Computing Exact Nash Equilibria in Graphical Games: A Geometric Approach to Paths, Stars, and Caterpillars

Evan D. Lucca[1] **Mohammad T. Irfan**[2] **Luis E. Ortiz**[3]

[1]Department of Computer Science, Dartmouth College, Hanover, New Hampshire, USA
[2]Department of Computer Science, Bowdoin College, Brunswick, Maine, USA
[3]Department of Computer and Information Science, University of Michigan - Dearborn, Dearborn, Michigan, USA

Abstract

Computing mixed-strategy Nash equilibria (MSNE) in graphical games being PPAD-complete, special types of graphical games have received a lot of attention over the years. We present a number of new results using a geometric approach to computing exact MSNE. We consider graphical games of several structures, such as paths, stars, and caterpillars. We also consider graphical polymatrix games on these structures. We provide a new tight upper bounding result for path-structured graphical games and a new quadratic-time algorithm for representing all MSNE and computing one for path-structured polymatrix games. For star graphical games, our algorithm is logarithmic time for non-degenerate cases (i.e., without symmetric players) and linear time for general cases with symmetric players. Interestingly, having more symmetric players makes the computation more expensive. For star polymatrix games, our algorithm is linear in the input size. For the open problems on caterpillar polymatrix and graphical games, our algorithms are polynomial time in the input size.

1 INTRODUCTION

Since Kearns et al. [2001] introduced graphical games in this conference, these games have been widely studied because of their natural ability to capture strategic interactions over networks. The inspiration came from Bayesian networks. In a graphical game, given the strategies of the neighbors of a node, the node's payoff is independent of the other nodes of the underlying network. This led to numerous applications, such as studying trades on networks [Kakade et al., 2004], modeling interdependent security games [Kearns and Ortiz, 2003], studying strategic influence in networks [Irfan and Ortiz, 2014], and even analyzing deep RL [Yan et al., 2025], just to name a few. Notably, the TreeNash algorithm proposed in Kearns et al. [2001] has become a standard algorithmic framework in computational game theory.

Another attractive feature of graphical games is their compact representation size. Whereas a 2-action n-player game in normal form has a representation size of $O(n2^n)$, a graphical game with n nodes and maximum degree d has a representation size of $O(n2^k)$, where typically $k \ll n$. Although compact representation is attractive in practice, it makes algorithm design challenging because the running time of an algorithm is measured with respect to the input size. In fact, graphical games played a key role in the PPAD-completeness results in game theory [Goldberg, 2011, Daskalakis et al., 2009, Chen and Deng, 2005, Rubinstein, 2015]. More recently, it has been shown that even tree polymatrix games, which are special types of graphical games, are PPAD-complete [Deligkas et al., 2020].

With all the hardness results, it is not surprising that a lot of attention has been given to computing approximate mixed-strategy Nash equilibria (MSNE) [Deligkas et al., 2014, Ortiz and Irfan, 2017, Tsaknakis and Spirakis, 2008]. Another direction in the literature is solving graphical games of special structures, such as paths and cycles, as well as solving special types of graphical games such as polymatrix games. This paper takes the latter direction. In fact, this paper is inspired by Elkind et al. [2006]'s seminal work on exact MSNE computation on paths, cycles, and other types of graphs. We present a geometric approach that opens up a new way of looking at graphical games of special structures. Our approach leads to the first known tight upper bounds for path-structured graphical games. For star graphical games without strategic symmetry we give a logarithmic time algorithm. For caterpillar graphs, which are listed as an open problem in Elkind et al. [2006], we give polynomial time algorithms. The PPAD-completeness results given in Elkind et al. [2006] and later bolstered in Deligkas et al. [2020] suggest that even slight extensions to simple caterpillar structures considered in this paper may be provably hard.

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:4027–4066.

Moreover, our approach is visual and easy to implement. As a result, it holds promise for future applications, such as yielding approximation or heuristic algorithms for more complex structures like trees and cyclic graphs.

2 PRELIMINARIES

We first define two classes of games studied in this paper.

Definition 1 (Graphical Games [Kearns et al., 2001])**.** *An n-player graphical game $\mathcal{GG}$ is defined by the following:*

Structure: An n-node undirected graph $G = (N, E)$, where each node $i \in N = [n]$ represents a player *of the game and each edge $(i, j) \in E$ signifies the interdependence between its endpoints i and j. We define $N_i \equiv \{i\} \cup \{j \mid (i, j) \in E\}$ to be the* neighborhood *of player i.*

Actions: A set of actions *A_i for each player i. We define $A_{N_i} = \times_{j \in N_i} A_j$ to be the set of* neighborhood action profiles *of i and $A_N = \times_{i \in N} A_i$ to be the set of* action profiles*.*

Payoffs: A payoff function *M_i for each player i, which directly depends only on the neighborhood action profile of i (which includes i's own action). That is, $M_i : A_{N_i} \to \mathbb{R}$.*

Definition 2 (Graphical Polymatrix Games [Janovskaja, 1968])**.** *A graphical polymatrix game is a special type of graphical game with an additively decomposable payoff function. For each player i and any of its neighbors j, we are given a* local *payoff function for i, $M_{i,j} : A_i \times A_j \to \mathbb{R}$ (which may be different from $M_{j,i}$). Player i's payoff function $M_i : A_{N_i} \to \mathbb{R}$ is defined as the sum of its local payoffs. That is, given a neighborhood action profile a_{N_i}, $M_i(a_{N_i}) = \sum_{j \in N_i} M_{i,j}(a_i, a_j)$.*

