Propositional Proof Systems and Their Canonical NP-pairs

Christian Glaßer * Alan L. Selman† Liyu Zhang‡

December 19, 2006

Abstract

We investigate the connection between of proof systems and their canonical pairs. The following list summarizes our results.

1. Let $\mathbf{d}_1 < \mathbf{d}_2$ be Turing degrees of disjoint NP-pairs and let f be a proof system such that $can(f) \in \mathrm{d}_1$. We construct a proof system g such that $can(g) \in \mathrm{d}_2$ and $f \leq_p g$.
2. Let $\mathbf{d}_1 < \mathbf{d}_2$ be many-one degrees of disjoint NP-pairs. We construct proof system f, g such that $can(f) \in \mathrm{d}_1$, $can(g) \in \mathrm{d}_2$, and $f \leq_p g$.
3. Under the assumption that P-inseparable NP-pairs exist, we construct proof system f, g such that $can(f) < can(g)$ but $f \not\leq g$.
4. For all disjoint NP-pairs (A, B) and (C, D) such that $(A, B) <^{pp}_T (C, D)$ there exist proof system f and g such that $can(f) \equiv^{pp}_m (A, B)$, $can(g) \equiv^{pp}_m (C, D)$, and $f \not\leq g$ and $g \not\leq f$.
5. If optimal proof system do not exist, then for every proof system f there exists a proof system g such that $can(f) \equiv^{pp}_m can(g)$ and $f < g$.
6. For every disjoint NP-pair (A, B) there exist proof system f, g, and h whose canonical pairs are equivalent to (A, B) such that $f < g$, $f < h$, $g \not\leq h$, and $h \not\leq g$.
7. If $can(f)$ is P-inseparable and $can(g) \leq^{pp}_m can(f)$, then "$g$ and f cannot differ too much."
8. Robustness of Razborov's implication: Suppose the proof system f and g simulate each other except on some $A \subseteq$ TAUT. If $A \in$ P, then $can(f) \equiv^{pp}_m can(g)$. We construct an $A \in$ NP such that $can(f) < can(g)$.
9. As corollary we obtain: If P $\neq$ NP $\cap$ coNP, then there exist proof system f and g whose canonical pairs are not many-one equivalent, but that simulate each other except on an NP-subset of TAUT.
10. For every disjoint NP-pair (A, B) there exist proof system f and g whose canonical pairs are equivalent to (A, B) such that for every P-subset $S \subseteq$ TAUT, f and g do not simulate each other on TAUT $- S$.

*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. Research partially supported by NSF grant CCR-0307077. Email: selman@cse.buffalo.edu

‡Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY 14260. Email: lzhang7@cse.buffalo.edu

1 Introduction

One reason it is important to study the class DisjNP of all disjoint NP-pairs is its relationship to the theory of proof systems for propositional calculus [GSZ06b]. Specifically, Razborov [Raz94] defined the canonical disjoint NP-pair, $(\mathrm{SAT}^*, \mathrm{REF}_f)$, for every propositional proof system f, and he showed that if there exists an optimal propositional proof system f, then its canonical pair is a complete pair for DisjNP. (We will explain this notation later.) In the same paper he asked for evidence of existence of a propositional proof system whose canonical disjoint NP-pair is not separable by a set belonging to the complexity class P, and, relatedly, he asked whether it is possible to reduce to canonical pairs $(\mathrm{SAT}^*, \mathrm{REF}_f)$, another disjoint NP-pair that we believe to be hard (i.e., not separable by a set in P). In our recent paper [GSZ06a] we answered these questions in the strongest possible way. We proved that every disjoint NP-pair is polynomial-time, many-one equivalent to the canonical disjoint NP-pair of some propositional proof system. It follows immediately that every disjoint NP-pair we believe to be P-inseparable (cannot be separated by a set in P) is many-one equivalent to some pair $(\mathrm{SAT}^*, \mathrm{REF}_f)$ that is also P-inseparable.

Another consequence of the above result by is that the degree structure of the class of all disjoint NP-pairs is identical to that of the class of all canonical disjoint NP-pairs of propositional proof system. Therefore, by examining the degree structure of the class DisjNP, one can understand the degree structure of canonical pairs $(\mathrm{SAT}^*, \mathrm{REF}_f)$. We studied this structure in the same paper and showed that assuming P-inseparable pairs exist, every countable distributive lattice can be embedded into every interval of polynomial degrees of disjoint pairs by maps that preserve the least and greatest element, respectively. Thus, assuming that P-inseparable disjoint NP-pairs exist, the class DisjNP has a rich, dense, degree structure—and each of these degrees contains a canonical pair.

