.

Games With Uniqueness Properties

by Shin Aida[1], Marcel Crasmaru[2], Kenneth Regan[3], and Osamu Watanabe[2]

1. Dept. of Knowledge-based Information Engineering,
Toyohashi University of Technology, Toyohashi, Japan 441-8580
2. Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology
Meguro-ku Ookayama, Tokyo 152-8552
3. Dept. of Computer Science, State University of New York at Buffalo
Buffalo, NY 14260-2000

Supported in part by JSPS/NSF cooperative research grant "Complexity Theory for Strategic Goals," INT 9726724, 1998–2001.

Abstract. For two-player games of perfect information such as Checker, Chess, and Go we introduce "uniqueness" properties. A game position has a uniqueness property if a winning strategy—should one exist—is forced to be unique. Depending on the way that winning strategy is forced, a uniqueness property is classified as (*bi-*)*weak*, (*bi-*)*strong*, or *global*. We prove that any reasonable two-player game G is extendable to a game G^* with the bi-strong uniqueness property, so that e.g., QBF remains PSPACE−complete under this reduction. For global uniqueness, we introduce a simple game over Boolean formulas with this property, and prove that any reasonable two-player game with the global uniqueness property is reducible to it. We show that the class of languages that reduce to globally-unique games equals Niedermeier and Rossmanith's unambiguous alternation class UAP [NR98], which is in an interesting region between FewP and SPP.

1. Introduction

Is it easier to analyze a position in a game to determine which player stands to win if one is told that the winning strategy for that player is unique? In general, in one-player games like "solitaire," where one player makes a series of moves and the resulting position is then adjudicated by an efficient referee, the answer is believed by many to be *yes*. A question of this type corresponds, in complexity theory, to the one asking whether problems in UP are easier than NP, or more precisely, whether the satisfiability of a given formula F is easier to solve under the promise that it has at most one satisfying assignment. For the case of NP problems, it is known [Val86] that the promise does not reduce the complexity of NP problems so much.

This paper analyzes theoretically whether the same intuition holds for two-player games. From a complexity theoretic point of view, our question is on the complexity of alternating computation, and it is to ask whether computation becomes easy if we may assume some sort of "uniqueness property" in the accepting computation.

Since the notion of unique winning strategy is not so clear for a two-player game or alternating computation we first explain uniqueness properties we will discuss in this paper by using some intuitive examples. In the game Chess, a problem of the "White to move and win" variety is called a *study*. For a study to be "publishable" it is generally an absolute requirement that the first winning move—called the "key"—be unique. Moreover, the study should have a "main line" in which the defender forces the attacker to find unique winning moves at every turn. The existence of a main line with unique play by the winner corresponds to our condition called *weak uniqueness*. More elegant and challenging are problems in which the defender has many options that force unique winning plays by the attacker. When *all* options by the defender force unique replies at every move, then the study position has the *strong uniqueness* property. A final element of study elegance puts uniqueness on the flip side of the coin. It is desirable that certain tempting mis-steps by the attacker—called "tries"—be punishable by unique defenses. When this applies not only to all poor first moves by the attacker but to every position reachable from the start, then we speak of the *global uniqueness* property.

If all positions of a game have this global uniqueness property, then we simply say that the game has the global uniqueness property. An example of such a game is the basic form of Nim in which each player may remove from 1 to m counters from a single pile initially with N counters, and the player taking the last counter wins. If N is not a multiple of $m + 1$, then the first player wins uniquely by removing $N \bmod (m + 1)$ counters. But if so, then the first player loses—note now that the opponent has a unique winning reply to every move. Similarly, a mistake by the first player in a winning position is punished uniquely by the opponent. Thus the global uniqueness property is typical of many Nim-style game.

Now consider any game and any position P of the game that does not have the global uniqueness property. Suppose furthermore that there is no winning strategy of the current side. Since we assume no "draw" in the game, this means that there is a winning strategy of the opponent side for any of P's succeeding positions. Here if some (resp., any) succeeding position of P has the weak (resp., strong) uniqueness property, then we say that P has the *weak* (resp., *strong*) *co-uniqueness* property. To simplify our statement, let us also regard a position with no winning strategy (of the current side) to have the weak and strong uniqueness property. Similarly, we regard a position with no winning strategy (of the opponent side) to have the weak and strong co-uniqueness property. Then it is possible that a position has the weak (resp., strong) uniqueness and co-uniqueness properties, in which case we say that this position has the *weak* (resp., *strong*) *bi-uniqueness* property.

In general, the problem of finding a winning move for a given position is computationally hard. In complexity theory, the *Winner Decision Problem*, namely whether a given position has a winning strategy for the player to move, usually has the same hardness as finding a winning move when one exists. For example, it has been shown that Winner Decision Problem is EXPTIME-complete for certain liberal generalizations to $n \times n$ boards of Chess and Go and several other games, for which see [Rob84], and becomes PSPACE-complete when those games are constrained to limit all plays to length polynomial in n. We would like to investigate whether this decision problem becomes easier if we can assume some of the uniqueness properties mentioned above. For example, we consider some subset of positions of Go that has the strong uniqueness property, and investigate the complexity of the Winner Decision Problem. (Recall that a position without a winning strategy is also regarded to have the strong uniqueness property. Thus, having the strong uniqueness property does not necessarily mean that there is a winning strategy of the current side.)

There have been some preliminary results on our question. Crasmaru and Tromp [CT00] proved that Winner Decision Problem for Go remains PSPACE-hard even when instances are restricted to very simple "ladder positions". Aida and Tsukiji [AT01] also discussed a way to define a game with some uniqueness property simulating a given alternating Turing machine.

In this paper, we consider our question in the most general case, and we first show the following result. We define "reasonable game G" in the next section, and "extension" means a game that includes all positions of G without changing their winners.

(1) For every reasonable game G, we can define its extension G^* in such a way that every position P of G has the strong bi-uniqueness property in G^*. Furthermore, the (maximal) depth of P, i.e., the depth of the game tree rooted by P, becomes at most three times of the original depth in G.

This result illustrates that the hardness of the problem of deciding which side of a given position wins does not change even if we may assume that the position has the strong bi-uniqueness property. In fact, by using this result, we can define a game G^{PS} (resp., G^{EXP}) and a set P^{PS} (resp., P^{EXP}) of strong bi-unique game positions such that it is PSPACE- (resp., EXPTIME-) complete to determine whether a given position in P^{PS} (resp., P^{EXP}) has a winning strategy. Then for any game G, in order to obtain a set of strong bi-unique game positions that are PSPACE- (resp., EXPTIME-) hard, it is enough to show the way to implement the game rule of G^{PS} (resp., G^{EXP}) in G while keeping the strong bi-uniqueness. One can show the previous results mentioned above in this way.

In contrast to the case of the strong bi-uniqueness property, it seems that the Winner Decision Problem becomes easier when the global uniqueness property can be assumed. Formally we define the following *promise problem* GUQBF:

INSTANCE: A quantified Boolean formula ψ, usually in the prenex form $\psi = (\exists x_d)(\forall x_{d-1}) \cdots \phi$.
PROMISE: The logic game on ψ has the global uniqueness property.
QUESTION: Is ψ true?

In the "logic game on ψ," the first player assigns $x_d := 0$ or $x_d := 1$, then the second player assigns a value to x_{d-1}, and so on. The first player wins if the resulting sentence ϕ is true, else the second player wins. The class SPP was introduced in [FFK94], while the more-familiar class FewP contains the decision version of factoring and other problems believed to be intractable.

(2) The class of languages that polynomial-time many-one reduce to GUQBF, by reductions whose range is contained in the promise set, equals the unambiguous alternation class UAP defined by Niedermeier and Rossmanith, which is located by FewP $\subseteq$ UAP $\subseteq$ SPP [NR98].

In particular, it follows from (2) that GUQBF cannot be PSPACE-hard unless, for example, PSPACE $= \oplus$P. Thus, it seems unlikely that every two player game of polynomial depth can be extended to a game of polynomial depth with the global uniqueness property. On the other hand, GUQBF is not trivial because it is hard for the problem of factoring integers.

2. Formal Definition of Game and Uniqueness Properties

Here we define the notion of game and several uniqueness properties and prepare some notations for our discussion.

For the notion of two-player game, we adopt the following definition given in [CS79].

Definition 2.1. A (*two person perfect-information*) game G is a triple (P_0^G, P_1^G, R^G), where P_0^G and P_1^G are sets such that $P_0^G \cap P_1^G = \emptyset$, and R^G is a subset of $P_0^G \times P_1^G \cup P_1^G \times P_0^G$.

