A Complete Axiomatization for Blocks World

Stephen A. Cook and Yongmei Liu

Department of Computer Science

University of Toronto

Toronto, ON, Canada M5S 3G4

{sacook,yliu}@cs.toronto.edu

Contact author: Yongmei Liu

Abstract

Blocks World (BW) has been one of the most popular model domains in AI history. However, there has not been serious work on axiomatizing the state constraints of BW and giving justification for its soundness and completeness. In this paper, we model a state of BW by a finite collection of finite chains, and call the theory of all these structures BW theory. We present seven simple axioms and prove that their consequences are precisely BW theory, using Ehrenfeucht-Fraïssé games. We give a simple decision procedure for the theory which can be implemented in exponential space, and prove that every decision procedure (even if nondeterministic) for the theory must take at least exponential time. We also give a characterization of all nonstandard models for the theory. Finally, we present an expansion of BW theory and show that it admits elimination of quantifiers. As a result, we are able to characterize all definable predicates in BW theory, and give simple examples of undefinable predicates.

Keywords: Blocks World, finite axiomatization, Ehrenfeucht-Fraïssé games, complexity of logical theories, definable predicates

A Complete Axiomatization for Blocks World

Stephen A. Cook and Yongmei Liu

Department of Computer Science

University of Toronto, Toronto, ON, Canada M5S 3G4

{sacook,yliu}@cs.toronto.edu

Abstract

Blocks World (BW) has been one of the most popular model domains in AI history. In this paper, we model a state of BW by a finite collection of finite chains, and call the theory of all these structures BW theory. We present seven simple axioms and prove that their consequences are precisely BW theory, using Ehrenfeucht-Fraïssé games. We present a simple decision procedure for the theory which can be implemented in exponential space, and prove that every decision procedure (even if nondeterministic) for the theory must take at least exponential time. We characterize both the definable predicates in the theory and its nonstandard models.

A preliminary version of this paper appeared in the Seventh International Symposium on Artificial Intelligence and Mathematics, Fort Lauderdale, Florida, January 2-4, 2002.

1 Introduction

In the history of artificial intelligence, Blocks World (BW) has been one of the most popular model domains. This domain consists of a set of blocks of various shapes, sizes and colors sitting on a table. A robot can pick up a block and move it to another position, either onto the table or on top of some other block. A simple and well-known version of BW, called Elementary BW, consists of cubic blocks of equal size. The use of BW dates back to 1970s when Winograd first used BW for his natural language understanding program [12], and then Waltz, Winston, etc. used BW for studies of computer vision [13].

BW is most extensively used in studies of planning. Roughly, the planning task is to find a sequence of actions which transforms a given initial state into a given goal state. Gupta and Nau [5] showed that optimal BW planning is NP-hard. Slaney and Thiébaux [10] presented linear time algorithms for near-optimal BW planning within a ratio of 2. Up to late 1990s, BW has been used as a benchmark for domain-independent planning techniques and systems. In a recent paper [11], Slaney and Thiébaux [10] further investigated BW planning. In particular, they presented methods for generating random problems for systematic experimentation.

BW is also used in studies of action theories, a subarea of AI concerned with representing and reasoning about actions and their effects. In recent years, founded on action theories, high-level programming languages for robots have been developed. An example is situation calculus-based GOLOG [9], which provides a way of defining complex actions in terms of primitive actions, by using programming constructs such as sequences, conditionals, loops and recursive procedures. In [7], Liu used BW in studies of verification of robot programs written in GOLOG. This research suggested the need for a formal theory of BW.

Such a theory would address two issues relevant to program verification in [7]. First, it would be useful to have a complete axiomatization of the state constraints of BW. By state constraints of BW, we mean those properties which hold in every state of BW. For example, "no block can be above itself" is such a property. Without such an axiomatization, it is even impossible to prove some simple properties of BW programs. Despite the popularity of BW, to the best of our knowledge, there has not been serious work on axiomatizing the state constraints of BW and giving justification for its soundness and completeness.

The second issue is the expressiveness of the BW language: What predicates are definable in this language? This resolves the question of what pre- and post-conditions of BW programs can be expressed by the BW language. For example, we can write a BW program which makes two towers of the same height, provided there is a single tower in the beginning and its height is even. Our intuition is that the pre- and post-conditions of this program can't be expressed in the BW language. (We justify this intuition below by Corollary 4.6.) Besides, this issue is related to the completeness issue of proof systems for verification of BW programs. In [2], Cook proved relative completeness of Hoare Logic for sufficiently expressive languages, that is, those languages L such that for any L-formula ϕ and any program P, there is an L-formula ψ expressing the postcondition corresponding to ϕ and P.

In this paper, we address these two issues. We model a state of Elementary BW by a finite collection of finite chains, and call the theory of all these structures BW theory. We present a finite axiomatization of BW theory, analyze the computational complexity of the theory, and characterize the definable predicates in the theory.

In Section 2, we introduce the syntax and semantics of BW theory. In Section 3, we give a set of axioms and prove by a game-theoretic argument that their consequences are precisely BW theory. We give a simple decision procedure for BW theory requiring exponential space, and prove that every decision procedure for BW theory requires at least exponential time, even if it is nondeterministic. We also give a characterization of all nonstandard models for the theory. In Section 4, we present an expansion of BW theory and show that it admits elimination of quantifiers. As a result, we are able to characterize all definable predicates in BW theory, and give simple examples of undefinable predicates.

2 Blocks World Theory

The notion of linear order (or chain) will play an important role in this paper. Since there is variation in the usage of this term, we first clarify the usage in this paper. By linear order (or chain), we mean a structure $(A, <)$ where A is a nonempty set and $<$ is a binary relation on A which is irreflexive, transitive and connected, i.e., for any $a, b \in A$, exactly one of $a < b$, $a = b$ and $a > b$ is true.

Definition 2.1 *Blocks World is a theory of the first-order predicate calculus with equality. The language of Blocks World is* $\mathcal{L}_{bw} = \{above, =\}$*, where* $above$ *is a binary predicate symbol. We say that a structure* $\mathcal{A}$ *for* $\mathcal{L}_{bw}$ *is* *a Blocks World model (BW model) if it is a finite disjoint union of finite chains, where* $above(x, y)$ *is intended to mean* $y < x$.