In this paper we continue this line of research and try to understand more precisely the correspondence between the simulation order of propositional proof systems and the degree structure of the corresponding canonical NP-pairs. The following is an easy known fact: if proof system f simulates proof system g, then the pair $(\mathrm{SAT}^*, \mathrm{REF}_g)$ is many-one reducible to the pair $(\mathrm{SAT}^*, \mathrm{REF}_f)$. The question is whether the converse also holds; i.e., whether the reductions between canonical NP-pairs imply the simulation order between the corresponding propositional proof systems. Pudlák [Pud03] gave a concrete example of two propositional proof systems with equivalent canonical NP-pairs where one does not simulate the other. Beyersdorff [Bey06] gave general conditions where one can construct such pair of propositional proof systems. Both results show that in general the reductions between canonical NP-pairs do not necessarily imply the simulation order between the corresponding propositional proof systems. In Section 3 we obtain this result in a stronger sense. More precisely, using only the necessary hypothesis that P-inseparable NP-pairs exist we show that there exist propositional proof systems f and g such that g does not simulate f and the degree of the canonical NP-pair of f is *strictly below* the degree of the canonical NP-pair of g. Furthermore, our proof is purely a complexity proof and does not rely on properties of concrete propositional proof systems. This allows us to further obtain the following stronger and much more general result: assuming that optimal propositional proof systems do not exist, then for *every* propositional proof system f there exists another propositional proof system g whose canonical NP-pair is equivalent to f's such that f is strictly below g in the simulation order. Under the same hypothesis, this also gives an infinite chain of propositional proof systems, all of whose canonical pairs lie in the degree

of (A, B), for *every* disjoint NP-pairs (A, B).[1]

Another interesting question related to the one we studied in Section 3 is to what extent two propositional proof systems with equivalent canonical NP-pairs can be different from each other. We study this question in Section 4. This question was studied earlier by Beyersdorff [Bey06]. He constructed proof systems with natural properties that have equivalent canonical NP-pairs but are not equivalent in the simulation order. We, however, not only construct propositional proof systems of arbitrary simulation order whose canonical NP-pairs are equivalent, but also try to explore the difference between propositional proof systems with equivalent canonical NP-pairs. We consider two propositional proof systems different a subset S of TAUT if the one proof system has much shorter proof than the other for almost all tautologies in S. We show in Section 4 that two propositional proof systems with equivalent canonical NP-pairs can be different on some set that belongs to $\mathrm{NP} - \mathrm{P}$, but not on the set of all tautologies. It is still unknown whether for every subset $S \notin \mathrm{P}$ of TAUT there exist two propositional proof systems f and g with equivalent canonical NP-pairs such that f and and g differ on S.

In Section 5, we investigate the degrees of canonical pairs for the strongly many-one reduction. Strongly many-one reductions were introduced by Köbler et al. [KMT03] and further studied by Glaßer et al. [GSS05] and Beyersdorff [Bey06]. They are equivalent to many-one reductions as regard to the existence of complete disjoint NP-pairs [GSS05]. We show in this paper that the question whether a disjoint NP-pair is strongly many-one equivalent to the canonical NP-pairs of some propositional proof system is closely related to the problem of whether the disjoint union of two NP-complete sets are still NP-complete. The latter has been an important open problem in the study of structural complexity [GPSS04, GSTW06]. The results in this section demonstrate new connections between proof systems and disjoint NP-pairs. This would provide further motivations for the study of disjoint NP-pairs.

In Section 6, we extend some of the results in Section 3 to the more general Turing reductions.

2 Preliminaries

A disjoint NP-pair is a pair (A, B) of nonempty sets A and B such that $A, B \in \mathrm{NP}$ 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 class 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*. As there has been plenty of evidence [GS88, GSSZ04] that P-inseparable disjoint NP-pairs do not exist, we will use it as our main hypothesis in the paper. The following proposition summarizes known results about P-inseparability.

Proposition 2.1

1. $\mathrm{P} \neq \mathrm{NP} \cap co\text{-}\mathrm{NP}$ *implies* NP *contains* P*-inseparable sets.*

[1] (How) should we mention the other result in Section 3?

2. P $\neq$ UP *implies* NP *contains* P*-inseparable sets [GS88].*

3. *If* NP *contains* P*-inseparable sets, then* NP *contains* NP*-complete* P*-inseparable sets [GS88].*

While it is probably the case that NP contains P-inseparable sets, there is an oracle relative to which P $\neq$ NP and P-inseparable sets in NP do not exist [HS92]. So P $\neq$ NP probably is not a sufficiently strong hypothesis to show existence of P-inseparable sets in NP.

We review the natural notions of reducibilities between disjoint pairs. The original notions are nonuniform [GS88]. Here we state only the known equivalent uniform versions [GS88, GSSZ04].

