Electronic Colloquium on Computational Complexity, Report No. 107 (2009)

Improved inapproximability factors for some Σ_2^p minimization problems

Kevin Dick*
Computer Science Division
Soda Hall
University of California
Berkeley, CA 94720-1776.
kgdick@eecs.berkeley.edu

Christopher Umans†
Department of Computer Science
California Institute of Technology
Pasadena, CA 91125.
umans@cs.caltech.edu

October 28, 2009

Abstract

We give improved inapproximability results for some minimization problems in the second level of the Polynomial-Time Hierarchy. Extending previous work by Umans [Uma99], we show that several variants of DNF minimization are Σ_2^p-hard to approximate to within factors of $n^{1/3-\epsilon}$ and $n^{1/2-\epsilon}$ (where the previous results achieved $n^{1/4-\epsilon}$), for arbitrarily small constant $\epsilon > 0$. For one problem shown to be inapproximable to within $n^{1/2-\epsilon}$, we give a matching $O(n^{1/2})$-approximation algorithm, running in randomized polynomial time with access to an NP oracle, which shows that this result is tight assuming the PH doesn't collapse.

1 Introduction

DNF minimization and related problems are among the most natural problems that are complete for levels of the Polynomial-Time Hierarchy (PH) above NP. DNF minimization itself ("Given a DNF formula and an integer k, is there an equivalent formula of size at most k?") was shown to be Σ_2^p-complete in [Uma01], and that problem and some surrounding minimization problems were further shown to be Σ_2^p-hard to approximate to within factors of the form n^δ in [Uma99]. In this paper we study the particular constant δ achievable in these inapproximability proofs.

There are quite a number of optimization problems in the literature that are complete for the second level of the PH, and for many of these some inapproximability is known (see the surveys [SU02a, SU02b]). For inapproximability proofs that rely on variants of the PCP Theorem for the levels of the PH, it seems technically challenging to obtain tight results. However, for the collection of problems in [Uma99] with approximation thresholds of the form n^δ, the inapproximability proofs used *dispersers* as their main technical tool, and subsequent disperser constructions [TSUZ07] improved the constant δ to (the optimal) $1-\epsilon$ for arbitrarily small $\epsilon > 0$ for the basic problem (SHORTEST IMPLICANT CORE, defined below) of [Uma99].

*This work was done as part of a Caltech Summer Undergraduate Research Fellowship (SURF), supported by the Paul K. Richter and Evalyn E. Cook Richter Memorial Funds, and NSF CCF-0346991.
†Supported by NSF grant CCF-0830787 and BSF grant 2004329.

ISSN 1433-8092

The constant δ for several other of the most important problems improved to only $1/4 - \epsilon$ due to losses in the reductions. There is thus an opportunity to further improve this constant by tightening the reductions from SHORTEST IMPLICANT CORE. This is what we do in this paper: we improve $1/4$ to $1/3$ or $1/2$ depending on the problem. For one problem (IRREDUNDANT*(LITERALS), defined below), we show that the $n^{1/2-\epsilon}$ factor is tight (assuming the PH doesn't collapse), which is somewhat surprising since it was natural to conjecture that $\delta = 1 - \epsilon$ was the correct answer for all of these problems.

1.1 Preliminaries

Suppose Π is a minimization problem. Then Π WITH GAP $r(n)$ is defined to be a promise variant of Π in which positive instances have an optimal solution of at most $k(n)$ and negative instances have an optimal solution greater than $r(n)k(n)$. If Π WITH GAP $r(n)$ is $\mathcal{C}$-hard for a class $\mathcal{C}$, it is said to be *$\mathcal{C}$-hard to approximate to within a factor* $r(n)$.

An *implicant* of a Boolean function $f(x_1, x_2, \ldots, x_n)$ is a conjunction of literals that implies f. A *prime implicant* C is a "minimal" implicant, i.e., no proper subset C' of C's literals implies f.

We define a number of minimization problems related to DNF minimization. The decision versions of all of these are known to be Σ_2^p-complete, and they are all Σ_2^p-hard to approximate to within factors of the form n^δ for some constant $\delta > 0$ [Uma99].

SHORTEST IMPLICANT CORE: Given a DNF formula ϕ and an implicant X of ϕ, what is a shortest implicant $X' \subseteq X$?