We define "on", "ontable" and "clear" as abbreviations as follows:

$on(x, y) \stackrel{def}{=} above(x, y) \wedge \neg(\exists z)(above(x, z) \wedge above(z, y))$;

$ontable(x) \stackrel{def}{=} \neg(\exists y)above(x, y)$;

$clear(x) \stackrel{def}{=} \neg(\exists y)above(y, x)$.

Intuitively, x is on y if x is the least element above y.

Definition 2.2 *We use* $Th(bw)$ *to denote the theory of all BW models, that is, the set of all* $\mathcal{L}_{bw}$*-sentences true in every BW model. We call* $Th(bw)$ *Blocks World theory.*

Of course, Blocks World theory is incomplete in the technical sense. Let $\phi = (\forall x, y)[ontable(x) \wedge ontable(y) \rightarrow x = y]$. Then neither ϕ nor $\neg\phi$ belongs to it.

3 An Axiomatization for Blocks World Theory

In this section, we give a set of axioms and prove by a game-theoretic argument that their consequences are precisely BW theory. We give a simple decision procedure for BW theory requiring exponential space, and prove that any decision procedure (even if nondeterministic) requires at least exponential time. Finally, we give a characterization of all nonstandard models for the theory.

Let $\mathcal{A}_{bw}$ be the set of the following axioms. In each axiom, any free variables are implicitly universally quantified. To reduce parentheses, we assume that $\rightarrow$ and $\leftrightarrow$ bind with lowest precedence.

(1). $\neg above(x, x)$,

(2). $above(x, y) \wedge above(y, z) \rightarrow above(x, z)$,

(3). $above(x, y) \wedge above(x, z) \rightarrow y = z \vee above(y, z) \vee above(z, y)$,

(4). $above(y, x) \wedge above(z, x) \rightarrow y = z \vee above(y, z) \vee above(z, y)$,

(5). $ontable(x) \vee (\exists y)(above(x, y) \wedge ontable(y))$,

(6). $clear(x) \vee (\exists y)(above(y, x) \wedge clear(y))$,

(7). $above(x, y) \rightarrow (\exists z)on(x, z) \wedge (\exists w)on(w, y)$.

The axioms $\mathcal{A}_{bw}$ ensure that every model of it has the following properties. The elements of the universe are partitioned into disjoint sets, which we shall call "towers", by the comparability relation. The elements in each tower are linearly ordered by the "above" relation, but elements in different towers are incomparable. Every tower has a top element and a bottom element. Every element that is not a top element has something on it, and every element that is not a bottom element is on something.

Clearly, every BW model is a model of $\mathcal{A}_{bw}$. Thus $Cn\mathcal{A}_{bw} \subseteq Th(bw)$, where $Cn\mathcal{A}_{bw}$ denotes the set of consequences of $\mathcal{A}_{bw}$. It is not obvious that equality holds. We will prove this by a game-theoretic argument. So we begin with an introduction of Ehrenfeucht-Fraïssé games.

3.1 Ehrenfeucht-Fraïssé Games

The following material is adapted from Immerman [6].

Definition 3.1 *Let $\mathcal{A}$ and $\mathcal{B}$ be structures of the same vocabulary, and let $k \in \mathbb{N}$. The k-round first-order Ehrenfeucht-Fraïssé game $G_k(\mathcal{A}, \mathcal{B})$ is played by two players called the spoiler and the duplicator. There are k rounds of moves. In the ith round, the spoiler first selects an element in one of the two structures, then the duplicator selects an element in the other structure. If the vocabulary contains constant symbols $c_1, \ldots, c_m$, then let p_i (q_i respectively) denote the interpretation of c_i in $\mathcal{A}$ ($\mathcal{B}$ respectively), for $1 \leq i \leq m$. For $1 \leq i \leq k$, let p_{m+i} (q_{m+i} respectively) denote the element selected in $\mathcal{A}$ ($\mathcal{B}$ respectively) in the ith round. The duplicator wins if the substructure of $\mathcal{A}$ induced by $p_1, \ldots, p_{m+k}$ is isomorphic to the substructure of $\mathcal{B}$ induced by $q_1, \ldots, q_{m+k}$ under the function that maps p_i onto q_i for $1 \leq i \leq m + k$. Otherwise, the spoiler wins. We say that the spoiler or the duplicator has a winning strategy if he can guarantee that he will win, no matter how the other player plays. We write $\mathcal{A} \sim_k \mathcal{B}$ if the duplicator has a winning strategy.*

The quantifier rank $qr(\varphi)$ of a formula φ is the depth of nesting of quantifiers in φ. Let $\mathcal{A}$ and $\mathcal{B}$ be structures of the same vocabulary, and let $k \in \mathbb{N}$. We say that $\mathcal{A}$ and $\mathcal{B}$ are k-equivalent, written $\mathcal{A} \equiv_k \mathcal{B}$, if they agree on all first-order sentences of quantifier rank up to k.

The following is the fundamental result of Ehrenfeucht-Fraïssé games. It holds for both finite and infinite structures.

Proposition 3.2 *Let $\mathcal{A}$ and $\mathcal{B}$ be structures of the same finite vocabulary without function symbols, and let $k \in \mathbb{N}$. Then $\mathcal{A} \sim_k \mathcal{B}$ iff $\mathcal{A} \equiv_k \mathcal{B}$.*

The following result is proved in [6]. For $n \in \mathbb{N}$, let L_n denote a linear order on n elements.

Proposition 3.3 *Let $k \in \mathbb{N}$, and let $n = 2^{k+1} + 1$. Then $L_n \sim_k L_{n+1}$.*

Here we extend the above result as follows.

Definition 3.4 *We say that a chain L is a tower if*

(1). L has least and greatest elements;

(2). every element that is not a greatest element has a successor;

(3). every element that is not a least element has a predecessor.

Obviously, every finite chain is a tower. Note that an infinite discrete chain with least and greatest elements is not necessarily a tower, since it may not satisfy (3) in the above definition. Let $n \in \mathbb{N} \cup \{\infty\}$, where ∞ denotes infinity (of any cardinality). We use P_n to denote a tower on n elements.

Proposition 3.5 *Let $k \in \mathbb{N}$. Then for any $n > 2^k$, including $n = \infty$, $P_n \sim_k P_{2^k+1}$.*

Proof: The proof is essentially the same as that of Proposition 3.3.