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)*,* $(A, B) \leq_m^{pp} (C, D)$*, if there exists a polynomial-time computable function* f *such that* $f(A) \subseteq C$ *and* $f(B) \subseteq D$*.*

2. (A, B) *is* Turing reducible in polynomial-time *to* (C, D)*,* $(A, B) \leq_T^{pp} (C, D)$*, 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)*.*

Definition 2.2 *For any* $(A, B) \in$ DisjNP*, the* polynomial-time NP-Turing-degree *(* NP*-Turing-degree for short) of* (A, B) *is defined as*

$$\mathbf{d}(A, B) = \{(C, D)\text{DisjNP} \mid (A, B) \equiv_T^{pp} (C, D)\}.$$

Let TAUT denote the set of tautologies. Cook and Reckhow [CR79] defined a *propositional proof system* (proof system for short) to be a function $f : \Sigma^* \to$ TAUT such that f is onto and $f \in$ PF. The canonical pair of f [Raz94, Pud03] is the disjoint NP-pair $(\text{SAT}^*, \text{REF}_f)$, denoted by $can(f)$, where

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

Let f and f' be two propositional proof systems. We say that f *simulates* 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 $|h(w)| \leq p(|w|)$. A proof system is *optimal* if it simulates every other proof system.

In Section 4, we will also need the following generalization of the concept "simulation". We say that f *simulates* f' (*except*, respectively) *on a subset* S of $TAUT$ if there is a polynomial p and a function $h : \Sigma^* \to \Sigma^*$ such that for every $w \in \Sigma^*$, $f'(w) \in S$ ($f'(w) \in$ TAUT $- S$, respectively) implies that $f(h(w)) = f'(w)$ and $|h(w)| \leq p(|w|)$. Apparently, a proof system f simulates a proof system f' if and only if f simulates f' on TAUT (or equivalently, except on $\emptyset$).

3 Proof Systems and Many-one Degrees of Canonical Pairs

The following known proposition gives a relation between simulation orders of proof systems and many-one degrees of their canonical NP-pairs.

Proposition 3.1 [2] *Let* f *and* g *be proof systems. If* g *simulates* f *(*$f \leq g$*), then* $can(f) \leq_m^{pp} can(g)$*.*

[2] Should we give some citation for this? For e.g., Pudl'ak [Pud03]

This gives rise to the interesting question whether the converse it true, as that would give a nice complexity characterization of the simulation order of proof systems. However, we have already known that the converse of Proposition 3.1 is probably not true, because relative to the oracle O_2 constructed in Glaßer et al. [GSSZ04], there exists an infinite, strictly increasing chain of proof systems $f_0 < f_1 < \cdots$ such that the canonical NP-pairs of each f_i is a many-one complete NP-pair [GSZ06a]. Beyersdorff [Bey06] also refuted the converse of Proposition 3.1 assuming existence of non-optimal proof systems that are closed under disjunction.

In the following theorem we refute the converse of Proposition 3.1 in a stronger sense. Note that Our hypothesis is necessary as otherwise the theorem is trivially false. Also, our proof only involves notions and techniques from complexity theory.

Theorem 3.2 *If there exists a P-inseparable disjoint* NP*-pair, then there exist proof system f and g such that*

1. *g does not simulate f and*
2. *$can(f) <_m^{pp} can(g)$.*

Proof. Define the following set of propositional formulas.

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

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

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

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

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

By assumption there exists a P-inseparable disjoint NP-pair (A, B). Hence, as shown in [GSZ05], there exists a proof system g' such that $can(g')$ and (A, B) are many-one equivalent. Now define another proof system.

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

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

Now we verify that $can(g') \leq_m^{pp} can(g)$ via the reduction that maps $(x, 0^n)$ to $(x, 0^{n+1})$. If $(x, 0^n) \in \mathrm{SAT}^*$, then $(x, 0^{n+1}) \in \mathrm{SAT}^*$ and we are done. Let $(x, 0^n) \in \mathrm{REF}_{g'}$. So there exists some w such that $|w| \leq n$ and $g'(w) = (\neg x)$. Note that $(\neg x) \notin \mathrm{EASY}$, 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 $(x, 0^{n+1}) \in \mathrm{REF}_g$.

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

The above theorem shows that the degree structure of canonical pairs do not necessarily reflect the simulation order of their corresponding proof systems. However, as the next theorem shows, for each pair of many-one degrees of canonical pairs, there do exist proof systems whose canonical pairs lie in respective degrees such that their simulation order is consistent with the degree structure of the corresponding canonical pairs.

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

- $can(f_1) \equiv_m^{pp} (A, B)$;
- $can(f_2) \equiv_m^{pp} (C, D)$;
- $f_1 \leq_p f_2$.

