B), the* polynomial-time Turing-degree *(Turing-degree for short) of (A, B) is defined as*

$$\mathbf{d}(A, B) = \{(C, D) \mid (C, D) \textit{ is a disjoint pair and } (A, B) \equiv_T^p (C, D)\}.$$

In an earlier paper [7] we investigated the restriction of Turing-degrees of disjoint pairs on DisjNP and showed that every countable distributive lattice can be embedded into the interval between any two comparable but inequivalent restricted Turing-degrees of disjoint NP-pairs. It follows trivially that every countable distributive lattice can be embedded into the interval between any two comparable but inequivalent Turing-degrees of disjoint pairs if both degrees contain some disjoint NP-pair.

Cook and Reckhow [2] defined a *propositional proof system* (proof system for short) to be a function $f : \Sigma^* \to \mathrm{TAUT}$ such that f is onto and f is polynomial-time computable. For every tautology α, if $f(w) = \alpha$, then we say w is an *f-proof* of α. Throughout this paper, we fix a polynomial-time computable and polynomial-time invertible pairing function $\langle\cdot,\cdot\rangle$ such that $|\langle v, w\rangle| = 2|vw|$.

The *canonical* NP*-pair* (canonical pair for short) of f [15, 14] is the disjoint NP-pair $(\mathrm{SAT}^*, \mathrm{REF}(f))$, denoted by $can(f)$, where

$$\begin{aligned} \mathrm{SAT}^* &= \{(x, 0^n) \,|\, x \in \mathrm{SAT} \text{ and } n \geq 0 \text{ is an integer}\} \quad \text{and} \\ \mathrm{REF}(f) &= \{(x, 0^n) \,|\, \neg x \in \mathrm{TAUT} \text{ and } \exists y[|y| \leq n \text{ and } f(y) = \neg x]\}. \end{aligned}$$

Conversely, for every disjoint NP-pair (A, B), we can define a proof system $f_{A,B}$ as follows. Choose a g that is polynomial-time computable and polynomial-time invertible such that $A \leq_m^p \mathrm{SAT}$ via g. Let N be an NP-machine that accepts B in time p.

$$f_{A,B}(z) \overset{df}{=} \begin{cases} \neg g(x) & : \text{ if } z = \langle x, w\rangle,\ |w| = p(|x|),\ N(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w\rangle,\ |w| \neq p(|x|),\ |z| \geq 2^{|x|},\ x \in \mathrm{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

Clearly, $f_{A,B}$ is a propositional proof system for every disjoint NP-pair (A, B).

Theorem 2.2 ([7]) *For every disjoint* NP*-pair (A, B), $(A, B) \equiv_m^p can(f_{A,B})$.*

Let f and f' be two propositional proof systems. We say that f *simulates* f' ($f' \leq f$) if there is a polynomial p and a function $h : \Sigma^* \to \Sigma^*$ such that for every $w \in \Sigma^*$, $f(h(w)) = f'(w)$ and $|h(w)| \leq p(|w|)$. Furthermore, if the function h can be computed in polynomial time, f *p-simulates* f' ($f' \leq_p f$). A proof system is *(p-)optimal* if it (p-)simulates every other proof system.

In Section 4, we will need the following generalization of the concept "simulation." We say that f *simulates f' on a subset S* of TAUT, if there is a polynomial p and a function $h : \Sigma^* \to \Sigma^*$ such that for every $w \in \Sigma^*$, $f'(w) \in S$ implies that $f(h(w)) = f'(w)$ and $|h(w)| \leq p(|w|)$. Moreover, f *simulates f' except on a subset S* of TAUT, if f simulates f' on $\mathrm{TAUT} - S$. Obviously, a proof system f simulates a proof system f' if and only if f simulates f' on TAUT. We say a proof system f *simulates* another proof system g *infinitely often* if f simulates g on an infinite set $S \subseteq \mathrm{TAUT}$.

We use $f < g$ to denote that $f \leq g$ and $g \not\leq f$. We use $(A, B) <_m^p (C, D)$ to denote that $(A, B) \leq_m^p (C, D)$ and $(C, D) \not\leq_m^p (A, B)$. We use similar notations for other reductions ($\leq_{sm}^{\mathrm{p}}$, $\leq_T^p$) between disjoint pairs.

3 Proof Systems and Many-One Degrees of Canonical Pairs

We recall the fundamental relation between the simulation order of proof systems and reducibility of their canonical NP-pairs.

Proposition 3.1 ([14, 7]) *Let f and g be proof systems.*

$$f \leq g \quad \Rightarrow \quad can(f) \leq_m^p can(g)$$

In this section, we investigate the converse of the above proposition. We show results that address both parts of Q1 and Q2.

We start our investigations with the observation that refuting an implication that is slightly weaker than (2) is equivalent to proving the existence of P-inseparable disjoint NP-pairs. This is done by a purely complexity theoretic proof that does not rely on specific properties of concrete proof systems.

Theorem 3.2 *The following statements are equivalent.*

1. P*-inseparable disjoint* NP*-pairs exist.*
2. *There exist proof systems f and g such that $can(f) <_m^p can(g) \not\Rightarrow f \leq g$.*

Proof. If P-inseparable disjoint NP-pairs do not exist, then all canonical pairs of proof systems are P-separable and hence are equivalent. This shows 2 ⇒ 1.

For the other direction, assume that P-inseparable disjoint NP-pairs exist and define the following set of propositional formulas.

$$\begin{array}{rl}\text{EASY} \overset{df}{=} \{x \,\big|\, & x \text{ is a propositional formula such that } x = (b \vee \bar{b} \vee y) \\ & \text{for a suitable variable } b \text{ and a suitable formula } y\}\end{array}$$

EASY is a subset of TAUT. Also, EASY $\in$ P. Let true $\overset{df}{=} (b \vee \bar{b} \vee b)$ and define a proof system as follows.

$$f(z) \overset{df}{=} \begin{cases} x & : \text{ if } z = \langle x, \varepsilon \rangle \text{ and } x \in \text{EASY} \\ x & : \text{ if } z = \langle x, y \rangle \text{ and } |y| > 2^{|x|} \text{ and } x \in \text{TAUT} \\ \text{true} & : \text{ otherwise.} \end{cases}$$

Note that f is a proof system. Observe that the elements in EASY are the only tautologies that have polynomial-size f-proofs. All other tautologies do not have polynomial-size f-proofs. This makes $can(f)$ P-separable which is witnessed by the following separator:

$$S = \{(x, 0^n) \,\big|\, [n \leq 2^{|x|} \text{ and } \neg x \notin \text{EASY}] \text{ or } [n > 2^{|x|} \text{ and } x \in \text{SAT}]\}$$

By assumption there exists a P-inseparable disjoint NP-pair (A, B). Hence, by Theorem 2.2 there exists a proof system g' such that $can(g')$ and (A, B) are many-one equivalent. Now define another proof system.

$$g(z) \overset{df}{=} \begin{cases} g'(w) & : \text{ if } z = 0w \text{ and } g'(w) \notin \text{EASY} \\ \text{true} & : \text{ if } z = 0w \text{ and } g'(w) \in \text{EASY} \\ x & : \text{ if } z = 1w,\, w = \langle x, y \rangle,\, |y| = 2^{|x|}, \text{ and } x \in \text{EASY} \\ \text{true} & : \text{ otherwise.} \end{cases}$$

Note that g is a proof system. Observe that formulas in EASY $-\{\text{true}\}$ do not have polynomial-size g-proofs. It follows that g does not simulate f, since f provides polynomial-size proofs for elements in EASY.