Let $i, j, d \in \mathbb{N}$. We use $i =_d j$ to mean that $i = j \vee i \geq d \wedge j \geq d$. We expand the vocabulary to contain constant symbols 0 and max, and their interpretations in P_n ($n \in \mathbb{N} \cup \{\infty\}$) are the least and greatest elements of P_n, respectively. We use $<$ to denote the ordering on P_n. Let $a, b \in P_n$. We use $dist(a,b)$ to denote the distance between a and b, which could be ∞. We say that a is to the left of b if $a < b$.

The duplicator's winning strategy in $G_k(P_n, P_{2^k+1})$ is to maintain the following invariant: After the mth move, and for all $1 \leq i, j \leq m+2$,

$$dist(p_i, p_j) =_{2^{k-m}} dist(q_i, q_j) \text{ and } p_i < p_j \text{ iff } q_i < q_j. \tag{1}$$

Note that when $m = k$, (1) implies that the duplicator wins the game.

We prove that (1) holds by induction on m. Basis: $m = 0$. (1) holds since $dist(0^{P_n}, max^{P_n}) \geq 2^k$, for $n > 2^k$. Induction step: Assume that (1) holds for m. Suppose that the spoiler selects p_{m+3}. Let p_i and p_j be the closest to the left and right of p_{m+3} among $p_1, \ldots, p_{m+2}$. By induction hypothesis, $dist(p_i, p_j) =_{2^{k-m}} dist(q_i, q_j)$. Assume without loss of generality that p_i is the closer of p_i and p_j to p_{m+3} or that they are equidistant. The duplicator selects q_{m+3} to the right of q_i so that $dist(q_i, q_{m+3}) = \min\{dist(p_i, p_{m+3}), \lfloor dist(q_i, q_j)/2 \rfloor\}$. It follows that $dist(p_i, p_{m+3}) =_{2^{k-m-1}} dist(q_i, q_{m+3})$ and $dist(p_{m+3}, p_j) =_{2^{k-m-1}} dist(q_{m+3}, q_j)$. So (1) holds for $m+1$. The case that the spoiler selects q_{m+3} is similar.

Therefore the duplicator wins the game, and hence $P_n \sim_k P_{2^k+1}$. ■

3.2 Adequacy of $\mathcal{A}_{bw}$

We will use $\mathcal{A}_{bw}$-models to mean models of $\mathcal{A}_{bw}$. We first analyze properties of $\mathcal{A}_{bw}$-models. Let M be an $\mathcal{A}_{bw}$-model. We use D_M to denote the domain of M, and we use $above^M$ to denote the interpretation of *above* in M.

Proposition 3.6 *Each $\mathcal{A}_{bw}$-model is a disjoint union of towers.*

Proof: Let M be an $\mathcal{A}_{bw}$-model. Define a relation $\approx_M$ on D_M as follows: for any $a, b \in D_M$, $a \approx_M b$ iff $a = b$ or $\langle a, b\rangle \in above^M$ or $\langle b, a\rangle \in above^M$. By Axioms (2), (3) and (4), $\approx_M$ is an equivalence relation. Also by definition of $\approx_M$, no two equivalence classes of $\approx_M$ are connected by $above^M$. By Axioms (1) and (2), each equivalence class of $\approx_M$ is a chain; besides, it is a tower by Axioms (5), (6) and (7). ■

Thus an $\mathcal{A}_{bw}$-model is a BW model iff it is finite.

Let M be an $\mathcal{A}_{bw}$-model. The height of a tower is the number of elements in the tower. Let e_1 and e_2 be elements of the same tower. We use $dist(e_1, e_2)$ to denote the distance between e_1 and e_2. Let e be an element. The height of e, written $height(e)$, is one plus the number of elements below e. The depth of e, written $depth(e)$, is one plus the number of elements above e. Note that both height of towers and distance between elements could be ∞.

Lemma 3.7 *For any $k \in \mathbb{N}^+$ and $\mathcal{A}_{bw}$-model M, there exists a BW model M' consisting of at most $k \cdot (2^k + 1)$ towers, each of height at most $2^k + 1$, such that $M \equiv_k M'$.*

Proof: We prove this in two steps.

Step 1. Let M^* be obtained from M by replacing each tower of height $> 2^k + 1$ by a tower of height $2^k + 1$. The duplicator's winning strategy in $G_k(M, M^*)$ is as follows. Copy the moves on all towers of height $\leq 2^k+1$, and on other towers use the winning strategy in $G_k(P_n, P_{2^k+1})$ (see Proposition 3.5), where $n > 2^k + 1$. Thus $M \sim_k M^*$.

Step 2. Let M' be obtained from M^* as follows: for $1 \leq h \leq 2^k+1$, if there are more than k towers of height h, keep only k of them. The duplicator's winning strategy in $G_k(M^*, M')$ is as follows. If the spoiler selects a new tower, then the duplicator selects a new tower of the same height, otherwise the duplicator uses the winning strategy in $G_k(P_n, P_n)$, where $n \in \mathbb{N}$. Thus $M^* \sim_k M'$.

Clearly, the height of each tower of M' is at most $2^k + 1$ and the number of towers of M' with the same height is at most k. By transitivity of $\sim_k$, $M \sim_k M'$, and hence $M \equiv_k M'$. ∎

Theorem 3.8 $Cn\mathcal{A}_{bw} = Th(bw)$.

Proof: Now we prove that $Th(bw) \subseteq Cn\mathcal{A}_{bw}$, that is, for any $\phi \in Th(bw)$ and any $\mathcal{A}_{bw}$-model M, $M \models \phi$. Let $k = qr(\phi)$. Then $k > 0$. By Lemma 3.7, there exists a BW model M' such that $M \equiv_k M'$. Since $M' \models \phi$, $M \models \phi$. ∎

3.3 Decidability of Blocks World Theory

A by-product of the completeness proof of $\mathcal{A}_{bw}$ is that $Th(bw)$ is decidable.