It is important to note how the payoff functions are represented for the above games because the running time of algorithms hinges on it. There are two predominant representations in the literature: normal form [Kearns et al., 2001] and parametric or functional form [Gottlob et al., 2005]. We use the normal form representation in this paper. That is, for graphical games, each player's payoff function will be represented in a tabular or matrix form. For graphical polymatrix games, the local payoff functions will be represented in a tabular form. This implies that the representation size (or input size) of a graphical game with n players, a maximum of m actions, and maximum neighborhood size k is $O(nm^k)$. In contrast, graphical polymatrix games have a representation size of $O(nm^2k)$. Both of these classes of games have compact representation because $k \ll n$ in practice and a player's payoff directly depends on the neighborhood action profile only and not on everyone's actions.

We next consider expected payoffs under mixed strategies. First, a mixed strategy p_i of player i is a probability distribution over the set of actions A_i. We use $p_{N_i} = (p_j)_{j \in N_i}$ to denote player i's *neighborhood mixed-strategy profile* (which includes p_i) and $p_N = (p_j)_{j \in N}$ to denote a *mixed-strategy profile* of everyone. Using these terms, the *expected payoff* of i under p_{N_i} is defined as $M_i(p_{N_i}) \equiv \sum_{a_{N_i} \in A_{N_i}} M_i(a_{N_i}) \prod_{j \in N_i} p_j(a_j)$.

Using the notation $S - i \equiv S - \{i\}$ for any set $S \subseteq [n]$ and $i \in [n]$, player i's expected payoff for playing action $a_i \in A_i$ while i's neighbors play mixed strategies $p_{N_i - i}$ is given by $M_i(a_i, p_{N_i - i}) \equiv \sum_{a_{N_i - i} \in A_{N_i - i}} M_i(a_{N_i}) \prod_{j \in N_i - i} p_j(a_j)$.

We next define a mixed-strategy Nash equilibrium (MSNE).

Definition 3 (Mixed-Strategy Nash Equilibrium (MSNE))**.** *An MSNE is given by a mixed strategy profile p^*_N such that for each player i and for any action $a_i \in A_i$, $M_i(p^*_{N_i}) \geq M_i(a_i, p^*_{N_i - i})$.*

Although the above definitions are for any number of actions, we study 2-action games in this paper, as in Elkind et al. [2006]. We use several well-known undirected graph structures, such as trees, paths, stars, and caterpillars. A *tree* is a connected acyclic graph. A *path graph*, also known as a line graph, is a tree with maximum degree 2. A *star graph* with n nodes is a tree where there is a *hub* node of degree $n - 1$, and each of the other nodes, called a *spoke* node, is connected to the hub node by an edge. A *caterpillar graph* is a tree with a path structure as its *spine* and every other node, called a *spoke*, connected to the spine by an edge. That is, the spokes are one hop away from the spine.

THE TREENASH FRAMEWORK

Devised by Kearns et al. [2001] in their UAI 2001 paper, *TreeNash* is a dynamic programming algorithm for computing Nash equilibria in a 2-action tree-structured graphical game. The algorithm operates in two passes, referred to as the upstream and downstream passes. The upstream pass goes from the leaves to the root. Each node V sends a message called the *best response policy* (defined below) to its parent W.[1] The message essentially conveys the best mixed strategy of V in response to any mixed strategy of W, given the messages V has received from its children. Here, for 2-action games with actions $\{0, 1\}$, we represent a mixed strategy by the probability of playing 1.

Definition 4 (Best Response Policy)**.** *The* best response policy *$B_{W,V} : [0, 1]^2 \to \{0, 1\}$ represents the messages from V to its parent W, where $B_{W,V}(w, v) = 1$ (or simply $B(w, v) = 1$ when clear from context) iff there exists a* witness vector *of mixed strategies $(u_1, \ldots, u_k)$ for V's children $U_1, \ldots, U_k$, respectively, such that:*

[1] Our definition (as a function) is different from Elkind et al. [2006] (as a set) and is more aligned with Kearns et al. [2001].

1. $B(v, u_i) = 1$, for $1 \le i \le k$, and

2. v is V's best response to p_{N_V} constituted of v, w, and $(u_1, \ldots, u_k)$; i.e., $M_V(p_{N_V}) \ge M_V(a_V, p_{N_V - V})$ for any $a_V \in A_V$.

For the special case of a leaf node V, $B_{W,V}$ represents V's best responses to W without requiring any witness vector.

In the above definition, $B(w, v) = 1$ basically means that V is "happy" to play v in response to the parent W playing w iff two conditions are satisfied. First, each child U_i of V told V that they are happy to play some strategy u_i in response to v, and second, v is V's best response in this scenario. Note that $B(w, v)$ does not consider W's best response.

Another perspective on the best response policy $B_{W,V}$ is that it represents MSNE in the subgame comprised of the subtree rooted at V and conditioned on the strategy of W. This is used by the downstream pass described next.