MIN DNF: Given a DNF formula ϕ, what is an equivalent DNF ϕ' with the fewest occurrences of literals?

MIN TERM DNF: Given a DNF formula ϕ, what is an equivalent DNF ϕ' with the least number of terms?

IRREDUNDANT: Given an n-term DNF formula $\phi = t_1 \vee \cdots \vee t_n$, what is a minimum cardinality index set $I \subseteq \{1, \ldots, n\}$ such that ϕ is equivalent to $\bigvee_{i \in I} t_i$?

IRREDUNDANT(LITERALS): Given an n-term DNF formula $\phi = t_1 \vee \cdots \vee t_n$, what is an index set $I \subseteq \{1, \ldots, n\}$ with $\bigvee_{i \in I} t_i$ equivalent to ϕ such that $\bigvee_{i \in I} t_i$ has the fewest occurrences of literals?

IRREDUNDANT*(LITERALS): Given an n-term DNF formula $\phi = t_1 \vee \cdots \vee t_n$ that depends on all of its variables, what is an index set $I \subseteq \{1, \ldots, n\}$ with $\bigvee_{i \in I} t_i$ equivalent to ϕ such that $\bigvee_{i \in I} t_i$ has the fewest occurrences of literals?

The problem IRREDUNDANT*(LITERALS) differs from IRREDUNDANT(LITERALS) only in the additional promise that the formula ϕ depends on all of its variables, meaning that for each variable x there exists an assignment of the remaining variables such that changing x from false to true changes whether the formula is satisfied. This is a natural demand (one can check that this promise holds on a given instance in polynomial time with an NP oracle, which is "easy" when discussing Σ_2^p-hard problems) that ensures any equivalent formula will mention all of the original variables at least once, and moreover, the known hardness reductions (with a few simple modifications) produce instances that fulfill this promise.

The previous results of [Uma99] together with the disperser constructions in [TSUZ07] show that SHORTEST IMPLICANT CORE is Σ_2^p-hard to approximate to within a factor $n^{1-\epsilon}$, and that the latter five problems are Σ_2^p-hard to approximate within a factor $n^{1/4-\epsilon}$.

1.2 Outline

In Section 2 we give improved inapproximability results for the above problems. Section 3 describes the approximation algorithm for IRREDUNDANT*(LITERALS). We end with some open problems in Section 4.

2 Improved inapproximability results

We build on the reduction from SHORTEST IMPLICANT CORE in [Uma99] to obtain the following:

Theorem 2.1. MIN TERM DNF WITH GAP $n^{1/3-\epsilon}$ *is* Σ_2^p*-hard.*

Proof. The proof follows the proof in [Uma99] closely, with the main difference being in the use of the z variables below and how we reason about them. Begin with an instance (ϕ, X) of SHORTEST IMPLICANT CORE, which we know to be Σ_2^p-hard to approximate to within a factor $n^{1-\epsilon}$. Suppose $(\phi \wedge \neg X) = s_1 \vee s_2 \vee \cdots \vee s_m$. Construct

$$\phi' \quad = \quad \bigvee_{i=1}^{m} s_i' \vee \bigvee_{x_i \in X} \bigvee_{j=1}^{m} u_i^j,$$

where $s_i' = s_i \wedge \bigwedge_{k \in S_i} z_k$ and $u_i^j = \overline{x_i} a_j z_1 z_2 \cdots z_{2 \log m}$, the a and z variables are new variables, and with $S_1, S_2, \ldots, S_m$ distinct subsets of order $\log m$ of $\{1, 2, \ldots, 2 \log m\}$. Note that $\binom{2\log m}{\log m} \geq \left(\frac{2 \log m}{\log m}\right)^{\log m} = m$, so we can select our desired number of subsets.