Where no ambiguity arises, we will abbreviate P_0^G as P_0, etc. The set P_0 (resp., P_1) stands for the set of positions in which *player* 0 (resp., 1) is to move, and R stands for the set of legal moves. We assume some reasonable coding for elements of $P_0 \cup P_1$; in the following, we simply identify elements of $P_0 \cup P_1$ and their description under this coding. We assume that P_0 and P_1 are disjoint; furthermore, for any position π in $P_0 \cup P_1$, we can easily compute the player that is to move at π, which we denote by $x(\pi)$. More precisely, the value $x(\pi)$ is computable within polynomial-time w.r.t. the length $|\pi|$ of the representation of π. For a generic player x, where $x \in \{0, 1\}$, we denote by $\overline{x}$ its opponent, i.e., $\overline{x} = 1 - x$, and subsequently we denote by P_x (resp., $P_{\overline{x}}$) the set of moves of the player x (resp., $\overline{x}$). A *move* for player x in the current position $\pi \in P_x$ consists in choosing of a pair $(\pi, \pi') \in R$, π' becoming the next position. For any π, we define $R(\pi)$ $= \{\, \pi' : (\pi, \pi') \in R \,\}$. For any $d \geq 1$, we inductively define R_d as follows.

$$R_1(\pi) \;=\; R(\pi), \quad \text{and} \quad R_{d+1}(\pi) \;=\; \bigcup_{\pi' \in R_d} R(\pi').$$

Define $R_\infty(\pi) = \bigcup_{d \geq 1} R_d(\pi)$.

For the requirements for "reasonable game", we consider the following polynomial-time computability. First, for any $\pi \in P_0 \cup P_1$ and any $\pi' \in R_\infty(\pi)$, it must hold that $|\pi'|$ $\leq q(|\pi|)$ for some polynomial q. Note that this condition implies that the set $R_\infty(\pi)$ is finite for any π. Second, there must be some algorithm that determines $R(\pi, \pi')$ within polynomial-time w.r.t. $|\pi| + |\pi'|$. Third, there is no loop; that is, for any $\pi, \pi' \in P_0 \cup P_1$, it must not be the case that both $\pi' \in R_\infty(\pi)$ and $\pi \in R_\infty(\pi')$ hold. Games satisfying these conditions are called *polynomially definable*.

By convention, a player who is unable to move is declared the loser. Thus a position π is *winning for player* x if:

1. the opponent $\overline{x}$ is to move at π but $\overline{x}$ is unable to move, or
2. x is to move at π and there exists a move driving to a winning position for x, or
3. $\overline{x}$ is to move at π and any available move drives to a winning position for x.

Formally, we define the set $W_x^G = W_x$ of winning positions for player x in the game G by $W_x = \cup_{d \geq 0} W_{x,d}$ and inductively:

$$\begin{aligned} W_{x,0} \;&=\; \{\, \pi \in P_{\overline{x}} \,:\, R(\pi) = \emptyset \,\}, \\ W_{x,d+1} \;&=\; W_{x,d} \,\cup\, \{\, \pi \in P_x \,:\, R(\pi) \cap W_{x,d} \neq \emptyset \,\} \,\cup\, \{\, \pi \in P_{\overline{x}} \,:\, R(\pi) \subseteq W_{x,d} \,\}. \end{aligned}$$

By definition, we clearly have $W_0 \cap W_1 = \emptyset$. Notice also that it holds that $W_0 \cup W_1 = P_0 \cup P_1$; that is, all positions are winning positions of either player 0 or player 1. This is due to the condition prohibiting the existence of a loop. For any $\pi \in P_0 \cup P_1$, the *depth* of π, denoted by $d(\pi)$, is the smallest d such that $\pi \in W_{x,d}$ for some $x \in \{0,1\}$. Intuitively, $d(\pi)$ is the smallest depth necessary to determine which player wins under their best moves. On the other hand, we will also use the *maximal depth* of π, i.e., the length of the longest path from π to a leaf of the game tree, which we denote by $d_{\max}(\pi)$. Precisely, $d_{\max}(\pi)$ is defined inductively as follows.

$$d_{\max}(\pi) \;=\; \begin{cases} 0, & \text{if } R(\pi) = \emptyset, \\ 1 + \max_{\pi' \in R(\pi)}(d_{\max}(\pi')), & \text{otherwise.} \end{cases}$$

Since we assume polynomially definable games, both $d(\pi)$ and $d_{\max}(\pi)$ are bounded by $2^{q(|\pi|)}$ for some polynomial q.

Informally, a position π has the *weak* (resp., *strong*) *uniqueness* property for player x if

1. π is a winning position for opponent $\overline{x}$, or
2. x is to move at π and there exists a unique winning move driving to a position having the *weak* (resp., *strong*) uniqueness property for x, or
3. the opponent $\overline{x}$ is to move at π and there is a move driving to a position (resp., *all* the moves are driving to positions) having the *weak* (resp., *strong*) uniqueness property for x.

Formally, the sets $WU_x^G = WU_x$ and $SU_x^G = SU_x$ of positions having the weak and strong uniqueness properties for player x, are defined inductively by $WU_0 = SU_0 = W_{x,0}$ and for $d \geq 0$,

$$\begin{aligned} WU_{x,d+1} &= \{\pi \in P_x : \pi \in W_x \Rightarrow [\,|R(\pi) \cap W_x| = 1 \,\wedge\, R(\pi) \cap W_x \subseteq WU_{x,d}\,]\} \\ &\quad \cup \{\pi \in P_{\overline{x}} : \pi \in W_x \Rightarrow R(\pi) \cap WU_{x,d} \neq \emptyset\}, \quad \text{and} \\ SU_{x,d+1} &= \{\pi \in P_x : \pi \in W_x \Rightarrow [\,|R(\pi) \cap W_x| = 1 \,\wedge\, R(\pi) \cap W_x \subseteq SU_{x,d}\,]\} \\ &\quad \cup \{\pi \in P_{\overline{x}} : \pi \in W_x \Rightarrow R(\pi) \subseteq SU_{x,d}\}. \end{aligned}$$

Define $WU_x = \bigcup_{d \geq 0} WU_{x,d}$ and $SU_x = \bigcup_{d \geq 0} SU_{x,d}$. We say that a position π has the *weak* (resp., *strong*) *uniqueness property* for player x if π belongs to the set WU_x (resp., SU_x). We say that a position π has the *weak* (resp., *strong*) *bi-uniqueness property* for player x if π belongs to the set $WU_x \cap WU_{\overline{x}}$ (resp., $SU_x \cap SU_{\overline{x}}$). Finally, we say that a position π has the *global uniqueness property* if every position π of $R_\infty(\pi)$ has the strong bi-uniqueness property, and that a game itself has the *global uniqueness property* if every position of the game has the global uniqueness property.

The following relations are immediate from the definition: $W_{\overline{x}} \subseteq SU_x \subseteq WU_x$ and $W_x \subseteq SU_{\overline{x}} \subseteq WU_{\overline{x}}$. The fact that a losing position π for player x has by definition the weak/strong uniqueness property for x is natural, because there is no way to force next move of x in π.

One can easily see that polynomially definable games can be used as a model of polynomially space bounded alternating computation. Furthermore, the uniqueness properties considered here correspond the uniqueness of accepting path at each alternation. For example, for any Σ_k^{P}-machine (for odd k), we can easily define a game simulating its computation so that the depth of any position corresponding to a valid initial configuration of the machine is always k. In particular, for any UP-machine, a game simulating it has the global unique solution property.

3. Conversion to a Game with the Strong Uniqueness Property

Here we prove our first main result, namely that many game problems remain complete for their respective classes even when the range of reductions is restricted to positions with the strong uniqueness property. We do so by converting every polynomially definable game G into a game G^* so that for each position π of G, (i) the conversion does not change π and its winner in G^*, and (ii) π has the strong uniqueness property in G^*. Furthermore $d^*_{\max}(\pi)$, namely the maximum depth of the game tree below π in G^*, is at most $3d_{\max}(\pi)$. The basic idea is that G^* adds a new "turn-the-tables" rule. Under some total ordering of the legal moves in any position π, if the move made in π is not least in the order, the opponent may change places, take back that move, and play a move that is lower in the order. Reductions from G^* then obey the desired restriction on their ranges.

We prepare some notation that is useful in our proof. For any game $G = (P_0, P_1, R)$, we define the *dual game* $\overline{G}$ of G by $\overline{G} = (\overline{P}_0, \overline{P}_1, \overline{R})$, where roughly speaking, $\overline{P}_0 = P_1$, $\overline{P}_1 = P_0$, and $\overline{R} = R$. That is, the dual of G is simply the game in which the players have "exchanged" only their places. Recall that we assume that the player to move is encoded in a position. Thus, precisely speaking, $\overline{P}_0$ is not the same as P_1, for example; instead, every position $\pi \in P_1$, its dual $\overline{\pi} \in \overline{P}_0$ is obtained from π by replacing the part encoding "player 0" with "player 1". From this, we may assume that $(P_0 \cup P_1) \cap (\overline{P}_0 \cup \overline{P}_1) = \emptyset$. For the dual game $\overline{G}$, we can define $W_x^{\overline{G}}$ in the same way as G. To simplify our notation, we will denote $W_x^{\overline{G}}$ by $\overline{W}_x$.

One can observe that the dual of the dual of a game is the game itself, i.e., $\overline{\overline{G}} = G$, and that player x wins in a position π of G if and only if x loses in the dual position $\overline{\pi}$ of $\overline{G}$. Moreover, one can easily show that for any position π of G, we have $d_{\max}(\pi) = d_{\max}(\overline{\pi})$.