Theorem 3.9 *$Th(bw)$ is decidable by a decision procedure requiring at most space $2^{O(n)}$ and time $2^{2^{O(n)}}$.*

Proof: The following is a decision procedure for $Th(bw)$. Given an arbitrary $\mathcal{L}_{bw}$-sentence ϕ, we check whether $\phi \in Th(bw)$ by (using lemma 3.7) checking whether ϕ is true in every BW model with at most $2^{O(k)}$ elements, where $k = qr(\phi)$. The space required to specify and check each model is $2^{O(n)}$. The run time of any deterministic Turing machine is at most exponential in its space. ∎

Theorem 3.10 *If M is a Turing machine with runtime bounded by $T(n)$ which accepts precisely the set $Th(bw)$, then there exists $\epsilon > 0$ such that $T(n) > 2^{\epsilon n}$ for infinitely many n. The same holds if M is a nondeterministic Turing machine.*

The proof uses methods [3, 4] established in the 1970's for proving lower bounds on the complexity of decidable theories. We begin by reviewing some definitions of complexity classes and reducibilities. To simplify this discussion, we fix a finite alphabet $\Sigma \supseteq \{0, 1\}$, and assume that all sentences over $\mathcal{L}_{bw}$ are coded as strings over Σ.

Definition 3.11 **NE** *(nondeterministic exponential time) is the set of all languages $L \subseteq \Sigma^*$ accepted in time $2^{O(n)}$ by some nondeterministic Turing machine.* **coNE** $= \{\bar{L} \mid L \in \mathbf{NE}\}$*, where $\bar{L} = \Sigma^* - L$.*

Definition 3.12 *Let $L_1, L_2 \subseteq \Sigma^*$. Then $L_1 \leq_{p\ell} L_2$ (L_1 is polynomial time linear expansion reducible to L_2) if there is a polynomial time computable function $f : \Sigma^* \to \Sigma^*$ such that for all $w \in \Sigma^*$*

$$w \in L_1 \iff f(w) \in L_2, \text{ and}$$

$$|f(w)| = O(|w|)$$

We say that a language L_0 is $\leq_{p\ell}$-hard for a complexity class $\mathcal{C}$ iff $L \leq_{p\ell} L_0$ for all $L \in \mathcal{C}$.

Theorem 3.13 *$Th(bw)$ is $\leq_{p\ell}$-hard for* **coNE**.

Proof of Theorem 3.10 from 3.13: Choose M_0 to be a nondeterministic universal Turing machine which behaves as follows. We use $\langle M \rangle$ to denote an appropriate string ending in 1 encoding a nondeterministic Turing machine M. Then we design M_0 so that on an input $\langle M\rangle 0^k$ of length n, M_0 runs for at most 2^n steps while simulating M for $2^{n/3}$ steps on the same input. Then for every nondeterministic Turing machine M and for all sufficiently large k, machine M_0 accepts input $\langle M\rangle 0^k$ within time 2^n iff M accepts $\langle M\rangle 0^k$ within time $2^{n/3}$, where $n = |\langle M\rangle 0^k|$.

Let $L_0 = L(M_0)$ be the language accepted by M_0. Then $L_0 \in \mathbf{NE}$, so $\bar{L}_0 \in \mathbf{coNE}$, so by the preceding theorem there is a polynomial time function $f : \Sigma^* \to \Sigma^*$ such that for all nonempty $w \in \Sigma^*$

$$M_0 \text{ accepts } w \Longleftrightarrow f(w) \notin Th(bw), \text{ and} \tag{2}$$

$$|f(w)| \leq c_0|w| \tag{3}$$

for some constant $c_0 > 0$.

Now suppose M_1 is a nondeterministic Turing machine which accepts precisely $Th(bw)$ and suppose contrary to Theorem 3.10 that M_1 runs in time $T(n) \leq 2^{n/(6c_0)}$ for all sufficiently large input lengths n. Design a machine M_2 which on input $\langle M\rangle 0^k$ applies the transformation f from (2) to obtain a formula $\phi = f(\langle M\rangle 0^k)$. Now M_2 runs M_1 on input ϕ, and M_2 accepts $\langle M\rangle 0^k$ iff M_1 accepts ϕ. Note that $|\phi| \leq c_0 n$ by (3), so the runtime of M_1 on input ϕ is at most $2^{c_0 n/(6c_0)} = 2^{n/6}$.

Thus for sufficiently large k, if $n = |\langle M\rangle 0^k|$ then M_2 accepts $\langle M\rangle 0^k$ within time $2^{n/3}$ iff M_1 accepts $f(\langle M\rangle 0^k)$ iff $f(\langle M\rangle 0^k) \in Th(bw)$ iff (by (2)) M_0 does not accept $\langle M\rangle 0^k$ iff M does not accept $\langle M\rangle 0^k$ within time $2^{n/3}$ (by the property of M_0). By taking $M = M_2$ we obtain a contradiction. ■

Proof of Theorem 3.13: Let $L \in$ **coNE**. Then $\bar{L}$ is accepted by some nondeterministic Turing machine M with run time at most $2^{c_1 n}$, for some integer $c_1 > 0$. For each input string $w \in \Sigma^*$ we will design a sentence $f(w)$ such that $|f(w)| = O(|w|)$ and

$$M \text{ accepts } w \iff f(w) \notin Th(bw)$$

We let $f(w) = \neg g(w)$, where the sentence $g(w)$ satisfies

$$M \text{ accepts } w \iff g(w) \text{ is satisfied by some BW model}$$

Thus it suffices to design the sentence $g(w)$ such that it is satisfied by some BW model iff M accepts w in a computation C with at most $2^{c_1 n}$ steps, where $n = |w|$. Each configuration I of the computation represents the current sequence of tape symbols, along with the current state and scanned square. We code I by a binary string of length 2^m, where $m = cn$, and c is an integer constant depending on M and c_1. The entire computation C is coded by a binary string X of length $L = 2^m(2^{c_1 n} + 1)$ which is the concatenation of $2^{c_1 n} + 1$ codes for the successive configurations in the computation. Let $X_i \in \{0, 1\}$ be the i-th bit of $X, 1 \leq i \leq L$. We say that a BW model B *represents* X iff for each i, $1 \leq i \leq L$, B has a tower of height i iff $X_i = 1$.