The downstream pass starts at the root and goes down to the leaves. Reusing symbols, the root node V uses the messages it received from its children $U_1, \ldots, U_k$ to determine a strategy v for itself and a witness vector $(u_1, \ldots, u_k)$ such that v is a best response to $(u_1, \ldots, u_k)$ and $B(v, u_i) = 1$ for $1 \le i \le k$. V then instructs each child U_i to play u_i. In turn, each node U_i instructs its children to play according to a witness vector for $B(v, u_i) = 1$. This process is repeated until all the leaves are assigned strategies. The assignments made in the downstream pass constitute an MSNE.

As we will soon visualize in the context of a path graph, $B(w, v) = 1$ consists of a set of axis-aligned rectangles within $[0, 1]^2$ [Kearns et al., 2001]. Therefore, the running time of TreeNash for computing an MSNE depends on how the number of rectangles grows in the upstream pass. Kearns et al. [2001] showed that this growth is exponential in n. This result, along with several hardness results [Elkind et al., 2006, Deligkas et al., 2020], leads us to consider special types of graphs. We begin with paths in the next section, but we first highlight two points about our usage of TreeNash.

First, we treat TreeNash as a *framework* because we use the two-pass message passing scheme proposed in the original TreeNash paper [Kearns et al., 2001], but the construction of our messages and the analysis are different from theirs. Second, the running time statements in this paper are for computing a representation of all MSNE, from which one MSNE can be computed in the same worst-case running time. Appendix D addresses computing an MSNE without representing all MSNE.

3 REVISITING PATHS THROUGH A GEOMETRIC LENS

It is well understood how to compute an MSNE in path graphical games in polynomial time [Elkind et al., 2006]. We present a new geometric approach to this problem. Although our geometric approach does not improve prior results on paths in terms of worst-case running time for MSNE computation, it tightens them and opens up a new way of thinking about more complex structures. Additionally, the geometric approach is easy to visualize, understand, and implement.

Consider a 2-action graphical game on an n-node path $U, V, W, \ldots$. That is, $\Delta_U(v) \equiv M_U(1, v) - M_U(0, v)$. We now make a connection between $\Delta_U(v)$ and U's best response to v. For that, we first express U's expected payoff for playing a mixed strategy u when V plays v (see Appendix A.1 for details).

$$\begin{aligned} M_U(u, v) &= uM_U(1, v) + (1 - u)M_U(0, v) \\ &= u\,(M_U(1, v) - M_U(0, v)) + M_U(0, v) \\ &= u\Delta_U(v) + M_U(0, v) \,. \end{aligned}$$

Observe that U has no control over $M_U(0, v)$ (because it does not depend on u). Therefore, when $\Delta_U(v) > 0$, U maximizes its payoff by setting $u = 1$. Similarly, when $\Delta_U(v) < 0$, U sets $u = 0$. When $\Delta_U(v) = 0$, U gets the same payoff of $M_U(0, v)$ for any $u \in [0, 1]$ and is therefore able to play mixed strategies. Assuming strategic interdependence, since $\Delta_U(v)$ is a linear function of v,[2] $\Delta_U(v) = 0$ for at most one value of v, which we denote by v^*. This is illustrated in Figure 1. U's best response policy $B_{V,U}$ contains a maximum of three segments only when $v^* \in (0, 1)$: one horizontal segment at each pure strategy of U and a vertical segment at the indifference point v^* connecting them. If $v^* \le 0$ or $v^* \ge 1$, $B_{V,U}$ contains one horizontal segment only. In the degenerate case of U's payoff not depending on V, $\Delta_U(v) = 0$ for all v, which makes $B_{V,U}$ the $[0, 1]^2$ box. We make the following assumption, which is not restrictive because as shown in Appendix C, interdependence can only lead to more segments than its absence. [3]

Assumption 1. *We assume strategic interdependence between any two nodes connected by an edge and as a result, do not consider rectangles in best response policies.*

Once V receives $B_{V,U}$ from U, it is V's turn to send its best response policy $B_{W,V}$ to W. Similar to the case of U, let $\Delta_V(u, w) \equiv M_V(u, 1, w) - M_V(u, 0, w)$. The expected payoff of V can be expressed in terms of Δ_V as follows.

$$\begin{aligned} M_V(u, v, w) &= vM_V(u, 1, w) + (1 - v)M_V(u, 0, w) \\ &= v\,(M_V(u, 1, w) - M_V(u, 0, w)) + M_V(u, 0, w) \\ &= v\Delta_V(u, w) + M_V(u, 0, w) \,. \end{aligned}$$

As before, V's best response policy $B_{W,V}$ is dictated by $\Delta_V(u, w)$. Of particular interest is the indifference case of

[2] $\Delta_U(v) \equiv M_U(1, v) - M_U(0, v) = M_U(1, 1)v + M_U(1, 0)(1 - v) - M_U(0, 1)v - M_U(0, 0)(1 - v)$.

[3] By the standard (Borel) measure over real-valued payoffs, almost all games satisfy this assumption measure-theoretically.

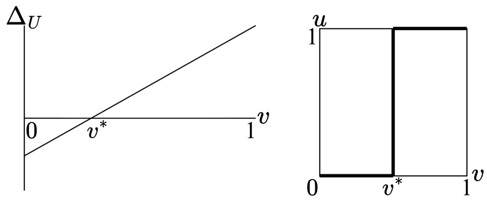

Figure 1: An example of Δ_U (left) and the resulting best response policy $B_{V,U}$ (right), which contains a horizontal segment at $u = 0$ over $v \in [0, v^*]$, another horizontal segment at $u = 1$ over $v \in [v^*, 1]$, and a vertical segment at $v = v^*$ over $u \in [0, 1]$ denoting U's indifference.

