The complexity of SPP formula minimization

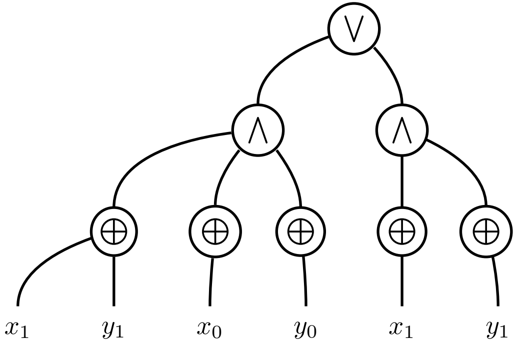
Dave Buchfuhrer

December 16, 2008

Outline

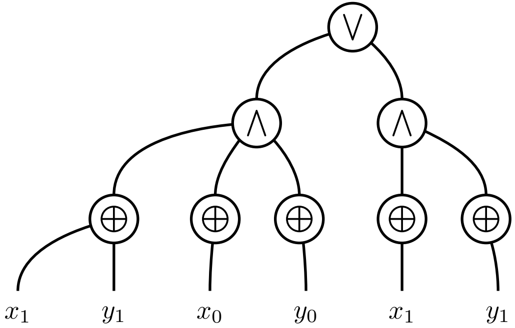
SPP Formulae

An SPP formula consists of 3 levels, the top of which is and OR gate, followed by AND gates then XOR (parity) gates



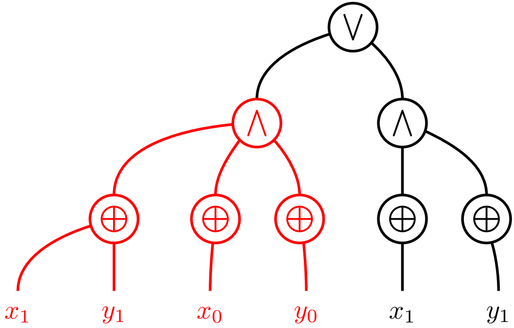
Pseudoproducts

The following SPP formula consists of 2 pseudoproducts



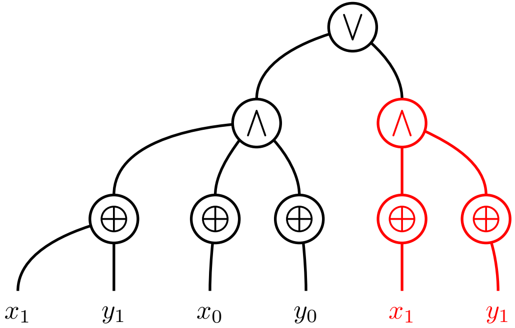
Pseudoproducts

The following SPP formula consists of 2 pseudoproducts



Pseudoproducts

The following SPP formula consists of 2 pseudoproducts



SPP Minimization

Problem (SPP Minimization)

Given an SPP S and an integer k, does there exist an SPP S' of size at most k such that $S \equiv S'$?

SPP Minimization

Problem (SPP Minimization)

Given an SPP S and an integer k, does there exist an SPP S' of size at most k such that $S \equiv S'$?

We show that this is complete for Σ_2^P under Turing reductions

Background Information

- DNF Minimization is Σ_2^P-complete (Umans, '98)
- Constant depth/unlimited depth $(\vee, \wedge, \neg)$ formula minimization is Σ_2^P-complete (Buchfuhrer, Umans, '08)
- SPP Minimization is clearly coNP-hard, but no matching upper-bound

Outline

Equivalence

Problem (SPP Equivalence)

Given two SPP formulae S, T, do both S and T compute the same function?

Equivalence

Problem (SPP Equivalence)

Given two SPP formulae S, T, do both S and T compute the same function?

coNP-complete because it contains DNF Equivalence as a special case

Irredundancy

Problem (SPP Irredundancy)

Given an SPP S and and an integer k, does there exist an SPP S' equivalent to S which is composed of exactly k pseudoproducts from S?

Irredundancy

Problem (SPP Irredundancy)

Given an SPP S and and an integer k, does there exist an SPP S' equivalent to S which is composed of exactly k pseudoproducts from S?

Σ_2^P-hard to approximate within n^ϵ because it contains DNF Irredundancy as a special case (shown hard by Umans, '99)

Prime Pseudoproducts

Definition (Prime Pseudoproduct)

A prime pseudoproduct of an SPP S is a pseudoproduct P which implies S, but does not imply any other pseudoproduct which implies S.

Prime Pseudoproducts

Definition (Prime Pseudoproduct)

A prime pseudoproduct of an SPP S is a pseudoproduct P which implies S, but does not imply any other pseudoproduct which implies S.

Problem

SPP Prime Pseudoproduct Given an SPP S and a pseudoproduct P, is P a prime pseudoproduct of S?