Our goal is to design $g(w)$ so that a BW model B satisfies $g(w)$ iff B represents a string X coding an accepting computation of M on input w of at most $2^{c_1 n}$ steps. We write $g(w) = F_1 \wedge F_2 \wedge F_3$, where sentence F_1 asserts that the first 2^m bits of X correctly represent the initial configuration of M on input w, F_2 asserts that for $0 \leq i < 2^{c_1 n}$, bits number $2^m(i+1)+1, ..., 2^m(i+2)$

code a possible successor configuration to the configuration coded by bits number $2^m \cdot i+1, ..., 2^m(i+1)$. Finally F_3 asserts that one of the configurations coded by X contains an accepting state.

Here are the basic formulas needed to construct F_1, F_2, F_3. We use $x < y$ for $above(y, x)$, $min(x)$ for $ontable(x)$, and $max(x)$ for $clear(x)$. Recall that if a and b are elements in the same tower of a BW model, then $dist(a, b)$ is the distance between a and b. Our first task is to design, for each $k \geq 0$, a formula $Dist_k(x, y)$ of length $O(k)$ asserting $x < y$ and $dist(x, y) = 2^k$. Thus $Dist_0(x, y) = on(y, x)$, and in general we want

$$Dist_{k+1}(x, y) \leftrightarrow \exists z(Dist_k(x, z) \wedge Dist_k(z, y))$$

However we cannot use the RHS for $Dist_{k+1}$ since it has two occurrences of $Dist_k$ and hence the formula would have length exponential in k. Instead we use a standard quantifier trick and define

$$Dist_{k+1}(x, y) = \exists z \forall u \forall v[((u = x \wedge v = z) \vee (u = z \wedge v = y)) \rightarrow Dist_k(u, v)]$$

Now the RHS has only one occurrence of $Dist_k$ and hence the number of symbols in $Dist_k$ grows linearly in k. Further the bound variable z on the RHS can also occur bound in $Dist_k$, and u, v can also occur bound in $Dist_{k-1}$, so five distinct bound variables u, v, x, y, z suffice in total for each $Dist_k$, although each is quantified many times. Thus the number of bits required to code $Dist_k$ is linear in k.

Next for each k we define a formula $EQ_k(x_1, y_1, x_2, y_2)$ with the intention

$$EQ_k(x_1, y_1, x_2, y_2) \leftrightarrow x_1 \leq y_1 \wedge x_2 \leq y_2 \wedge dist(x_1, y_1) = dist(x_2, y_2) \leq 2^k$$

Thus

$$EQ_0(x_1, y_1, x_2, y_2) = [(x_1 = y_1 \wedge x_2 = y_2) \vee (on(y_1, x_1) \wedge on(y_2, x_2))]$$

and in general

$$EQ_{k+1}(x_1, y_1, x_2, y_2) \leftrightarrow \exists z_1 \exists z_2 [EQ_k(x_1, z_1, x_2, z_2) \wedge EQ_k(z_1, y_1, z_2, y_2)]$$

Again we use the quantifier trick so that EQ_k has bit size $O(k)$.

Now we define a formula $EQH_k(x, y)$ to assert that elements x and y have the same height and that height is at most $2^k + 1$. Thus

$$EQH_k(x, y) = \exists x' \exists y' [min(x') \wedge min(y') \wedge EQ_k(x', x, y', y)]$$

Recall that a string X encoding an accepting computation has length $L = 2^{m+c_1 n} + 2^m$. We define a formula $H_L(x)$ to assert x has height $L+1$ as follows:

$$H_L(x) = \exists y \exists z (min(y) \wedge Dist_{m+c_1 n}(y, z) \wedge Dist_m(z, x))$$

Now we sketch the method for building the sentence F_2, which asserts that each successive step in the computation is correct. F_2 begins with the prefix $\exists x H_L(x)$ asserting that some element x has height $L+1$, and now for each $y < x$, y can be used as an index for the y-th bit of X. For example, to assert that $X_y = 1 \rightarrow X_{y+2^m} = 1$ we assert that if there is a tower of height y, then there is a tower of height $y + 2^m$; thus

$$\exists z (max(z) \wedge EQH_{m+c_1 n}(z, y)) \rightarrow$$
$$\exists z_1 \exists z_2 (max(z_2) \wedge EQH_{m+c_1 n}(z_1, y) \wedge Dist_m(z_1, z_2))$$

With proper coding of states and symbols, the sentence F_3 simply asserts that some constant length bit pattern (representing the accepting state) occurs in X, and this is easily done using the above tools.

It remains to design F_1, which asserts that the first 2^m bits of X are correct. The initial r bits $b_1...b_r$ code the initial state and the input string w, where $r = O(n)$ (recall $n = |w|$). The remaining $2^m - r$ bits for the first configuration code blanks, and we can assume these bits are all 0.

We explain how to design a sentence of size linear in r which asserts that bits $X_1...X_r$ coincide with $b_1...b_r$. For $k \leq r$, let A_k be the assertion that for $1 \leq i \leq k$, $b_i = 1$ iff there is a tower of height i. For $1 \leq k \leq r$ we define a formula $G_k(x,y)$ with the intention $G_k(x,y) \leftrightarrow height(x) = k \wedge height(y) \neq k \wedge A_{k-1}$. The recurrence is

$$G_1(x,y) = min(x) \wedge \neg min(y); \qquad G_{k+1}(x,y) \leftrightarrow C_1 \wedge C_2$$

where C_1 asserts $height(x) = k+1 \wedge height(y) \neq k+1 \wedge A_{k-1}$, and C_2 asserts $X_k = b_k$. Thus

$$C_1 \leftrightarrow \exists x' \exists y' [on(x,x') \wedge G_k(x',y') \wedge (min(y) \vee on(y,y'))]$$

There are two cases for C_2, depending on whether b_k is 0 or 1. We give the case $b_k = 0$, since this illustrates the use of the negative clause in the intended meaning of G_k. In this case, C_2 asserts that there is no tower of height k.

$$C_2 \leftrightarrow \exists x' \forall z [on(x,x') \wedge (max(z) \rightarrow G_k(x',z))]$$

Notice that G_k occurs only positively on the right hand side of the recurrence, so the quantifier trick can be applied to obtain a sentence $\exists x \exists y G_r(x,y)$ of size $O(n)$ which asserts that the first r bits of X are correct. ∎

3.4 Nonstandard Models of Blocks World Theory

A nonstandard model of $Th(bw)$ is any model of $Th(bw)$ which is not a BW model. Here we give a complete characterization of all these nonstandard models.

