Strong Bounds on the Approximability of Two PSPACE-hard Problems in Propositional Planning

Peter Jonsson
Department of Computer and Information Science
Linköping Universitet
S-581 83 Linköping, Sweden
email: petej@ida.liu.se
phone: +46 13 28 24 15
fax: +46 13 28 26 06

Abstract

The computational complexity of planning with STRIPS-style operators has received a considerable amount of interest in the literature. However, the approximability of such problems has only received minute attention. We study two PSPACE-hard optimization versions of propositional planning and provide tight upper and lower bounds on their approximability.

Keywords: Computational complexity; Approximation; Planning; PSPACE.

AMS Classification: Primary 68T20, Secondary 68Q25.

1 Introduction

Recently, there has been an increasing interest in studying the approximation properties of PSPACE-hard problems. In similarity to NP-hard approximation problems, PSPACE-hard approximation problems display a wide variation in the precision with which they can be approximated efficiently. Results in the literature show that some PSPACE-hard problems have efficient approximation algorithms while others have such if and only if some complexity-theoretic inclusion such as P=NP or P=PSPACE hold.

The PSPACE-complete problem *propositional planning* is investigated in this article. This problem has been the subject of intensive study in the knowledge representation, control theory[1] and artificial intelligence communities for at least 25 years (for a collection of papers, see [1]). However, the approximability of problems in connection with propositional planning has been given very little attention in the literature. The objective of this paper is to show nonapproximability results (which are close to optimal) for two optimization versions of this problem.

[1] Planning is generally denoted *sequential control* in the control theory literature.

1.1 Results

The problems we will study are Number of Satisfied Goals (NSG) and Plan Length (PL). In the NSG problem, an instance of the propositional planning problem is given as input. The objective is to determine how many of the goals can (at most) be simultaneously satisfied by any plan. Clearly, the planning problem is solvable in the ordinary sense if and only if *all* goals can be satisfied. The PL problem has the same set of instances as NSG but the goal is to determine the length of the shortest plan solving the instance (if such a plan exists).

We say that a function f approximates a function g within a function α iff $g(x)/\alpha(|x|) \leq f(x) \leq g(x) \cdot \alpha(|x|)$ for all instances x. We show that any function which approximates NSG within $n^{1/2-\varepsilon}$ for some $\varepsilon > 0$ is Pspace-hard to compute. Furthermore, every function f which approximates NSG with one-sided error (*i.e.*, $g(x)/\alpha(|x|) \leq f(x) \leq g(x)$) within $n^{1-\varepsilon}$ is Pspace-hard. Since NSG is trivially approximable within n, this bound is essentially optimal.

For the PL problem, we prove that any function which approxiamtes PL within

$$\sqrt{2^{n(1-n^{-\varepsilon})}} \approx 1.41^{n(1-n^{-\varepsilon})}$$

for some $\varepsilon < 1$ is Pspace-hard. By restricting our attention to functions with one-sided error, we show that any function which approximates PL within

$$2^{n(1-n^{-\varepsilon})}$$

for some $\varepsilon < 1$ is Pspace-hard. Since PL can be approximated within 2^n in polynomial time, this bound is close to optimal.

1.2 Related Work

The literature on approximating Pspace-hard problems is rapidly growing so we do not attempt to give a full survey of the results. An early example of a nonapproximability result is Orlin's [20] paper on approximating a periodic version of the Knapsack problem. Periodic and hierarchically defined structures have then been studied quite extensively since such problems arise in scheduling and VLSI applications. Marathe *et al.* [19] contains several results of this type. Nonapproximability results for some other problems, including Max QDones and Finite Function Generation, can be found in Hunt *et al.* [16] and Jonsson [17].

The techniques of *probabilistically checkable proofs* which have been so important in understanding the approximability of NP-hard problems have also influenced the study of Pspace-hard problems. Condon *et al.* [8] introduced the *probabilistically checkable debate systems* (PCDS) and used them for showing nonapproximability of optimization versions of Pspace-hard problems such as Quantified Satisfiability, Generalized Geography and Finite Automata Intersection. With similar techniques, they have also showed nonapproximability of certain stochastic problems including Dynamic Graph Reliability and Mah-Jong [9].

Approximation results for propositional planning have not appeared very frequently in the literature. Selman [22] has provided a weak lower bound on the approximability of PL—the problem cannot be approximated within any constant in polynomial time. PL for a certain class of planning problems, known as the *blocks-world problem*, has been shown to be approximable within 2 but not within every constant between 1 and 2 [22]. However, this problem is NP-complete and thus not directly comparable with the problems studied in this article. Other approximation results for problems in propositional planning which are not directly related to those we consider can be found in Bäckström [2] and Bäckström & Jonsson [4].

In a paper by Ginsberg [15], the phrase 'approximate planning' is used but it has a meaning which differs from the usual interpretation of approximation. Ginsberg's idea is that a planning algorithm approximates the planning problem if most plans (measured in some suitable way) that are returned are correct. We will not consider this notion of approximation in this article.

1.3 Structure of the Article

The article is organized as follows: In Section 2 we define the propositional planning problem formally and give some basic complexity-theoretic results. Section 3 contains definitions of approximability and related concepts together with the reduction techniques used. Section 4 collects the nonapproximability results for NSG and Section 5 the results for PL. Some concluding remarks and directions for future research can be found in Section 6.