Proof. Let $\langle \cdot, \cdot \rangle$ be a polynomial-time computable, polynomial-time invertible pairing function such that $|\langle v, w \rangle| = 2|vw|$. Choose g_1 that is polynomial-time computable and polynomial-time invertible such that $A \leq_m^p \mathrm{SAT}$ via g_1. Let N_1 be an NP-machine that accepts B in time p_1. Define the following function f_1.

$$f_1(z) \stackrel{df}{=} \begin{cases} \neg g_1(x) & : \text{ if } z = \langle x, w \rangle,\ |w| = p_1(|x|),\ M_1(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w \rangle,\ |w| \neq p_1(|x|),\ |z| \geq 2^{|x|},\ x \in \mathrm{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

The proof of Theorem 3.1 in Glaßer et al. [GSZ05] shows that f_1 is a proof system and $can(f_1) \equiv_m^{pp} (A, B)$. Now choose g_2 that is polynomial-time computable and polynomial-time invertible such that $C \leq_m^p \mathrm{SAT}$ 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.

$$f_2(z) \stackrel{df}{=} \begin{cases} \neg g_1(x) & : \text{ if } z = \langle x, w \rangle,\ |w| = p_1(|x|),\ M_1(x) \text{ accepts along path } w \\ \neg g_2(x) & : \text{ if } z = \langle x, w \rangle,\ |w| = p_2(|x|),\ M_2(x) \text{ accepts along path } w \\ x & : \text{ if } z = \langle x, w \rangle,\ |w| \neq p_i(|x|) \text{ for } i = 1, 2,\ |z| \geq 2^{|x|},\ x \in \mathrm{TAUT} \\ \text{true} & : \text{ otherwise} \end{cases}$$

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 a f_1-proof z is also a 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 $can(f_2) \equiv_m^{pp} (C, D)$. We only show $can(f_2) \leq_m^{pp} (C, D)$. The proof for $(C, D) \leq_m^{pp} can(f_2)$ is the same as that for $(A, B) \leq_m^{pp} can(f_1)$, which we refer the reader to Glaßer et al. [GSZ05].

Let g many-one reduces (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 ≥ 2^{|y|+1} then
3     if y ∈ 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 $h \in \text{PF}$.

Assume $(y, 0^n) \in \text{SAT}^*$. 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), then $g_1^{-1}(y) \in A$ and so, $g(g_1^{-1}(y)) \in C$. Therefore in either case (output is made in line 5 or line 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 $f(z) = \neg y$. 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, $f(z) = \neg y$ 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 $can(f_2) \leq_m^{pp} (C, D)$ via h and finishes the proof of Theorem 3.3. □

Corollary 3.4 *For every proof systems f and g such that $can(f) \leq_m^{pp} can(g)$, there exists a proof system g' such that $can(g') \equiv_m^{pp} can(g)$ and $f \leq_p g'$.*

Proof. Apply Theorem 3.3 with $(A, B) = can(f)$ and $(C, D) = can(g)$, and the obtained proof system f_2 has the desired property. □

3.1 Well-behaved Proof Systems

The proof system g constructed in proving Theorem 3.2 might seem "pathological", as it has super-polynomially long proofs for tautologies in an easy subset of TAUT. One might wonder whether we could prove Theorem 3.2 by constructing proof systems without such pathology. Such proof systems can be formalized as follows:

Definition 3.1 *A proof system f is* well-behaved *if for any $S \subseteq$ TAUT and $S \in$ P, there exists a polynomial p such that for every $x \in S$,*

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

However, well-behaved proof systems probably do not even exist because Meßner [Meß00] showed that existence of well-behaved proof systems implies existence of optimal proof systems and there is plenty of evidence (see, for example, Meßner [MT98] or Glaßer et al. [GSSZ04]) that optimal proof systems do not exist. Therefore, it is probably the case that no proof system is well-behaved, i.e., every proof system has super-polynomially long proofs on some P-subset of TAUT. This shows that the proof system constructed in the proof of Theorem 3.2 is not as uncommon as one might thought.

Now it is easy to see that the arguments used in the proof of Theorem 3.2 apply to every non-well-behaved proof system. This allows us to obtain the following results on non-well-behaved proof systems.

Proposition 3.5 *Let f be a proof system that is not well-behaved. Then for every $(A, B) \in$ DisjNP, there exists a proof system g such that*

- $g \not\leq f$ *and*
- $can(g) \equiv_m^{pp} (A, B)$.