Since $Cn\mathcal{A}_{bw} = Th(bw)$, models of $Th(bw)$ are exactly the same as $\mathcal{A}_{bw}$-models. The set of towers of an $\mathcal{A}_{bw}$-model can have any cardinality.

A chain is a Z-chain if it is isomorphic to the chain $\ldots, -2, -1, 0, 1, 2, \ldots$. A chain is a Z^+-chain (Z^--chain respectively) if it is isomorphic to the chain $1, 2, \ldots$ ($\ldots, -2, -1$ respectively).

Definition 3.14 *We say that a chain is a "Z^+Z^--chain", if it is isomorphic to the concatenation of a Z^+-chain and a Z^--chain. We say that a chain is a "$Z^+Z^\lambda Z^-$-chain", if it is isomorphic to the concatenation of a Z^+-chain, any ordered set (countable or uncountable) of Z-chains, and a Z^--chain.*

Proposition 3.15 *Let L be a $Z^+Z^\lambda Z^-$-chain. Then L is a Z^+Z^--chain iff there is no element in L which separates L into two infinite parts.*

Proposition 3.16 *Any infinite tower of an $\mathcal{A}_{bw}$-model is a $Z^+Z^\lambda Z^-$-chain.*

Proof: Let T be an infinite tower of an $\mathcal{A}_{bw}$-model. We define a partition of T using the equivalence relation: two elements are equivalent iff there are finitely many elements between them. By Axiom (7), the equivalence class containing the least (greatest respectively) element of T is a Z^+-chain (Z^--chain respectively), and any other equivalence class is a Z-chain. ■

4 Definability in Blocks World Theory

In this section, we give a complete characterization of all the predicates definable in BW theory. We achieve this by the method of elimination of quantifiers. The theory $Th(bw)$ itself does not admit elimination of quantifiers. We overcome this by expanding BW models with additional relations.

Definition 4.1 *The expanded language of Blocks World is*

$$\begin{aligned}\mathcal{L}_{bw}^{+} = \mathcal{L}_{bw} \cup \{H_k \mid k \geq 2\} \cup \{D_k \mid k \geq 2\} \cup \{above_k \mid k \geq 2\} \\ \cup \{R_k^h \mid h, k \geq 1\} \cup \{T_k^h \mid h, k \geq 1\},\end{aligned}$$

where the new symbols are predicate symbols with the following arities: $H_k : 1$, $D_k : 1$, $above_k : 2$, $R_k^h : 0$, *and* $T_k^h : 0$.

Let M be a model of $\mathcal{A}_{bw}$. The intended expansion of M to $\mathcal{L}_{bw}^{+}$ is as follows:

(1). $H_k(x)$: the height of x is at least k;

(2). $D_k(x)$: the depth of x is at least k;

(3). $above_k(x, y)$: x is above y and their distance is at least k;

(4). R_k^h: there are at least k towers with height h;

(5). T_k^h: there are at least k towers with height at least h.

The intended expansion can be easily axiomatized using an explicit definition of each new predicate as follows. For each $h, k \geq 1$,

(1). $above_{k+1}(x, y) \leftrightarrow (\exists x_1 \ldots x_k)[on(x, x_1) \wedge \bigwedge_{1 \leq i < k} on(x_i, x_{i+1}) \wedge above(x_k, y)]$, (2). $H_{k+1}(x) \leftrightarrow (\exists y)(ontable(y) \wedge above_k(x, y))$, where $above_1$ is $above$,

(3). $D_{k+1}(x) \leftrightarrow (\exists y)(clear(y) \wedge above_k(y, x))$,

(4). $R_k^h \leftrightarrow (\exists x_1 \dots x_k)[\bigwedge_{1 \le i < j \le k} x_i \neq x_j \wedge \bigwedge_{1 \le i \le k} (ontable(x_i) \wedge D_h(x_i) \wedge \neg D_{h+1}(x_i))]$,

(5). $T_k^h \leftrightarrow (\exists x_1 \dots x_k)[\bigwedge_{1 \le i < j \le k} x_i \neq x_j \wedge \bigwedge_{1 \le i \le k} (ontable(x_i) \wedge D_h(x_i))]$.

We use $\mathcal{A}_{bw}^+$ to denote $\mathcal{A}_{bw}$ extended by the above axioms. It is easy to see that $\mathcal{A}_{bw}^+$ is a conservative extension of $\mathcal{A}_{bw}$, since each model of $\mathcal{A}_{bw}$ can be expanded to a model of $\mathcal{A}_{bw}^+$ by defining the new predicates using the new axioms above. We will prove that $Cn\mathcal{A}_{bw}^+$ admits elimination of quantifiers by using the following theorem from Marker *et al* [8].

Proposition 4.2 *Let $\mathcal{L}$ be a language containing at least one constant symbol. Let T be an $\mathcal{L}$ theory, and let $\phi(\vec{x})$ be an $\mathcal{L}$ formula with free variables $\vec{x}$. Then the following are equivalent:*

(1). There is a quantifier-free formula $\psi(\vec{x})$ such that $T \models (\forall \vec{x})(\phi(\vec{x}) \leftrightarrow \psi(\vec{x}))$;

(2). If M_1 and M_2 are models of T, and M is a common substructure of M_1 and M_2, then for any $\vec{a} \in D_M$, $M_1 \models \phi[\vec{a}]$ iff $M_2 \models \phi[\vec{a}]$.

It can be shown that the above theorem still holds when $\mathcal{L}$ does not contain any constant symbol, assuming that we introduce a propositional connective 1 for "true".

Theorem 4.3 *The theory $Cn\mathcal{A}_{bw}^+$ admits quantifier elimination.*

Proof: It suffices to show that Condition (2) of Proposition 4.2 holds for every formula of the form $(\exists y)\phi(\vec{x}, y)$ where $\phi(\vec{x}, y)$ is a conjunction of literals. (To see this, place the ϕ of Proposition 4.2 in prenex form, and eliminate the quantifiers one at a time, starting with the innermost quantifier. We may assume that this is $\exists y$ by considering the negation, if necessary. Place the quantifier-free part in disjunctive normal form, and distribute $\exists y$ over the $\vee$'s.) We may suppose that the variable y occurs in each literal. Let M_1 and M_2 be models of $\mathcal{A}^+_{bw}$, and let M be a common substructure of M_1 and M_2 such that $\vec{a} \in D_M$. Then the predicates R^h_k and T^h_k guarantee that M_1 and M_2 each have the same number of towers of each height (where ∞ is a possible number), and the other predicates guarantee that any tower containing any of the constants in $\vec{a}$ is the same in both models, with respect to finite and infinite distances between constants and the top and bottom of the tower.