2 Propositional Planning

We base this article on the common propositional STRIPS (CPS) formalism [14], which is equivalent to most other variants of propositional planning [3].

Definition 2.1 An instance of the *propositional planning problem* is a quadruple $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ where

- $\mathcal{P}$ is a finite set of *atoms*;
- $\mathcal{O}$ is a finite set of *operators* of the form $\langle pre, add, del \rangle$, where $pre \subseteq \mathcal{P}$ denote the *precondition* and $add, del \subseteq \mathcal{P}$ denote the *add* and *delete lists* respectively, satisfying $add \cap del = \varnothing$;
- $\mathcal{I}, \mathcal{G} \subseteq \mathcal{P}$ denote the *initial state* and *goal state*, repsectively.

Let CPS denote the set of instances of the propositional planning problem. For $o = \langle pre, add, del \rangle \subseteq \mathcal{O}$, we write $pre(o)$, $add(o)$ and $del(o)$ to denote pre, add and del respectively. Let $\mathcal{S}(\Pi) = 2^{\mathcal{P}}$ denote the set of *states* over

Π and let $\mathcal{S}_\perp(\Pi) = \mathcal{S}(\Pi) \cup \{\perp\}$ where $\perp$ is the undefined element. The operators are functions on $\mathcal{S}_\perp$, defined as follows: let o be an operator and $S \in \mathcal{S}_\perp$, then

$$o(S) = \begin{cases} (S - del(o)) \cup add(o) & \text{if } pre(o) \subseteq S \\ \perp & \text{otherwise} \end{cases}$$

Given a set of operators $\mathcal{O}$, we define $Seqs(\mathcal{O})$ as the set of all finite sequences over $\mathcal{O}$. We say that the members of $Seqs(\mathcal{O})$ are the *plans* over Π and we define the notion of a plan being a solution to an instance as follows: A plan $\omega = \langle o_1, \ldots, o_n \rangle \in Seqs(\mathcal{O})$ solves Π iff

$$\mathcal{G} \subseteq o_n(o_{n-1}(\ldots(o_1(\mathcal{I}))\ldots))$$

We use $\omega(\mathcal{I})$ as short-hand for the expression $o_n(o_{n-1}(\ldots(o_1(\mathcal{I}))\ldots))$.

We have the following basic computational problem for propositional planning.

Definition 2.2 The Plan Existence Problem (Pe) is: given a CPS instance Π, decide whether Π has a solution or not.

The following theorem is a consequence of Bylander [6] and Bäckström [3].

Theorem 2.3 **PE** is Pspace-complete.

The next lemma shall be needed several times in the sequel.

Lemma 2.4 If $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ is a CPS instance with a shortest solution ω then $|\omega| \leq 2^{|\mathcal{P}|}$.

Proof: Since ω is a shortest solution, $|\omega| \leq \mathcal{S}(\Pi) = 2^{|\mathcal{P}|}$. □

3 Approximations and Reductions

This section contains some basic definitions regarding the approximability of functions together with the reduction techniques which will be used in the nonapproximability proofs. We begin by defining the general optimization problem.

Definition 3.1 An *optimization problem* A is a quadruple $\langle I, sol, m, type \rangle$ such that

1. I is the set of instances of A.
2. Given an instance $x \in I$, $sol(x)$ denotes the set of feasible solutions of x.
3. Given an instance x and a feasible solution y of x, $m(x, y)$ denotes the positive integer measure of y (often also called the value of y).

4. $type \in \{\min, \max\}$.

Define the *optimal value function* $opt : I \to \mathbb{Z}^+$ as follows:

$$opt(x) = type\{m(x,y) \mid y \in sol(x)\}.$$

The goal of an optimization problem with respect to an instance x is to compute $opt(x)$. An optimization problem such that $type = \min$ is said to be a *minimization problem*. Otherwise, the optimization problem is a *maximization problem*.

Definition 3.2 Given an optimization problem $A = \langle I, sol, m, type\rangle$ with optimal value function opt and a function $\alpha : \mathbb{N} \to (1,\infty)$, we say that a function $f : I \to \mathbb{N}$ *approximates* A *within* α iff for every instance $x \in I$ of size n, $opt(x)/\alpha(n) \leq f(x) \leq \alpha(n)opt(x)$.

Definition 3.3 Let $A = \langle I, sol, m, type\rangle$ be an optimization problem with optimal value function opt, let $\mathcal{C}$ be a complexity class and let $\alpha : \mathbb{N} \to (1,\infty)$ be a function. We say that A *is* $\mathcal{C}$*-hard to approximate* within α iff every function $f : I \to \mathbb{N}$ which approximates A within α is $\mathcal{C}$-hard.