We establish the following intermediate result:

Claim 1. *There exists an implicant $X' \subseteq X$ of ϕ of length k if and only if ϕ' has an equivalent DNF ϕ'' with at most $m(k+1)$ terms.*

Proof. Let $X' \subseteq X$ be an implicant of ϕ of length k. Construct

$$\phi'' \quad = \quad \bigvee_{i=1}^{m} s_i' \vee \bigvee_{x_i \in X'} \bigvee_{j=1}^{m} u_i^j.$$

As every term of ϕ'' also appears in ϕ', ϕ' will be true whenever ϕ'' is true. We need to check that ϕ' being true implies that ϕ'' is true. Suppose assignment A has $\phi'(A) = 1$. If A has at least one z variable false, or all a variables false, then it is inconsistent with all u_i^j and hence must be consistent with some s_i'. Since all s_i' terms are included in ϕ'', $\phi''(A) = 1$. Otherwise, all z variables are true and at least one a variable is true. If A is consistent with X', then $\phi(A) = 1$. Every term of ϕ' has a negated x variable, so A cannot be consistent with X. But then an s_i' term must be true, implying $\phi''(A) = 1$. If A is inconsistent with X', there must be some $x_i \in X'$ set to false. Then some u term with a true a_j and a negated x_i will be true, and will be included in ϕ''.

In the other direction, we start with some ϕ'' equivalent to ϕ'. Suppose term t of ϕ'' contains no true a variables. For any true assignment A of the variables in ϕ'' with all a variables false, some s_i' term from ϕ' must be true. This s_i' term will remain true if some of the a variables are flipped to true, as no s_i' term contains any a variable. Since any assignment on which t is true will also force $\bigvee s_i'$ to be true, we can say that t implies $\bigvee s_i'$.

Let A_i be an assignment of ϕ' consistent with s_i' that sets all z variables not in S_i to false and all a variables to false. As ϕ' will be true on A_i, some term t in ϕ'' must be true. Term t cannot have any true a

variables, and it must include all z variables in S_i, since otherwise ϕ'' would be true on an assignment with less than $\log m$ of the z variables true. Moreover, t cannot contain any z variables not in S_i, as t is true on assignment A_i with these z variables false. Thus, each value of i gives us a term in ϕ'' with no true a variables that includes precisely the z variables from S_i. As the sets S_i are distinct and of equal size, there must be at least m terms in ϕ'' with no true a variables. We now discard all terms in ϕ'' that imply $\bigvee s'_i$. This removes at least m terms from ϕ'', and all remaining terms must have no positive a variables.

Suppose we are left with R terms. Every assignment accepted by ϕ' has a false x variable, so each term of ϕ'' must contain some negated x_i. Label each of these R terms with a pair (i, j), where i is the index of a negated x variable in the term and j is the index of a positive a variable in the term. We must have some j that labels no more than $\lceil R/m \rceil$ terms. For this j, take X' to be the x_i such that some term is labeled with (i, j).

It remains to argue that this X' is an implicant of ϕ, and that $|X'|$ is small. Suppose X' were not an implicant of ϕ. Then we have some assignment A consistent with X' on which ϕ is false. Set all z variables to true in A, set a_j to true for the j selected above, and set all other a variables to false. As ϕ does not include any a or z variables, ϕ will remain false on A. As X is an implicant of ϕ, some x_i must be set to false. The u_i^j term of ϕ' will be true, so ϕ' is true on A. Since $\vee s'_i = (\phi \wedge \neg X)$, every s'_i term will be false. If one of the discarded terms above were true, then some s'_i would have to be true, so no discarded term of ϕ'' is true. Moreover, the R remaining terms are unsatisfied, because each has a positive a literal with that a set to false in A, or have a negated x_i literal with that x_i appearing in X' and hence set to true in A. Thus, ϕ'' is false on A while ϕ' is true, violating the equivalence of ϕ' and ϕ''. Hence X' must be an implicant of ϕ.

We have that $|X'| \leq \lceil R/m \rceil$, and that $R \leq |\phi''| - m$. Let k be the least integer such that $|\phi''| \leq m(k+1)$, so that $R \leq mk$. This gives $|X'| \leq \lceil R/m \rceil \leq \lceil mk/m \rceil = k$, completing the claim. □

Let n be the total size of the original instance (ϕ, X) of SHORTEST IMPLICANT CORE. In creating the instance ϕ' of MIN DNF, m will be at most $|\phi| \cdot |X| \leq n^2$. Each term of ϕ' has $O(\log n)$ literals, so the size of ϕ' will then be $|\phi'| \leq O(n^2 \log n + n \cdot n^2 \log n) = O(n^{3+o(1)})$. Expressed in terms of the size of the MIN DNF instance, our gap is $n^{1/3-\epsilon}$, for any constant $\epsilon > 0$. □

We use a nearly identical argument to prove inapproximability for IRREDUNDANT. The crucial difference is that we don't need to form $(\phi \wedge \neg X)$ to begin with, because we can rely on the fact that the terms of the solution are required to be terms of the original DNF formula. This simplifies some other steps in the proof as well. For completeness we record the entire proof (with these simplifications) below:

Theorem 2.2. IRREDUNDANT WITH GAP $n^{1/2-\epsilon}$ *is* Σ_2^p*-hard*.