Now we verify that $can(g') \leq_m^p can(g)$ via the reduction that maps $(x, 0^n)$ to $(x, 0^{n+1})$. If $(x, 0^n) \in$ SAT*, then $(x, 0^{n+1}) \in$ SAT* and we are done. Let $(x, 0^n) \in \text{REF}(g')$. So there exists some w such that $|w| \leq n$ and $g'(w) = (\neg x)$. Note that $(\neg x) \notin$ EASY, since formulas in EASY do not start with a negation. From the definition of g it follows that $g(0w) = g'(w) = (\neg x)$. So $(x, 0^{n+1}) \in \text{REF}(g)$.

So $can(g') \leq_m^p can(g)$ and therefore, $(A, B) \leq_m^p can(g)$. Hence $can(g)$ is P-inseparable. This shows $can(f) <_m^p can(g)$. □

The examples given by Pudlák [14] and Beyersdorff [1] show that the simulation order of proof systems is not necessarily reflected by the reducibility of their canonical pairs. However, as the next theorem shows, the canonical pairs of proof systems that satisfy implication (2) in a non-trivial way, vary over all degrees of disjoint NP-pairs. More precisely, for each pair of many-one degrees of disjoint NP-pairs, there do exist proof systems whose canonical pairs lie in the respective degrees such that their simulation order is consistent with the reducibility of the canonical pairs. This answers the first part of Q1 in the sense that implication (2) can be satisfied non-trivially for arbitrary canonical pairs.

Theorem 3.3 *Let (A, B) and (C, D) be disjoint* NP*-pairs such that $(A, B) \leq_m^p (C, D)$. Then there exist proof systems f_1 and f_2 such that all of the following holds.*

- $can(f_1) \equiv_m^p (A, B)$
- $can(f_2) \equiv_m^p (C, D)$
- $f_1 \leq_p f_2$

Proof. Choose g_1 that is polynomial-time computable and polynomial-time invertible such that $A \leq_m^p$ SAT via g_1. Let N_1 be an NP-machine that accepts B in time p_1. Define the following function f_1.

$$f_1(z) \overset{df}{=} \begin{cases} \neg g_1(x) & : \text{ if } z = \langle x, w \rangle,\ |w| = p_1(|x|),\ N_1(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w \rangle,\ |w| \neq p_1(|x|),\ |z| \geq 2^{|x|},\ x \in \text{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

The proof of Theorem 2.2 shows that f_1 is a proof system and $can(f_1) \equiv_m^p (A, B)$. Now choose g_2 that is polynomial-time computable and polynomial-time invertible such that $C \leq_m^p$ SAT via g_2. Let N_2 be an NP-machine that accepts D in time p_2. Without loss of generality, we assume for every $n \geq 0$, $p_1(n) \neq p_2(n)$ and $range(g_1) \cap range(g_2) = \emptyset$. Define the following function f_2.

$$f_2(z) \overset{df}{=} \begin{cases} \neg g_1(x) & : \text{ if } z = \langle x, w\rangle,\ |w| = p_1(|x|),\ N_1(x) \text{ accepts along path } w \\ \neg g_2(x) & : \text{ if } z = \langle x, w\rangle,\ |w| = p_2(|x|),\ N_2(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w\rangle,\ |w| \neq p_i(|x|) \text{ for } i = 1, 2,\ |z| \geq 2^{|x|},\ x \in \text{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

Clearly f_2 is also a proof system, since for every tautology y,

$$f_2(\langle y, 0^{2^{|y|}}\rangle) = y.$$

Also, we notice that each f_1-proof z is also an f_2-proof for the same tautology except for $z \in \{\langle x, w\rangle \mid |w| = p_2(|x|) \wedge |\langle x, w\rangle| \geq 2^{|x|} \wedge x \in \text{TAUT}\}$, which is a finite set. So, $f_1 \leq_p f_2$.

It remains to show $can(f_2) \equiv_m^p (C, D)$. We only show $can(f_2) \leq_m^p (C, D)$. The proof for $(C, D) \leq_m^p can(f_2)$ is the same as that for $(A, B) \leq_m^p can(f_1)$, for which we refer the reader to [7].

Let g many-one reduce (A, B) to (C, D). Choose elements $c \in C$ and $d \in D$. Define a reduction function h as follows.

```
1   input $(y, 0^n)$
2   if $n \geq 2^{|y|+1}$ then
3      if $y \in \mathrm{SAT}$ then output c else output d
4   endif
5   if $g_1^{-1}(y)$ exists then output $g(g_1^{-1}(y))$
6   if $g_2^{-1}(y)$ exists then output $g_2^{-1}(y)$
7   output c
```

The exhaustive search in line 3 is possible in quadratic time in n. So h is polynomial-time computable.

Assume $(y, 0^n) \in \text{SAT}^*$. Then $y \in \text{SAT}$. If we reach line 3, then we output $c \in C$. Otherwise we reach line 5. If $g_1^{-1}(y)$ exists (hence, $g_2^{-1}(y)$ does not exist, since the ranges of g_1 and g_2 are disjoint), then $g_1^{-1}(y) \in A$ and so, $g(g_1^{-1}(y)) \in C$. Otherwise we reach line 6. If $g_2^{-1}(y)$ exists, then $g_2^{-1}(y) \in C$ as $y \in \text{SAT}$. Otherwise we reach line 7. Therefore, in all cases (output made in line 5, 6 or 7), we output an element in C.

Assume $(y, 0^n) \in \text{REF}(f_2)$ (in particular $y \in \text{UNSAT}$). So there exists z such that $|z| \leq n$ and $f_2(z) = \neg y$. If we reach line 3, then we output $d \in D$. Otherwise we reach line 5. So far we have $\neg y \neq true$ and $|z| \leq n < 2^{|y|+1}$. Therefore, $f_2(z) = \neg y$ must be due to line 1 or line 2 in the definition of f_2. It follows that either $g_1^{-1}(y)$ exists or $g_2^{-1}(y)$ exists (but not both). If $g_1^{-1}(y)$ exists, then $g_1^{-1}(y) \in B$ (by line 1 of f_2's definition) and we output $g(g_1^{-1}(y))$, which belongs to D. Otherwise, $g_2^{-1}(y)$ exists and we output $g_2^{-1}(y)$, which belongs to D as well (by line 2 of f_2's definition). This shows $can(f_2) \leq_m^p (C, D)$ via h and finishes the proof of Theorem 3.3. $\square$

The proof system g constructed in Theorem 3.2 might seem "pathological," since in this proof system, tautologies from a polynomial-time decidable subset of TAUT have proofs of super-polynomial length. One might wonder whether Theorem 3.2 can be proved without such pathology. The corresponding proof systems are formalized as follows.