$\Delta_V(u, w) = 0$. For some constants a, b, c, and d, we can derive the following (details are in Appendix A.2).

$$\Delta_V(u, w) = cuw - au + dw - b \,.$$

$$\Delta_V(u, w) = 0 \implies w = \frac{au + b}{cu + d} \,.$$

Here, w is a linear rational function of u. In non-degenerate cases,[4] its geometric shape is hyperbolic with two distinct curves separated by a vertical asymptote at $u = -\frac{d}{c}$ and a horizontal asymptote at $w = \frac{a}{c}$ (see Figure 2 and Appendix A.3). Since $u, w \in [0, 1]$, we are only interested in the hyperbola curves within the region $[0, 1]^2$ which we refer to as the *(hyperbolic) indifference curves* because $\Delta_V(u, w) = 0$ along the curves. *To our knowledge, the geometry of hyperbolas was not directly exploited by prior literature.*

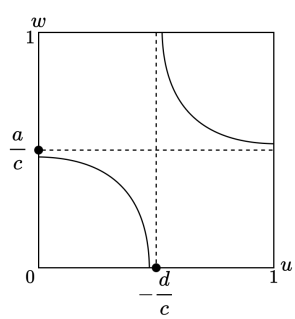

Figure 2: An example of $w = \frac{au+b}{cu+d}$. Two distinct hyperbolic curves are separated by vertical and horizontal asymptotes (dotted). The curves may also appear in the other quadrants.

To construct $B_{W,V}$, we combine the information from $B_{V,U}$ (Figure 1) with the indifference curves (Figure 2). There are three cases: $\Delta_V(u, w) = 0$ (signifying indifference), $\Delta_V(u, w) > 0$ (implying $v = 1$), and $\Delta_V(u, w) < 0$ (implying $v = 0$). While we leave a full treatment of these cases in Appendix A.4, we sketch the most impactful case of $\Delta_V(u, w) = 0$ here. See Figure 3 for an illustration. We map each segment of $B_{V,U}$ to segment(s) in $B_{W,V}$ as follows. Consider the vertical segment of $B_{V,U}$ at $v = v^*$, where $u \in [0, 1]$. Using the indifference curves, we identify the intervals of w values (blue dashes in Figure 3) such that (u, w) is on the curves. These intervals of w values lead to two horizontal segments (blue lines) in $B_{W,V}$ at $v = v^*$.

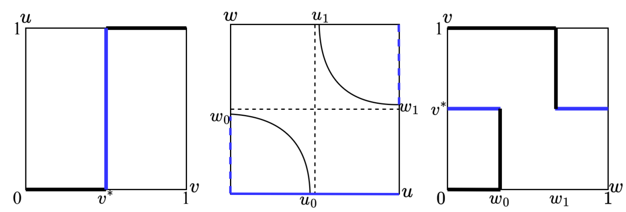

Figure 3: **(Left)** $B_{V,U}$ with the vertical segment highlighted in blue: $u \in [0, 1]$ at $v = v^*$. **(Center)** Indifference curves with highlighted intervals of w values $[0, w_0]$ and $[w_1, 1]$ such that (u, w) is on the curve for $u \in [0, 1]$. **(Right)** $B_{W,V}$ with the highlighted horizontal segments that result from the vertical segment of $B_{V,U}$. The segments represent $w \in [0, w_0]$ and $w \in [w_1, 1]$ at $v = v^*$. Other segments of $B_{W,V}$ result from various other cases.

Not surprisingly, the number of segments in best response policies grow as we go along a path. The construction of $B_{W,V}$ shows that the number of segments is maximized when both curves of the hyperbola are present within $[0, 1]^2$. Therefore, for upper bounding the number of segments, we will assume such hyperbolas. Figure 4, generated using an implementation of our algorithm presented in Section 4, illustrates segment growth in a worst-case tight example.

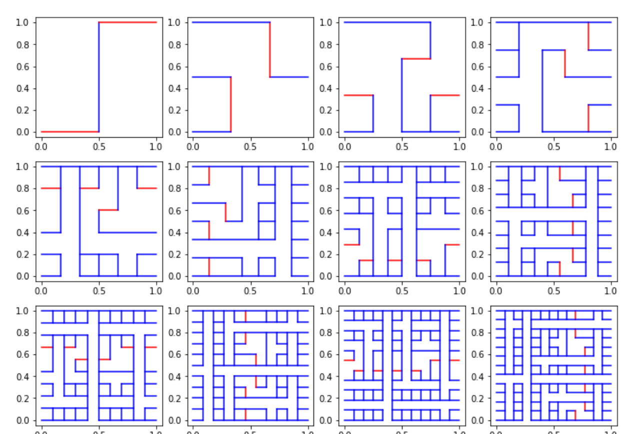

Figure 4: A worst-case tight example of segment growth as we go along a path. This is generated using an implementation of our geometric approach and is proved to be worst case in Section 4. The segments are evenly spaced for clarity. The blue and red colors signify how vertical segments lead to horizontal segments (and vice versa) in successive plots.

Having seen examples of horizontal and vertical segments, the following definition of indifference points arises naturally and helps us with technical results.

[4] Please see Appendix C for a treatment of degenerate cases.

