ON THE COMPUTATIONAL COMPLEXITY OF THE OPTIMAL YOSE MOVE IN GO VIA COMBINATORIAL GAME THEORY

LIGHTMAN CHANG

Abstract. We study the complexity of computing an optimal endgame (*yose*) move in $n \times n$ Go under three formal readings of "optimal": maximum combinatorial game theory (CGT) temperature ($X = B$); maximum swing LS − RS ($X = C$); and maximum (or minimum) value-of-being-forced-to-play under optimal sequel ($X = D$-must, $X = D$-avoid). Working within the local decomposition framework of Berlekamp and Wolfe, we prove that for every $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$ the problem Best-Move-X is PSPACE-hard, by a reduction from the Generalized Ladder Game (Crâşmaru–Tromp 2000). The reduction is calibrated by an explicit *swing-booster gadget* (Lemma 3.2) that yields a polynomial gap $D_Y - D_N = \Omega(|B|)$ between yes-instances and no-instances. We complement the lower bound with a matched algorithmic upper bound: under the standard Berlekamp–Wolfe shape promise, ko-freeness, and local decomposability, Best-Move-X admits an $O(n^2 + m \log m)$ algorithm based on closed-form region values and an $O(m)$ prefix-sum evaluation of the optimal-sequel scores. The two results together identify the precise barrier between tractable and intractable yose, and clarify which formalizations of "best move" are algorithmically equivalent.

1. Introduction

The closing phase of a game of Go—the *yose*—admits a natural formalization within combinatorial game theory (CGT), where the board decomposes into disjoint local games whose values are added under the disjunctive sum operation [3, 1]. In the classical Berlekamp–Wolfe theory of cooled Go positions [2], every "standard" shape (corridor, simple switch, sente sequence, etc.) admits explicit polynomial-time stops and temperature, and the player who maximizes the global score is the player who maximizes a known function of these local values. This raises a basic complexity question:

> *Given a Go position in the yose phase, how hard is it to compute the local move that is optimal under a precisely specified definition of "best"?*

Three definitions of "best move" are standard in the CGT and Go literature, and they need not coincide:

(B) *Hottest move*: the local game R_i with the largest CGT temperature $\mathrm{T}(R_i)$.
(C) *Largest swing*: the local game with the largest swing $\Delta(R_i) = \mathrm{LS}(R_i) - \mathrm{RS}(R_i)$.
(D) *Win-deciding move*: the local game whose forced selection most affects the final score under optimal subsequent play.

Definition (D) admits at least three precise readings (must-play, avoid-play, sign-flipping locus); see Definition 2.3. Throughout the paper we restrict attention to the must-play and avoid-play readings; the sign-flipping reading involves a self-referential baseline V_0 and is left open.

Background. Robson [5] proved that $n \times n$ Go is EXPTIME-complete using nested ko constructions. This bound is *global*: the entire board, with all the complications of ko and life-and-death, is encoded into a single instance. For the *yose* phase, ko has typically been resolved and the position admits a CGT decomposition. In this restricted regime, Crâşmaru and Tromp [4] showed that even the single-region problem of deciding the outcome of a Go ladder is PSPACE-complete,

Date: May 5, 2026.
2020 *Mathematics Subject Classification.* Primary 91A46; Secondary 68Q17, 91A05.
Key words and phrases. Combinatorial game theory, Go endgame, computational complexity, PSPACE-hardness, thermograph, Berlekamp–Wolfe theory.

by encoding QBF into the ladder dynamics. Their construction is a natural building block for a complexity theory of the *yose* that lies strictly below the $\mathsf{EXPTIME}$ bound of Robson.

On the upper-bound side, Berlekamp and Wolfe [2] catalogued a family of standard shapes whose CGT canonical form is computable by closed formulas. Modulo the catalogue and the disjunctive-sum law of Conway, this yields polynomial-time algorithms for (B) and (C) on positions composed entirely of standard shapes.

The gap. Two questions remain unresolved in this picture. First, under the *generic* local-decomposition promise (no shape restriction), is computing the optimal yose move actually as hard as deciding QBF? Second, do the three definitions (B), (C), (D) produce the same top-1 choice on standard shapes, and if not, where do they diverge?