Note the difference between our definition of optimization and the definition of NP optimization given by, for instance, Crescenzi and Panconesi [11] or Crescenzi and Kann [10]. They require that there exists an algorithm which produces a feasible solution y such that $m(x,y)$ is within the given bounds. Furthermore, they require that m is computable in polynomial time. Our definition of approximation merely requires that the approximation algorithm produces a number within these bounds and we do not limit m to be a polynomial-time computable function. There is a good reason for this difference since given an NP optimization problem instance x, any feasible solution y satisfies $|y| \leq p(|x|)$ for some polynomial p. Further, m is computable in polynomial time so we can compute $m(x,y)$ in time bounded by $q(|x|)$ (where q is some polynomial) for any y. This is not the case for a general optimization problem where there does not exist any polynomial p such that $|y| \leq p(|x|)$ for arbitrary instances x and feasible solutions y. Hence, we do not require that a feasible solution be explicitly generated.[2]

Also note that the definition of NP optimization problems implies that the approximation algorithm has one-sided error: Given an instance x of a maximization problem A, the feasible solution y generated by an approximation algorithm cannot be such that $opt_A(x) < m(x,y)$. Clearly, this guarantee does not hold for general optimization problems. We have thus motivated the next definition.

Definition 3.4 Let $A = \langle I, sol, m, type\rangle$ be a maximization problem with optimal value function opt, let $\mathcal{C}$ be a complexity class and let $\alpha : \mathbb{N} \to (1,\infty)$

[2] Approximation algorithms which do not produce a feasible solution are often called *non-constructive*, *cf.* Crescenzi and Trevisan [12].

be a function. We say that A *is $\mathcal{C}$-hard to approximate with one-sided error* within α iff every function $f : I \to \mathbb{N}$ satisfying

$$opt(x)/\alpha(|x|) \leq f(x) \leq opt(x)$$

is $\mathcal{C}$-hard. The definition for minimization problems is analogous.

We can now present the reduction techniques which will be used in this article and we begin with a reduction suitable for maximization problems. For an arbitrary function $f : X \to Y$, let $Img(f)$ denote the set $\{f(x) \mid x \in X\}$.

Lemma 3.5 Let the following be given:

1. a maximization problem A;
2. a complexity class $\mathcal{C}$ such that $P \subseteq \mathcal{C}$;
3. a function $\ell : \mathbb{N} \to \mathbb{N}$ and two functions $a, b : \mathbb{N} \to (1, \infty)$ with the following property: for arbitrary $i \in \mathbb{N}$ and $j \in Img(\ell)$, it can be decided whether $i > a(j)$ and it can be decided whether $i > a(j) \cdot b(j)$ in polynomial time.

Let T be a polynomial-time transformation from a $\mathcal{C}$-hard decision problem F to A such that for an arbitrary instance I of F of size n, the size of $T(I)$ is $\ell(n)$. Furthermore, assume that T has the following two properties:

1. If $I \in F$ then $opt_A(T(I)) > b(|T(I)|)$; and
2. If $I \notin F$ then $opt_A(T(I)) < a(|T(I)|)$.

Then, A is $\mathcal{C}$-hard to approximate within $\alpha(n) = \sqrt{b(n)/a(n)}$ and $\mathcal{C}$-hard to approximate with one-sided error within $\alpha(n) = b(n)/a(n)$.

Proof: Assume that K approximates A within $\sqrt{\frac{b(|x|)}{a(|x|)}}$, *i.e.*, for an arbitrary instance x of A,

$$\sqrt{\frac{a(|x|)}{b(|x|)}} \cdot opt_A(x) \leq m_A(x, K(x)) \leq \sqrt{\frac{b(|x|)}{a(|x|)}} \cdot opt_A(x).$$

Arbitrarily select an instance I of F and let $I' = T(I)$ and $s = |I'| = \ell(I)$. Note that the value of s does not depend on whether I is in F or not and $s \in Img(\ell)$. If $I \in F$ then

$$\begin{aligned} K(I') &\geq \sqrt{\tfrac{a(s)}{b(s)}} \cdot opt_A(I') \\ &> \sqrt{\tfrac{a(s)}{b(s)}} \cdot b(s) \\ &= \sqrt{a(s) \cdot b(s)}. \end{aligned}$$

Otherwise,

$$\begin{aligned} K(I') &\leq \sqrt{\tfrac{b(s)}{a(s)}} \cdot opt_A(I') \\ &< \sqrt{\tfrac{b(s)}{a(s)}} \cdot a(s) \\ &= \sqrt{a(s) \cdot b(s)}. \end{aligned}$$

Hence, $I \in F$ iff $K(I') > \sqrt{a(s) \cdot b(s)}$. Note that $K(I') > \sqrt{a(s) \cdot b(s)}$ iff $K(I')^2 > a(s) \cdot b(s)$. Thus, we can decide F in polynomial time with an oracle for K which implies that K is $\mathcal{C}$-hard.

Now, assume that K approximates A with one-sided error. Then, for an arbitrary instance x of A,

$$\frac{a(|x|)}{b(|x|)} \cdot opt_A(x) \leq m_A(x, K(x)) \leq opt_A(x).$$

Arbitrarily select an instance I of F and let $I' = T(I)$ and $s = |I'|$. Note that the value of s does not depend on whether I in F or not. If $I \in F$ then

$$\begin{aligned} K(I') &\geq \tfrac{a(s)}{b(s)} \cdot opt_A(I') \\ &> \tfrac{a(s)}{b(s)} \cdot b(s) \\ &= a(s). \end{aligned}$$

Otherwise,

$$\begin{aligned} K(I') &\leq opt_A(I') \\ &< a(s). \end{aligned}$$

Hence, $I \in F$ iff $K(I') > a(s)$ and we can decide F in polynomial time with an oracle for K. $\mathcal{C}$-hardness of K follows immediately. $\square$

In the previous lemma, the nonapproximability of a problem is described as a function of the size of the problem instance. However, it can also be described as a function of some size parameter of the instance. For all the problems considered in this article, we will assume that $|\cdot|$ denotes the number of propositional atoms in the given instance.