Proof. Begin with an instance (ϕ, X) of SHORTEST IMPLICANT CORE, which we know to be Σ_2^p-hard to approximate to within a factor $n^{1-\epsilon}$. Suppose $\phi = s_1 \vee s_2 \vee \cdots \vee s_m$. Construct

$$\phi' \quad = \quad \bigvee_{i=1}^{m} s'_i \vee \bigvee_{x_i \in X} \bigvee_{j=1}^{m} u_i^j,$$

where $s'_i = s_i \wedge \bigwedge_{k \in S_i} z_k$ and $u_i^j = \overline{x_i} a_j z_1 z_2 \cdots z_{2\log m}$, the a and z variables are new variables, and with $S_1, S_2, \ldots, S_m$ distinct subsets of order $\log m$ of $\{1, 2, \ldots, 2\log m\}$. Note that $\binom{2\log m}{\log m} \geq \left(\frac{2\log m}{\log m}\right)^{\log m} = m$, so we can select our desired number of subsets. We establish the following intermediate result:

Claim 2. *There exists an implicant $X' \subseteq X$ of ϕ of length k if and only if ϕ' has an equivalent DNF ϕ'' with at most $m(k+1)$ terms.*

Proof. Let $X' \subseteq X$ be an implicant of ϕ of length k. Construct

$$\phi'' \quad = \quad \bigvee_{i=1}^{m} s'_i \vee \bigvee_{x_i \in X'} \bigvee_{j=1}^{m} u_i^j.$$

As every term of ϕ'' also appears in ϕ', ϕ' will be true whenever ϕ'' is true. We need to check that ϕ' being true implies that ϕ'' is true. Suppose assignment A has $\phi'(A) = 1$. If A has at least one z variable false, or all a variables false, then it is inconsistent with all u_i^j and hence must be consistent with some s'_i. Since all s'_i terms are included in ϕ'', $\phi''(A) = 1$. Otherwise, all z variables are true and at least one a variable is true. If A is consistent with X', then $\phi(A) = 1$. But then an s'_i term must be true, implying $\phi''(A) = 1$. If A is inconsistent with X', there must be some $x_i \in X'$ set to false. Then some u term with a true a_j and a negated x_i will be true, and will be included in ϕ''.

In the other direction, we start with some ϕ'' equivalent to ϕ', with all of its terms coming from ϕ'. Since each s'_i term in ϕ' has an assignment on which it's true and no other term is true (namely, an assignment with s_i true and all z_k with $k \in S_i$ true, with the remaining z_k false), each s'_i term must be included in ϕ''. Discard all of these s'_i terms from ϕ'', and suppose we are left with R terms. Every remaining term is of the form $u_i^j = \overline{x_i} a_j z_1 z_2 \cdots z_{2\log m}$. Fix some j such that no more than $\lceil R/m \rceil$ of the remaining terms contain a_j, and then take X' to be the x_i such that some remaining term contains $\overline{x_i}$ along with a_j.

It remains to argue that this X' is an implicant of ϕ, and that $|X'|$ is small. Suppose X' were not an implicant of ϕ. Then we have some assignment A consistent with X' on which ϕ is false. Set all z variables to true in A, set a_j to true for the j selected above, and set all other a variables to false. As ϕ does not include any a or z variables, ϕ will remain false on A. As X is an implicant of ϕ, some x_i must be set to false. The u_i^j term of ϕ' will be true, so ϕ' is true on A. Since $\vee s'_i = \phi$, every s'_i term will be false. If one of the discarded terms above were true, then some s'_i would have to be true, so no discarded term of ϕ'' is true. Moreover, the R remaining terms are unsatisfied, because each has a positive a literal with that a set to false in A, or have a negated x_i literal with that x_i appearing in X' and hence set to true in A. Thus, ϕ'' is false on A while ϕ' is true, violating the equivalence of ϕ' and ϕ''. Hence X' must be an implicant of ϕ.