Now we are ready to prove our first main result.

Theorem 3.1. For any game $G = (P_0, P_1, R)$, we can define a game $G^* = (P_0^*, P_1^*, R^*)$ such that, for each $x \in \{0, 1\}$, we have (i) $P_x \subseteq P_x^*$ and $W_x \subseteq W_x^*$, and (ii) $W_x \subseteq SU_x^*$. (Here we abbreviate $P_0^{G^*}$ as P^*, etc.) Furthermore, if G is polynomially definable, then so is G^*, and for every $\pi \in P_0 \cup P_1$, we have $d^*_{\max}(\pi) \le 3d_{\max}(\pi)$.

Before stating the proof formally, we explain our conversion idea intuitively. Here let G and G^* denote an original game and its conversion, and let π be any position of G for a player x. Suppose that $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$; that is, $\{\pi_1, \pi_2, \ldots, \pi_k\}$ is the set of the possible next move from π in G. (See Figure 1.) Assume that there are some winning moves of x in $\{\pi_1, \pi_2, \ldots, \pi_k\}$. That is, x can win the game by choosing one of these winning moves. On the other hand, in our new game G^*, player x has to choose the *leftmost* winning move, i.e., the winning move with the smallest index. Furthermore, x has to verify (if it is requested by the opponent player $\overline{x}$) that his choice is indeed the leftmost.

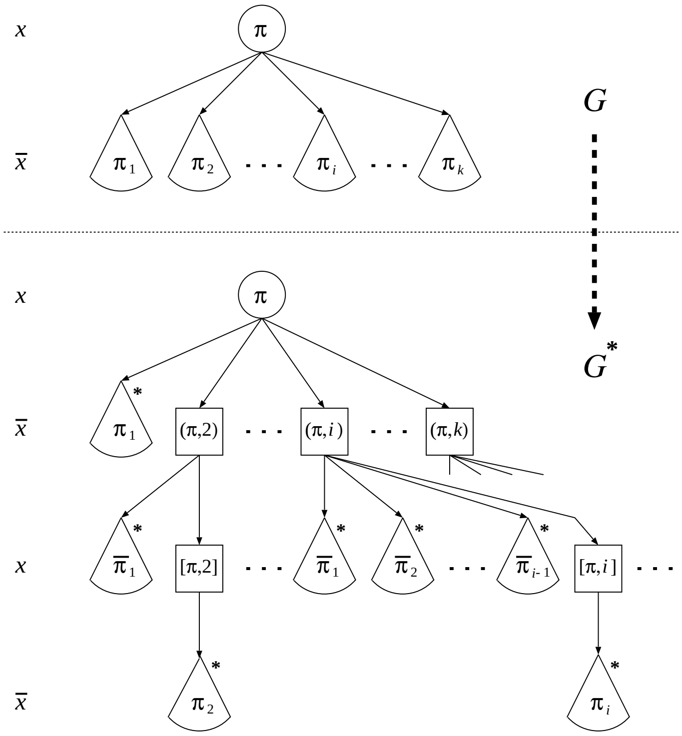

Figure 1. Conversion from G to G^*

Assume for example, π_3 is the leftmost winning move. Then to win in the new game G^*, x has to choose (π, π_3) as a next position of π, where (π, π_3) is a newly introduced position. Intuitively, choosing this position is to declare that the third position (i.e., π_3) in the set of the next moves from π is the leftmost winning move. Then in the next move, the opponent player $\overline{x}$ can choose $\overline{\pi}_1$, $\overline{\pi}_2$, or $[\pi, \pi_3]$. Intuitively, choosing $[\pi, \pi_3]$ means to continue the game as in the original game; that is, $\overline{x}$ accepts π_3 as a next position. Then x continues the game by choosing π_3, and hence, the next move of x from $[\pi, \pi_3]$ is always π_3. (See Figure 1.) On the other hand, choosing, e.g., $\overline{\pi}_1$, intuitively means that player x has to show that π_1 is not the winning move in G; that is, he would have lost the game by choosing π_1 in G (unless the opponent makes any mistake). This is because x loses from π_1 if and only if x wins from $\overline{\pi}_1$.

We keep this conversion inductively throughout the game tree. (The star symbol in Figure 1 means that the same conversion is made in the corresponding subtrees.) Since the leftmost winning move is unique, it is intuitively clear that some unique solution property is guaranteed in the new game, and indeed, it is not so difficult to prove that the strong bi-unique solution property holds. Also it is clear that the maximal depth increases at most three times from the original maximal depth.

Proof. We first state how to define G^* from G formally. For any $x \in \{0,1\}$, let $\pi \in P_x$ be a position of the game G, and let $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$ be the set of obtainable positions from π in one move in game G. (Whenever the sequence "$\pi_1, \pi_2, \ldots, \pi_k$" appears in the proof, suppose that it is previously ordered lexicographically.) Then we define $AuxP(\pi)$ and $AuxP'(\pi)$ by

$$AuxP(\pi) \; = \; \{\, (\pi, \pi_i) \,:\, 2 \le i \le k \,\}, \quad \text{and} \quad AuxP'(\pi) = \; \{\, [\pi, \pi_i] \,:\, 2 \le i \le k \,\}.$$

For any $\pi \in \overline{P}_x$, we define $AuxP(\pi)$ and $AuxP'(\pi)$ in the same way. These states are newly added in the game G^*. Precisely, in G^*, we define the sets P_0^* and P_1^* of positions in the following way.

$$\begin{aligned} P_0^* &= P_0 \cup \overline{P}_0 \cup \bigcup_{\pi \in P_0 \cup \overline{P}_0} AuxP'(\pi) \cup \bigcup_{\pi \in P_1 \cup \overline{P}_1} AuxP(\pi), \quad \text{and} \\ P_1^* &= P_1 \cup \overline{P}_1 \cup \bigcup_{\pi \in P_1 \cup \overline{P}_1} AuxP'(\pi) \cup \bigcup_{\pi \in P_0 \cup \overline{P}_0} AuxP(\pi). \end{aligned}$$

The new game relation R^* is defined as follows in terms of $AuxR(\pi)$.

$$\begin{aligned} R^* &= \bigcup_{\pi \in P_0 \cup P_1 \cup \overline{P}_0 \cup \overline{P}_1} AuxR(\pi), \quad \text{and} \\ AuxR(\pi) &= \{\, (\pi, \pi_1)\} \,\cup\, \{(\pi, (\pi, \pi_i)) \,:\, 2 \le i \le k\,\} \\ &\quad \cup\, \{\, ((\pi, \pi_i), \overline{\pi}_j) \,:\, 1 \le j < i \le k\,\} \,\cup\, \{\, ((\pi, \pi_i), [\pi, \pi_i]) \,:\, 2 \le i \le k\,\} \\ &\quad \cup\, \{\, ([\pi, \pi_i], \pi_i) \,:\, 2 \le i \le k\,\}. \end{aligned}$$

Observe that equality $P_0^* \cap P_1^* = \emptyset$ holds by assuming that $(P_0 \cup P_1) \cap (\overline{P}_0 \cup \overline{P}_1) = \emptyset$. It is obvious that $P_0 \subseteq P_0^*$, and $P_1 \subseteq P_1^*$.

If P_0, P_1 and R are polynomial-time computable, then so are P_0^* and P_1^*, because P_0^* (resp. P_1^*) is the union of polynomial-time computable sets, P_0, $\overline{P}_0$, $\bigcup_{\pi \in P_0 \cup \overline{P}_0} AuxP'(\pi)$ and $\bigcup_{\pi \in P_1 \cup \overline{P}_1} AuxP(\pi)$ (resp. P_1, $\overline{P}_1$, $\bigcup_{\pi \in P_1 \cup \overline{P}_1} AuxP'(\pi)$ and $\bigcup_{\pi \in P_0 \cup \overline{P}_0} AuxP(\pi)$.) Similarly one can check polynomial-time computability of R^*. Therefore if G is polynomially definable, then so is G^*.

We can show that the new game has the desired properties by induction on the maximal depth.

For this, we first define $Q(d)$ to be the set of π with $d_{\max}(\pi) = d$. Then clearly, we have $\bigcup_{d \ge 0} Q(d) = P_0 \cup P_1$. Again we use $\overline{Q}(d)$ to denote the same set for the dual game. Below we prove that, equalities $W_x \cap Q(d) = W_x^* \cap Q(d)$ and $\overline{W}_x \cap \overline{Q}(d) = W_x^* \cap \overline{Q}(d)$ hold for any integer $d \ge 0$.

Consider the case when $\pi \in W_x \cap Q(0)$. Then since $R(\pi) = \emptyset$, we have $R^*(\pi) = \emptyset$, and by definition of W_x^*, we have $\pi \in W_x^*$. If $\pi \in W_x^* \cap Q(0)$, then we also have $R(\pi) = \emptyset$, which is equivalent to $\pi \in W_x$. Thus, we conclude that $W_x \cap Q(0) = W_x^* \cap Q(0)$. The corresponding equality holds in the same way for the dual game.