Our contribution. We address the first question for all four readings $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$. Our main result (Theorem 3.1) is:

Theorem (Informal)**.** *For every $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$, the problem* Best-Move-X *is* PSPACE*-hard, even under the local-decomposition promise. Conversely, under the additional Berlekamp–Wolfe standard-shape and ko-free promises,* Best-Move-X *admits an $O(n^2 + m \log m)$ algorithm, where n is the board side and m is the number of regions.*

The key technical step in the lower bound is a *swing-control lemma* (Lemma 3.2): given any Generalized Ladder Game instance (B, s_0), we construct in polynomial time a region R_Φ with an explicit *escape booster* and a *capture booster* gadget, such that

$$\mathsf{GLG}(B, s_0) = \text{YES} \iff \Delta(R_\Phi) = D_Y \geq 6k, \qquad \mathsf{GLG}(B, s_0) = \text{NO} \iff \Delta(R_\Phi) = D_N \leq 4,$$

where $k = \text{poly}(|B|)$ is the chase length of the ladder. The polynomial gap $D_Y - D_N = \Omega(|B|)$ closes a calibration gap left in prior informal arguments and allows us to instantiate a single reduction template for all four definitions of "best move".

The matching upper bound (Theorem 4.3) reduces optimal-sequel scores to a closed-form prefix-sum expression over the swing-sorted permutation of regions, giving $O(m)$ work per region after sorting.

Comparison with prior work. Robson's $\mathsf{EXPTIME}$-completeness theorem [5] operates on the unrestricted Go-position problem and uses ko, while our lower bound holds in the ko-free yose regime—hence our bound is strictly weaker (PSPACE vs. $\mathsf{EXPTIME}$) but applies to a strictly more restricted class of inputs. The Crâşmaru–Tromp result [4] established PSPACE-completeness for a single ladder, but did not address the comparison-of-regions structure that distinguishes "best move" problems from outcome problems. Our swing-booster construction (Lemma 3.2) and the dominance analysis of its CGT canonical form (Lemma 3.3) are new.

Paper structure. Section 2 fixes CGT notation and defines the four problems Best-Move-X. Section 3 states and proves the swing-control lemma and the PSPACE-hardness theorem. Section 4 gives the polynomial-time algorithm under the standard-shape promise. Section 5 discusses the gap between the two bounds and open problems.

2. Preliminaries

2.1. **Combinatorial game values.** We work throughout in the framework of Conway [3] and Berlekamp–Conway–Guy [1]. A *game* G is recursively defined by a set of *Left options* G^L and *Right options* G^R. For a game G, the *Left stop* and *Right stop* are

$$\text{LS}(G) = \max_{G^L \in \mathcal{G}^L} \text{RS}(G^L), \qquad \text{RS}(G) = \min_{G^R \in \mathcal{G}^R} \text{LS}(G^R),$$

with the base case $\text{LS}(G) = \text{RS}(G) = G$ when G is a (dyadic rational) number. The *swing* of G is $\Delta(G) = \text{LS}(G) - \text{RS}(G) \geq 0$, its *mean* is $\mu(G) = (\text{LS}(G) + \text{RS}(G))/2$, and its *temperature* $\text{T}(G)$ is the height at which the two scaffolds of the thermograph of G meet [1, Ch. 6]. For a simple switch $S = \{a \mid b\}$ with $a > b$ both numbers, $\text{T}(S) = (a - b)/2$ and $\mu(S) = (a + b)/2$.

2.2. **Local decomposition and the yose phase.**

Definition 2.1 (Local decomposability)**.** A Go position s on an $n \times n$ board is *locally decomposable* if its set of empty intersections partitions into disjoint regions $\{R_1, \dots, R_m\}$ such that

(i) for every legal play sequence, the liberties and stones inside R_i depend only on play inside R_i;
(ii) the global Japanese-rules score satisfies $V(s) = \sum_{i=1}^{m} V_i(R_i)$;
(iii) each R_i defines a well-formed CGT subgame.

Definition 2.2 (Ko-free promise)**.** A region R_i is *ko-free* if its local game tree contains no node whose subboard (restricted to R_i together with its boundary) recurs at any descendant. This is strictly stronger than the simple ko rule of Japanese rules.