Definition 3.1 *A proof system f is* well-behaved *if for every polynomial-time decidable $S \subseteq \mathrm{TAUT}$ there exists a polynomial p such that for all $x \in S$,*

$$\min\{|w| \mid f(w) = x\} \leq p(|x|).$$

However, well-behaved proof systems probably do not even exist. It has been known [11, 12] that the existence of well-behaved proof systems is equivalent to the existence of optimal proof systems, which we believe not to exist (Messner and Torán [13] and Glaßer et al. [4] give evidence for this). So it is probably the case that no proof system is well-behaved and therefore, every proof system has long proofs on some polynomial-time decidable subset of TAUT. This shows that the proof system constructed in Theorem 3.2 is not uncommon. Even more, we can apply the arguments used in Theorem 3.2 to every non-well-behaved proof system. This allows us to obtain the following general result.

Theorem 3.4 *Let f be a proof system that is not well-behaved. For every $(A, B) \in \mathrm{DisjNP}$, there exists a proof system g such that*

- *$can(g) \equiv^p_m (A, B)$ and*
- *$g \not\leq f$.*

Proof. Let f be a proof system that is not well-behaved. Then there exists a set $S \subseteq \mathrm{TAUT}$ such that $S \in \mathrm{P}$ and for every polynomial p, there exists $x \in S$ and $\min\{|w| \mid f(w) = x\} > p(|x|)$.

Let $(A, B) \in \mathrm{DisjNP}$. By Theorem 3.1 in Glaßer et al. [7], there exists a proof system g' such that $can(g') \equiv^p_m (A, B)$. Now define proof system g as follows:

$$g(z) \overset{df}{=} \begin{cases} g'(w) & : \text{ if } z = 0w, \\ w & : \text{ if } z = 1w \text{ and } w \in S, \\ true & : \text{ otherwise.} \end{cases}$$

Clearly, $g' \leq g$. So by Proposition 3.1, it holds that $can(g') \leq^p_m can(g)$ and hence, $(A, B) \leq^p_m can(g)$.

Now we show $can(g) \leq^p_m can(g')$ and hence $can(g) \leq^p_m (A, B)$. Let $n_0 = \min\{|w| \mid g'(w) = \neg false\}$. So $(false, 0^{n_0}) \in \mathrm{REF}(g')$. The reduction is the following function h:

$$h(x, 0^n) \overset{df}{=} \begin{cases} (false, 0^{n_0}) & : \text{ if } \neg x \in S, \\ (x, 0^{n-1}) & : \text{ otherwise.} \end{cases}$$

Clearly h is polynomial-time computable. Assume $(x, 0^n) \in \mathrm{SAT}^*$, then $x \in \mathrm{SAT}$ and hence, $\neg x \notin S$. So $h(x, 0^n) = (x, 0^{n-1}) \in \mathrm{SAT}^*$. Now assume $(x, 0^n) \in \mathrm{REF}(g)$. So $\neg x \in \mathrm{TAUT}$ and there exists z such that $|z| \leq n$ and $g(z) = \neg x$. If $\neg x \in S$, then $h(x, 0^n) = (false, 0^{n_0}) \in \mathrm{REF}(g')$. If $\neg x \notin S$, then for every w, $g(1w) \neq \neg x$. So it must hold that $z = 0w$ for some w and $g(z) = g'(w) = \neg x$. Since $|w| = |z| - 1$, $(x, 0^{n-1}) \in \mathrm{REF}(g')$. This shows $can(g) \leq_m^p can(g')$ and hence, $can(g) \leq_m^p (A, B)$.

It remains to show $g \not\leq f$. Suppose $g \leq f$. Then there exists a polynomial q such that for every w, there exists w' such that $|w'| \leq q(|w|)$ and $g(w) = f(w')$. Let $p(n) = q(n+1)$. Let $x \in S$ be such that $\min\{|w| \mid f(w) = x\} > p(|x|) = q(|x|+1)$. By the definition of g, $g(1x) = x$. Now for $w = 1x$, since $g \leq f$, there exists w' such that $|w'| \leq q(|1x|) = p(|x|)$ and $f(w') = g(w) = x$. This contradicts the fact that $\min\{|w| \mid f(w) = x\} > p(|x|) = q(|x|+1)$. □

With help of Theorem 3.4 we can now give an answer to Q2: All non-well-behaved proof systems provide examples for proof systems that have equivalent canonical pairs, but that differ with respect to their strengths. Moreover, we can answer the second part of Q1 in the sense that all non-well-behaved proof systems provide counter examples for implication (2).

Corollary 3.5 *For every proof system f that is not well-behaved, there exists a proof system g such that $can(f) \equiv_m^p can(g)$ and $f < g$. In particular,*

$$can(g) \leq_m^p can(f) \quad \not\Rightarrow \quad g \leq f.$$

Proof. The proof is the same as Theorem 3.4 except that we take (A, B) to be $can(f)$ and g' to be f. □

If we assume that optimal proof systems do not exist, then Corollary 3.5 provides even stronger answers: With regard to Q1, *all* proof systems provide counter examples for the implication (2). With regard to Q2, *all* proof systems provide examples that have equivalent canonical pairs, but that differ with respect to their strengths. Even more, each proof system is the origin of an infinite, strictly ascending chain of proof systems whose canonical pairs are equivalent.

Corollary 3.6 *The following statements are equivalent.*

1. *Optimal proof systems do not exist.*
2. *For every proof system f there exists a proof system g such that $can(f) \equiv_m^p can(g)$ and $f < g$.*
3. *For every proof system f there exists an infinite chain of proof systems $g_0, g_1, \ldots$ such that $f < g_0 < g_1 < \cdots$ and $can(f) \equiv_m^p can(g_0) \equiv_m^p can(g_1)\ \cdots$.*
4. *For every proof system f there exists a proof system g such that*

$$can(g) \leq_m^p can(f) \quad \not\Rightarrow \quad g \leq f.$$

Proof. The implication 3 ⇒ 4 is trivial by choosing $g = g_0$. If an optimal proof system f exists, then $can(f)$ is $\leq_m^p$-complete for DisjNP. So if f is an optimal proof system, then for every proof system g it holds that $can(g) \leq_m^p can(f)$ and $g \leq f$. This shows 4 ⇒ 1. The implication 1 ⇒ 2 follows from Corollary 3.5 together with Messner's result [12] that the non-existence of optimal proof systems implies the non-existence of well-behaved proof systems. The remaining implication 2 ⇒ 3 follows by a repeated application of statement 2. □

4 Proof Systems With Equivalent Canonical Pairs

We have seen in the last section that the degree structure of canonical pairs does not necessarily reflect the simulation order of the corresponding proof systems. In this section we study the following related question.