We have that $|X'| \leq \lceil R/m \rceil$, and that $R \leq |\phi''| - m$. Let k be the least integer such that $|\phi''| \leq m(k+1)$, so that $R \leq mk$. This gives $|X'| \leq \lceil R/m \rceil \leq \lceil mk/m \rceil = k$, completing the claim. □

Let n be the total size of the original instance (ϕ, X) of SHORTEST IMPLICANT CORE. In creating the instance ϕ' of IRREDUNDANT, m will be at most $|\phi| \leq n$. Each term of ϕ' has $O(\log n)$ literals, so the size of ϕ' will then be $|\phi'| \leq O(n \log n + n \cdot n \log n) = O(n^{2+o(1)})$. Expressed in terms of the size of the MIN DNF instance, our gap is $n^{1/2-\epsilon}$, for any constant $\epsilon > 0$. □

Similar arguments give the following:

Theorem 2.3. MIN DNF WITH GAP $n^{1/3-\epsilon}$, IRREDUNDANT(LITERALS) WITH GAP $n^{1/2-\epsilon}$ *and* IRREDUNDANT*(LITERALS) WITH GAP $n^{1/2-\epsilon}$ *are all* Σ_2^p*-hard, for any constant* $\epsilon > 0$.

Proof. A reduction of [Uma99] using the disperser of [TSUZ07] proves that SHORTEST IMPLICANT CORE is Σ_2^p-hard to approximate to within $n^{1-\epsilon}$ by reducing from QSAT$_2$. We describe a modification to this reduction to ensure that ϕ depends on all of its variables. This extra property is needed only for the reduction to IRREDUNDANT*(LITERALS). An instance of QSAT$_2$ consists of a 3-DNF formula $\phi(x, y)$, and the instance is a positive one if and only if there exists an x such that for all y, $\phi(x, y)$ is satisfied. Without loss of generality we can assume that every variable in x and y *appears* in ϕ. Begin by removing from ϕ all terms with contradictory literals, meaning all terms with pairs of literals such that one is the negation of the other.

These terms can never be satisfied, so removing them does not affect the satisfiability of ϕ. Then augment to each term t in ϕ a new variable x_t by replacing t with $t \wedge x_t$, and add to the set x each new x_t variable. To see that this modified formula, which is now a 4-DNF formula, depends on all of its variables, pick a term t and set all literals in t to true while setting all $x_{t'}$ with $t' \neq t$ to false (and setting the remaining variables arbitrarily). Since each term $t' \neq t$ will then be set to false, we have an assignment that depends on all the variables in t. Since every variable appears in some term of ϕ, we find that ϕ depends on all of its variables. Moreover, if the original ϕ was a positive instance, then the modified ϕ is as well, by setting all of the new x_t variables to true. And in the other direction, if the modified ϕ is a positive instance, then there exists a setting of the original x variables so that for all y variables the *subformula* of the original ϕ selected by the true x_t variables is true; this implies that the original ϕ itself is a positive instance (since adding additional non-trivial terms cannot flip the output value to false).

Continuing the reduction from [Uma99] until we have an instance (ϕ', X) of SHORTEST IMPLICANT CORE, we see that the resulting DNF ϕ' has only a constant number of literals per term. This means the terms of ϕ'' in the proofs of Theorems 2.1 and 2.2 are each of length $O(\log m)$. Moreover, an assignment with fewer than $\log m$ of the z variables true can never evaluate to true, so each term in any equivalent DNF formula will have length at least $\log m$. A formula with a minimum number of terms can therefore differ by at most a multiplicative constant in size from a formula with minimum size, so the analysis above also shows that MIN DNF WITH GAP $n^{1/3-\epsilon}$ is Σ_2^p-hard. As with MIN TERM DNF and MIN DNF, an IRREDUNDANT formula with a minimum number of terms can differ by at most a multiplicative constant in size from an IRREDUNDANT(LITERALS) formula with a minimum number of literal occurrences, giving Σ_2^p-hardness for IRREDUNDANT(LITERALS) WITH GAP $n^{1/2-\epsilon}$.