We assume throughout that the input is locally decomposable (a promise). The yose phase is, by definition, the regime in which this promise holds.

2.3. Definitions of optimal move.

Definition 2.3 (Three readings of "best move")**.** Let s have decomposition $\{R_i\}_{i=1}^{m}$. Let

$$V_j \;=\; \mathrm{RS}\Big(R_j^{L^*} + \sum_{i \neq j} R_i\Big), \quad \text{where } R_j^{L^*} = \arg\max_{R_j^L} \mathrm{RS}\Big(R_j^L + \sum_{i \neq j} R_i\Big).$$

That is, V_j is the final score after Black is forced to play in R_j and both sides play optimally thereafter. We define:

(D-must) $j^* = \arg\max_j V_j$. "The move Black should play to maximize the final score."
(D-avoid) $j^* = \arg\min_j V_j$. "The move Black must not play."
(D-sign) Order regions by $|V_j - V_0|$, where $V_0 = \mathrm{LS}(\sum_i R_i)$ is the unconstrained optimal-play value.

Definition 2.4 (The problems Best-Move-X)**.** For $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$, the problem Best-Move-X takes as input a locally decomposable position s and an integer $k \geq 1$, and outputs the top-k regions ranked by

$$X = B:\ \mathrm{T}(R_i) \text{ descending}; \qquad X = C:\ \Delta(R_i) \text{ descending};$$

$$X = D\text{-must}:\ V_i \text{ descending}; \qquad X = D\text{-avoid}:\ V_i \text{ ascending}.$$

The sign-flipping reading (D-sign) requires the baseline V_0, which is itself the answer to a Best-Move-X-style optimization, leading to a circular dependence we do not resolve here; see Section 5.

2.4. Crâşmaru–Tromp Generalized Ladder Game.

We rely on the following theorem.

Theorem 2.5 (Crâşmaru–Tromp [4])**.** *The Generalized Ladder Game (*GLG*) is* PSPACE*-complete: given a rectangular Go region B (enclosed by a 2-stone-wide live wall) with a chaser stone, an escaper stone, and obstruction stones at fixed positions, deciding whether the escaper has a winning strategy under the standard ladder dynamics is* PSPACE*-complete.*

The proof reduces from QBF, the prototypical PSPACE-complete problem [6]: each quantifier alternation is encoded as a turn in the ladder where the current player's choice corresponds to a variable assignment; obstructions enforce the "wrong choice ⇒ forced capture" structure. The complete construction occupies the 30 pages of [4]; we use it as a black box. The overall reduction structure is in the spirit of the polynomial-time many-one reductions of [7].

3. Lower Bound: PSPACE-Hardness

The main lower bound is the following theorem; the heart of the argument is the swing-control lemma below.

Theorem 3.1. *For every $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$, the problem* Best-Move-X *is* PSPACE*-hard, even under the local-decomposition promise.*

3.1. Swing-control via booster gadgets.

Lemma 3.2 (Swing-control lemma). *There is a polynomial-time algorithm that, given any* GLG *instance* (B, s_0) *with chase length* $k = \mathrm{poly}(|B|)$*, constructs a Go region* R_Φ *embedded inside a 2-stone-wide live white wall, such that*

$$\mathsf{GLG}(B, s_0) = \text{YES} \implies \text{LS}(R_\Phi) = 4k,\ \text{RS}(R_\Phi) = -2k,\ \Delta(R_\Phi) =: D_Y = 6k,$$

$$\mathsf{GLG}(B, s_0) = \text{NO} \implies \text{LS}(R_\Phi) = -2k + \varepsilon,\ \text{RS}(R_\Phi) = -2k - \varepsilon',\ \Delta(R_\Phi) =: D_N = \varepsilon + \varepsilon' \le 4,$$

where $\varepsilon, \varepsilon' \in [0, 2]$*. In particular,* $D_Y - D_N \ge 6k - 4 = \Omega(|B|)$*.*

Proof. We give an explicit construction. The region R_Φ embeds the ladder block B with its obstruction stones, plus two booster gadgets attached at the two distinguished endpoints of the ladder dynamics:

- *Escape booster*: a $1 \times 2k$ corridor attached at the "successful escape" terminal of the ladder. If the escaper succeeds, Black additionally collects $2k$ points by occupying the corridor in a forced sente sequence (standard corridor evaluation, cf. [2, Ch. 3]).
- *Capture booster*: a small dead shape (e.g., a 4-stone "one-eye" configuration) attached at the "failed escape" terminal. Both colors play to no effect inside this gadget, contributing LS = RS = 0.