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

Let $(A, B) \in$ DisjNP. By Theorem 3.1 in Glaßer et al. [GSZ05], there exists a proof system g' such that $can(g') \equiv_m^{pp} (A, B)$. Now define proof system g as follows:

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

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

Now we show $can(g) \leq_m^{pp} can(g') \leq_m^{pp} (A, B)$. Without loss of generality, assume $\min\{|w| \mid g'(w) = \neg false\} = c$ for some constant $c > 0$. So $(false, c) \in REF(g')$. The reduction is the following function h:

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

Clearly h is polynomial-time computable. Assume $(x, 0^n) \in \text{SAT}^*$, then $x \in SAT$ and hence, $\neg x \notin S$. So $h(x, 0^n) = (x, 0^{n-1}) \in \text{SAT}^*$. Now assume $(x, 0^n) \in REF(g)$. So $\neg x \in$ TAUT and there exists z such that $|z| \leq n$ and $g(z) = \neg x$. If $\neg x \in S$, then $h(x, 0^n) = (false, c) \in REF(g')$. If $\neg x \notin S$, then for any w, $g(1w) \neq \neg x$. So it must hold that $z = 0w$ for some w and $g(z) =$

$g'(w) = \neg x$. Since $|w| = |z| - 1$, $(x, 0^{n-1}) \in REF(g')$. This shows $can(g) \leq_m^{pp} can(g')$ and hence, $can(g) \leq_m^{pp} (A, B)$.

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 $w = 1x$ there exists w' such that $|w'| \leq q(|1x|) = p(|x|)$ and $f(w') = g(w) = x$. This contradicts the fact that $\min\{|w| \mid f(w) = x\} > p(|x|) = q(|x|+1)$.

□

Corollary 3.6 *For every proof system f that is not well-behaved, there exists a proof system g such that*

- $can(f) \equiv_m^{pp} can(g)$ *and*
- $f < g$.

Proof. The proof is the same as Proposition 3.5 except that g' is replaced by f. □

Corollary 3.7 *Assume optimal proof system do not exist. Then for every proof system f, there exists a proof system g such that*

- $can(f) \equiv_m^{pp} can(g)$ *and*
- $f < g$.

4 Proof Systems With Equivalent Canonical Pairs

We have seen in the last section that the degree structure of canonical pairs does not necessary reflect the simulation order of the corresponding proof systems. One related interesting question is how different two proof systems can be from each other if they have equivalent canonical pairs. We investigate this question in this section.

As one would expect from Theorem 3.2, the simulation order of two proof systems with equivalent canonical pairs is probably very arbitrary. This is verified by the following proposition.

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

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

Proof. Consider the proof system f constructed in the proof of Theorem 3.1 in Glaßer et al. [GSZ05]:

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

where g_1 is a polynomial-time computable and invertible many-one reduction from A to SAT, and N_1 is an NP-machine that accepts B in time p_1.

It is known that $can(f) \equiv_m^{pp} (A, B)$. Define two subsets of TAUT as follows:

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

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

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 both EASY1 and EASY2.

Now define proof system g and h as follows:

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

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

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

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

□

However, from another point of view, the proof systems defined in the above proof are actually quite "similar" in the sense that they differ super-polynomially only on an easy subset of TAUT. More precisely, the construction of proof systems with equivalent canonical pairs but with arbitrary simulation order as in the proof of Theorem 4.1 hinges on the following easy-to-prove fact:

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

So the question here is whether we can construct proof systems f and g with equivalent canonical pairs such that they are "more different" or "very different". For example, do there exist proof systems f and g such that $can(f) \equiv_m^{pp} can(g)$ and f is super-polynomially stronger than g almost everywhere? The following theorem shows that such proof systems exist only when both $can(f)$ and $can(g)$ are P-separable.