Suppose now that the equalities $W_x \cap Q(d') = W^* \cap Q(d')$ and $\overline{W}_x \cap \overline{Q}(d') = W^* \cap \overline{Q}(d')$ hold for any positive integer $d' \leq d$, and we will show that the same equalities hold for $d+1$.

Let π be a position of the game G such that $\pi \in P_x \cap W_x \cap Q(d+1)$, and $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$. By the definition of W_x, there exists an index i_0 such that $1 \leq i_0 \leq k$, $\pi_{i_0} \in W_x$, $d_{\max}(\pi_{i_0}) \leq d$, and $\pi_j \in W_{\overline{x}}$ for each $j < i_0$. Using the property of the dual game, we have that $\overline{\pi}_{i_0} \in \overline{W}_{\overline{x}}$, and $\overline{\pi}_j \in \overline{W}_x$ for each $j < i_0$. Because $d_{\max}(\pi_j) = d_{\max}(\overline{\pi}_j) \leq d$ for each $1 \leq j \leq i_0$, from our inductive hypothesis we obtain that $\pi_{i_0} \in W^*_x$, $\overline{\pi}_{i_0} \in W^*_{\overline{x}}$, $\pi_j \in W^*_{\overline{x}}$, and $\overline{\pi}_j \in W^*_x$, where $1 \leq j < i_0$. Then $[\pi, \pi_{i_0}] \in W^*_x$, because $R^*([\pi, \pi_{i_0}]) \cap W^*_x = \{\pi_{i_0}\}$. We then have that $R^*([\pi, \pi_{i_0}]) = \{\overline{\pi}_1, \ldots, \overline{\pi}_{i_0-1}, [\pi, \pi_{i_0}]\} \subseteq W^*_x$, therefore $(\pi, \pi_{i_0}) \in W^*_x$, by the definition of W^*_x. The inclusion $\{(\pi, \pi_{i_0})\} \subseteq R^*(\pi) \cap W^*_x$ implies that $\pi \in W^*_x$.

On the other hand, take a position $\pi \in P_x \cap W^*_x \cap Q(d+1)$ such that $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$. Using the definition of W^*_x and the rule R^*, we have either $\pi_1 \in W^*_x$, or there exists an index i_0 such that $2 \leq i_0 \leq k$, $(\pi, \pi_{i_0}) \in W^*_x$, and $\{\pi_1, (\pi, \pi_2), \ldots, (\pi, \pi_{i_0-1})\} \subseteq W^*_{\overline{x}}$. The case when $\pi_1 \in W^*_x$ is trivial, following directly from our inductive hypothesis, observing that $d_{\max}(\pi_1) \leq d$. Let us consider the other case; that is, $R^*((\pi, \pi_{i_0})) = \{\overline{\pi}_1, \ldots, \overline{\pi}_{i_0-1}, [\pi, \pi_{i_0}]\} \subseteq W^*_x$, following that $[\pi, \pi_{i_0}] \in W^*_x$. We then have $\{\pi_{i_0}\} = R([\pi, \pi_{i_0}]) \cap W^G_x$; thus $\pi_{i_0} \in W^*_x$. Observing that $d_{\max}(\pi_{i_0}) \leq d$, and applying our inductive hypothesis, follows that $\pi_{i_0} \in W^*_x$. But the last relation shows that $R(\pi) \cap W_x \neq \emptyset$; therefore $\pi \in W_x$.

Consider now the case when π is a position of the game G such that $\pi \in P_x \cap W_{\overline{x}} \cap Q(d+1)$, and $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$. By the definition of $W_{\overline{x}}$, we have that $\pi_j \in W_{\overline{x}}$, and $d_{\max}(\pi_j) \leq d$ for $1 \leq j \leq k$. Our inductive hypothesis implies that $\pi_j \in W^*_{\overline{x}}$ for $1 \leq j \leq k$. Then $[\pi, \pi_j] \in W^*_{\overline{x}}$, because $R^*([\pi, \pi_j]) = \{\pi_j\} \subseteq W^*_{\overline{x}}$ and $[\pi, \pi_j] \in P^*_x$, for each $2 \leq j \leq k$. It follows that $(\pi, \pi_j) \in W^*_{\overline{x}}$, observing that $[\pi, \pi_j] \in R^*((\pi, \pi_i)) \cap W^*_{\overline{x}} \cap P^*_{\overline{x}}$ for each $2 \leq j \leq k$. We then have $R^*(\pi) = \{\pi_1, (\pi, \pi_2), \ldots, (\pi, \pi_k)\} \subseteq W^*_{\overline{x}}$; therefore $\pi \in W^*_{\overline{x}}$, by the definition of $W^*_{\overline{x}}$.

Conversely, consider a position π of the game G such that $R^G(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$, and $\pi \in P_x \cap W^*_{\overline{x}} \cap Q(d+1)$. Using the definition of $W^*_{\overline{x}}$, we have that $R^*(\pi) = \{\pi_1, (\pi, \pi_2), \ldots, (\pi, \pi_k)\} \subseteq W^*_{\overline{x}}$. Then $\pi_1 \in W^*_{\overline{x}}$ and $\overline{\pi}_1 \in W^*_x$. Suppose now that we have proved that $\pi_1, \ldots, \pi_j \in W^*_{\overline{x}}$, where $1 \leq j < k$. Then $\overline{\pi}_i \in W^*_x$, for each $1 \leq i < j+1$. Moreover $R^*((\pi, \pi_{j+1})) = \{\overline{\pi}_1, \ldots, \overline{\pi}_j, [\pi, \pi_{j+1}]\} \cap W^*_{\overline{x}} \neq \emptyset$ implies that $[\pi, \pi_{j+1}] \in W^*_{\overline{x}}$. Because of the equality $\{\pi_{j+1}\} = R^*([\pi, \pi_{j+1}]) \subseteq W^*_{\overline{x}}$, we have that $\{\pi_{j+1}\} \subseteq W^*_{\overline{x}}$. Hence, we conclude that $\{\pi_1, \ldots, \pi_k\} \subseteq W^*_{\overline{x}}$ and $d_{\max}(\pi_j) \leq d$ for each $1 \leq j \leq k$. From our inductive hypothesis, we obtain that $\{\pi_1, \ldots, \pi_k\} \subseteq W_{\overline{x}}$, which implies that $\pi \in W_{\overline{x}}$.

We just proved by mathematical induction that $W_x \cap Q(d) = W^*_x \cap Q(d)$, for any $d \geq 0$. The corresponding dual relation $\overline{W}_x \cap \overline{Q}(d) = W^*_x \cap \overline{Q}(d)$ can be proved in the same way. Therefore, we can conclude that $W_x \subseteq W^*_x$ holds. What follows now is the proof that any position of the game G has the strong bi-uniqueness property in the game G^*.

First let us observe that any position $\pi \in W_x \cap Q(0)$ has the strong bi-uniqueness property in the game G^*, because $R^*(\pi) = \emptyset \subseteq SU_x^*$. For its dual position $\overline{\pi}$, we also have $\overline{\pi} \in SU_{\overline{x}}^*$. Then relations $W_x \cap Q(0) \subseteq SU_x^*$, and $\overline{W}_x \cap \overline{Q}(0) \subseteq SU_x^*$ hold.

Suppose now that the relations $W_x \cap Q(d) \subseteq SU_x^*$, and $\overline{W}_x \cap \overline{Q}(d) \subseteq SU_x^*$ hold for any integer d' such that $0 \leq d' \leq d$.

Let π be a position such that $\pi \in P_x \cap W_x \cap Q(d+1)$ and $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$. As we did in the above proof, we can assume that there exists an index i_0 such that $1 \leq i_0 \leq k$, $\pi_{i_0} \in W_x$, and $\pi_j \in W_{\overline{x}}$ for each $j < i_0$. Then as shown above, we have $\pi_{i_0} \in W_x^*$, and $\pi_j \in W_{\overline{x}}^*$, where $1 \leq j < i_0$. One can observe that $R^*([\pi, \pi_j]) = \{\pi_j\} \subseteq W_{\overline{x}}^*$; that is, $[\pi, \pi_j] \in W_{\overline{x}}^*$ for $2 \leq j < i_0$. Then $R^*((\pi, \pi_j)) \cap W_{\overline{x}}^* = \{[\pi, \pi_j]\}$; hence, $(\pi, \pi_j) \in W_{\overline{x}}^*$ for $2 \leq j < i_0$. On the other hand, $\overline{\pi}_{i_0} \in W_{\overline{x}}^*$ implies that $\{\overline{\pi}_{i_0}\} \subseteq R^*((\pi, \pi_j)) \cap W_{\overline{x}}^* \neq \emptyset$; thus $(\pi, \pi_j) \in W_{\overline{x}}^*$ for each $i_0 < j \leq k$. Remembering that $(\pi, \pi_{i_0}) \in W_x^*$, we can conclude that $R^*(\pi) \cap W_x^* = \{(\pi, \pi_{i_0})\}$. That is, $|R^*(\pi) \cap W_x^*| = 1$.