Q3: What do (non-)equivalent canonical pairs tell about the corresponding proof systems?

We answer Q3 in different ways. Equivalent canonical pairs do not tell much about the mere simulation order of two proof systems (Theorem 4.1). However, inequivalent canonical pairs tell us that the corresponding proof systems do not simulate each other except on a P-subset of TAUT (Proposition 4.2). Hence such systems are not "very similar." In contrast, equivalent, P-inseparable canonical pairs tell us that none of the corresponding proof systems is almost everywhere super-polynomially stronger than the other one (Theorem 4.3). So the proof systems are not "very different."

Assuming the hypothesis that P-inseparable disjoint NP-pairs exist, we show the following boundaries for the statements above: There exist proof systems whose canonical pairs are not many-one equivalent, but they simulate each other except on an NP-subset of TAUT (Corollary 4.9). For every P-inseparable NP-pair (A, B), there exist proof systems f and g whose canonical pairs are many-one equivalent to (A, B), and for every P-subset S of TAUT it holds that f and g do not simulate each other on TAUT $- S$ (Theorem 4.11).

We first show that equivalent canonical pairs do not tell much about the simulation order of two proof systems.

Theorem 4.1 *For every disjoint* NP*-pair (A, B), there exist proof systems f, g, and h such that*

- $can(f) \equiv_m^p can(g) \equiv_m^p can(h) \equiv_m^p (A, B)$,
- $f < g$ *and* $f < h$,
- $g \not\leq h$ *and* $h \not\leq g$.

Proof. Let $f = f_{A,B}$. By Theorem 2.2, $can(f) \equiv_m^p (A, B)$. Define two subsets of TAUT as follows.

$$\text{EASY1} \overset{df}{=} \{x \mid x \text{ is a propositional formula such that } x = (b \vee (\neg b) \vee y) \text{ for a suitable variable } b \text{ and a suitable formula } y\}$$

$$\text{EASY2} \overset{df}{=} \{x \mid x \text{ is a propositional formula such that } x = (b \vee b \vee (\neg b) \vee y) \text{ for a suitable variable } b \text{ and a suitable formula } y\}$$

It is obvious that EASY1 $\cap$ EASY2 $= \emptyset$ and both EASY1 and EASY2 belong to P. Also note that proof system f does not have polynomial-bounded proofs for tautologies in EASY1 $\cup$ EASY2 (since formulas having short f-proofs either equal *true* or start with the symbol $\neg$).

Now define proof systems g and h as follows.

$$g(z) \overset{df}{=} \begin{cases} f(w) & : \text{ if } z = 0w \\ w & : \text{ if } z = 1w \text{ and } w \in \text{EASY1} \\ \mathit{true} & : \text{ otherwise} \end{cases}$$

$$h(z) \overset{df}{=} \begin{cases} f(w) & : \text{ if } z = 0w \\ w & : \text{ if } z = 1w \text{ and } w \in \text{EASY2} \\ \mathit{true} & : \text{ otherwise} \end{cases}$$

Clearly, both proof systems, g and h, simulate f, since for every w, $f(w) = g(0w) = h(0w)$. Also note that proof system g (resp., h) has polynomial-size proofs for tautologies in EASY1 (resp., EASY2), but does not have polynomial-size proofs for tautologies in EASY2 (resp., EASY1). Therefore, f does not simulate g or h. Neither do proof systems g and h simulate each other.

An argument similar to that in the proof of Theorem 3.2 shows that $can(g)$ and $can(h)$ are many-one equivalent to $can(f)$. Therefore, $can(f) \equiv_m^p can(g) \equiv_m^p can(h) \equiv_m^p (A, B)$. □

However, from another point of view, the proof systems defined in the proof of Theorem 4.1 are actually quite "similar" to each other. They differ only super-polynomially on a polynomial-time decidable subset of TAUT. More precisely, the construction of proof systems with equivalent canonical pairs but arbitrary simulation order hinges on the following fact.

Proposition 4.2 *If proof systems f and g simulate each other except on a* P*-subset of* TAUT*, then $can(f) \equiv_m^p can(g)$.*

So here the question is whether we can construct proof systems f and g with equivalent canonical pairs such that the proof systems are "very different." For example, do there exist proof systems f and g such that $can(f) \equiv_m^p can(g)$ and f is almost everywhere super-polynomially stronger than g? The following theorem shows that such an extreme difference is only possible for proof systems whose canonical pairs are P-separable.

Theorem 4.3 *Let f and g be proof systems such that $can(g) \leq_m^p can(f)$. If for almost all tautologies x and for every polynomial p, the length of the shortest f-proof of x is not bounded by p in the length of the shortest g-proof of x, then $can(f)$ and $can(g)$ are* P*-separable.*

Proof. Fix proof system f and g that satisfy the premise of the theorem. Without loss of generality, we assume $g(\lambda)$ does not start with $\neg$. Let h many-one reduce $can(g)$ to $can(f)$. Assume g and h can be computed in time n^k and n^l, respectively. By hypothesis, let n_0 be the minimal integer such that for every tautology x with $|x| > n_0$ the length of the shortest f-proof of x is not bounded by the polynomial $n^{(k+1)l+1}$ in the length of the shortest g-proof of x.

We use the following algorithm to separate $can(g)$:

```
0   input $\langle x, 0^n \rangle$
1   $y = x$, $m = n$
2   while ($|y| \leq m^k$) and ($m > 0$)
3       compute $\langle y', 0^{m'} \rangle = h(\langle y, 0^m \rangle)$
4       if $|y'| \leq n_0$ then (accept iff $y' \in \mathrm{SAT}$)
5       let $\langle y, 0^m \rangle = \langle y', 0^{(m')^{\frac{1}{(k+1)l+1}}} \rangle$
6   accept
```

We first claim that the while-loop in the above algorithm runs for at most n times. To see this, we just need to observe by line 3 that

$$m' \leq |\langle y, 0^m \rangle|^l \leq (m^k + m)^l < m^{(k+1)l+1}.$$

So the assignment in line 5, which sets $m = (m')^{\frac{1}{(k+1)l+1}}$, decrements m. Since initially m is set to n and decremented in each iteration of the while-loop, the claim holds. This implies that the above algorithm is polynomial-time computable.

Now we prove the correctness of the algorithm. Assume the input $\langle x, 0^n \rangle \in \mathrm{SAT}^*$. By line 3, it holds for each iteration of the while-loop that $y \in \mathrm{SAT}$ implies $y' \in \mathrm{SAT}$, since h many-one reduces $can(g)$ to $can(f)$. Since initially $y = x \in \mathrm{SAT}$ and y is set to y' by line 5 in each iteration, the loop-invariant $y' \in \mathrm{SAT}$ holds. Therefore, the algorithm either accepts in line 4 or accepts in line 6.