We continue by defining a reduction for proving non-approximability of minimization problems. The proof is omitted since it is similar to the proof of Lemma 3.5.

Lemma 3.6 Let the following be given:

1. a minimization problem A;
2. a complexity class $\mathcal{C}$ such that $P \subseteq \mathcal{C}$;
3. two functions $a, b : \mathbb{N} \rightarrow (1, \infty)$ with the following property: for arbitrary $i, j \in \mathbb{N}$, it can be decided whether $i > a(j)$ and it can be decided whether $i > a(j) \cdot b(j)$ in polynomial time.

Let T be a polynomial-time transformation from a $\mathcal{C}$-hard decision problem F to A such that for an arbitrary instance I of F of size n, the size of $T(I)$ is $\ell(n)$ for some $\ell : \mathbb{N} \rightarrow \mathbb{N}$. Furthermore, assume that T has the following two properties:

1. If $I \in F$ then $opt_A(T(I)) < a(|T(I)|)$; and
2. If $I \notin F$ then $opt_A(T(I)) > b(|T(I)|)$.

Then, A is $\mathcal{C}$-hard to approximate within $\alpha(n) = \sqrt{b(n)/a(n)}$ and $\mathcal{C}$-hard to approximate with one-sided error within $\alpha(n) = b(n)/a(n)$.

4 Number of Satisfied Goals

Recall that the objective of the NSG problem is to determine how many of the goals in a planning problem that can be satisfied. The formal definition is as follows.

Definition 4.1 The maximization problem NUMBER OF SATISFIED GOALS (NSG) $\langle I, sol, m, type\rangle$ is defined as follows:

1. $I = \mathsf{CPS}$.
2. Given an instance $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G}\rangle \in I$, $sol(\Pi) = Seqs(\mathcal{O})$.
3. Given an instance $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G}\rangle \in I$ and $\omega \in Seqs(\mathcal{O})$, $m(\Pi, \omega) = 1 + |\mathcal{G} \cap \omega(\mathcal{I})|$ if $\omega(\mathcal{I}) \neq \perp$. Otherwise $m(\Pi, \omega) = 1$.
4. $type = \max$.

We denote the optimal value function for NUMBER OF SATISFIED GOALS by NSG. Since $|\mathcal{G} \cap \omega(\mathcal{I})|$ may equal 0 for certain choices of $\mathcal{G}$, $\mathcal{I}$ and ω, we let $m(\Pi, \omega) = 1 + |\mathcal{G} \cap \omega(\mathcal{I})|$ so m is a positive measure function. Observe that NSG(Π) = 1 if Π has no solution. Hence, we implicitly assume the existence of trivial solutions and ensure that NSG is a total function.

Also note that NSG can be computed in polynomial space. This follows from the fact that we can nondeterministically guess the plan one operator at a time and compute the resulting state after applying each new operator, using only polynomial space. Furthermore, if the instance contains n atoms, the length of the shortest solution must be less than 2^n by Lemma 2.4. Hence, no more than 2^n nondeterministic choices are required and the problem is in NSPACE and, by Savitch's [21] theorem, in PSPACE.

To prove the nonapproximability of NSG, we use a construction which is presented in Definition 4.2. With the aid of this construction, planning instances having large gaps in the number of satisfied goals can be generated and this is proved in Lemma 4.3. We use this fact in Lemma 4.4 and Theorem 4.5 to show the desired result.

Definition 4.2 Arbitrarily choose an integer $n > 0$ and a CPS instance $\Pi = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle$. Let $\mathcal{Q} = \{q_1, \ldots, q_n\}$ be a set of atoms such that $\mathcal{Q} \cap \mathcal{P}_\Pi = \varnothing$. Define $\Pi_{\mathrm{NSG}}(\Pi, n) = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ such that

- $\mathcal{P} = \mathcal{P}_\Pi \cup \mathcal{Q}$,
- $\mathcal{O} = \mathcal{O}_\Pi \cup \{o_*\}$ where $o_* = \langle \mathcal{G}_\Pi, \mathcal{P}, \varnothing \rangle$,
- $\mathcal{I} = \mathcal{I}_\Pi$,
- $\mathcal{G} = \mathcal{G}_\Pi \cup \mathcal{Q}$.

Lemma 4.3 Arbitrarily choose a CPS instance $\Pi = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle$ and an integer $n > 0$. Let $\Sigma = \Pi_{\mathrm{NSG}}(\Pi, n)$. Then $\mathrm{NSG}(\Sigma) < |\mathcal{G}_\Pi| + 1$ iff Π has no solution. Otherwise, $\mathrm{NSG}(\Sigma) = n + |\mathcal{G}_\Pi| + 1$.