One can observe that $d_{\max}(\pi_{i_0}) \leq d$, and $d_{\max}(\overline{\pi}_j) \leq d$. Then from our inductive hypothesis it follows that $\pi_{i_0} \in SU_x^*$, and $\overline{\pi}_j \in SU_x^*$ for $1 \leq j < i_0$. Because $|R^*([\pi, \pi_{i_0}])| = |\{\pi_{i_0}\}| = 1$, and $\{\pi_{i_0}\} \subseteq SU_x^*$, we have that $[\pi, \pi_{i_0}] \in SU_x^*$. It follows that $R^*((\pi, \pi_{i_0})) = \{\overline{\pi}_1, \ldots, \overline{\pi}_{i_0-1}, [\pi, \pi_{i_0}]\} \subseteq SU_x^*$; that is, $(\pi, \pi_{i_0}) \in SU_x^*$. This and the fact $|R^*(\pi) \cap W_x^*| = |\{(\pi, \pi_{i_0})\}| = 1$ imply that $\pi \in SU_x^*$.

Consider now a position π such that $\pi \in P_x \cap W_{\overline{x}} \cap Q(d+1)$ and $R(\pi) = \{\pi_1, \pi_2, \ldots, \pi_k\}$. We have that $d_{\max}(\pi_j) \leq d$; then from our inductive hypothesis it follows that $\pi_j \in SU_{\overline{x}}^*$, and $\overline{\pi}_j \in SU_x^*$ for $1 \leq j \leq k$. Because $R^*([\pi, \pi_j]) = \{\pi_j\} \subseteq SU_{\overline{x}}^*$, we have $[\pi, \pi_j] \in SU_{\overline{x}}^*$ for each $2 \leq j \leq k$. We then obtain that $|R^*((\pi, \pi_j)) \cap W_{\overline{x}}^*| = |\{[\pi, \pi_j]\}| = 1$; that is, $(\pi, \pi_j) \in SU_{\overline{x}}^*$, where $2 \leq j \leq k$. Hence, we have $R^*(\pi) = \{\pi_1, (\pi, \pi_2), \ldots, (\pi, \pi_k)\} \subseteq SU_{\overline{x}}^*$; therefore, $\pi \in SU_{\overline{x}}^*$. This finishes our proof. □

When using polynomially definable games as a computation model, we may usually assume that the depth is the same as the maximal depth. Thus, from the above result, we can show that any alternating computation can be simulated by some alternating computation with (i) the same order of alternations, and (ii) the strong bi-uniqueness property.

4. Global Uniqueness Property and Globally Unique QBFs

Given any G, it is possible to create a game G^{**} with the global uniqueness property by extending the "turn the tables and takeback" rule of the last section to allow rewinding any number of moves. While G^{**} always has *optimal* strategies of polynomial depth (given that *all* plays in G—in particular, all leftmost winning plays—have polynomial depth), it will no longer hold that *all* plays in G^{**} have polynomial depth. This section will show that the difference between these clauses is that between PSPACE and a subclass of PP called UAP for "alternating UP" by Niedermeier and Rossmanith [NR98]. We will show that UAP is captured by reduction to GUQBF, i.e. by global uniqueness for the standard logic game on quantified Boolean formulas, but first we give their natural definition in terms of alternating Turing machines (ATMs).

Definition 4.1. (cf. [NR98]) An ATM M has the *global uniqueness property* if the naturally-associated game does. Namely, in every accepting non-final existential configuration of M, exactly one move leads to an accepting configuration, and in every rejecting non-final universal configuration, exactly one move leads to a rejecting configuration.

UAP denotes the class of languages of languages accepted by polynomial-time bounded ATMs with the global uniqueness property.

Our first main theorem is that UAP equals the class of languages that many-one reduce in polynomial time to the promise problem GUQBF given in the Introduction. We re-cast this as a promise problem about Boolean formulas F with labeled variables $x_1, \ldots, x_d$, by regarding F as inducing the quantified Boolean formula

$$\phi \;=\; \exists x_d \, \forall x_{d-1} \, \exists x_{d-2} \cdots Q_d x_1 \, F,$$

where Q_d is $\forall$ if d is even, $\exists$ if d is odd. The strict alternation of $\exists$ and $\forall$ quantifiers thus defined makes a strict alternation of moves by the player controlling the variables with $\exists$ quantifers, whom we call the "E-player," and the opponent controlling the universal quantifiers, whom we call the "U-player." This definition makes the E-player always go first, but we will sometimes consider (sub-)formulas to include (sub-)games in which the U-player goes first. All plays of the logic game have the same depth d, as variables from x_d down to x_1 are successively instantiated 0 or 1, and the E-player wins iff the final sentence F is true. We call ϕ a "PQBF" for "prenex QBF," and use "GUPQBF" for GUQBF emphasized thus as a promise problem about *Boolean* formulas:

INSTANCE: A Boolean formula F with labeled variables $x_d, \ldots, x_1$.
PROMISE: The game on the induced PQBF ϕ has the global uniqueness property.
QUESTION: Is ϕ true?

Abstractly, a promise problem (Q, R) has *promise set* Q and *property set* R, and any language S such that $Q \cap R \subseteq S$ and $Q \setminus R \subseteq \overline{S}$ is a *solution*. The definition of reducibility between promise problems given by Selman [Sel88] entails that a language A polynomial-time many-one reduces to (Q, R) if $A \leq_m^p S$ for every solution S of (Q, R). It follows that there is a single polynomial-time computable function f such that for all x, $f(x)$ is in Q, and $f(x) \in R \Leftrightarrow x \in A$. (To see this, build a solution S such that for every polynomial-time computable function g with $Ran(f) \setminus Q$ infinite, g does not reduce A to S. The leftover f giving $A \leq_m^p S$ all have $Ran(f) \setminus Q$ finite, and patching does the rest.) We note that the promise problem "given a Boolean formula F, with the promise that F has at most one satisfying assignment, is F satisfiable?" yields UP as the class of languages that reduce to it in this sense. Likewise BPP, RP, and other "promise classes" can be characterized by reducibility to promise problems.

We note first that GUPQBF is invariant under equivalence of Boolean formulas F and G, since the PQBFs induced from F and G define the same logic game. Thus we can characterize game positions with global uniqueness by selecting representatives from each equivalence class. Let TRUE (resp., FALSE) denote a constant Boolean formula whose value is 1 (resp., 0). We define inductively the following sets of Boolean formulas:

$$\begin{aligned}
A_0 &= \{\,\text{FALSE}\,\}, \\
B_0 &= \{\,\text{TRUE}\,\}, \quad \text{and for } d \geq 1, \\
A_d &= \{\,(x_d \wedge \neg F_1) \vee (\bar{x}_d \wedge \neg F_0) : F_0, F_1 \in B_{d-1}\,\}, \\
B_d &= \{\,(x_d \wedge \neg F_1) \vee (\bar{x}_d \wedge \neg F_0) : \\
&\qquad (F_0 \in A_{d-1} \text{ and } F_1 \in B_{d-1}) \text{ or } (F_0 \in B_{d-1} \text{ and } F_1 \in A_{d-1})\,\}.
\end{aligned}$$

Here B_d comprises those d-variable Boolean formulas that induce globally-unique logic games with the E-player to move that are wins for the E-player, while A_d comprises those in which the E-player is to move but loses. Note that A_d says that both substitutions for x_d leave Boolean formulas whose *negations* are in B_{d-1}, meaning the negations are unique wins with the E-player to move, which implies that the resulting formulas themselves are unique U-player wins with the U-player to move. The recursion for B_d is interpreted similarly.

For example, $A_1 = \{\,(x_1 \wedge \text{FALSE}) \vee (\bar{x}_1 \wedge \text{FALSE})\,\}$, which is equivalent to $A_1 = \{\,\text{FALSE}\,\}$ and (mentioning x_1) to $A_1 = \{\,x_1 \wedge \bar{x}_1\,\}$. Also

$$B_1 = \{\,(x_1 \wedge \text{TRUE}) \vee (\bar{x}_1 \wedge \text{FALSE}), \quad (x_1 \wedge \text{FALSE}) \vee (\bar{x}_1 \wedge \text{TRUE})\,\},$$

which reduces to $B_1 = \{\,x_1, \bar{x}_1\,\}$. For simplicity, let us use $\pm x$ to denote either x or $\bar{x}$. Then syntactically, A_2 consists of the four formulas $(x_2 \wedge \pm x_1) \vee (\bar{x}_2 \wedge \pm x_1)$ over the four possible combinations of $\pm x_1$ in both places. Semantically this reduces to $\{\,x_1, \bar{x}_1\,\}$, which is the same as B_1. Also

$$\begin{aligned}
B_2 = \{ \;\; & (x_2 \wedge \text{TRUE}) \vee (\bar{x}_2 \wedge x_1), \;\; (x_2 \wedge \text{TRUE}) \vee (\bar{x}_2 \wedge \bar{x}_1), \\
& (x_2 \wedge x_1) \vee (\bar{x}_2 \wedge \text{TRUE}), \;\; (x_2 \wedge \bar{x}_1) \vee (\bar{x}_2 \wedge \text{TRUE}) \;\},
\end{aligned}$$

which is equivalent to the set of four formulas $\pm x_1 \vee \pm x_2$. The following characterization confirms the above interpretation.