Assume the input $\langle x, 0^n \rangle \in \mathrm{REF}(g)$. We first show the loop invariant $(y, 0^m) \in \mathrm{REF}(g)$. By line 1 the invariant holds initially. So suppose the invariant holds at the beginning of some iteration of the loop. If the algorithm reaches line 3, then $\langle y', 0^{m'} \rangle \in \mathrm{REF}(f)$, since h many-one reduces $can(g)$ to $can(f)$. So in line 4, $y' \in \mathrm{UNSAT}$ and hence, the algorithm rejects if $|y'| \leq n_0$. Otherwise, the algorithm reaches line 5. By the choice of n_0, it holds that $(y', 0^{m'}) \in \mathrm{REF}(f)$ implies $(y', 0^{(m')^{\frac{1}{(k+1)l+1}}}) \in \mathrm{REF}(g)$. This proves the invariant $(y, 0^m) \in \mathrm{REF}(g)$.

Suppose the algorithm reaches line 6. Then either $|y| > m^k$ or $m = 0$. By the loop invariant $(y, 0^m) \in \mathrm{REF}(g)$, there exists w with $|w| \leq m$ and $g(w) = \neg y$. This shows $|y| \leq m^k$, since g can be computed in time n^k. So it must hold that $m = 0$ and hence $g(\lambda) = \neg y$. This is a contradiction, because by our hypothesis, $g(\lambda)$ does not start with $\neg$. Therefore, we do not reach line 6. So the algorithm must exit in line 4 where it rejects. This shows that $can(g)$ is P-separable.

As f has longer proofs than g for almost all tautologies, it follows trivially that $f \leq g$. Hence, by Proposition 3.1 it holds that $can(f) {\leq}_m^p can(g)$. This implies that $can(f)$ is P-separable too. $\square$

Corollary 4.4 *Let f and g be propositional proof systems such that $can(f) {\equiv}_m^p can(g)$ and both $can(f)$ and $can(g)$ are* P*-inseparable. Then f and g must simulate each other infinitely often.*

Proof. The corollary follows immediately from the contrapositive of Theorem 4.3. $\square$

Let us summarize what we have seen so far: Proposition 4.2 says that if two proof systems are "very similar," then they have equivalent canonical pairs. Theorem 4.3 tells us that if two proof systems are "very different" from each other, then either they have P-separable canonical pairs or their canonical pairs are inequivalent.

We continue to follow the question to what extent can proof systems differ while still having equivalent canonical pairs. Under the hypothesis that P-inseparable disjoint NP-pairs exist, we show that Proposition 4.2 does not hold when the P-subset is replaced with an NP-subset (Corollary 4.9). So altering f-proofs on a P-subset of TAUT does not change the many-one degree of $can(f)$, but altering f-proofs on an NP-subset of TAUT can do so.

Theorem 4.5 *Let f be a proof system such that $can(f)$ is not m-complete for* DisjNP*. Then there exists a proof system f' such that $can(f) {<}_m^p\, can(f')$ and f and f' simulate each other except on an* NP*-subset of* TAUT*.*

Proof. Let f be a proof system such that $can(f)$ is not m-complete for DisjNP. Since $can(f)$ is not m-complete, there exists a disjoint NP-pair (A, B) such that $(A, B) \not\leq_m^p can(f)$. Let $(C, D) = (0A \cup 1\mathrm{SAT}^*, 0B \cup 1\mathrm{REF}(f))$. Clearly, $can(f) {\leq}_m^p (C, D)$, but $(C, D) \not\leq_m^p can(f)$.

Choose g that is polynomial-time computable and polynomial-time invertible such that $C \leq_m^p \mathrm{SAT}$ via g. Let N be an NP-machine that accepts D in time p. Define a function f' as follows.

$$f'(z) \overset{\mathrm{df}}{=} \begin{cases} \neg g(x) & : \text{ if } z = 0\langle x, w\rangle,\ |w| = p(|x|),\ N(x) \text{ accepts along path } w \\ x & : \text{ if } z = 1w, \text{ and } f(w) = x \\ \mathit{true} & : \text{ otherwise} \end{cases}$$

We first observe that f' is a proof system. Clearly f' is polynomial-time computable. To see $range(f') \subseteq$ TAUT, we just need to observe in the first case in the definition of f' that $g(x) \in$ UNSAT, since $N(x)$ accepts along path w implies $x \in D \subseteq \overline{C}$. Also, it is obvious that $range(f') \supseteq$ TAUT since $range(f') \supseteq range(f) =$ TAUT.

Note that f' is similar to the proof system $f_{C,D}$. The difference is only that the trivial proofs in $f_{C,D}$ are replaced by f-proofs here. This makes f' at least as strong as f.

Claim 4.6 *$f \le f'$.*

Proof. The proof is trivial since for every w, $f'(1w) = f(w)$. □

Claim 4.7 *$(C, D) {\le}_m^p can(f')$.*

Proof. The reduction is given by $h(x) = (g(x), 0^{2(|x|+p|x|)+1})$. Clearly h is polynomial-time computable. Assume $x \in C$. Then $g(x) \in \mathrm{SAT}$ and hence, $h(x) \in \mathrm{SAT}^*$. Assume $x \in D$. Let w be a witness of x of length $p(|x|)$. Then for $z = 0\langle x, w\rangle$ with $|z| = 2(|x| + p(|x|)) + 1$, it holds that $f'(z) = \neg g(x)$. So $h(x) \in \mathrm{REF}(f')$. □

Claim 4.8 *For all tautologies $x \notin \neg g(D) \cup \{true\}$, x has an f'-proof of length n implies x has an f-proof of length $n - 1$.*

Proof. This is clear from the definition of f'. □

Claim 4.6 implies $can(f) {\le}_m^p can(f')$ (Proposition 3.1). Claim 4.7 implies $can(f') \not\le_m^p can(f)$, since otherwise $(C, D) {\le}_m^p can(f)$ which is not true. Claim 4.8 together with Claim 4.6 shows that f and f' simulate each other on all tautologies except on $\neg g(D) \cup \{true\}$. The latter is an NP-subset of TAUT, since g is polynomial-time invertible and $D \in \mathrm{NP}$. □

Corollary 4.9 *The following statements are equivalent.*

1. P*-inseparable* NP*-pairs exist.*
2. *There exist proof systems f and g whose canonical pairs are not many-one equivalent, but that simulate each other except on an* NP*-subset of* TAUT*.*

Proof. This follows from Theorem 4.5, since if P-inseparable NP-pairs exist, then P-separable NP-pairs are not m-complete for DisjNP. □

Corollary 4.10 *If* $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$*, then there exist proof systems f and g whose canonical pairs are not many-one equivalent, but that simulate each other except on an* NP*-subset of* TAUT*.*

Proof. This follows from Corollary 4.9, since $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$ implies that P-inseparable NP-pairs exist. □

Under the hypothesis that P-inseparable disjoint NP-pairs exist, we now show that proof systems that do not simulate each other except on P-subsets of TAUT may still have equivalent canonical pairs. Hence, the converse of Proposition 4.2 does not hold, unless P-inseparable disjoint NP-pairs do not exist.