Theorem 4.3 *Let f and g be propositional proof systems such that $can(g) \leq_m^{pp} can(f)$. 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 α, then both $can(f)$ 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 the empty string λ is neither a f-proof nor a g-proof. 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. Let n_0 be the minimal integer such that for every tautology x with $|x| > n_0$ that 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 $can(g)$:

```
   Input ⟨x, 0^n⟩
0  y = x, m = n;
1  While (|y| ≤ m^k) and (m > 0)
2     Compute ⟨y', 0^{m'}⟩ = h(⟨y, 0^m⟩);
3     If |y'| ≤ n_0 then ACCEPT iff y' ∈ SAT;
4     Set ⟨y, 0^m⟩ = ⟨y', 0^{(m')^{1/((k+1)l+1)}}⟩;
5  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 2 that

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

So the assignment in line 4, which sets $m = (m')^{\frac{1}{(k+1)^l+1}}$, decrements m. Since m is initially set to n and decremented in each iteration of the while-loop, the claim holds. It is easy to see the claim implies the above algorithm is polynomial-time computable.

Now we prove the correctness of the algorithm. Assume the input $\langle x, 0^n \rangle \in \text{SAT}^*$. By line 2, it holds for each iteration of the while-loop that $y \in \text{SAT}$ implies $y' \in \text{SAT}$ since h many-one reduces

$can(g)$ to $can(f)$. Since initially $y = x \in \text{SAT}$ and y is set to y' by line 4 in each iteration, the loop-invariant $y' \in \text{SAT}$ holds. Therefore, the algorithm either accepts in line 3 or accepts in line 5.

Assume the input $\langle x, 0^n \rangle \in \text{REF}(g)$. So by line 0 it holds initially that $\langle y, 0^m \rangle \in \text{REF}(g)$. By line 2 it holds for each iteration of the while-loop that $\langle y', 0^{m'} \rangle \in \text{REF}(f)$ if the algorithm reaches line 2, since h many-one reduces $can(g)$ to $can(f)$. So in line 3 it holds that $y' \in \text{UNSAT}$ and hence, the algorithm rejects if $|y'| \leq n_0$. Otherwise, the algorithm reaches line 4. Then by the choice of n_0, it holds that $(y', 0^{m'}) \in \text{REF}(f)$ implies $(y', 0^{(m')^{\frac{1}{(k+1)l+1}}}) \in \text{REF}(g)$. So the loop invariant $(y, 0^m) \in \text{REF}(g)$ holds. Suppose the algorithm reaches line 5. Then either $|y| > m^k$ or $m = 0$. By the loop invariant $(y, 0^m) \in \text{REF}(g)$ that we showed above, there exists w with $|w| = m$ 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 $m = 0$ in line 5. However, this is a contradiction because by our hypothesis, λ, the only string of length 0, is neither a g-proof nor a f-proof and hence, it holds for any $y \in \text{TAUT}$ that $\langle y, 0^0 \rangle \notin \text{REF}(g)$. Therefore, line 5 cannot be reached on input $(x, 0^n) \in \text{REF}(g)$ and the algorithm must exit from line 3 and reject.

□

Now let us summarize what we have just seen: Proposition 4.2 says that if two proof systems are "very similar" then they have equivalent canonical pairs while Theorem 4.3 tells us that proof systems that are "very different" from each other cannot have equivalent canonical pairs. Therefore, a natural question is to exactly what extent two proof systems can be different from each other while still having equivalent canonical pairs. We show in Theorem 4.4 that Proposition 4.2 does not hold any more when P-subset is replaced with NP-subset unless P-inseparable disjoint NP-pairs do not exist. This means altering f-proofs on a P-subset of TAUT does not change the many-one degree of $can(f)$, but altering f-proofs on a NP-subset of TAUT might do. On the other hand, we also show in Theorem 4.10 that proof systems that differ super-polynomially on more than a P-subset of TAUT may still have equivalent canonical pairs. It is still unknown whether two proof systems can have equivalent canonical pairs if they differ super-polynomially on a set whose complexity lies between P and NP (for example, $\text{NP} \cap \text{coNP}$.)

Theorem 4.4 *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 $can(f) < can(f')$ and f and f' simulate each other except on a* NP*-subset of* TAUT*.*

Proof. Let f be a proof system such that $can(f)$ is not m-complete for DisjNP. Note that such proof system exist if P-inseparable NP-pairs exist. Since $can(f)$ is not m-complete, there exists a disjoint NP-pair (C, D) such that $can(f) \leq_m^{pp} (C, D)$ but $(C, D) \not\leq_m^{pp} can(f)$ (take (A, B) such that $(A, B) \not\leq_m^{pp} can(f)$ and let $(C, D) = (A, B) \oplus can(f)$.)

Let $\langle \cdot, \cdot \rangle$ be a polynomial-time computable, polynomial-time invertible pairing function such that $|\langle v, w \rangle| = 2|vw|$. Choose g that is polynomial-time computable and polynomial-time invertible such that $C \leq_m^p \text{SAT}$ via g. Let N be an NP-machine that accepts D in time p. Define a function f' as follows:

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

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 $x \in D \subseteq \overline{C}$.

Note that f' is similar to the proof system "f" constructed in the main theorem in Glaßer et al. [GSZ05] The difference is only that the trivial proofs in "f" are replaced by f-proofs here. This makes f' at least as strong as f.

Claim 4.5 $(C, D) \leq_m^{pp} can(f')$.