Definition 5 (Indifference Point)**.** *Given $B_{W,V}$, any point w_1 on the w-axis is considered a w*-indifference point *or a* vertical indifference point *if there is at least one vertical segment at $w = w_1$; i.e., formally, if $\{w_1\} \times \{v \mid B(w_1, v) = 1\}$ contains a segment.*

Similarly, any point v_1 on the v-axis is considered a v-indifference point *or a* horizontal indifference point *if there is at least one horizontal segment at $v = v_1$; i.e., formally, if $\{w \mid B(w, v_1) = 1\} \times \{v_1\}$ contains a segment.*

Although the meaning of a *segment* should be clear in the above definition, we make it explicit below.

Definition 6 (Segments and Floating Status)**.** *Given $y_1 \in [0, 1]$, we say that a set $S \subset [0, 1]^2$ is a* horizontal segment *at $y = y_1$) if $S = [x_1, x_2] \times \{y_1\}$ for some x*-interval *$[x_1, x_2]$ with $x_1 < x_2$. A horizontal segment is* floating *if $0 < x_1 < x_2 < 1$; otherwise, we say it is* non-floating. *A* vertical segment *is similarly defined (except rotated $90°$).*

We define a *pure-strategy segment* of $B_{W,V}$ as a horizontal segment at either $v = 0$ or $v = 1$.

4 GRAPHICAL GAMES ON PATHS: A TIGHT UPPER BOUND

We present the first known tight upper bound on the number of segments needed to represent best response policies on paths. This is significant because the number of segments is the dominating factor in the running time of TreeNash. The result also gives an insight into the type of path-structured games that are tight examples. We first analyze the worst-case growth of the number of segments and then create instances with that growth rate.

Although our geometric approach is easy to visualize, as we just saw in the previous section, formalizing our statements requires phrasing in words what is more naturally visualized. Hence, we would like to sketch the key properties leading to our results at a high level. These properties, which we exploit algorithmically, follow from the problem's geometry. We have the following in the worst-case. (1) Every horizontal segment leads to exactly one vertical segment in the next message. (2) For any point v_1, exactly one vertical segment at v_1 may lead to exactly two non-floating horizontal segments at v_1 in the next message; and if such vertical segment exists at v_1, then any other vertical segment at v_1 leads to exactly one horizontal segment at v_1. (3) The number of horizontal-axis indifference points is one for the first message and increases by exactly one for each subsequent message. (4) The number of pure-strategy segments is exactly two (one at each pure strategy) in the first two messages and exactly three (one at a pure strategy and two at the other) in subsequent messages.

4.1 WORST CASE ANALYSIS OF GROWTH RATE

The maximum number of segments in all future best response policies can be determined inductively with the help of the following lemmas. Assume T, U, V, and W are the nodes on a path with node number $j-2$, $j-1$, j, and $j+1$, respectively. Let $B_{W,V}$ contain R_j horizontal segments and C_j vertical segments, with the sum $S_j = R_j + C_j$.

Here is our formal plan. We first show how horizontal segments in $B_{V,U}$ lead to vertical segments in $B_{W,V}$. We then show the other case: vertical to horizontal, followed by pure-strategy considerations and the final accounting. Complete proofs are in Appendix B.

4.1.1 Horizontal Segments to Vertical Segments

Lemma 1. *All horizontal segments at the same u-indifference point in $B_{V,U}$ result in vertical segments at the same w-indifference point in $B_{W,V}$. In the worst case, each of these horizontal segments results in a distinct vertical segment.*

Proof Sketch. Consider the u-indifference point of a horizontal segment in $B_{V,U}$. There is at most one w value on the hyperbolic indifference curves corresponding to it. That w value becomes the w-indifference point in $B_{W,V}$ for *all* horizontal segments having the same u-indifference point. Additionally, a horizontal segment cannot lead to multiple vertical segments because of the same range of v values. □

The next lemma follows from Lemma 1.

Lemma 2. $C_j \leq R_{j-1}$.

4.1.2 Vertical Segments to Horizontal Segments

Lemma 3. *A vertical segment from $B_{V,U}$ can result in at most two horizontal segments in $B_{W,V}$, and if it results in two horizontal segments, they will both be non-floating.*

Proof Sketch. The u-range of a vertical segment of $B_{V,U}$ can intersect V's hyperbolic indifference curves at at most two w-intervals, leading to two horizontal segments in $B_{V,W}$. They are non-floating because the w-intervals are cut off at $w = 0$ and $w = 1$ (see Figure 3). □

Lemma 4. *If the u-interval of a vertical segment at $v = v_1$ of $B_{V,U}$ does not overlap with both of V's hyperbolic indifference curves, then it will result in at most one horizontal segment s in $B_{W,V}$. Furthermore, if s is non-floating, then no other u-interval at $v = v_1$ of $B_{V,U}$ can overlap with both of V's indifference curves.*

Proof Sketch. The proof follows from the geometry of hyperbolas and the property that two vertical segments at $v = v_1$ cannot overlap (otherwise, they become one segment). □

Lemma 5. *All vertical segments at $v = v_1$ in $B_{V,U}$ result in horizontal segments at $v = v_1$ in $B_{W,V}$.*

Proof Sketch. For any vertical segment at $v = v_1$ in $B_{V,U}$, its u-interval, together with V's indifference curves, determines the w-interval(s) of the corresponding horizontal segment(s) in $B_{W,V}$ which also occur at $v = v_1$. □

4.1.3 Pure-Strategy Segments

The following holds for any graphical game.