The total size of the boosters is $O(k)$, which fits inside the 2-stone-wide white wall enclosure of R_Φ for sufficiently large n, and remains decoupled from the rest of the board.

Computation of $\text{LS}(R_\Phi)$ **and** $\text{RS}(R_\Phi)$**.** Let k be the chase length of the ladder.

Case GLG = YES*.* If Black plays first in R_Φ, she initiates the chase, escapes successfully (saving k stones, $+2k$ points), and then collects the escape booster ($+2k$ points), giving $\text{LS}(R_\Phi) = 4k$. If White plays first, the chase forces the capture of Black's k stones ($-2k$ points) and the boosters are inert, giving $\text{RS}(R_\Phi) = -2k$.

Case GLG = NO*.* If Black plays first, she still attempts the chase but the escape fails; the forced sequence captures her k stones, with a small move-parity correction $\varepsilon \in [0, 2]$ from the boundary of the chase. So $\text{LS}(R_\Phi) = -2k+\varepsilon$. Symmetrically, $\text{RS}(R_\Phi) = -2k-\varepsilon'$ with $\varepsilon' \in [0, 2]$.

The values $\varepsilon, \varepsilon'$ are fully determined by the ladder dynamics on the boundary stones and can be computed in polynomial time from (B, s_0). In particular, $D_N = \varepsilon + \varepsilon' \le 4$. □

Lemma 3.3 (CGT canonical form of R_Φ). *Under the construction of Lemma 3.2, the canonical form of* R_Φ *in the sense of* [1, Ch. 4] *is the simple switch* $\{\text{LS}(R_\Phi) \mid \text{RS}(R_\Phi)\}$*.*

Proof. We show that all Left options of R_Φ except the "initiate-chase" option are dominated, and similarly for Right options.

The Left options of R_Φ are:

(L_1) Initiate the ladder chase. Final value: $4k$ (YES case), $-2k + \varepsilon$ (NO case).

(L_2) Play directly inside the escape booster corridor. The corridor is sealed by a live white wall at this point in the game tree, so a Black stone there has no liberties and is immediately captured. The chase still proceeds and is forced by White. Final value $\le -2k - 1$.

(L_3) Play directly inside the capture-booster dead shape. The shape is dead, so the stone is wasted, and White still drives the chase. Final value $= -2k$.

In the YES case, (L_1) strictly dominates: $4k$ versus $\le -2k - 1$ and $-2k$. In the NO case, (L_1) achieves $-2k + \varepsilon \ge -2k$, which weakly dominates (L_3) and strictly dominates (L_2). The canonical-form algorithm of [1, Ch. 4] eliminates dominated options under the weak inequality $\ge$, so $(L_2), (L_3)$ are removed and only (L_1) remains.

By symmetry (the Right options (R_1), (R_2), (R_3) are analogous), only the symmetric "initiate-chase" option (R_1) survives among Right options. Each surviving option's value is a number, so the canonical form is the simple switch $\{\text{LS}(R_\Phi) \mid \text{RS}(R_\Phi)\}$. □

3.2. Reduction template.

Fix a GLG instance (B, s_0) and apply Lemma 3.2 to obtain R_Φ with parameters D_Y, D_N. Let $a := (D_Y + D_N)/4$, which is computable in polynomial time, and define a second region R_* to be a simple switch $\{a \mid -a\}$ (realizable in Go as a standard $1 \times \ell$

corridor with ℓ chosen so that $\Delta(R_*) = 2a$; see [2, Ch. 3] for the catalogue). Both R_Φ and R_* sit inside disjoint, sealed sub-boards, so the two-region position is locally decomposable in the sense of Definition 2.1.

Proof of Theorem 3.1. We show that for each X, an oracle for Best-Move-X on the two-region instance $\{R_\Phi, R_*\}$ decides GLG.