Proof: If Π has no solution, then the preconditions of the operator o_* cannot be satisfied. Hence, no plan can lead to a state S such that an atom $q \in \mathcal{Q}$ is in S and $\mathrm{NSG}(\Sigma) < |\mathcal{G}_\Pi| + 1$. Otherwise, the preconditions of o_* can be satisfied and the state $\mathcal{P}$ can be reached. Consequently, $\mathrm{NSG}(\Sigma) = n + |\mathcal{G}_\Pi| + 1$. □

Lemma 4.4 Let $C > 0$ be a fixed integer. Then NSG is PSPACE-hard to approximate within $\sqrt{\frac{n}{\sqrt[C]{n}}}$ where n is the number of atoms in the given instance. Furthermore, NSG is PSPACE-hard to approximate with one-sided error within $\frac{n}{\sqrt[C]{n}}$.

Proof: The proof is by a polynomial-time reduction from the PSPACE-complete problem PE. Define functions $a, b : \mathbb{N} \to (1, \infty)$ such that $a(n) = \sqrt[c]{n}$ and $b(n) = n$. For arbitrary $i, j \in \mathbb{N}$, we can decide (in polynomial time) whether $i > a(j)$ since this is equivalent to $i^c > j$. Similarly, it is decidable in polynomial time whether $i > a(j) \cdot b(j)$ since

$$i > a(j) \cdot b(j) \Leftrightarrow i > \sqrt[C]{j(j)^C} \Leftrightarrow i^C > j^{C+1}.$$

Furthermore, let $\ell : \mathbb{N} \to \mathbb{N}$ be such that $\ell(n) = n^C$.

Arbitrarily choose a CPS instance $\Sigma = \langle \mathcal{P}_\Sigma, \mathcal{O}_\Sigma, \mathcal{I}_\Sigma, \mathcal{G}_\Sigma \rangle$. Let

$$\Pi = \Pi_{\mathrm{NSG}}(\Sigma, |\mathcal{P}_\Sigma|^C - |\mathcal{P}_\Sigma|) = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle.$$

This is a polynomial-time transformation since C is fixed. By this construction, $|\mathcal{P}_\Pi| = |\mathcal{P}_\Sigma|^C = \ell(|\mathcal{P}_\Sigma|)$.

If Σ is not solvable, then

$$\mathrm{NSG}(\Pi) < |\mathcal{G}_\Sigma| + 1 \leq |\mathcal{P}_\Sigma| = \sqrt[C]{|\mathcal{P}_\Pi|} = a(|\mathcal{P}_\Pi|)$$

by Lemma 4.3. If Σ is solvable, then

$$\mathrm{NSG}(\Pi) = |\mathcal{P}_\Pi| + 1 > b(|\mathcal{P}_\Pi|)$$

by Lemma 4.3. By noting that PE is PSPACE-hard, it follows from Lemma 3.5 that NSG is PSPACE-hard to approximate within $\sqrt{\frac{n}{\sqrt[C]{n}}}$ where n is the number of atoms in the given instance. Furthermore, NSG is PSPACE-hard to approximate with one-sided error within $\frac{n}{\sqrt[C]{n}}$. □

Theorem 4.5 Arbitrarily choose $\varepsilon > 0$. Approximating NSG within $n^{1/2-\varepsilon}$ is PSPACE-hard where n is the number of atoms in the instance. Approximating NSG with one-sided error within $n^{1-\varepsilon}$ is PSPACE-hard.

Proof: Choose $C > 0$ such that $(C-1)/2C > 1/2 - \varepsilon$. By the previous lemma, NSG is PSPACE-hard to approximate within

$$\sqrt{\frac{n}{\sqrt[C]{n}}} = n^{\frac{C-1}{2C}} > n^{1/2-\varepsilon}.$$

Now, assume that we only consider functions which approximate NSG with one-sided error. Choose $C > 0$ such that $(C-1)/C > 1-\varepsilon$. By the previous lemma, NSG is PSPACE-hard to approximate with one-sided error within

$$\frac{n}{\sqrt[C]{n}} = n^{\frac{C-1}{C}} > n^{1-\varepsilon}.$$

Since ε was arbitrarily chosen, the theorem follows. □

Note that the function $f(\Pi) = 1$ approximates NSG with one-sided error within n in polynomial time. Thus, the previous theorem cannot be substantially improved.

5 Plan Length

We begin this section by formally defining the optimization problem PLAN LENGTH.

Definition 5.1 The minimization problem PLAN LENGTH (which we abbreviate PL) $\langle I, sol, m, type\rangle$ is defined as follows:

1. $I = \mathsf{CPS}$.
2. Given an instance $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G}\rangle \in I$, $sol(\Pi) = Seqs(\mathcal{O})$.
3. Given an instance $\Pi = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G}\rangle \in I$ and $\omega \in Seqs(\mathcal{O})$, $m(\Pi, \omega) = 1 + |\omega|$ if $\omega(\mathcal{I}) \neq \perp$. Otherwise $m(\Pi, \omega) = 2^{|\Pi|}$.
4. $type = \min$.

We denote the optimal value function for PLAN LENGTH by PL. Note that the measure function m is adjusted to make it positive and we assume the existence of trivial solutions in order to make PL total. Showing the membership of PL in PSPACE is analogous to showing that NSG can be computed in polynomial space.

We prove the nonapproximability of PL in two steps. In Lemma 5.3, we prove the existence of planning instances which have extremely long shortest solution. This proof is inspired by Bäckström and Klein [5]. Such planning instances are then used to construct planning instances which have large gaps in their plan lengths; the main proof of this can be found in Lemma 5.5. After a few results of a technical nature, the nonapproximability of PL is proved in Theorem 5.8.