Proposition 1. *[Kearns et al., 2001] Every pure-strategy segment of $B_{W,V}$ is non-floating.*

Proof. Because Δ_V is linear when viewed as a function of w only, for all u, the set $\{w \mid \Delta_V(u,w) \geq 0\} \cap [0,1]$ is either empty, or an interval of the form $[0, w_1]$ or $[w_1, 1]$ for some $w_1 \in [0,1]$; similarly for the case of $\Delta_V \leq 0$. □

Lemma 6. *$B_{W,V}$ can have ≤ 3 pure-strategy segments.*

Proof. If the interiors of V's two hyperbolic indifference curves are positive (supporting $v = 1$), then the region between them is negative (supporting $v = 0$), and vice versa. This leads to three possible w-intervals. □

Using the lemmas above, we get the following result.

Lemma 7. *In the worst case, the j-th message $B_{W,V}$ has j w-indifference points and $j+1$ v-indifference points.*

4.1.4 Final Accounting

Recall that j is the node number for V. Interestingly, we get a pattern of different behaviors depending on whether j is odd or even.

Lemma 8. *If j is odd, in the worst case, a horizontal line at some $v = v^*$ crosses the interior of one vertical segment at each w-indifference point of $B_{W,V}$.*

Proof Sketch. Assuming inductively that this holds for an odd j represented by $B_{U,T}$, compute the t-interval of the intersection among the vertical segments in question. Using Lemmas 3 and 5, in the worst case, this t-interval can cross the vertical asymptote of U's indifference curves and result in two non-floating horizontal segments at each mixed-strategy u-indifference point of $B_{V,U}$. Additionally, by Lemma 6 and Proposition 1, there are at most three non-floating pure-strategy segments of U in $B_{V,U}$. Using the geometric properties of U's hyperbolic indifference curves, there is a v value in $B_{V,U}$ that overlaps with a horizontal segment at each u-indifference point. Use Lemma 1 to convert these horizontal segments to vertical in $B_{W,V}$ (which represents the next odd number after j). The v-intervals of the horizontal segments of $B_{V,U}$ are the same as the v-intervals of the corresponding vertical segments of $B_{W,V}$. Therefore, the statement holds. □

Lemma 9. *If j is even, in the worst case, a horizontal line at some $v = v^*$ crosses the interior of a vertical segment at all but one w-indifference point of $B_{W,V}$, and no horizontal line crosses a vertical segment at each w-indifference point.*

Proof Sketch. We present the intuition here. The base case is the second message, as shown in Figure 3. The interiors of the two vertical segments do not overlap. In the worst case, one of the two will lead to two non-floating horizontal segments in the next message and the other to a floating horizontal segment. The interiors of the three resulting horizontal segments do not overlap (see Figures 14 and 15 in the Appendix). Next, these horizontal segments will lead to vertical segments whose interiors do not overlap either. In fact, as shown in red in Figure 4, none of their respective descendant segments will ever have overlapping interiors in future messages. Therefore, for any even j, no horizontal line can cross the interior of a vertical segment from each of the two descendant sets. □

Lemma 10. *$R_j \leq C_{j-1} + j + 2 - j\%2$.*

Proof Sketch. C_{j-1} vertical segments of $B_{V,U}$ lead to $\leq C_{j-1} + j - 1 - j\%2$ horizontal segments in $B_{W,V}$ by Lemmas 7, 8, and 9. There are at most three additional pure-strategy horizontal segments by Lemma 6. □

The above lemmas lead to the following upper bound.

Theorem 1. *$S_1 \leq 3$, $S_2 \leq 6$, and for any $j \geq 3$, $S_j = S_{j-1} + j + 2 - j\%2 = \frac{1}{2}j^2 + \frac{5}{2}j - \lfloor \frac{j+1}{2} \rfloor$.*

Proof. We use induction. *Base Case:* The upper bounds of 3 and 6 for the first and second policies, respectively, are shown in Section 3. *Induction Step:* For $j \geq 3$, assume $B_{V_j,V_{j-1}}$ contains S_{j-1} total segments, R_{j-1} horizontal segments and C_{j-1} vertical segments. B_{V_{j+1},V_j} has at most R_{j-1} vertical segments by Lemma 2 and at most $C_{j-1} + j + 2 - j\%2$ horizontal segments by Lemma 10. Therefore, the total number of segments in B_{V_{j+1},V_j} is $S_j = R_{j-1} + C_{j-1} + j + 2 - j\%2$. Using the definition of S_{j-1}, $S_j = S_{j-1} + j + 2 - j\%2$. In closed form, $S_j = \frac{1}{2}j^2 + \frac{5}{2}j - \lfloor \frac{j+1}{2} \rfloor$. □

We get the following runtime result from Theorem 1 which matches the result of Elkind et al. [2006], but importantly, our geometric analysis leads to the first known tight examples presented next. Also, recall that our runtime results are for representing all MSNE and computing one of them.

Theorem 2. *The TreeNash algorithm runs in $O(n^3)$ time on n-player graphical games on paths.*

4.2 TIGHT EXAMPLES

We devise an algorithm to construct worst-case examples where the number of segments matches the theoretical upper bound given by Theorem 1. Figure 4 shows a worst-case instance generated by an implementation of our algorithm.