Case $X = B$ (temperature). Since R_Φ has canonical form $\{\mathrm{LS}(R_\Phi) \mid \mathrm{RS}(R_\Phi)\}$ by Lemma 3.3, we have $\mathrm{T}(R_\Phi) = \Delta(R_\Phi)/2$. Also $\mathrm{T}(R_*) = a$. Hence

$$\mathrm{T}(R_\Phi) > \mathrm{T}(R_*) \iff \Delta(R_\Phi) > 2a \iff D_Y > 2a > D_N \iff \mathsf{GLG} = \mathrm{YES}.$$

The middle equivalence uses $a = (D_Y + D_N)/4$, which lies strictly between $D_N/2$ and $D_Y/2$ since $D_Y > D_N$ (Lemma 3.2).

Case $X = C$ (swing). Identical to $X = B$ since $\Delta(R_\Phi) > \Delta(R_*) = 2a$ iff $\mathsf{GLG} = \mathrm{YES}$.

Case $X = D$-must. For the two-region instance,

$$V_{R_\Phi} = \mathrm{LS}(R_\Phi) + \mathrm{RS}(R_*) = \mathrm{LS}(R_\Phi) - a,$$
$$V_{R_*} = \mathrm{LS}(R_*) + \mathrm{RS}(R_\Phi) = a + \mathrm{RS}(R_\Phi).$$

Hence

$$V_{R_\Phi} - V_{R_*} = \big(\mathrm{LS}(R_\Phi) - \mathrm{RS}(R_\Phi)\big) - 2a = \Delta(R_\Phi) - 2a,$$

which is positive iff $\mathsf{GLG} = \mathrm{YES}$.

Case $X = D$-avoid. The same calculation gives $V_{R_\Phi} < V_{R_*} \iff \Delta(R_\Phi) < 2a \iff \mathsf{GLG} = \mathrm{NO}$, so the avoid-play top-1 identifies R_Φ iff $\mathsf{GLG} = \mathrm{NO}$.

In all four cases the reduction is polynomial in $|B|$ and decides GLG. By Theorem 2.5, Best-Move-X is PSPACE-hard. ☐

4. Upper Bound under the Standard-Shape Promise

We now show that the lower bound is essentially tight: under a shape-restriction promise that excludes the ladder constructions of Section 3, Best-Move-X becomes polynomial.

Definition 4.1 (Berlekamp–Wolfe standard shape)**.** A region R is a *Berlekamp–Wolfe standard shape* if its CGT canonical form is a finite disjunctive sum of the following components:

(a) a number (a region with no hot move);
(b) a simple switch $\{a \mid b\}$ with a, b numbers and $a > b$;
(c) a corridor of length ℓ with standard live-stone closure [2, Ch. 3];
(d) a finite sente sequence terminating in (a) or (b);
(e) a dyadic infinitesimal in $\{0, *, \uparrow, \downarrow, \uparrow *, \cdot 2, \dots\}$.

For each component type, closed-form values $(\mathrm{LS}, \mathrm{RS}, \mathrm{T}, \mu)$ and an optimal move are computable in $O(|R|)$ by standard formulas; the infinitesimal components contribute $\mathrm{LS} = \mathrm{RS} = 0$ and do not affect top-k rankings of hot regions. We refer to the full procedure as BWFormula(R).

4.1. The hotstrat-on-switches lemma.

A key combinatorial step is the following identity, which expresses the optimal-sequel value of a sum of simple switches in closed form.

Lemma 4.2 (Hotstrat on simple switches)**.** *Let $G = \sum_{i=1}^{m} S_i$ with $S_i = \{a_i \mid b_i\}$ simple switches and $\Delta_i = a_i - b_i > 0$. Sort the indices in descending order of Δ: $\Delta_{\sigma(1)} \geq \cdots \geq \Delta_{\sigma(m)}$. Then*

(a) under optimal play with Black to move, Black's first move is in $S_{\sigma(1)}$;
(b) the entire optimal play alternates by selecting the largest remaining Δ, with Black taking $a_{\sigma(l)}$ at odd positions l and White taking $b_{\sigma(l)}$ at even positions;
(c) $\mathrm{LS}(G) = \sum_{l \text{ odd}} a_{\sigma(l)} + \sum_{l \text{ even}} b_{\sigma(l)}$, and the symmetric formula holds for $\mathrm{RS}(G)$.