Definition 5.2 For a given integer $n > 0$, define $\Pi_{\mathrm{EXP}}(n) = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ such that $\mathcal{P} = \{p_1, \ldots, p_n\}$, $\mathcal{O} = \{o_1^+, \ldots, o_n^+\}$, $\mathcal{I} = \varnothing$ and $\mathcal{G} = \mathcal{P}$. The operators in $\mathcal{O}$ are defined as follows:

$o_1^+ = \langle \varnothing, \{p_1\}, \varnothing \rangle$; and

$o_k^+ = \langle \{p_1, \ldots, p_{k-1}\}, \{p_k\}, \{p_1, \ldots, p_{k-1}\} \rangle$ for $k > 1$.

Lemma 5.3 For arbitrary $n > 0$, there exists a minimal solution of $\Pi_{\mathrm{EXP}}(n)$ and this solution contains $2^n - 1$ operators.

Proof: Consider $\Pi_{\mathrm{EXP}}(n) = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ for some $n > 1$. Note that $p_1, \ldots, p_{n-1}$ must be in a state S to make o_n^+ applicable and $o_n^+(S) = \{p_n\}$. Hence, the minimal plan for solving $\Pi_{\mathrm{EXP}}(n)$ is the following:

$(\omega_{n-1}; o_n^+; \omega_{n-1})$

where ω_{n-1} is a minimal plan for solving the problem $\langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} - \{p_n\} \rangle$. Obviously, $\omega_1 = o_1^+$ so the length of ω_n satisfies the following recursive equation:

$|\omega_1| = 1$;

$|\omega_n| = 2|\omega_{n-1}| + 1$.

Thus, $|\omega_n| = 2^n - 1$. □

The solution ω_n in the above proof corresponds to a Hamiltonian path from $\mathcal{I}$ to $\mathcal{G}$ in the state space $\mathcal{S}(\Pi_{\mathrm{EXP}}(n))$. In fact, if we interpret the states as binary numbers where p_1 denotes the least significant bit and p_n the most significant bit, the states are "visited" in numerical order, *i.e.*, 0, 1, 2...

Definition 5.4 Let $\Pi = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle$ be an arbitrary CPS instance.

Choose an arbitrary integer $n > |\mathcal{P}_\Pi|$ and let $\Sigma = \Pi_{\mathrm{EXP}}(n - |\mathcal{P}_\Pi|) = \langle \mathcal{P}_{\mathrm{EXP}}, \mathcal{O}_{\mathrm{EXP}}, \mathcal{I}_{\mathrm{EXP}}, \mathcal{G}_{\mathrm{EXP}} \rangle$. Assume that $\mathcal{P}_\Pi$ and $\mathcal{P}_{\mathrm{EXP}}$ are disjoint sets.

Define $\Pi_{\mathrm{PL}}(\Pi, n) = \langle \mathcal{P}, \mathcal{O}, \mathcal{I}, \mathcal{G} \rangle$ such that

- $\mathcal{P} = \mathcal{P}_\Pi \cup \mathcal{P}_{\mathrm{EXP}}$,
- $\mathcal{O} = \mathcal{O}_\Pi \cup \mathcal{O}_{\mathrm{EXP}} \cup \{o_\Pi^+, o_{\mathrm{EXP}}^+\}$,
- $\mathcal{I} = \mathcal{I}_\Pi$,
- $\mathcal{G} = \mathcal{P}$.

where $o_\Pi^+ = \langle \mathcal{G}_\Pi, \mathcal{P}, \varnothing \rangle$ and $o_{\mathrm{EXP}}^+ = \langle \mathcal{G}_{\mathrm{EXP}}, \mathcal{P}, \varnothing \rangle$.

Lemma 5.5 For arbitrary CPS instances $\Pi = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle$ and $n > |\mathcal{P}_\Pi|$, $\Sigma = \Pi_{\mathrm{PL}}(\Pi, n)$ has a minimal solution ω. Furthermore, if Π has a solution, then $|\omega| \leq 2^{|\mathcal{P}_\Pi|}$. Otherwise, $|\omega| = 2^{|\mathcal{P}_\Sigma| - |\mathcal{P}_\Pi|}$.

Proof: There are two possible minimal plans solving Σ, namely $\omega_\Pi = (\psi_\Pi; o_\Pi^+)$ and $\omega_{\mathrm{EXP}} = (\psi_{\mathrm{EXP}}; o_{\mathrm{EXP}}^+)$ where ψ_Π is a minimal plan solving Π (if such a plan exists) and ψ_{EXP} is the minimal solution to $\Pi_{\mathrm{EXP}}(n)$. Since $\Pi_{\mathrm{EXP}}(n)$ is solvable for all n (by Lemma 5.3), Σ is solvable.

Obviously, if ω_Π exists, then $|\omega_\Pi| \leq 2^{|\mathcal{P}_\Pi|} - 1$ Hence, if Π has a solution, then the minimal plan ω solving Σ satisfies $|\omega| \leq 2^{|\mathcal{P}_\Pi|}$ and, otherwise, $|\omega| = 2^{|\mathcal{P}_\Sigma| - |\mathcal{P}_\Pi|}$. □

Lemma 5.6 Assume $i, j, k \in \mathbb{Z}^+$ and $i = j^k$. If i and k are given then j can be determined in polynomial time.