Theorem 4.11 *Let (A,B) be a* P*-inseparable* NP*-pair. Then there exist proof systems f and f' such that $can(f) \equiv_m^p can(f') \equiv_m^p (A, B)$ and for every* P*-subset S of* TAUT *it holds that f and f' do not simulate each other on* TAUT $- S$.

Proof. Consider the proof system $f = f_{A,B}$. Clearly f has short proofs on $\neg g(B)$ and trivial (but long) proofs otherwise, where g is a polynomial-time computable and invertible many-one reduction from A to SAT. It is easy to define another polynomial-time computable and invertible many-one reduction g' from A to SAT such that $range(g) \cap range(g') = \emptyset$. Let f' be the proof system that is obtained from $f_{A,B}$ when g is replaced by g'. Then by Theorem 2.2 $can(f) \equiv_m^p (A, B) \equiv_m^p can(f')$. We show that f cannot simulate f' on TAUT $- S$ for any P-subset S of TAUT.

Assume that for some P-subset S of TAUT it holds that f simulates f' on TAUT $- S$. Note that on $\neg g'(B)$, f-proofs have exponential length while f'-proofs have polynomial length. Therefore, (TAUT $- S$) $\cap \neg g'(B)$ is a finite set. So $\overline{S}$, which is a set in P, separates a finite variation of the NP-pair $(\overline{\mathrm{TAUT}}, \neg g'(B))$. However, $(\overline{\mathrm{TAUT}}, \neg g'(B))$ cannot be P-separable, because $(A, B) \leq_m^p (\overline{\mathrm{TAUT}}, \neg g'(B))$ via the reduction $h(x) = \neg g'(x)$.

Symmetric arguments show that f' cannot simulate f on TAUT $- S$ for any P-subset S of TAUT. □

5 Strongly Many-One Degrees of Canonical Pairs

Every disjoint NP-pair is many-one equivalent to the canonical pair of some proof system [7]. We ask the same question for strongly many-one reductions. Note that if a disjoint NP-pair (A, B) is strongly many-one equivalent to the canonical pair of some proof system, then trivially A must be NP-complete. So we arrive at the following question.

Q4: Is every NP-pair (A, B), where A is NP-complete, strongly many-one equivalent to the canonical pair of some proof system?

Surprisingly, this question is closely related to the following open problem, which has been studied for quite a while [16, 3, 5].

Q5: Is the union of two disjoint NP-complete sets NP-complete?

For this, we first translate Q4 into the question whether certain NP-pairs are many-one hard for NP (Corollary 5.5). From this we show under the hypothesis NP $\neq$ coNP that if Q4 has an affirmative answer, then Q5 has an affirmative answer. In order to show this implication, it turned out that it suffices to demand that Q4 has answer 'yes' only for $A =$ SAT (Corollary 5.6).

Theorem 5.1 *Let (A, B) be a disjoint* NP*-pair. If $(A, \overline{A \cup B})$ is $\leq_m^p$-hard for* NP*, then there exists a proof system f such that $(\mathrm{SAT}^*, \mathrm{REF}(f)) \equiv_{sm}^p (A, B)$.*

Proof. Choose a one-one, length-increasing, polynomial-time computable, polynomial-time invertible function g such that $A \leq_m^p \mathrm{SAT}$ via g. Such a g exists, since SAT is a paddable NP-complete set. Moreover, let r be a polynomial-time-computable function that witnesses $(\mathrm{SAT}, \overline{\mathrm{SAT}}) \leq_m^p (A, \overline{A \cup B})$. Let N be an NP-machine that accepts B in time p, and choose a constant $c > 0$ such that for all n, $p(n) < 2^n + c$.

$$f(z) \overset{df}{=} \begin{cases} \neg g(x) & : \text{ if } z = \langle x, w \rangle,\ |w| = p(|x|),\ N(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w \rangle,\ |w| = 2^{|x|} + c,\ x \in \mathrm{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

The function is polynomial-time computable, since in the second case, $|z|$ is large enough so that $x \in \mathrm{TAUT}$ can be decided by the brute force algorithm in deterministic time $O(|z|^2)$. In the first case of f's definition, $x \in B$ and so $g(x) \notin \mathrm{SAT}$. It follows that $f : \Sigma^* \to \mathrm{TAUT}$. The mapping is onto, since for every tautology y,

$$f(\langle y, 0^{2^{|y|}+c} \rangle) = y.$$

Therefore, f is a propositional proof system.

Claim 5.2 $(\mathrm{SAT}^*, \mathrm{REF}(f)) \leq_{sm}^{\mathrm{p}} (A, B)$.

Proof. Choose elements $a \in A$ and $b \in B$. The reduction function h is defined as follows.

```
0  input $(y, 0^n)$
1  if $n \geq 2(|\neg y| + 2^{|\neg y|} + c)$ then
2     if $y \in \mathrm{SAT}$ then output a else output b
3  endif
4  if $g^{-1}(y)$ exists and $n \geq 2(|g^{-1}(y)| + p(|g^{-1}(y)|))$ then output $g^{-1}(y)$
5  output $r(y)$
```

Observe that the exhaustive search in line 2 is possible in quadratic time in n. So h is computable in polynomial time. We show that h achieves the asserted reduction.

Assume $(y, 0^n) \in \mathrm{SAT}^*$, i.e., $y \in \mathrm{SAT}$. If we reach line 2, then we output $a \in A$. Otherwise we reach line 4. If $g^{-1}(y)$ exists, then it belongs to A, since g reduces A to SAT. Moreover, $r(y) \in A$. So any output made in lines 4 or 5 belongs to A.

Assume $(y, 0^n) \in \mathrm{REF}(f)$ and hence $\neg y \in \mathrm{TAUT}$. If $n \geq 2(|\neg y| + 2^{|\neg y|} + c)$, then the output is $b \in B$. Otherwise, $n < 2(|\neg y| + 2^{|\neg y|} + c)$ and we reach line 4. By assumption, there exists a string z of length $\leq n$ such that $f(z) = \neg y$. Note that $f(z)$ is not defined according to the third line of f's definition, since the expression 'true' does not start with the character '$\neg$'. Also, $f(z)$ is not defined according to the second line of f's definition, since there, $n \geq |z| = 2(|\neg y| + 2^{|\neg y|} + c)$. So $f(z)$ must be defined according to the first line of f's definition. Therefore, for some $x \in B$, $y = g(x)$ and $n \geq |z| = 2(|x| + p(|x|))$. This shows that $g^{-1}(y)$ exists, that $g^{-1}(y) = x \in B$, and that $n \geq 2(|g^{-1}(y)| + p(|g^{-1}(y)|))$. So if the algorithm reaches line 4, then the output is made in line 4 and this output is a string from B.