Next, we deal with the reduction to IRREDUNDANT*(LITERALS). We first argue that the formula produced in the reduction of [Uma99] from QSAT_2 to SHORTEST IMPLICANT CORE depends on all of its variables. That reduction first produces a "co-ND" circuit on a larger set x' of "x-variables" (so-labeled because they encode an assignment to the original x variables), and polynomially many copies of the y variables. By setting all of the copies of the y variables to an assignment that witnesses the dependence of ϕ on some particular y_i and setting x' so that it encodes the associated assignment to the x variables, we find that toggling the i-th y variable in any copy changes the output of the co-ND circuit; hence it depends on all of its y variables as long as ϕ does. To see that the co-ND circuit depends on all of the x' variables, we select an assignment encoding a given x, and extend this to an assignment of the y variables, for which the circuit outputs 1. To ensure that such an assignment of the y variables always exists, we apply an easy transformation to the original formula that adds to the set y a single new variable y^*. We require ϕ to be satisfied whenever y^* is true, and whenever y^* is false we require ϕ to be satisfied if and only if the original formula is satisfied. Then for any setting of the existentially quantified variables, there exists an assignment of the universally quantified variables, such that the formula is satisfied. The encoding used by the circuit is such that toggling any of the true x' variables to false makes the circuit output 0, so it depends on each x' variable that is true in this encoding. We then easily verify that every x' variable is true in some such encoding, and hence the co-ND circuit depends on all of its x' variables. Finally, the composition of the co-ND circuit with itself described in Section 5.1 of [Uma99] preserves the property of depending on all variables, and the (standard) transformation to a 3-DNF that produces an instance of SHORTEST IMPLICANT CORE continues to depend on all of its variables.

Lastly, we verify that if the ϕ coming from an instance of SHORTEST IMPLICANT CORE depends on all of its variables, then the DNF ϕ' produced in the reduction in the proof of Theorem 2.2 also depends on all of its variables. The formula ϕ' becomes equivalent to ϕ under the restriction that sets all of the z variables to true and all of the a variables to false; thus it depends on all of the variables in ϕ, since ϕ depends on all of its variables. If we consider the assignment that sets exactly the literals in term s'_i to true, we see that toggling any of the z variables in the set S_i changes the output of ϕ', and every z variable appears in some S_i, so ϕ' depends on all of the z variables. Finally, since ϕ depends on all of its variables, it cannot be computing the trivial function 1. X is an implicant of ϕ, so there exists an assignment on which ϕ is false which also has some x_i set to false. Restricting ϕ' to this assignment, we see that the resulting formula depends on all of the a variables, so ϕ' must as well.

The conclusion is that IRREDUNDANT(LITERALS) WITH GAP $n^{1/2-\epsilon}$ remains Σ_2^p-hard even when demanding that the formula depends on all of its variables; in our language this gives Σ_2^p-hardness for IRREDUNDANT*(LITERALS) WITH GAP $n^{1/2-\epsilon}$.

□

3 An approximation algorithm

In this section we will need to approximate the size of a set recognizable in P using randomness and an NP oracle. The following lemma follows from well-known methods for approximate counting via hashing (see, e.g., [BGP00]):

Lemma 3.1. *There exists a constant c, and a polynomial-time randomized algorithm with access to an NP oracle, which, when given a set S specified implicitly as the ones of a Boolean circuit C, returns an integer t satisfying $|S|/c \leq t \leq c|S|$ with probability at least $1 - o(1/n^2)$.*

In fact c can be made arbitrarily close to 1, but we will not need that here. The following theorem shows that the inapproximability factor for IRREDUNDANT*(LITERALS) in Theorem 2.3 is tight:

Theorem 3.2. IRREDUNDANT*(LITERALS) *can be approximated to within a factor $O(n^{1/2})$ in randomized polynomial time with access to an NP oracle.*

Proof. We describe the algorithm. Suppose we have a DNF formula ϕ with R variables. If $R \geq n^{1/2}$, we can simply return the formula as is. Any formula equivalent to ϕ must at least mention all R of these variables because ϕ depends on all of its variables, forcing it to have size at least $R \geq n^{1/2}$. Thus, returning the original formula will give a result within an $n^{1/2}$ factor of optimal.