Proof: Clearly, $1 \leq j \leq i$. Given i and k, j can be computed by binary search. In the search process, $\lceil \log_2 i \rceil$ integers in the interval $[1, i]$ must be raised to the k:th power. This is a polynomial-time task and the lemma follows. □

Lemma 5.7 Let $C > 0$ be a fixed integer. Then PL is PSPACE-hard to approximate within

$$\sqrt{\frac{2^{n - \sqrt[C]{n}}}{2^{\sqrt[C]{n}} + 2}}$$

where n is the number of atoms in the given instance. Furthermore, PL is PSPACE-hard to approximate with one-sided error within

$$\frac{2^{n - \sqrt[C]{n}}}{2^{\sqrt[C]{n}} + 2}$$

Proof: Define functions $a, b : \mathbb{N} \to (1, \infty)$ such that $a(n) = 2^{\sqrt[C]{n}} + 2$ and $b(n) = 2^{n - \sqrt[C]{n}}$. Furthermore, let $\ell : \mathbb{N} \to \mathbb{N}$ be such that $\ell(n) = n^C$.

Arbitrarily choose a CPS instance $\Pi = \langle \mathcal{P}_\Pi, \mathcal{O}_\Pi, \mathcal{I}_\Pi, \mathcal{G}_\Pi \rangle$. Let

$$\Sigma = \Pi_{\mathrm{PL}}(\Pi, |\mathcal{P}_\Pi|^C - |\mathcal{P}_\Pi|) = \langle \mathcal{P}_\Sigma, \mathcal{O}_\Sigma, \mathcal{I}_\Sigma, \mathcal{G}_\Sigma \rangle.$$

Obviously, this is a polynomial-time transformation since C is fixed. Observe that, by the construction of Σ, $|\mathcal{P}_\Sigma| = |\mathcal{P}_\Pi|^C = \ell(|\mathcal{P}_\Pi|)$.

Observe that if $j \in \mathit{Img}(\ell)$ then $j = k^C$ for some $k \in \mathbb{N}$. By Lemma 5.6, we can in polynomial time compute k which equals $\sqrt[C]{j}$. For arbitrary $i \in \mathbb{N}$, we can consequently decide (in polynomial time) whether $i > a(j)$ or $i > a(j) \cdot b(j)$.

If Π is solvable, then Σ has a minimal solution ω such that $|\omega| \leq 2^{|\mathcal{P}_\Pi|}$ by Lemma 5.5. Consequently,

$$\mathrm{PL}(\Sigma) \leq 1 + 2^{|\mathcal{P}_\Pi|} < a(|\mathcal{P}_\Sigma|)$$

Assume to the contrary that Π is not solvable. Then, by Lemma 5.5, Σ has a minimal solution ω of length $2^{|\mathcal{P}_\Sigma| - \sqrt[C]{|\mathcal{P}_\Sigma|}}$. It follows that

$$\mathrm{PL}(\Sigma) = 1 + 2^{|\mathcal{P}_\Sigma| - \sqrt[C]{|\mathcal{P}_\Sigma|}} > 2^{|\mathcal{P}_\Sigma| - \sqrt[C]{|\mathcal{P}_\Sigma|}} = b(|\mathcal{P}_\Sigma|)$$

Since PE is PSPACE-hard, it follows from Lemma 3.5 that PL is PSPACE-hard to approximate within $\sqrt{\frac{2^{n-\sqrt[C]{n}}}{2^{\sqrt[C]{n}}+2}}$ where n is the number of atoms in the given instance. Furthermore, PL is PSPACE-hard to approximate with one-sided error within $\frac{2^{n-\sqrt[C]{n}}}{2^{\sqrt[C]{n}}+2}$. □

Theorem 5.8 Arbitrarily choose $0 < \varepsilon < 1$. Approximating PL within $\sqrt{2^{n(1-n^{-\varepsilon})}}$ is PSPACE-hard where n is the number of atoms in the instance. Approximating PL with one-sided error within $2^{n(1-n^{-\varepsilon})}$ is PSPACE-hard.

Proof: Arbitrarily choose C such that $(C-2)/C > \varepsilon$ and assume, without loss of generality, that n is so large that

$$2^{\sqrt[C+1]{n}} + 2 < 2^{\sqrt[C]{n}}$$

and $2 < n^{1/C}$.

By the previous lemma, PL is PSPACE-hard to approximate within

$$\sqrt{\frac{2^{n-\sqrt[C+1]{n}}}{2^{\sqrt[C+1]{n}}+2}} > \sqrt{\frac{2^{n-\sqrt[C+1]{n}}}{2^{\sqrt[C]{n}}}} = \sqrt{2^{n-\sqrt[C+1]{n}-\sqrt[C]{n}}} \geq \sqrt{2^{n-2\sqrt[C]{n}}}.$$

Now, note that

$$n - 2\sqrt[C]{n} = n(1-2n^{-\frac{C-1}{C}}) > n(1-n^{1/C}n^{-\frac{C-1}{C}})$$

and

$$n(1-n^{1/C}n^{-\frac{C-1}{C}}) = n(1-n^{-\frac{C-2}{C}}) > n(1-n^{-\varepsilon})$$

which implies that PL is PSPACE-hard to approximate within $\sqrt{2^{n(1-n^{-\varepsilon})}}$. Showing the bound for approximations with one-sided error is analogous.
□

Note that the function $f(\Pi) = 2^{|\Pi|}$ approximates PL with one-sided error within 2^n in polynomial time.