Assume $(y, 0^n) \notin \mathrm{SAT}^* \cup \mathrm{REF}(f)$ and hence $\neg y \in \mathrm{TAUT}$. For $z = \langle \neg y, 0^{2^{|\neg y|+c}} \rangle$ it holds that $|z| = 2(|\neg y| + 2^{|\neg y|} + c)$ and $f(z) = \neg y$. So $n < 2(|\neg y| + 2^{|\neg y|} + c)$, since otherwise $(y, 0^n) \in \mathrm{REF}(f)$ witnessed by z. Hence we reach line 4. Assume that the output is made in line 4, i.e., $g^{-1}(y)$ exists and $n \geq 2(|g^{-1}(y)| + p(|g^{-1}(y)|))$. Note that $g^{-1}(y) \notin A$, since g reduces A to SAT. Suppose $x \overset{df}{=} g^{-1}(y)$ belongs to B. Let $z = \langle x, w \rangle$ where w is an accepting path of $N(x)$. So $f(z) = \neg g(x) = \neg y$ and

$$n \geq 2(|g^{-1}(y)| + p(|g^{-1}(y)|)) = 2(|x| + p(|x|)) = |z|.$$

Hence $(y, 0^n) \in \mathrm{REF}(f)$ which contradicts our assumption. Therefore, $x = g^{-1}(y) \notin B$. This shows that any output that is made in line 4 does not belong to $A \cup B$. It remains the case where the output is made in line 5. Here $r(y) \notin A \cup B$, since $(\mathrm{SAT}, \overline{\mathrm{SAT}}) \leq_m^p (A, \overline{A \cup B})$ via reduction r.

This shows $(\mathrm{SAT}^*, \mathrm{REF}(f)) \leq_{sm}^{\mathrm{p}} (A, B)$ via h, which proves Claim 5.2. □

Claim 5.3 $(A, B) \leq_{sm}^{\mathrm{p}} (\mathrm{SAT}^*, \mathrm{REF}(f))$.

Proof. The reduction is $h(x) \overset{df}{=} (g(x), 0^{2(|x|+p(|x|))})$.

If $x \in A$, then $g(x) \in \mathrm{SAT}$ and therefore, $h(x) \in \mathrm{SAT}^*$. Assume now $x \in B$. Let w be an accepting path of $N(x)$ and define $z \overset{df}{=} \langle x, w \rangle$. So $|z| = 2(|x| + p(|x|))$ and hence $f(z) = \neg g(x)$. Therefore, $h(x) \in \mathrm{REF}(f)$.

Finally, let us assume $x \notin A \cup B$. Hence $g(x) \notin \mathrm{SAT}$ and so $h(x) \notin \mathrm{SAT}^*$. Suppose $h(x) \in \mathrm{REF}(f)$, i.e., there exists a z such that $|z| \leq 2(|x| + p(|x|))$ and $f(z) = \neg g(x)$. Note that $f(z)$ is not defined according to the third line of f's definition, since the expression 'true' does not start with the character '$\neg$'. Also, $f(z)$ is not defined according to the second line of f's definition, since there, $|z| = 2(|\neg g(x)| + 2^{|\neg g(x)|} + c) > 2(|x| + p(|x|))$ (recall that g is length-increasing). So $f(z)$ must be defined according to the first line of f's definition. Hence $z = \langle x', w' \rangle$ such that $|w'| = p(|x'|)$ and $M(x')$ accepts along path w'. So $\neg g(x) = f(z) = \neg g(x')$. From the fact that g is one-one we obtain $x = x'$. Therefore, $x \in B$ which contradicts our assumption. This shows $h(x) \notin \mathrm{REF}(f)$ and hence $h(x) \notin \mathrm{SAT}^* \cup \mathrm{REF}(f)$. This finishes the proof of Claim 5.3. □

The theorem follows from the Claim 5.2 and 5.3. □

Proposition 5.4 *Let (A, B) be a disjoint* NP*-pair such that $A \cup B \neq \Sigma^*$. If there exists a proof system f such that* $(\mathrm{SAT}^*, \mathrm{REF}(f)) \equiv_{sm}^{p} (A, B)$*, then* $(A, \overline{A \cup B})$ *is $\leq_m^p$-hard for* NP.

Proof. Let g be a reduction that witnesses $(\mathrm{SAT}^*, \mathrm{REF}(f)) \leq_{sm}^{\mathrm{p}} (A, B)$. Fix some $c \in \overline{A \cup B}$. We claim that $(\mathrm{SAT}, \overline{\mathrm{SAT}}) \leq_m^p (A, \overline{A \cup B})$ via the following reduction.

$$h(x) \overset{df}{=} \begin{cases} g(x,\lambda) & : \text{ if } f(\lambda) \neq \neg x \\ c & : \text{ otherwise} \end{cases}$$

If $x \in \mathrm{SAT}$, then $(x,\lambda) \in \mathrm{SAT}^*$ and hence $g(x,\lambda) \in A$. Note that $f(\lambda) \neq \neg x$. Therefore, $h(x) \in A$.

Assume now that $x \in \overline{\mathrm{SAT}}$. So $(x,\lambda) \notin \mathrm{SAT}^*$ and hence $g(x,\lambda) \notin A$. Together with $c \notin A$ we obtain $h(x) \notin A$. Suppose $h(x) \in B$. So $f(\lambda) \neq \neg x$ and $h(x) = g(x,\lambda)$. Thus $g(x,\lambda) \in B$ and $(x,\lambda) \in \mathrm{REF}(f)$. It follows that $f(\lambda) = \neg x$ which is a contradiction. Therefore, $h(x) \notin A \cup B$. □

Corollary 5.5 *The following are equivalent for a disjoint* NP*-pair* (A,B) *where* $A \cup B \neq \Sigma^*$.

1. $(A, \overline{A \cup B})$ *is* $\leq_m^p$*-hard for* NP.
2. *There exists a proof system* f *such that* $(\mathrm{SAT}^*, \mathrm{REF}(f)) \equiv_{sm}^p (A,B)$.

Proof. This follows from Theorem 5.1 and Proposition 5.4. □

Corollary 5.6 *Assume* NP $\neq$ coNP. *If for all disjoint* NP*-pairs* (SAT, B) *there exists a proof system* f *such that* $(\mathrm{SAT}^*, \mathrm{REF}(f)) \equiv_{sm}^p (\mathrm{SAT}, B)$, *then unions of disjoint* NP*-complete sets are* NP*-complete.*

Proof. Assume NP $\neq$ coNP and assume that for all disjoint NP-pairs (SAT, B) there exists a proof system f such that $(\mathrm{SAT}^*, \mathrm{REF}(f)) \equiv_{sm}^p (\mathrm{SAT}, B)$. Fix some $B \in \mathrm{NP}$ such that $\mathrm{SAT} \cap B = \emptyset$. From NP $\neq$ coNP it follows that $A \cup B \neq \Sigma^*$. By Corollary 5.5, $(\mathrm{SAT}, \overline{\mathrm{SAT} \cup B})$ is $\leq_m^p$-hard for NP. Therefore, $\mathrm{SAT} \cup B$ is NP-complete. So we have shown that for every $B \in \mathrm{NP}$, if $\mathrm{SAT} \cap B = \emptyset$, then $\mathrm{SAT} \cup B$ is NP-complete. By Theorem 5.7 in Glaßer et al. [3], unions of disjoint NP-complete sets are NP-complete. □

6 Proof Systems and Turing-Degrees of Canonical Pairs

In this section, we consider the connection between proof systems and the more general Turing-degrees of their canonical pairs. We try to generalize some of the results in previous sections on many-one reductions to Turing reductions.