Given the current policy, the algorithm generates the worst-case next policy. The idea is to sweep a horizontal line across the current policy to find a location that maximizes the number of vertical segments intersected. For this, we sort the endpoints of the vertical segments and position the horizontal line at each midpoint between successive endpoints in the sorted list.

We then construct a function for the hyperbolic indifference curves with the following properties: (1) the vertical asymptote is at the chosen midpoint, and (2) the two endpoints closest to the chosen midpoint on either side of it are in opposite regions of the hyperbola. These guarantee that the maximum number of splits occur and no segment is lost due to lying entirely between the two curves. The algorithm then constructs the next best response policy using the hyperbola. Notably, we can instantiate the payoff functions accordingly. We leave further details and pseudocode to Appendix B.2.

5 POLYMATRIX GAMES ON PATHS

We apply our geometric approach to polymatrix games on paths and present some new results. Consider a player V who has received the policy from U and is sending its policy to W. V's expected payoff is the sum of the expected payoffs from U and W. This leads to the following Δ function (derivation is in Appendix E).

$$\Delta_V(u, w) = c_2 w + c_1 u + c_0 = 0$$
$$w = \frac{(-c_1)u + (-c_0)}{c_2} = au + b$$

Observe that the polymatrix indifference curve takes the form of a linear function. Figure 5 gives an example. The linear indifference curve gives us a simpler version of two of the key properties of segments in the worst-case presented earlier for general games, but now for the polymatrix case, and leads to our result. (2') Every vertical segment at v_1 always leads to exactly 1 horizontal segment at v_1 in the next message. (4') The number of pure-strategy segments is always exactly 2 (one at each pure strategy).

Theorem 3. *In polymatrix games on paths, the jth best response policy contains at most $2j + 1$ segments.*

Proof Sketch. We prove it by induction. Each segment in $B_{V,U}$ leads to at most one segment in $B_{W,V}$. The additional two segments of $B_{W,V}$ come from V's pure strategies. □

This leads to the following result.

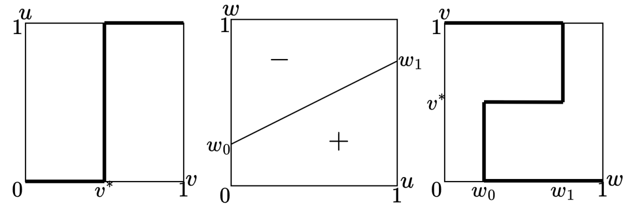

Figure 5: Polymatrix game on a path: $B_{U,V}$ (left), V's linear indifference function used to map this message (center), the resulting message $B_{W,V}$ (right).

Theorem 4. *The TreeNash algorithm runs in $O(n^2)$ time on n-player polymatrix games on paths.*

As we discuss in Appendices D and E in detail, the proof of this result gives us something a bit more interesting. Elkind et al. [2006] showed how to reduce the $B_{W,V}$ messages for graphical games on path to compute a single MSNE. They call the reduced message a "breakpoint policy" (see AppendixD). That reduction leads to a quadratic time algorithm for general graphical games on paths. Our proof shows that we cannot reduce $B_{W,V}$ further when the game is polynatrix. Hence, our results yield a lower bound on the minimum number of segments required to compute a single MSNE for graphical games on paths and imply as a corollary that we attain the worst case using a strictly smaller game class, mathematically speaking: i.e., we do not need to construct a general graphical games on paths to attain the lower bound; polymatrix versions are sufficient. (See Appendix E.)

6 STAR-STRUCTURED GAMES

We present several new results on star-structured games of graphical and polymatrix variants. We first give a general treatment of the best response policies. Consider a star with the hub node V and spokes $U_1, \ldots, U_{n-1}$. For TreeNash, we consider the hub as the root and spokes as the leaves. Each leaf sends a message to the root in the same manner as the first node does in a path, as illustrated in Figure 1. We assume U_i is indifferent at $v = v_i^*$.

Let $V^* = \{v_1^*, \ldots, v_{n-1}^*\}$. For any $v \notin V^*$, every leaf U_i has a pure-strategy best response, and therefore V has a singular payoff determined by a fixed value of Δ_V. At each $v \in V^*$, one or more leaf is indifferent, implying a range of Δ_V values. Figure 6 illustrates these two cases by plotting Δ_V against v. For every continuous non-v_i^* interval, Δ_V is fixed, with transitions between these fixed outputs at v_i^*. For an MSNE, it must be the case that $v = 0$ and $\Delta_V \leq 0$, or $v = 1$ and $\Delta_V \geq 0$, or $v \in (0, 1)$ and $\Delta_V = 0$.

We now investigate graphical and polymatrix games on

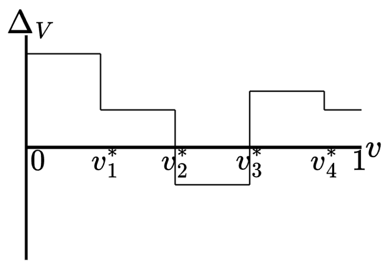

Figure 6: Δ_V of the hub node V of a star as a function of v. $v = v_i^*$ makes leaf U_i indifferent, leading to a range of Δ_V.

stars. These results have important implications because few classes of non-trivial games afford linear or sublinear time computation. In generic scenarios (analogous to Assumption 1), they also imply that we can characterize the set of all MSNE exactly, that it is finite, and that we can compute it in the same asymptotic worst-case time that we can compute just one. Please see Appendix F for details.