Proof. We argue by an adjacent-swap exchange. View the play as a sequence $\pi : [m] \to [m]$ that selects the $\pi(l)$-th switch at position l, with Black at odd positions and White at even positions. The score is $\mathrm{Score}(\pi) = \sum_{l \text{ odd}} a_{\pi(l)} + \sum_{l \text{ even}} b_{\pi(l)}$. For any $l \in [1, m-1]$, swapping positions l and $l+1$ yields

$$\mathrm{Score}(\pi) - \mathrm{Score}(\pi') = \pm\big(\Delta_{\pi(l)} - \Delta_{\pi(l+1)}\big),$$

with sign + when l is odd (Black's choice) and − when l is even (White's choice). At a saddle point the controlling player gains nothing by swapping, which forces $\Delta_{\pi^*(l)} \geq \Delta_{\pi^*(l+1)}$ in both cases. Hence $\pi^* = \sigma$, and the formulas in (a)–(c) follow directly. □

4.2. The polynomial-time algorithm.

Theorem 4.3. *Suppose the input position s on an $n \times n$ board is locally decomposable, ko-free, and every region is a Berlekamp–Wolfe standard shape. Then for every $X \in \{B, C, D\text{-must}, D\text{-avoid}\}$ the problem* Best-Move-X *is solvable in time $O(n^2 + m \log m)$.*

Proof. We give algorithms for each X.

Algorithm for $X = B$.

1. Decompose s into $\{R_1, \ldots, R_m\}$ via BFS in $O(n^2)$.
2. For each R_i, compute $(\mathrm{LS}_i, \mathrm{RS}_i, \mathrm{T}_i, m_i^*) \leftarrow \text{BWFormula}(R_i)$ in $O(|R_i|)$ total $O(n^2)$.
3. Sort regions by T_i in descending order; output top k.

The sort costs $O(m \log m)$. Total: $O(n^2 + m \log m)$.

Algorithm for $X = C$. Same as $X = B$ but sort by $\Delta_i = \mathrm{LS}_i - \mathrm{RS}_i$.

Algorithm for $X = D$-must. We use Lemma 4.2. Let σ be the descending order of Δ over the hot regions, and write $a_l = \mathrm{LS}_{\sigma(l)}$, $b_l = \mathrm{RS}_{\sigma(l)}$. By the lemma applied to "Black is forced to play in region $\sigma(p)$ first", the optimal-sequel total is

$$V_{\sigma(p)} = a_p + \sum_{\substack{l \leq p-1 \\ l \text{ even}}} a_l + \sum_{\substack{l \leq p-1 \\ l \text{ odd}}} b_l + \sum_{\substack{l \geq p+1 \\ l \text{ odd}}} a_l + \sum_{\substack{l \geq p+1 \\ l \text{ even}}} b_l.$$

Precompute the four prefix/suffix arrays

$$P_a^{\text{ev}}[p] = \sum_{l \leq p,\, l \text{ even}} a_l, \quad P_b^{\text{od}}[p] = \sum_{l \leq p,\, l \text{ odd}} b_l, \quad S_a^{\text{od}}[p] = \sum_{l \geq p,\, l \text{ odd}} a_l, \quad S_b^{\text{ev}}[p] = \sum_{l \geq p,\, l \text{ even}} b_l,$$

in $O(m)$ by single sweeps. Then each $V_{\sigma(p)}$ is computable in $O(1)$ via

$$V_{\sigma(p)} = a_p + P_a^{\text{ev}}[p-1] + P_b^{\text{od}}[p-1] + S_a^{\text{od}}[p+1] + S_b^{\text{ev}}[p+1],$$

giving $O(m)$ total over all regions, and a final $O(m \log m)$ sort.

Algorithm for $X = D$-avoid. Same as $X = D$-must but sort $V_{\sigma(p)}$ in ascending order.

In each case, the total time is $O(n^2 + m \log m)$. □

4.3. Numerical sanity checks.

We verify the prefix-sum formula on small m.