Suppose $M_1 \models \phi[\vec{a}, b]$ for some $b \in D_{M_1}$. We must show that there exists $e \in D_{M_2}$ such that $M_2 \models \phi[\vec{a}, e]$. To prove $M_2 \models \phi[\vec{a}, e]$, it suffices to prove that for each atom α occurring in ϕ, $M_1 \models \alpha[\vec{a}, b]$ iff $M_2 \models \alpha[\vec{a}, e]$. Let $m = max\{k \mid H_k, D_k \text{ or } above_k \text{ occurs in } \phi\}$. We use $[b]$ to denote the tower of M_1 containing b. There are two cases.

Case 1: The tower $[b]$ contains no a_i. If $[b]$ is finite, let T be a tower of M_2 containing no a_i and with the same height as $[b]$, and let $e \in T$ such that $height(e) = height(b)$. Otherwise, let T be a tower of M_2 containing no a_i and with height at least $2m + 2$. If $height(b)$ is finite, let $e \in T$ such that $height(e) = min\{height(b), m + 1\}$, otherwise let $e \in T$ such that $depth(e) = min\{depth(b), m + 1\}$. Now if α is an atom from a binary predicate occurring in ϕ, then $M_1 \not\models \alpha[\vec{a}, b]$ and $M_2 \not\models \alpha[\vec{a}, e]$; if α is a unary atom occurring in ϕ, it is easy to check that $M_1 \models \alpha[\vec{a}, b]$ iff $M_2 \models \alpha[\vec{a}, e]$.

Case 2: The tower $[b]$ contains some a_i. Let T be the tower of M_2 which contains that a_i. Let $A_1 = \{a_i \mid b \text{ is above } a_i\}$, and let $A_2 = \{a_i \mid a_i = b \text{ or } a_i \text{ is above } b\}$. If A_1 is not empty, let b_1 and e_1 be the greatest element of A_1, otherwise let b_1 be the least element of $[b]$, and let e_1 be the least element of T. Similarly, if A_2 is not empty, let b_2 and e_2 be the least element of A_2, otherwise let b_2 be the greatest element of $[b]$, and let e_2 be the greatest element of T. We use $[e_1, e_2]$ to denote the set of elements between e_1 and e_2, including e_1 and e_2. Now if $dist(b1, b)$ is finite, let $e \in [e1, e2]$ such that $dist(e1, e) = dist(b1, b)$; otherwise if $dist(b, b2)$ is finite, let $e \in [e1, e2]$ such that $dist(e, e_2) = dist(b, b_2)$, otherwise let $e \in [e1, e2]$ such that $dist(e, e_2) = m + 1$. It is straightforward to check that for each atom α occurring in ϕ, $M_1 \models \alpha[\vec{a}, b]$ iff $M_2 \models \alpha[\vec{a}, e]$. ■

We now start to discuss definability of predicates in BW theory. Intuitively, we feel that the predicate "the height of x is 11" is definable, but the predicate "x is in the center of its tower" is not definable. In what follows, we shall formalize this intuitive idea. We will use the following definition of definability from Chang and Keisler [1].

Definition 4.4 *Let $\mathcal{L}$ be a language, and let P be a new n-ary predicate symbol not in $\mathcal{L}$. Let $\Sigma(P)$ be a set of $\mathcal{L} \cup \{P\}$ sentences. We say that P is (explicitly) definable in $\Sigma(P)$ if there exists an $\mathcal{L}$-formula $\phi(\vec{x})$ such that $\Sigma(P) \models (\forall \vec{x})[P(\vec{x}) \leftrightarrow \phi(\vec{x})]$.*

Now let P be a new n-ary predicate symbol not in $\mathcal{L}_{bw}$. For every BW model M, we assume that there is an intended interpretation P^M of P in

M. We use $Th_{bw}(P)$ to denote the theory of all expanded BW models, that is, the set of all $\mathcal{L}_{bw} \cup \{P\}$ sentences true in every expanded BW model $\langle M, P^M \rangle$. We are concerned about whether P is definable in $Th_{bw}(P)$. By definition of $Th_{bw}(P)$, it follows that $Th_{bw}(P) \models (\forall \vec{x})[P(\vec{x}) \leftrightarrow \phi(\vec{x})]$ iff for every BW model M, P^M is equal to the relation that ϕ defines in M.

For example, let H be a new unary predicate symbol, and let $H^M = \{e \in D_M \mid \mathit{height}(e) = 11\}$ for any BW model M. Then H is definable in $Th_{bw}(H)$ by the formula $H_{11}(x) \wedge \neg H_{12}(x)$.

The following is a direct corollary of Theorem 4.3.

Corollary 4.5 *A new predicate symbol P is definable in $Th_{bw}(P)$ iff it is definable in $Th_{bw}(P) \cup \mathcal{A}^+_{bw}$ by a quantifier-free $\mathcal{L}^+_{bw}$-formula.*

Now we can use this corollary to prove the non-definability of some new predicate symbols.