Lemma 4.2. For every Boolean formula F on $d \geq 1$ variables, let ϕ be the QBF induced from F, and consider ϕ as a logic-game position with the E-player to move.

(a) The U-player wins at ϕ with global uniqueness if and only if F is equivalent to a Boolean formula in A_d.

(b) The E-player wins at ϕ with global uniqueness if and only if F is equivalent to a Boolean formula in B_d.

Proof. By inspection this is true for $d = 1$. Let $d \geq 2$. For a given F, define $F_0 = F[x_d = 0]$, $F_1 = F[x_d = 1]$,

$$\begin{aligned} \phi_0 &= \forall x_{d-1}\, \exists x_{d-2} \cdots Q_d x_1\, F_0, \quad \text{and} \\ \phi_1 &= \forall x_{d-1}\, \exists x_{d-2} \cdots Q_d x_1\, F_1. \end{aligned}$$

For showing the $(\Rightarrow)$ direction, first suppose that ϕ has the global uniqueness property and that U-player wins on ϕ. Then both ϕ_0 and ϕ_1 are false, so $\neg\phi_0$ and $\neg\phi_1$ are true. Thus the Boolean formulas $\neg F_0$ and $\neg F_1$ both induce QBFs such that E-player wins in the $(d-1)$-round logic game, and by global uniqueness for F, follows that they have global uniqueness. Hence by induction hypothesis, $\neg F_0$ and $\neg F_1$ are respectively equivalent to Boolean formulas G_0 and G_1 in B_{d-1}. Then the formula

$$G = (x_d \;\wedge\; G_1) \;\vee\; (\bar{x}_d \;\wedge\; G_0) \tag{1}$$

belongs to A_d and is equivalent to F as a Boolean formula.

Now suppose instead that E-player wins on ϕ. Then exactly one of ϕ_0 and ϕ_1 is true; w.l.o.g. suppose it is ϕ_0. Then $\neg F_0$ induces a QBF from which E-player loses and $\neg F_1$ induces a QBF from which E-player wins, both with global uniqueness. By induction hypothesis, there are formulas $G_0 \in A_{d-1}$ and $G_1 \in B_{d-1}$ equivalent to F_0 and F_1 respectively. Then G defined as in (1) belongs to B_d and is again equivalent to F. The case where ϕ_1 is true similarly yields the other defining case of membership in B_d.

The reasoning in the other direction is immediate, since equivalent Boolean formulas induce equivalent logic games. $\square$

This lemma characterizes true/false PQBFs with global uniqueness up to equivalence of the inducing Boolean formulas. Note that distinct formulas in $A_d \cup B_d$ are inequivalent. We can count these sets via the recursion:

$$|A_d| = |B_{d-1}|^2, \quad |B_d| = 2|A_{d-1}| \cdot |B_{d-1}|.$$

With $|A_0| = |B_0| = 1$, this yields for d even $|A_d| = |B_d| = 2^{(2/3)(2^d-1)}$, and for d odd, $|A_d| = 2^{(2/3)(2^d-2)}$, $|B_d| = 2|A_d|$. Thus there are members of A_d and B_d that have only formulas of bit-size $\Omega(2^d)$.

However, we will see that *some* members of A_d and B_d have equivalents of linear size in d. The exponential size in general makes it challenging to give an upper bound for the promise problem GUPQBF. We first utilize some familiar counting classes. PP is known to be polynomial-time Turing equivalent to Valiant's class #P of functions f such that for some polynomial-time bounded NTM N and all x, $f(x)$ equals the number of accepting computations of N on input x. A language L belongs to $\mathrm{C_=P}$ iff there is a #P function f and a polynomial-time computable function $g : \Sigma^* \longrightarrow \mathbf{N}$ such that for all x, $x \in L \Leftrightarrow f(x) = g(x)$. $\oplus$P is the class defined by stipulating $x \in L \Leftrightarrow f(x)$ is odd.

Theorem 4.3. The promise problem GUPQBF has solutions in $\mathrm{PP} \cap \oplus\mathrm{P}$ and in $\mathrm{C_=P} \cap \text{co-}\mathrm{C_=P}$.

Proof. The basic point of everything is the following fact, which is proved by straightforward induction.

Fact 1. For even d, every formula in A_d has $N_d = (2/3)(2^d - 1)$ satisfying assignments, while for odd d, every formula in A_d has $N_d = N_{d-1}$ satisfying assignments. On the other hand, for all d, every formula in B_d has $N_d + 1$ satisfying assignments.

Now let N_0 be the familiar NTM that checks satisfiability. We define an NTM N as follows. For a given input PQBF ϕ with global uniqueness, the machine N first reads d from the formula and obtains the unquantified formula F that indeced ϕ. If d is even, $d \geq 2$, then N makes a nondeterministic choice between simulating N_0 on F or entering a routine that has exactly $(1/3)(2^d - 1)$ accepting computations and the same number of rejecting computations. Then by the above fact, if ψ is true, then $N(\phi)$ has exactly 2^d accepting computations, while if ψ is false, then $N(\phi)$ has exactly $2^d - 1$ accepting computations. Then defining $g(\phi) = 2^d$ makes N, g define a language in $\mathrm{C_=P}$ that is a solution to the promise problem, while defining $g(\phi) = 2^d - 1$ likewise defines a solution in co-$\mathrm{C_=P}$. (Note that these solutions can diverge outside the promise set and may not be in $\mathrm{C_=P} \cap \text{co-}\mathrm{C_=P}$.)

Further padding yields an N' such that on such inputs ϕ, the accepting computations outnumber the rejecting ones iff ϕ is true, and this determination for N' can be made with one call to a language in PP. We can also pad to add exactly one satisfying assignment, thus making ϕ true iff the number of satisfying assignments is the odd value $2^d + 1$ versus the even value 2^d, thus yielding a solution in $\oplus$P. The case of d odd is similar. $\square$

We can upper-bound languages L such that $L \leq_m^p$ GUPQBF more sharply. The class SPP was defined in [FFK94] to comprise those languages L such that for some polynomial-time bounded NTM N and all $x \in \Sigma^*$,

$$\begin{aligned} x \in L &\Rightarrow \#acc(N,x) - \#rej(N,x) = 1, \\ x \notin L &\Rightarrow \#acc(N,x) - \#rej(N,x) = 0, \end{aligned}$$

where $\#acc(N,x)$ and $\#rej(N,x)$ denote the numbers of accepting and rejecting computations of N on input x, respectively. By applying "Closure Property 6" in [FFK94], it suffices [Fen02] to replace the former condition by

$$x \in L \Rightarrow \#acc(N,x) - \#rej(N,x) = 2.$$

Theorem 4.4. Every language L such that $L \leq_m^p$ GUPQBF belongs to SPP.

It is clear that $L \leq_m^p$ GUPQBF implies that L belongs to UAP as defined above, so we could use the result UAP $\subseteq$ SPP from [NR98]. The proof in [NR98] adds one accepting and one rejecting path to every rejecting final configuration, and one rejecting path to every universal configuration, observing that the resulting NTM N' gives $\#acc(N',x) - \#rej(N',x) = 0$ or 1 for every x. However, we prefer to use Fact 1 about the number of satisfying assignments to the formulas.

Proof. Take N to compute $f(x)$ and then guess an assignment of the resulting formula. Fact 1 gives us polynomial-time computable functions $g, g' : \Sigma^* \longrightarrow \mathbf{N}$ such that for all $x \in \Sigma^*$, regardless of whether d is even or odd:

$$\begin{aligned} x \in L &\Rightarrow \#acc(N,x) = g(x) + 1 \ \wedge \ \#rej(N,x) = g'(x), \\ x \notin L &\Rightarrow \#acc(N,x) = g(x) \ \wedge \ \#rej(N,x) = g'(x) + 1. \end{aligned}$$

By padding N with $g'(x) + 1$ additional accepting computations and $g(x)$ additional rejecting computations on any input x, we obtain a polynomial-time bounded NTM N' such that for all x,

$$\begin{aligned} x \in L &\Rightarrow \#acc(N',x) = g(x) + g'(x) + 2 \ \wedge \ \#rej(N',x) = g(x) + g'(x), \\ x \notin L &\Rightarrow \#acc(N',x) = \#rej(N',x) = g(x) + g'(x) + 1. \end{aligned}$$

This meets the "difference 2 or 0" condition for membership of L in SPP cited just above. $\square$

Since PP is believed to be a proper subclass of PSPACE, and SPP is regarded as even lower (to wit, $\mathrm{PP}^{\mathrm{SPP}} = \mathrm{PP}$, i.e. "SPP is low for PP" [FFK94]), it seems unlikely that truth of globally-unique formulas is as hard as PSPACE. Whether the complexity of the global-uniqueness property itself—i.e. the complexity of the *promise*—obeys these bounds is open, however, as is whether GUPQBF has any solutions in SPP itself. However, we show that GUPQBF is at least as hard as factoring integers by reducing UP to it. This is apparently tantamount to showing the equivalence between reduction to GUPQBF and membership in UAP as defined above, which requires first some padding techniques for converting alternating Turing machines into PQBFs that preserve global uniqueness.