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 $g(x) \in$ SAT and hence, $h(x) \in$ 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 $h(x) \in \text{REF}(f')$. □

Claim 4.6 $f \leq_s f'$.

Proof. Trivial since for every w, $f'(1w) = f(w)$. □

Claim 4.7 *For all tautologies $x \notin \neg g(D)$, x has a f'-proof of length n implies x has a f-proof of length $n - 1$.*

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

Claim 4.6 implies $can(f) \leq_m^{pp} can(f')$ by Proposition 3.1. Claim 4.5 implies $can(f') \not\leq_m^{pp} can(f)$ since otherwise we will have the following chain of many-one reductions:

$$(C, D) \leq_m^{pp} can(f') \leq_m^{pp} can(f),$$

which contracts to the fact that $(C, D) \not\leq_m^{pp} can(f)$. Claim 4.7 together with Claim 4.6 shows that f and f' simulate each other on all tautologies except on $\neg g(D)$, which is an NP-subset of TAUT where f' has shorter proofs.

□

Corollary 4.8 *The following is equivalent.*

1. P-*inseparable* NP-*pairs exist.*

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

Corollary 4.9 *If* $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$*, then there exist proof system f and g whose canonical pairs are not many-one equivalent, but that simulate each other except on an* NP-*subset of* TAUT.

On the other hand, as the next theorem shows, two proof systems that differ on every P-subset of TAUT can still have equivalent canonical pairs.

Theorem 4.10 *Let (A,B) be a* P-*inseparable* NP-*pair. Then there exist proof system f and f' such that $can(f) \equiv_m^{pp} can(f') \equiv_m^{pp} (A, B)$ and f and f' does not simulate each other on* $\mathrm{TAUT} - S$ *for every* P-*subset S of* TAUT.

Proof. Consider the proof system f we constructed in the main theorem in Glaßer et al. It's easy to see that f has short proofs on $\neg g(B)$ and trivial 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 $range(g) \cap range(g') = \emptyset$. Define proof system f' using g' as in the previous paper. Then clearly $can(f) \equiv_m^{pp} (A, B) \equiv_m^{pp} can(f')$. we claim that f cannot simulate f' on $\mathrm{TAUT} - S$ for any P-subset S of TAUT. This can be seen as follows:

Assume for some P-subset S of TAUT that f simulates f' on $\mathrm{TAUT} - S$. Since f has exponentially long proofs on $\neg g'(B)$ while f' has polynomially-bounded proofs on $\neg g(B)$, $(\mathrm{TAUT} - S) \cap \neg g'(B)$ 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 $(A, B) \leq_m^{pp} (\overline{\mathrm{TAUT}}, \neg g'(B))$ via the reduction $h(x) = \neg g'(x)$.

Symmetric arguments show f' cannot simulate f on $\mathrm{TAUT} - S$ for any P-subset S of TAUT.

□

5 Strongly Many-one degrees of Canonical Pairs

Glaßer et el. [GSZ06a] showed that every disjoint NP-pair is many-one equivalent to the canonical pair of some proof system. We ask the same question for the strongly many-one reduction, i.e., whether every disjoint NP-pair (A, B), where A is NP-complete[3], is strongly many-one equivalent to the canonical pair of some proof system. We show that this question is closely related to the problem of whether the disjoint union of two NP-complete sets are still NP-complete. The latter has been an important open problem in the study of structural complexity [GPSS04, GSTW06]. The results in this section demonstrate new connections between proof systems and disjoint NP-pairs. This would provide further motivations for the study of disjoint NP-pairs.

[3]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.

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

Proof. Choose a one-one, length-increasing, polynomial-time computable, polynomial-time invertible function g such that $A \leq^p_m \mathrm{SAT}$ 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 $(\mathrm{SAT}, \overline{\mathrm{SAT}}) \leq^{pp}_m (A, \overline{A \cup B})$. Let $\langle \cdot, \cdot \rangle$ be a one-one, polynomial-time computable, polynomial-time invertible pairing function such that $|\langle v, w \rangle| = 2|vw|$. Let M be an NP-machine that accepts B in time p, and choose a constant $c > 0$ such that for all n, $p(n) < 2^n + c$.

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

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 $(\mathrm{SAT}^*, \mathrm{REF}_f) \leq^{\mathrm{pp}}_{\mathrm{sm}} (A, B)$.

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 ≥ 2(|¬y| + 2^|¬y| + c) then
2     if y ∈ SAT then output a else output b
3  endif
4  if g^-1(y) exists and n ≥ 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 $x \stackrel{df}{=} g^{-1}(y)$ belongs to B. Let $z = \langle x, w \rangle$ where w is an accepting path of $M(x)$. 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 $(\mathrm{SAT}, \overline{\mathrm{SAT}}) \leq_m^{pp} (A, \overline{A \cup B})$ via reduction r.

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

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

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

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 $M(x)$ and define $z \stackrel{df}{=} \langle x, w \rangle$. 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 '¬'. 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 $(\mathrm{SAT}^*, \mathrm{REF}_f) \equiv_{\mathrm{sm}}^{\mathrm{pp}} (A, B)$, then $(A, \overline{A \cup B})$ is* NP*-hard.*