If $R < n^{1/2}$, we employ a different strategy. Let OPT be the fewest number of literals needed to give a formula ϕ'' equivalent to ϕ, with each term of ϕ'' coming from ϕ. We construct an equivalent formula ϕ' by essentially implementing a greedy algorithm for weighted set cover. The setting is a bit non-standard, since we only have approximate set sizes and succinct representations of the sets, so we go through the details.

The algorithm operates in iterations. At each iteration, we select a term t in ϕ and not in ϕ'. Our set-size estimation procedure from Lemma 3.1 is used to (approximately) compute the number of true assignments k_t of ϕ not yet covered by ϕ' that are covered by t. We add to ϕ' a term t such that $|t|/k_t$ is estimated to be minimum. (Here $|t|$ denotes the number of literals in term t.) These iterations are repeated until all true assignments of ϕ are covered by ϕ' (which we can check with an NP oracle).

Suppose ϕ has $m \leq 2^R$ true assignments of its variables. Label these assignments $A_1, A_2, \ldots, A_m$ in the order that they're covered by ϕ', breaking ties arbitrarily. If assignment A_i was first covered by term t, which covered a total of k_t previously uncovered assignments, we assign to A_i a cost $c(A_i) = |t|/k_t$. The sum of the costs will be the number of literals in our resulting subformula ϕ'. Observe that, at every iteration, the new assignments covered by ϕ' will have been, to within the uncertainty of our estimation, of minimum cost. When selecting the term that first covers assignment A_i, we have already covered $i - 1$ assignments, and we know that all assignments can be covered with only OPT literals, so we must be able to pick A_i such that $c(A_i) \le c^2 \cdot \mathrm{OPT}/(m-(i-1))$. The factor of c^2 results from the chance of overestimating the costs of the cheapest terms and underestimating the costs of more expensive ones. Summing this inequality for all i from 1 to m, we get

$$\begin{aligned}\sum_{i=1}^{m} c(A_i) &\le \sum_{i=1}^{m} c^2 \cdot \mathrm{OPT}/(m-(i-1)) \\ &= c^2 \cdot \mathrm{OPT} \sum_{i=1}^{m} 1/(m-(i-1)) \\ &\le c^2 \cdot \mathrm{OPT} \ln m \\ &= O(\log m) \cdot \mathrm{OPT}.\end{aligned}$$

As $m \le 2^R$, $\log m \le R < n^{1/2}$, so this gives the desired approximation ratio. A union bound over our $O(n^2)$ estimations, each of which will be inaccurate with probability $o(n^2)$, ensures that we err with negligible probability. This completes the approximation algorithm for IRREDUNDANT*(LITERALS). □

4 Conclusions and open problems

The most natural remaining open problems related to the results in this paper are to close the remaining gaps in the approximability of these problems. Our results for IRREDUNDANT*(LITERALS) suggest (by analogy) that perhaps the correct answer for some of these problems is an approximation threshold of n^δ for a constant δ bounded away from 1. If this is the case, then settling the approximability question will require not just tighter reductions, but clever approximation algorithms (that are of course allowed access to an NP oracle, since we are dealing with Σ_2^p minimization problems).

We also note that there are numerous optimization problems listed in [SU02a] with large gaps in the best known upper and lower bounds on their approximability.

References

[BGP00] M. Bellare, O. Goldreich, and E. Petrank. Uniform generation of NP-witnesses using an NP-oracle. *Information and Computation (formerly Information and Control)*, 163, 2000.

[SU02a] M. Schaefer and C. Umans. Completeness in the Polynomial-Time Hierarchy: A Compendium. *SIGACT News*, 33(3):32–49, 2002.

[SU02b] M. Schaefer and C. Umans. Completeness in the Polynomial-Time Hierarchy: Part II. *SIGACT News*, 33(4):22–36, 2002.

[TSUZ07] A. Ta-Shma, C. Umans, and D. Zuckerman. Lossless Condensers, Unbalanced Expanders, And Extractors. *Combinatorica*, 27(2):213–240, 2007.

[Uma99] C. Umans. Hardness of approximating Σ_2^P minimization problems. In *40th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 465–474, 1999.

[Uma01] C. Umans. The minimum equivalent DNF problem and shortest implicants. *J. Comput. Syst. Sci.*, 63(4):597–611, 2001. FOCS 1998 Special Issue.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de