Corollary 4.6 *Let P be a new 0-ary predicate symbol with intended interpretation P^M for each BW model M. If P is definable in $Th_{bw}(P)$, then there exists $B \in \mathbb{N}$ such that the following holds: for any M such that $P^M = true$ and for any tower of height $> B$ in M, let M' be obtained from M by increasing the height of that tower by some finite amount. Then $P^{M'} = true$.*

Proof: By Corollary 4.5, P is definable by a quantifier-free $\mathcal{L}^+_{bw}$-formula ϕ in negation normal form (i.e. $\neg$'s govern only atoms), where ϕ has no variables. We show by structural induction on ϕ that each such ϕ defines a predicate P satisfying the condition stated in the corollary. For the base case, ϕ has one of the four forms $R^h_k, \neg R^h_k, T^h_k, \neg T^h_k$, and in each case the condition is satisfied by taking $B = h$. For the induction step, ϕ is either $(\phi_1 \wedge \phi_2)$ or $(\phi_1 \vee \phi_2)$, where ϕ_i satisfies the condition for B_i, $i = 1, 2$. In either case ϕ satisfies the condition for $B = \max\{B_1, B_2\}$. ■

Example 4.1 *By the above corollary, it is easy to see that the predicates "there are two towers with the same height" and "there is a tower of even height" are not definable in BW theory.*

Corollary 4.7 *Let P be a new unary predicate symbol with an intended interpretation P^M for each BW model M. If P is definable in $Th_{bw}(P)$, then there exists $B \in \mathbb{N}$ such that for any BW model M, either the following holds when Q is P^M or it holds when Q is $(\neg P)^M$:*

$$\textit{for any } e \in Q,\ \textit{height}(e) < B \textit{ or } \textit{depth}(e) < B \tag{4}$$

Proof: By Corollary 4.5, $P(x)$ is definable by a quantifier-free $\mathcal{L}^+_{bw}$-formula $\phi(x)$. We prove the corollary by structural induction on $\phi(x)$. For the base case $\phi(x)$ is atomic, and must have one of the forms $H_k(x)$, $D_k(x)$, $above(x, x)$, $above_k(x, x)$, R^h_k, or T^h_k. For the first two cases, (4) holds with $B = k$ and $Q = \neg P^M$. For the other cases B is irrelevant, since either P^M or $\neg P^M$ is identically false, so (4) is vacuously true.

For the induction step, it suffices to consider the two cases $\phi(x)$ is $\neg\phi_1(x)$ and $\phi(x)$ is $(\phi_1(x) \wedge \phi_2(x))$, where $\phi_i(x)$ defines $P_i(x)$ and the induction hypothesis applies with bound B_i, $i = 1, 2$. The case $\neg\phi_1(x)$ is immediate. For the case $(\phi_1(x) \wedge \phi_2(x))$ either for some i (4) holds with $Q = P_i^M$ and $B = B_i$ (so (4) also holds when $Q = P^M$ and $B = B_i$), or (4) holds for both $i = 1$ and $i = 2$ when $Q = \neg P_i^M$ and $B = B_i$, so (4) also holds when $Q = \neg P^M$ and $B = \max\{B_1, B_2\}$. ∎

Example 4.2 *Let P be a new unary predicate symbol, and let $P^M = \{e \in D_M \mid height(e) = depth(e)\}$ for any BW model M. We will show that P is not definable in $Th_{bw}(P)$. Suppose to the contrary. Let B be the natural number in Corollary 4.7. Now let M_0 be a BW model with only one tower with height $2B + 3$. Then there exists $e_1 \in P^{M_0}$ such that $height(e_1) = depth(e_1) = B + 2$, and there exists $e_2 \in \neg P^{M_0}$ such that $height(e_2) = B + 3$ and $depth(e_2) = B + 1$. Thus we get a contradiction.*

Corollary 4.8 *Let P be a new 0-ary predicate symbol with intended interpretation P^M for each BW model M. If P is definable in $Th_{bw}(P)$, then there exists $B \in \mathbb{N}$ such that either the following holds for $Q = P^M$ or it holds for $Q = \neg P^M$: For any M such that $Q^M = true$, there exists h such that $1 \leq h \leq B$ and the number of towers of M with height h is less than B.*

Proof: The proof is similar to that of Corollary 4.7. The only atoms that we need consider are R_k^h and T_k^h, for which we can take $B = \max(h, k) + 1$. ∎

Example 4.3 *The predicate "the number of towers is even" is not definable in BW theory, since for each height h there are BW models consisting of an arbitrarily large even number of towers of height h, and also models consisting of an arbitrarily large odd number of towers of height h.*

5 Conclusions

Blocks World has been extensively studied in the planning literature [5, 10, 11]. Motivated by the use of BW in program verification [7], this paper presents a formal study of BW. We model a state of BW by a finite collection of finite chains, and call the theory of all these structures BW theory. We present a finite axiomatization of BW theory. We give a simple decision procedure for BW theory which can be implemented in exponential space, and prove that every decision procedure for the theory must take at least exponential time. Also, we give a complete characterization of all predicates definable in BW theory. It remains to be seen whether the proof techniques used in this paper can be applied to other similar problems in AI.

Acknowledgments

The authors are grateful to Leonid Libkin for reading an earlier version of this paper and pointing out that the use of games would substantially simplify our proofs, and we thank Charles Rackoff for help with the complexity lower bound proof (Theorem 3.10).

References

[1] C. C. Chang and H. J. Keisler. *Model Theory*, page 90. North Holland, 3rd edition, 1990.

[2] S. A. Cook. Soundness and Completeness of an Axiom System for Program Verification. *SIAM J. Comput.*, 7(1):70–90, 1978.

[3] J. Ferrante and C. Rackoff. *The Computational Complexity of Logical Theories. Lecture Notes in Mathematics* 718, Springer-Verlag, 1979.

[4] M. Fischer and M. Rabin. Super-Exponential Complexity of Presburger Arithmetic. *Proc. AMS Symp. on Complexity of Real Computational Processes*, vol. VII, 1974.

[5] N. Gupta and D. S. Nau. On the Complexity of Blocks-World Planning. *Artificial Intelligence*, 56(2-3):223–254, 1992.

[6] N. Immerman. *Descriptive Complexity.*

[7] Y. Liu. A Hoare-Style Proof System for Robot Programs. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-2002)*, pages 74–79.

[8] D. Marker, M. Messmer and A. Pillay. *Model Theory of Fields*, page 3. Springer-Verlag, 1996.

[9] R. Reiter. *Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems.* MIT Press, 2001.

[10] J. Slaney and S. Thiébaux. Linear Time Near-Optimal Planning in the Blocks World. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-1996)*, pages 1208–1214.

[11] J. Slaney and S. Thiébaux. Blocks World Revisited. *Artificial Intelligence*, 125:119–153, 2001.

[12] T. Winograd. *Understanding Natural Language.* Academic Press, New York, 1972.

[13] P. H. Winston. *The Psychology of Computer Vision.* McGraw-Hill, New York, 1975.