Proposition 6.1 *Let* f *and* g *be proof systems such that* $can(f) \leq_T^p can(g)$. *Then there exists a proof system* g' *such that* $can(g') \equiv_T^p can(g)$ *and* $f \leq_p g'$.

Proof. Define proof system g' as follows:

$$g'(w) = \begin{cases} f(w') & \text{if } w = 0w' \\ g(w') & \text{if } w = 1w' \end{cases}$$

Clearly, both f and g are p-simulated by g'. So, $can(f) \leq_T^p can(g) \leq_m^p can(g')$. It remains to show $can(g') \leq_T^p can(g)$. Let $can(f)$ be Turing reducible to $can(g)$ via a polynomial-time oracle Turing machine M. Then $can(g')$ is Turing reducible to $can(g)$ via the following polynomial-time oracle Turing machine M': On input $(x, 0^n)$, if $(x, 0^{n-1}) \notin S$, then reject; otherwise accept if and only if M^S accepts $(x, 0^{n-1})$, where S is the oracle set.

The correctness of M' can be seen as follows. Let S be a separator of $can(g)$. If $(x, 0^n) \in \mathrm{SAT}^*$, then $x \in \mathrm{SAT}$. This implies $(x, 0^{n-1}) \in S$ and $M^S(x, 0^{n-1})$ accepts and hence, $(M')^S(x, 0^n)$ accepts. On the other hand, if $(x, 0^n) \in \mathrm{REF}(g')$, then by the definition of g', either $(x, 0^{n-1}) \in \mathrm{REF}(f)$ or $(x, 0^{n-1}) \in \mathrm{REF}(g)$. If $(x, 0^{n-1}) \in \mathrm{REF}(g)$, then $(x, 0^{n-1}) \notin S$ and hence, $(M')^S(x, 0^n)$ rejects. Otherwise, $(x, 0^{n-1}) \in \mathrm{REF}(f)$ and hence $(M)^S(x, 0^{n-1})$ rejects. This implies that $(M')^S(x, 0^n)$ rejects. □

Corollary 6.2 *Let $\mathbf{d}_1 < \mathbf{d}_2$ be two Turing-degrees of disjoint* NP*-pairs. Then for every proof system f such that $can(f) \in \mathbf{d}_1$, there exists a proof system g such that $can(g) \in \mathbf{d}_2$ and $f < g$.*

Proof. By Theorem 2.2, we can choose a proof system g' such that $can(g') \in \mathbf{d}_2$. By Proposition 3.1, $g' \not\leq f$. By Proposition 6.1, there exists a proof system g such that $f \leq_p g$ and $g \in \mathbf{d}_2$. Also, $g \not\leq f$, since otherwise $can(g) \leq_m^p can(f)$. □

Proposition 6.3 *For all disjoint* NP*-pairs (A, B) and (C, D) such that $(A, B) <_T^p (C, D)$, there exist proof systems f and g such that*

- *$can(f) \equiv_m^p (A, B)$,*
- *$can(g) \equiv_m^p (C, D)$, and*
- *$f \not\leq g$ and $g \not\leq f$.*

Proof. By Theorem 2.2 we obtain a proof system g such that $can(g) \equiv_m^p (C, D)$. From the proof of this theorem it is clear that g is not well-behaved. (For example, the tautologies in the set $S = \{a \vee (\neg a) \vee y \mid y \text{ is a propositional formula}\}$ do not have g-proofs of polynomial length.) Now apply Theorem 3.4 to g and (A, B). We obtain a proof system f such that $can(f) \equiv_m^p (A, B)$ and $f \not\leq g$. Note that $g \not\leq f$ as well, since otherwise, by Proposition 3.1, $(C, D) \leq_m^p (A, B)$. □

References

[1] O. Beyersdorff. Disjoint NP-pairs from propositional proof systems. In *Proceedings 3rd Conference on Theory and Applications of Models of Computation*, volume 3959 of *Lecture Notes in Computer Science*, pages 236–247, 2006.

[2] S. Cook and R. Reckhow. The relative efficiency of propositional proof systems. *The Journal of Symbolic Logic*, 44(1):36–50, 1979.

[3] C. Glaßer, A. Pavan, A. Selman, and S. Sengupta. Properties of NP-complete sets. *SIAM Journal on Computing*, 36(2):516–542, 2006.

[4] C. Glaßer, A. Selman, S. Sengupta, and L. Zhang. Disjoint NP-pairs. *SIAM Journal on Computing*, 33(6):1369–1416, 2004.

[5] C. Glaßer, A. Selman, S. Travers, and K. Wagner. The complexity of unions of disjoint sets. *Journal of Computer and System Sciences*, 74:1173–1187, May 2008.

[6] C. Glaßer, A. Selman, and L. Zhang. Survey of disjoint NP-pairs and relations to propositional proof systems. In O. Goldreich, A. L. Rosenberg, and A. L. Selman, editors, *Theoretical Computer Science - Essays in Memory of Shimon Even*, volume 3895 of *Lecture Notes in Computer Science*. Springer, 2006.

[7] C. Glaßer, A. Selman, and L. Zhang. Canonical disjoint NP-pairs of propositional proof systems. *Theoretical Computer Science*, 370:60–73, 2007.

[8] J. Grollmann and A. Selman. Complexity measures for public-key cryptosystems. *SIAM Journal on Computing*, 17(2):309–335, 1988.

[9] S. Homer and A. Selman. Oracles for structural properties: The isomorphism problem and public-key cryptography. *Journal of Computer and System Sciences*, 44(2):287–301, 1992.

[10] J. Köbler, J. Messner, and J. Torán. Optimal proof systems imply complete sets for promise classes. *Information and Computation*, 184(1):71–92, 2003.

[11] J. Krajíček and P. Pudlák. Propositional proof systems, the consistency of first order theories and the complexity of computations. *The Journal of Symbolic Logic*, 54(3):1063–1079, Sep 1989.

[12] J. Messner. *On the Simulation order of proof systems*. PhD thesis, Universität Ulm, Abteilung Theoretische Informatik, December 2000.

[13] J. Messner and J. Torán. Optimal proof systems for propositional logic and complete sets. In *Proceedings 15th Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science, pages 477–487. Springer Verlag, 1998.

[14] P. Pudlák. On reducibility and symmetry of disjoint NP-pairs. *Theoretical Computer Science*, 295:323–339, 2003.

[15] A. Razborov. On provably disjoint NP-pairs. Technical Report TR94-006, Electronic Computational Complexity Colloquium, 1994.

[16] A. Selman. Natural self-reducible sets. *SIAM Journal on Computing*, 17(5):989–996, October 1988.