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

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

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

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

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

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

Proof. 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* $(\mathrm{SAT}^*, \mathrm{REF}_f) \equiv^{\mathrm{pp}}_{\mathrm{sm}} (\mathrm{SAT}, B)$*, then unions of disjoint* NP*-complete sets are* NP*-complete.*

Proof. Assume NP $\neq$ coNP and that for all disjoint NP-pairs (SAT, B) there exists a proof system f such that $(\mathrm{SAT}^*, \mathrm{REF}_f) \equiv^{\mathrm{pp}}_{\mathrm{sm}} (\mathrm{SAT}, B)$. Fix some $B \in$ 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 NP-hard. Therefore, $\mathrm{SAT} \cup B$ is NP-complete. So we have shown that $B \in$ NP, if $\mathrm{SAT} \cap B = \emptyset$, then $\mathrm{SAT} \cup B$ is NP-complete. By Theorem 5.9[4] in Glaßer et al. [GPSS04], it follows that unions of disjoint NP-complete sets are NP-complete. □

6 Proof Systems and Turing-degrees of Canonical Pairs

In this section, we try to generalize some of the results from Section 3 to Turing reductions.

Proposition 6.1 *Let* f *and* g *be proof systems such that* $can(f) \leq^{pp}_T can(g)$*. Then there exists a proof system* g' *such that* $can(g') \equiv^{pp}_T can(g)$ *and* $f \leq_p g'$.

[4] Christian: please check this

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, $can(f) \leq_T^{pp} can(g) \leq_m^{pp} can(g')$. It remains to show $can(g') \leq_T^{pp} can(g)$. Let $can(f)$ Turing reduces 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 $S \in Sep(can(g))$.

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, $M'^S(x, 0^{n-1})$ rejects also, since $(x, 0^{n-1}) \in \mathrm{REF}(f)$. □

Corollary 6.2 *Let* $\mathbf{d}_1 < \mathbf{d}_2$ *be two* NP*-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 3.1 in Glaßer et al. [GSZ05], we can pick two proof systems f and g such that $can(f) \in \mathbf{d}_1$ and $can(g') \in \mathbf{d}_2$. By Proposition 3.1, $g' \not\leq f$. By Proposition 6.1, there exists a proof system g such that $f \leq g' \leq g$ and $g \in \mathbf{d}_2$. Clearly, $g \not\leq f$ also. □

Corollary 6.3 *For every disjoint NP-pairs* (A, B) *and* (C, D) *such that* $(A, B) <_T^{pp} (C, D)$*, there exist proof system* f *and* g *such that*

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

Proof. By Theorem 3.1 in Glaßer et al. [GSZ05], we can define a proof system g such that $can(g) \equiv_m^{pp} (C, D)$. Also, it is easy to observe that g is not well-behaved from the proof of the theorem (Take $S = \{a \vee (\neg a) \vee y \mid y \text{ is a propositional formula}\}$, for example). Now apply Proposition 3.5 to g and (A, B), then we obtain a proof system f such that $can(f) \equiv_m^{pp} (A, B)$ and $f \not\leq g$. Note that $g \not\leq f$ as well since otherwise it would imply $(C, D) \leq_m^{pp} (A, B)$ by Proposition 3.1, which contradicts the premise. □

References

[Bey06] 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.

[CR79] Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *The Journal of Symbolic Logic*, 44(1):36–50, 1979.

[GPSS04] Christian Glaßer, Aduri Pavan, Alan L. Selman, and Samik Sengupta. Properties of np-complete sets. In *19th Annual IEEE Conference on Computational Complexity*, pages 184–197, 2004.

[GS88] J. Grollmann and A. Selman. Complexity measures for public-key cryptosystems. *SIAM Journal on Computing*, 17(2):309–335, 1988.

[GSS05] C. Glaßer, A. L. Selman, and S. Sengupta. Reductions between disjoint NP-pairs. *Information and Computation*, 200(2):247–267, 2005.

[GSSZ04] C. Glaßer, A. Selman, S. Sengupta, and L. Zhang. Disjoint NP-pairs. *SIAM Journal on Computing*, 33(6):1369–1416, 2004.

[GSTW06] G. Glaßer, A. Selman, S. Travers, and K. Wagner. The complexity of unions of disjoint sets. Technical Report TR06-069, Electronic Computational Complexity Colloquium, 2006.

[GSZ05] C. Glaßer, A. Selman, and L. Zhang. Canonical pairs of proof systems and disjoint NP-pairs. In *Proceedings of the 30th International Symposium on Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, 2005.

[GSZ06a] C. Glaßer, A. Selman, and L. Zhang. Canonical pairs of proof systems and disjoint NP-pairs. *Theoretical Computer Science*, 2006. To appear.

[GSZ06b] 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.

[HS92] 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.

[KMT03] Johannes Köbler, Jochen Meßner, and Jacobo Torán. Optimal propositional proof systems imply complete sets for promise classes. *Information and Computation*, 2003. To appear.

[Meß00] Jochen Meßner. *On the Simulation order of proof systems*. PhD thesis, Uiversität Ulm, Abteilun Theoretische Informatik, December 2000.

[MT98] 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.

[Pud03] P. Pudlák. On reducibility and symmetry of disjoint NP-pairs. *Theoretical Computer Science*, 295:323–339, 2003.

[Raz94] A. A. Razborov. On provably disjoint NP-pairs. Technical Report TR94-006, Electronic Computational Complexity Colloquium, 1994.