Prime Pseudoproducts

Definition (Prime Pseudoproduct)

A prime pseudoproduct of an SPP S is a pseudoproduct P which implies S, but does not imply any other pseudoproduct which implies S.

Problem

SPP Prime Pseudoproduct Given an SPP S and a pseudoproduct P, is P a prime pseudoproduct of S?

DP-Hard by same reduction showing DNF version is DP-Hard in (GHM, 08)

Outline

Outline

Modified Succinct Set Cover (MSSC)

Given a DNF formula D on variables

$$v_1, \ldots, v_m, x_1, \ldots, x_n$$

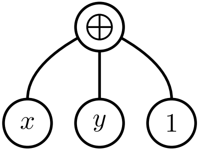
and an integer k, is there a subset $I \subseteq \{1, 2, \ldots, n\}$ with $|I| \leq k$ and for which

$$D \vee \bigvee_{i \in I} \neg x_i \equiv \left(\bigvee_{i=1}^{m} \neg v_i \vee \bigvee_{i=1}^{n} \neg x_i \right)?$$

Outline

Weighting

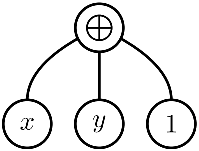
Under the usual size function, the formula



has size 2.

Weighting

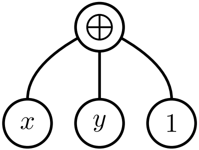
Under the usual size function, the formula



has size 2. If we give weights $w(x) = 2$, $w(y) = 3$, it has size 5.

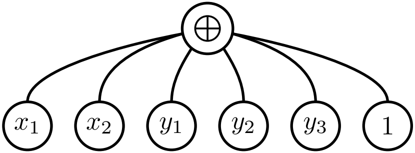
Weighting by Substitution

What if we replace x by the XOR of $w(x)$ new variables below?



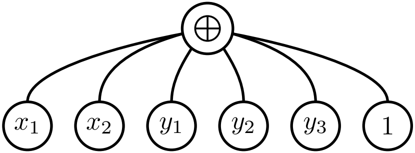
Weighting by Substitution

What if we replace x by the XOR of $w(x)$ new variables below?



Weighting by Substitution

What if we replace x by the XOR of $w(x)$ new variables below?



It is not difficult to see that this substitution preserves minimum formula size

Outline

High-Level Description of the Reduction

- We start with an MSSC instance $\langle D, x_1, \ldots, x_n, k \rangle$ of MSSC

High-Level Description of the Reduction

- We start with an MSSC instance $\langle D, x_1, \ldots, x_n, k\rangle$ of MSSC
- We create new variables $z_1, \ldots, z_\ell$ for large (but poly) ℓ

High-Level Description of the Reduction

- We start with an MSSC instance $\langle D, x_1, \ldots, x_n, k \rangle$ of MSSC
- We create new variables $z_1, \ldots, z_\ell$ for large (but poly) ℓ
- Weight the x_i variables by $k\ell$ and the v_i variables over k times higher than this

High-Level Description of the Reduction

- We start with an MSSC instance $\langle D, x_1, \ldots, x_n, k \rangle$ of MSSC
- We create new variables $z_1, \ldots, z_\ell$ for large (but poly) ℓ
- Weight the x_i variables by $k\ell$ and the v_i variables over k times higher than this
- Check the minimum size for $D \vee (Z \wedge \neg x_1) \vee \cdots \vee (Z \wedge \neg x_n)$ where $Z = z_1 \wedge \cdots \wedge z_\ell$

High-Level Description of the Reduction

- We start with an MSSC instance $\langle D, x_1, \ldots, x_n, k \rangle$ of MSSC
- We create new variables $z_1, \ldots, z_\ell$ for large (but poly) ℓ
- Weight the x_i variables by $k\ell$ and the v_i variables over k times higher than this
- Check the minimum size for $D \vee (Z \wedge \neg x_1) \vee \cdots \vee (Z \wedge \neg x_n)$ where $Z = z_1 \wedge \cdots \wedge z_\ell$
- The minimum equivalent formula should look like

$$D \vee \bigvee_{i \in I} (z_1 \wedge \cdots \wedge z_\ell \wedge \neg x_i)$$

Outline

Why the z_i Separate the D and the $\neg x_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

Why the z_i Separate the D and the $\neg x_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

Proof (sketch)

- With each variable z_i associate the vector $\hat{z}_i \in \mathbb{Z}_2^\beta$ which contains a 1 in position j iff z_i is in the jth XOR gate

Why the z_i Separate the D and the $\neg x_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

Proof (sketch)

- With each variable z_i associate the vector $\hat{z}_i \in \mathbb{Z}_2^\beta$ which contains a 1 in position j iff z_i is in the jth XOR gate
- Also create such a vector $\hat{c}$ for the constants