6 Conclusions and Future Work

We have presented tight upper and lower bounds on two optimization versions of propositional planning. The results indicate that both problems are extremely hard to approximate since the upper bounds are obtained by trivial algorithms.

This work can be extended in a number of different ways. We present a selection of possible directions below.

- The hardness of approximating PL and NSG may be used for showing the hardness of approximating other PSPACE-hard problems by reductions. Plausible candidates are certain problems on Turing machines which show similarities with propositional planning.

- There are other optimization versions of propositional planning which deserve study. One example is the *planning diameter* problem [18] where the objective is to find the diameter of the state transition graph induced by a propositional planning problem.
- The direct-reduction techniques used in this article have provided tight lower bounds. However, nonapproximability results based on PCDS's often fail to do so. One immediate question is whether the results in, for instance, Condon *et al.* [8] or Condon *et al.* [9] can be strengthened by using direct-reduction arguments. Another question is if their proofs can be simplified by employing such techniques. This last question has an affirmative answer in at least one case. Condon *et al.* [8] have provided nonapproximability results for MAX-GGEOG (by using PCDS's) and they report that Bodlaender has given a simpler proof of this result by not using PCDS's.

Acknowledgements

Thanks to Christer Bäckström and Patrik Haslum for helpful comments. This work has been sponsored by the *Swedish Research Council for the Engineering Sciences* (TFR) under grants 96-737 and 97-301.

References

[1] J. Allen, J. Hendler, and A. Tate, editors, *Readings in Planning* (Morgan Kaufmann, San Mateo, 1990).

[2] C. Bäckström, Computational aspects of reordering plans, *J. Artif. Intell. Res.*, **9** (1998) 99–137.

[3] C. Bäckström, Expressive equivalence of planning formalisms, *Artif. Intell.*, **76**(1–2) (1995) 17–34.

[4] C. Bäckström and P. Jonsson, Planning with abstraction hierarchies can be exponentially less efficient, in: *Proc. IJCAI'95*, (Morgan Kaufmann, 1995) 1599–1604.

[5] C. Bäckström and I. Klein, Planning in polynomial time: The SAS-PUBS class, *Comput. Intell.* **7**(3) (1991) 181–197.

[6] T. Bylander, The computational complexity of propositional STRIPS planning, *Artif. Intell.* **69** (1994) 165–204.

[7] D. Chapman, Planning for conjunctive goals, *Artif. Intell.* **32** (1987) 333–377.

[8] A. Condon, J. Feigenbaum, C. Lund, and P. W. Shor, Probabilistically checkable debate systems and nonapproximability of PSPACE-hard functions, *Chicago Journal of Theoretical Computer Science* (1995) article 4.

[9] A. Condon, J. Feigenbaum, C. Lund, and P. W. Shor, Random debaters and the hardness of approximating stochastic functions, *SIAM J. Comput.* **26**(2) (1997) 369–400.

[10] P. Crescenzi and V. Kann, A compendium of NP optimization problems, Technical Report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Universitá di Roma "La Sapienza", 1995.

[11] P. Crescenzi and A. Panconesi, Completeness in approximation classes, *Inf. and Comput.*, **93**(2) (1991) 241–262.

[12] P. Crescenzi and L. Trevisan, On approximation scheme preserving reducibility and its applications, in: *Proc. STACS'94*, Lecture Notes in Computer Science, Vol. 880 (Springer, Berlin, 1994) 330–341.

[13] K. Erol, D. S. Nau, and V. S. Subrahmanian, On the complexity of domain-independent planning, in: *Proc. AAAI'92* (AAAI Press/MIT Press, 1992) 381–386.

[14] R. E. Fikes and N. J. Nilsson, STRIPS: A new approach to the application of theorem proving to problem solving, *Artif. Intell.* **2** (1971) 189–208.

[15] M. L. Ginsberg, Approximate planning, *Artif. Intell.* **76** (1–2) (1995) 89–123.

[16] H. Hunt III, M. Marathe, and R. Stearns, Generalized CNF satisfiability problems and non-efficient approximability, *in: Proc. 9th Conference on Structure in Complexity Theory* (IEEE Computer Society Press, 1994) 356–366.

[17] P. Jonsson, A nonapproximability result for finite function generation, *Inf. Process. Lett.* **63** (1997) 143–145.

[18] R. E. Korf, Planning as search: A quantitative approach, *Artif. Intell.* **33** (1987) 65–88.

[19] M. Marathe, H. Hunt III and S. Ravi, The complexity of approximating PSPACE-complete problems for hierarchical specifications, *Nordic Journal of Computing* **1** (1994) 275–316.

[20] J. Orlin, The complexity of dynamic languages and dynamic optimization problems, *in: Proc. STOC'81* (ACM Press, New York, 1981) 218–227.

[21] W. J. Savitch, Relationships between nondeterministic and deterministic tape complexities, *J. Comput. Syst. Sci.* **4**(2) (1970) 177–192.

[22] B. Selman, Near-optimal plans, tractability, and reactivity, *in: Proc. KR'94* (Morgan Kaufmann, 1994) 521–529.