To begin, consider the conversion of any (prenex) QBF to one in PQBF, for instance $\exists x_3 \exists x_2 \forall x_1 \, F(x_1, x_2, x_3)$ to $\exists x'_4 \forall x'_3 \exists x'_2 \forall x'_1 \, F'(x'_1, x'_2, x'_3, x'_4)$ in PQBF. For this, we make use of the following set of formulas, which we call *trivial unique game positions.* (Here we write the $\wedge$ operation like an invisible tighter-binding multiplication for better visual impact.)

$$\begin{aligned} t_0 &= \textsc{True} \\ t_2 &= x_1 \vee x_2 \\ t_4 &= x_1 \vee x_2(x_3 \vee x_4) \\ t_6 &= x_1 \vee x_2(x_3 \vee x_4(x_5 \vee x_6)) \\ t_8 &= x_1 \vee x_2(x_3 \vee x_4(x_5 \vee x_6(x_7 \vee x_8))) \\ &\vdots \end{aligned}$$

and so on. Also for odd $d \geq 1$, define t_d by substituting $x_{d+1} = \text{False}$ in t_{d+1}. Let τ_d denote the QBF induced from t_d. Notice that these formulas are much simpler than those in $A_d \cup B_d$; in fact, the size of the formula τ_d is linear in d.

Lemma 4.5. For even d, the formula τ_d is a game position from which E-player wins with global uniqueness, while for odd d, U-player wins from τ_d with global uniqueness.

Proof. Consider any even d. In t_d, if E-player sets $x_d = \text{True}$, the game reduces to t_{d+2} regardless of U-player's move. But if E-player plays $x_d = \text{False}$, the result is t_{d-1}, and U-player can uniquely avoid reducing to t_{d-2} by setting $x_{d-1} = \text{False}$. That setting of x_{d-1} annihilates x_{d-2}, leaving E-player powerless to avoid reduction to t_{d-3} and so on to 0, i.e., a U-player's win. The same argument holds for every sequence of e moves; the formula is reduced either t_{d-e} or t_{d-e-1} and so global uniqueness holds. The odd case can be analyzed similarly. □

Now we see how these formulas fill an essential padding role.

Lemma 4.6. There is a polynomial-time procedure that, given any prenex form QBF ϕ' in d' variables with global uniqueness, produces a Boolean formula F in $d \leq 2d'$ variables that induces a formula ϕ in PQBF with global uniqueness that is equivalent to ϕ'. (In fact, it holds that F is equivalent to some formula in B_d if ϕ' is true and equivalent to some formula in A_d if ϕ' is false.)

Proof. By renumbering variables in ϕ', we may assume that the innermost variable is numbered x_1 (resp., x_2) if it is universally (resp., existentially) quantified, and that variables are numbered going outward so that every universally (resp., existentially) quantified variable has odd (resp., even) index. Let x_d (resp., x_{d-1}) be the last index if x_d is existentially (resp., universally) quantified. (Thus, we assume that d is even.) Now we define a recursive procedure $\mathcal{R}$ that works on prenex QBFs with free variables allowed. $\mathcal{R}$ does not change on single-quantifier QBFs; otherwise, it works as follows according to the first two quantifiers.

$$\begin{aligned}
\mathcal{R}(\exists x_d \exists x_{d-2}\psi) &= \exists x_d \forall x_{d-1} [(\bar{x}_{d-1} \wedge \tau_{d-2}) \vee (x_{d-1} \wedge \mathcal{R}(\exists x_{d-2}\psi))], \\
\mathcal{R}(\exists x_d \forall x_{d-1}\psi) &= \exists x_d \mathcal{R}(\forall x_{d-1}\psi), \\
\mathcal{R}(\forall x_{d-1} \exists x_{d-2}\psi) &= \forall x_{d-1} \mathcal{R}(\exists x_{d-2}\psi), \quad \text{and} \\
\mathcal{R}(\forall x_{d-1} \forall x_{d-3}) &= \forall x_{d-1} \exists x_{d-2} [(\bar{x}_{d-2} \wedge \tau_{d-3}) \vee (x_{d-2} \wedge \mathcal{R}(\forall x_{d-3}\psi))].
\end{aligned}$$

We claim that the logic game on $\mathcal{R}(\eta)$ is equivalent to that on η and retains global uniqueness. First consider the case that E-player is to make the next move from η. Suppose that E-player wins from ψ, and consider the first two moves. If E-player makes the right choice for x_d, then U-player with the inserted turn x_{d-1} has only the choice between continuing the game on η or entering the trivial unique game τ_{d-1} that E-player wins. If E-player makes the wrong choice for x_d, then U-player can punish this only by avoiding τ_{d-2} and continuing with η by setting $x_{d-1} = 1$. Now suppose that E-player loses from η. Even this case, U-player must still respond uniquely to either choice for x_d by setting $x_{d-1} = 1$.

Consider the case where η starts with a universal quantifier on a variable x_{d-1}, which means that it is U-player's turn. Suppose again that E-player wins from η. Then E-player must set $x_{d-2} = 1$ (no matter how x_{d-1} is assigned) to avoid the trivial unique game position t_{d-3}. On the other hand, if U-player wins from η, E-player must always set $x_{d-2} = 1$ to retain his chances.

Now we must transform $\mathcal{R}(\phi)$ into a prenex formula. We claim that we can simply move all quantifiers in $\mathcal{R}(\phi)$ to the far left without changing the semantics! This is because the invariant that existentially quantified variables have even index and the others have odd index is maintained throughout the computation of $\mathcal{R}(\phi)$, and because the inserted variables (such as x_{d-1} in the $\exists\exists$ case) enforce a separation that offsets the change in scope. $\square$

This facilitates our proof of one of the main results of this section.

Theorem 4.7. Consider any polynomially definable game $G = (P_0, P_1, R)$ with the global uniqueness property. Suppose that its maximal depth function $d_{\max}$ is polynomially bounded; that is, with some polynomial p, we have $d_{\max}(\pi) \leq p(|\pi|)$ for any $\pi \in P_0 \cup P_1$. Then we can define a polynomial-time computable procedure $\mathcal{RED}$ such that given position $\pi \in P_0 \cup P_1$, the output $\mathcal{RED}(\pi)$ is a formula ϕ in PQBF with global uniqueness such that ϕ is true if and only if π is a winning position for P_0.

Proof. We can modify the game G, while keeping global uniqueness, to a new polynomially definable game $G' = (P_0', P_1', R')$ so that (i) every original game position $\pi \in P_0 \cup P_1$ has some corresponding position (which is denoted by $[\pi]$) in G', and (ii) the game starting from any such position $[\pi]$ always terminates at the $2p(|\pi|)$th step with some special constant positions either π_0 or π_1, where this final position is π_0 (resp., π_1) if and only if π is a winning position for player 0 (resp., player 1) in the original game. Furthermore, we may assume that every position appearing in the game tree under $[\pi]$ is represented by a binary string of length $p(|\pi|)$ for some polynomial q. (Since the modified game (starting from $[\pi]$) terminates at even steps, *by definition* player 1 always wins from $[\pi]$. But here we want to keep the original win/loss criteria, and so, we will determine the win/loss by the final position. The global uniqueness property is defined under this win/loss criteria.)

Let us consider some game position π in G, and let $n = q(\pi)$ and $d = 2p(\pi)$. We use a binary vector $\boldsymbol{x} = x_1, ..., x_n$ to represent a game position in the game tree rooted $[\pi]$. Since R' is polynomial-time computable, by the standard technique, we can define a Boolean formula *DELTA* such that $\exists \boldsymbol{w} \mathit{DELTA}(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{w})$ holds if and only if a position $\boldsymbol{z}$ follows from $\boldsymbol{y}$ in G'. Furthermore, we may assume that the choice of $\boldsymbol{w}$ for having $\mathit{DELTA}(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{w}) = 1$ is unique if it indeed exists. Also we define a Boolean formula *WIN0* such that $\mathit{WIN0}(\boldsymbol{x})$ holds if and only if $\boldsymbol{x}$ is π_0.