Why the z_i Separate the D and the $\neg x_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

Proof (sketch)

- With each variable z_i associate the vector $\hat{z}_i \in \mathbb{Z}_2^{\beta}$ which contains a 1 in position j iff z_i is in the jth XOR gate
- Also create such a vector $\hat{c}$ for the constants
- A pseudoproduct accepts when $\hat{c} + \sum z_i \hat{z}_i = \vec{1}$

Why the z_i Separate the D and the $\neg x_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

Proof (sketch)

- With each variable z_i associate the vector $\hat{z}_i \in \mathbb{Z}_2^{\beta}$ which contains a 1 in position j iff z_i is in the jth XOR gate
- Also create such a vector $\hat{c}$ for the constants
- A pseudoproduct accepts when $\hat{c} + \sum z_i \hat{z}_i = \vec{1}$
- Since $\bigwedge_{i=1}^{\alpha} z_i$ accepts exactly one assignment, the $\hat{z}_i$ are linearly independent □

Why the z_i Separate the D and the $\neg x_i$

Formula to Minimize

$D \vee \bigvee_i (z_1 \wedge \cdots \wedge z_\ell \wedge \neg x_i)$

Consider a pseudoproduct P that accepts some assignment σ not accepted by D.

- Suppose that z_{i^*} occurs in an XOR gate shared only by other z_i variables.
- By appropriately restricting the other z_i, P implies $z_{i^*} = \mathrm{true}$
- Can we find a single index i^* that works for all P?

Why the z_i Separate the D and the $\neg x_i$

- Let H_P be the set of z_i variables in P which only appear in XORs containing non-z_i variables

Why the z_i Separate the D and the $\neg x_i$

- Let H_P be the set of z_i variables in P which only appear in XORs containing non-z_i variables
- If we restrict to σ on every variable outside of H_P, P becomes $\bigwedge_{z_i \in H_P} z_i$

Why the z_i Separate the D and the $\neg x_i$

- Let H_P be the set of z_i variables in P which only appear in XORs containing non-z_i variables
- If we restrict to σ on every variable outside of H_P, P becomes $\bigwedge_{z_i \in H_P} z_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

- So by the above lemma, there are at least H_P non-z_i variables in P

Why the z_i Separate the D and the $\neg x_i$

- Let H_P be the set of z_i variables in P which only appear in XORs containing non-z_i variables
- If we restrict to σ on every variable outside of H_P, P becomes $\bigwedge_{z_i \in H_P} z_i$

Lemma

A pseudoproduct which computes $\bigwedge_{i=1}^{\alpha} z_i$ must contain at least α XOR gates.

- So by the above lemma, there are at least H_P non-z_i variables in P
- Thus, if no index i^* works for all variables, the size is at least $k\ell^2$

Outline

Finishing Up

Now, we have an SPP formula of the form

$$D \vee \bigvee_i (z_1 \wedge \cdots \wedge z_\ell \wedge X_i)$$

for some set of pseudoproducts $\{X_i\}$, where

$$D \vee \bigvee_i X_i \equiv \bigvee_i \neg v_i \vee \bigvee_i \neg x_i$$

Finishing Up

We now take the simple lemma

Lemma

The smallest SPP formula accepting every assignment in a set S but not the all true assignment is of the form $\bigvee_i \neg y_i$

and apply it to

$$D \vee \bigvee_i X_i \equiv \bigvee_i \neg v_i \vee \bigvee_i \neg x_i$$

to see that $\bigvee_i X_i$ is at least as large as $\bigvee_{i \in I} \neg x_i$ for the smallest possible I

Done

Recall that the problem we reduced from asks whether there is a subset $I \subseteq \{1, 2, \ldots, n\}$ with $|I| \leq k$ and for which

$$D \vee \bigvee_{i \in I} \neg x_i \equiv \left(\bigvee_{i=1}^{m} \neg v_i \vee \bigvee_{i=1}^{n} \neg x_i \right) ?$$

Done

Recall that the problem we reduced from asks whether there is a subset $I \subseteq \{1, 2, \ldots, n\}$ with $|I| \leq k$ and for which

$$D \vee \bigvee_{i \in I} \neg x_i \equiv \left(\bigvee_{i=1}^{m} \neg v_i \vee \bigvee_{i=1}^{n} \neg x_i \right)?$$

So we want to know the total size of the X_i in

$$D \vee \bigvee_{i} (z_1 \wedge \cdots \wedge z_\ell \wedge X_i)$$

Outline

Open Problems

- Can the result be shown under many-one reductions?
- Hardness of approximation
- Limited fanout XOR gates

Open Problems

- Can the result be shown under many-one reductions?
- Hardness of approximation
- Limited fanout XOR gates

For a full version of this paper, visit
http://www.cs.caltech.edu/~dave/papers/