Example 4.4. Take $m = 3$ with $\Delta = (10, 8, 6)$ and $\mu_i = 0$, so $(a_l, b_l) = (5, -5), (4, -4), (3, -3)$. Then $P_a^{\text{ev}} = [0, 0, 4, 4]$, $P_b^{\text{od}} = [0, -5, -5, -8]$, $S_a^{\text{od}} = [0, 8, 3, 3, 0]$ (indices 0–4), $S_b^{\text{ev}} = [0, -4, -4, 0, 0]$. Direct evaluation gives $V_1 = 5+0+0+3-4 = 4$, $V_2 = 4+0-5+3+0 = 2$, $V_3 = 3+4-5+0+0 = 2$, matching the closed-form computation.

Example 4.5. Take $m = 4$ with $\Delta = (10, 8, 6, 4)$ and $\mu_i = 0$. The same formula gives $V_1 = 2$, $V_2 = V_3 = 0$, $V_4 = -2$, again matching the recursive evaluation. Note the tie $V_2 = V_3$, foreshadowing the structural-ties phenomenon studied in the companion paper.

5. Discussion

5.1. The gap between the bounds. Combining Theorems 3.1 and 4.3, we have a clean dichotomy: the polynomial-time tractability of Best-Move-X is governed precisely by the Berlekamp–Wolfe standard-shape promise. Without the shape promise, the embedding of a ladder into a single region suffices to make Best-Move-X as hard as deciding QBF.

An intermediate regime is also of interest. If we permit arbitrary regions but require $|R_i| = O(\log n)$, then a brute-force traversal of each local game tree gives an upper bound of $n^{O((\log n)^2)}$—quasi-polynomial. The condition $|R_i| = O(\log n)$ thus appears to be the natural fine-grained boundary: above it (constant-region-size, BW-standard) one is in P, below it (unrestricted, polynomial-region-size) one is PSPACE-hard.

5.2. **The fourth reading: sign-flipping.** The sign-flipping reading (D-sign) orders regions by $|V_j - V_0|$, where $V_0 = \mathrm{LS}(\sum_i R_i)$. The lower-bound reduction of Section 3 does not directly apply, because the calibration constant a depends on the sign of V_0, which itself depends on the GLG answer through the booster-corridor contribution. A careful analysis would either eliminate this circularity or show that (D-sign) is intrinsically harder than the must/avoid readings. We leave this as an open question.

5.3. **Open problems.**

1. EXPTIME-*hardness of yose under ko-freeness.* Robson's EXPTIME-hardness theorem [5] uses global ko constructions. Is there a yose construction (in our sense) that is EXPTIME-hard?
2. *Multi-ko in the BW framework.* Extending the standard-shape catalogue to allow multi-ko situations is a long-standing open problem; even a partial polynomial-time algorithm under bounded ko-threat structure would be of independent interest.
3. *Strong NP-hardness of top-k.* On general (non-standard) shapes, is the top-k version of Best-Move-X NP-hard via an explicit Partition-style reduction? The closed-form ranking on standard shapes (Lemma 4.2) rules out the most direct attempt.
4. *Empirical coverage.* What fraction of typical 19×19 endgame positions falls into the BW standard-shape class? An empirical study would clarify the practical relevance of Theorem 4.3.

References

[1] E. Berlekamp, J. H. Conway, and R. K. Guy. *Winning Ways for Your Mathematical Plays.* Academic Press, 1982. (Second edition, A K Peters, 2001–2004.)

[2] E. Berlekamp and D. Wolfe. *Mathematical Go: Chilling Gets the Last Point.* A K Peters, 1994.

[3] J. H. Conway. *On Numbers and Games.* Academic Press, 1976. (Second edition, A K Peters, 2001.)

[4] M. Crâşmaru and J. Tromp. Ladders are PSPACE-complete. In *Proceedings of the 2nd International Conference on Computers and Games (CG 2000)*, Lecture Notes in Computer Science, vol. 2063, pp. 241–249. Springer, 2001.

[5] J. M. Robson. The complexity of Go. In *Information Processing 83 (Proc. IFIP World Computer Congress)*, pp. 413–417. North-Holland, 1983.

[6] L. J. Stockmeyer and A. R. Meyer. Word problems requiring exponential time. In *Proceedings of the 5th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 1–9. ACM, 1973.

[7] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pp. 85–103. Plenum Press, 1972.

Independent Researcher
Email address: lightman.chang@gmail.com