Now we define, for our given π, a Boolean formula F and its quantified version ψ. Variables of F are

$$\begin{array}{lllll} x_1^m, \dots, x_n^m & , \; x_1^{m-1}, \dots, x_n^{m-1} & , \dots , & x_1^1, \dots, x_n^1, & \text{and} \\ w_1^m, \dots, w_n^m & , \; w_1^{m-1}, \dots, w_n^{m-1} & , \dots , & w_1^1, \dots, w_n^1. & \end{array}$$

Intuitively, variables $\boldsymbol{x}^{m-i+1} = x_1^{m-i+1}, \dots, x_n^{m-i+1}$ are used to encode a game position at the ith steps. For any $j \le m-1$, we use a formula $\boldsymbol{t}_j$ that is defined in the same way as t_j by substituting every x_i, $1 \le i \le j$, with $x_1^i \vee \cdots \vee x_n^i \vee w_1^i \vee \cdots \vee w_n^i$ if i is odd, and with $x_1^i \wedge \cdots \wedge x_n^i \wedge w_1^i \wedge \cdots \wedge w_n^i$ if i is even. For example,

$$\boldsymbol{t}_3 \;=\; (x_1^1 \vee \cdots \vee x_n^1 \vee w_1^1 \vee \cdots \vee w_n^1) \vee (x_1^2 \wedge \cdots \wedge x_n^2 \wedge w_1^2 \wedge \cdots \wedge w_n^2)(x_1^3 \vee \cdots \vee x_n^3 \vee w_1^3 \vee \cdots \vee w_n^3).$$

Then our F is defined as follows.

$$\begin{aligned} F = {} & [\,\neg \mathit{DELTA}(\boldsymbol{x}^{m+1}, \boldsymbol{x}^m) \;\wedge\; \boldsymbol{t}_{m-1}\,] \\ & \vee\; [\,\mathit{DELTA}(\boldsymbol{x}^{m+1}, \boldsymbol{x}^m) \;\wedge\; \neg \mathit{DELTA}(\boldsymbol{x}^m, \boldsymbol{x}^{m-1}) \;\wedge\; \boldsymbol{t}_{m-2}\,] \\ & \vee\; [\,\mathit{DELTA}(\boldsymbol{x}^{m+1}, \boldsymbol{x}^m) \;\wedge\; \mathit{DELTA}(\boldsymbol{x}^{m-1}, \boldsymbol{x}^{m-2}) \;\wedge\; \neg \mathit{DELTA}(\boldsymbol{x}^{m-2}, \boldsymbol{x}^{m-3}) \;\wedge\; \boldsymbol{t}_{m-3}\,] \\ & \vdots \\ & \vee\; \left[\left(\bigwedge_{i=m}^{1} \mathit{DELTA}(\boldsymbol{x}^{i+1}, \boldsymbol{x}^i) \right) \;\wedge\; \mathit{WIN0}(\boldsymbol{x}^1) \right]. \end{aligned}$$

Here we use the symbol $\boldsymbol{x}^{m+1}$ for simplifying the expression, it is a binary representation of $[\pi]$ that is a constant vector in F. A quantified formula ψ is obtained from F by simply quantifying variables with even superscripts existentially and variables with odd superscripts universally in the order of their superscripts. That is, $\phi = \exists x_1^m \, \exists x_2^m \, \cdots \, F$.

Our definition of F is based on the following interpretation: If E-player tries to cheat by setting the values of the existentially-quantified variables $\boldsymbol{x}^m$ to form a non-legal game position or he fails to give a unique witness for $\boldsymbol{w}^m$, then E-player loses uniquely according to the trivial game $\boldsymbol{t}_{m-1}$ on the remaining variables. If E-player give correct assignments to $\boldsymbol{x}^m$ and $\boldsymbol{w}^m$, then U-player must give correct assignments to $\boldsymbol{x}^{m-1}$ and $\boldsymbol{w}^{m-1}$ on pain of otherwise losing the trivial global unique game $\boldsymbol{t}_{m-2}$. This compulsion holds all the way down until the end.

As in the last proof, we apply the padding of Lemma 4.6 between adjacent like quantifiers in ψ to obtain a final Boolean formula ϕ in PQBF. Clearly, the whole process can be done within polynomial-time in $|\pi|$. $\square$

Corollary 4.8. The class UAP coincides with the class of languages that polynomial-time many-one reduce to GUQBF. $\square$

Now is finally the place to rejoin the remarks about the globally-unique extension "G^{**}" of an arbitrary game G that were made at the outset of this section. Picture G^{**} as being played "on" the game tree of possible plays from a position π_0 of G. At any time in G^{**}, a player can revisit a position π along the *current branch of play*, and play a move that is *more leftward* than the move made there (by either player) on the previous visit. This goes down a new branch of play from π. Thus any play of G^{**} sweeps out a subtree of the total game tree of G below π_0 from right to left. It is not hard to see that G^{**} has the global uniqueness property: the unique winning move is always the leftmost winning move in G from the highest position along the current branch of play that has a more-leftward winning move than the move last taken (if any).

In one sense, optimal play in G^{**} is no deeper than optimal play in G, since it involves selecting the leftmost winning move in G whwnever a win exists. It is not quite true that all optimal plays in G^{**} obey the same length bound as in G, as the opponent can delay losing exponentially long in a "silly" way by revisiting earlier positions and making more-leftward but losing moves. What difference do these "silly" plays make? Precisely put, they upset the counting mechanism in Fact 1 preceding Theorem 4.4. If QBF** had polynomial depth—or if long non-optimal plays could be canceled out of the counting—then PSPACE would collapse to SPP, which is widely disbelieved.

5 Conclusions and Open Problems

We have formalized and developed several concepts of unique solvability for games and game positions, extending natural notions such as "study" puzzles and Nim-style games. We have proved that many combinatorial games retain their hardness under strong conditions of unique solvability, but evidently not when uniqueness applies "globally" to every reachable position in the game. The class of languages reducible to game-decision problems with global uniqueness, i.e. reducible to GUQBF, is a natural analogue of UP for alternating classes. It is located fairly precisely between FewP and SPP [NR98]. (For more on these complexity-class environs, see [FFK94, FFL96].)

The main structural-complexity open problems are whether UAP = SPP, and whether the property of global uniqueness itself (of an ATM or of a quantified Boolean formula) has complexity equivalent to UAP. Does the promise problem GU(P)QBF have solutions in UAP? The self-reducibility structure of these problems also seems interesting. Perhaps most notably, however, can we say more about the concrete classes "A_d" and "$B - d$" of Boolean formulas defined in the last section? Our recursive definition of these formulas produces representatives of exponential size, but we showed that *some* of them have equivalent formulas of linear size (in d). Can we characterize formulas of polynomial size that are equivalent to ones in A_d or B_d, and/or relate questions about these formulas to complexity questions about Boolean formulas in general?

Finally, is there any larger mathematical significance of our uniqueness concepts, the strongest one (i.e., global uniqueness) in particular? Perhaps it can be shown to imply *unitarity* of some transition/value matrix associated to plays of the game, and thence to relate to quantum complexity classes—some of whom have been characterized in the same rough neighborhood of counting classes, as e.g. in [FR98]. An equivalence between UAP and a quantum complexity class might have the pleasing theological interpretation that exactly when we think God is feasibly "rolling dice," He is equivalently following a unique Plan.

Acknowledgments We thank Steve Fenner for helpful conversations at Complexity'02, in particular for the characterization of SPP. We thank Christian Glasser and Samik Sengupta for further ideas on global uniqueness and for bringing Niedermeier and Rossmanith [NR98] to our attention in November 2002.

References

[AT01] S. Aida and T. Tsukiji, Complexity of two person games with unique winning strategy, Technical Report of IEICE, COMP2000-87, March 2001.

[CS79] A.K. Chandra and L.J. Stockmayer, Provably difficult combinatorial games, *SIAM J. Comput.* **8** (1979), 151–174.

[CT00] M. Crasmaru and J. Tromp, Ladders are PSPACE-complete, in *Proceedings of the 2nd Int'l Conference on Computers and Games*, Lecture Notes in Computer Science **2063** (2000), pp 253–263.

[Cra01] M. Crasmaru, *PSPACE versus EXP-TIME and the game of GO*, Master Thesis, Tokyo Institute of Technology, March 2001.

[FFK94] S. Fenner, L. Fortnow, and S. Kurtz, Gap-definable counting classes, *J. Comp. Sys. Sci.* **48** (1994), 116–148.

[FFL96] S. Fenner, L. Fortnow, and L. Li, Gap-definability as a closure property, *Inform. and Comp.* **130** (1996), 1–17.

[Fen02] S. Fenner, personal communication, May 2002.

[FR98] L. Fortnow and J. Rogers, Complexity limitations on quantum computation, in *Proceedings of the 13th Annual IEEE Conference on Computational Complexity*, 1998, pp 202–206.

[NR98] R. Niedermeier and P. Rossmanith, Unambiguous computations and locally definable acceptance types, *Theor. Comp. Sci.* **194** (1998), 137–161.

[Rob84] J.M. Robson, Combinatorial games with exponential space complete decision problems, in *Proc. of Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science **176** (1984), pp 498–506.

[Sav72] W.J. Savitch, Relationship between nondeterministic and deterministic tape classes, *J. Comput. Syst. Sci.* **4** (1970), 177–192.

[Sel88] A. Selman, Promise problems complete for complexity classes, *Information and Computation* **78** (1988), 87–98.

[Val86] L.G. Valiant and V.V. Vazirani, NP is as eay as detecting unique solutions, *Theoret. Comput. Sci.* **47** (1986), 85–93.