6.1 GRAPHICAL GAMES ON STARS

The input size (or representation size) of a graphical game on a star is $O(2^n)$ due to the size of the root player's payoff matrix. This plays a role in running time analysis. We consider the following cases.

Case 1. *The leaves have distinct indifference points.*

We compute an MSNE in three steps, each taking $O(n)$ time. First, we compute each leaf U_i's indifference point v_i^*, which is a rational number. Second, we compute the Δ_V vs. v plot. For this, we radix sort the indifference points into $v_1^*, \ldots, v_{n-1}^*$ (wlog) in $O(n)$ time. We then find the range of Δ_V at v_1^* in $O(n)$ time by computing the pure-strategy best responses of $U_2, \ldots, U_{n-1}$ and considering $u_1 \in \{0, 1\}$. After this, the range of Δ_V at v_2^* is computed in $O(1)$ time because we only need to compute the pure-strategy best response of U_1 while reusing those of $U_3, \ldots, U_{n-1}$. We repeat this for v_3^* onward. Between successive indifference points, the Δ_V value is constant, as mentioned above.

Third, we compute an MSNE in $O(n)$ time. For the pure-strategy cases of $v \in \{0, 1\}$, we look up the plot and check the sign of Δ_V. Failing that, we next examine each v_i^* in any order. If the range of Δ_V values cross 0, we have found an MSNE. This is because V will only play $v_i^* \in (0, 1)$ if $\Delta_V = 0$. Recall that only U_i is indifferent at v_i^* and other leaves have pure-strategy best responses that we have already computed. This makes Δ_V a linear function of u_i. Solving $\Delta_V = 0$ for u_i completes the MSNE computation. Since we do $O(1)$ work at each v_i^*, this step takes $O(n)$.

Therefore, MSNE computation is in this case is $O(n)$, which is logarithmic in the input size.

Case 2. *Multiple leaves have the same indifference point.*

We may first think that having $m > 1$ leaves share the same indifference point adds to the flexibility of computing an MSNE. However, in this case, Δ_V depends on a 2^{m+1}-sized submatrix of V's payoff matrix, leading to a running time of $O(2^m)$, which is still linear in the input size. The idea is to first compute $\Delta_V^{\max}$ and $\Delta_V^{\min}$ at the indifference point. This takes $O(2^m)$ time because the extrema must occur at pure-strategy assignments for those m leaves. An MSNE consistent with the indifferences exists iff $\Delta_V^{\max}$ and $\Delta_V^{\min}$ have different signs or either is 0. In the case of different signs, we can perform binary search over the $[0, 1]^m$ space to compute an MSNE. Each step of the search takes $O(2^m)$, and the total number of steps required is bounded by the precision of the MSNE. We summarize the results below.

Theorem 5. *In the absence of strategic symmetry among the spokes, we can compute an MSNE in star graphical games in time logarithmic in the input size. When m spokes share the same indifference point, we can compute an MSNE in time exponential in m, which is still linear in the input size.*

6.2 POLYMATRIX GAMES ON STARS

The input size of a polymatrix game on a star is $O(n)$. The case of distinct indifference points for the leaves is the same as that for graphical games on stars, leading to $O(n)$ MSNE computation. We turn to the more challenging case where multiple leaves have the same indifference point and give a local-search algorithm for MSNE computation.

Due to the additive decomposition of payoff functions in polymatrix games, the Δ_V function has a separate linear term for each u_i. The pure-strategy cases of $v \in \{0, 1\}$ are simple. Given a v-indifference point $v^* \in (0, 1)$ shared by leaves $U_1, \ldots, U_m$ (wlog), the other leaves have pure-strategy best responses to v^*. Therefore, Δ_V at v^* takes the form $c_m u_m + ... + c_1 u_1 + c_0$, where $c_0, \ldots, c_m$ are constants. Using the signs of $c_1, \ldots, c_m$, we choose the pure-strategy for each leaf U_i that achieves the minimum Δ_V denoted by $\Delta_V^{\min}$. Here, $\Delta_V^{\min} > 0$ means there is no MSNE at v^*, and $\Delta_V^{\min} = 0$ means we have found an MSNE. If $\Delta_V^{\min} < 0$, we keep changing the pure strategies of the indifferent nodes one at a time until either $\Delta_V \geq 0$ or all the changes are made. The former case indicates an MSNE at v^*. In particular, if Δ_V flips from negative to positive once we change the pure strategy of U_j, we compute the MSNE by expressing Δ_V as a function of u_j (all other leaves have fixed pure strategies) and solving $\Delta_V = 0$ for u_j. This algorithm gives us the following result.

Theorem 6. *We can compute an MSNE in polymatrix stars in time linear in the input size.*

Beyond the broader significance stated earlier, an important corollary of this result is that unlike graphical stars in general, every MSNE of a generic polymatrix star is rational.

7 CATERPILLAR GAMES

An s-spoke *caterpillar* is a path-like graph where each node along the path, also called the spine, is connected by edges to at most s nodes of degree 1, termed spoke nodes. We call a 1-spoke caterpillar graph a *simple caterpillar*, where $V_1, \ldots, V_n$ are the nodes on the spine, and each V_j is connected to a spoke U_j.

It is known that TreeNash runs in exponential time in the worst case on general trees. For this reason, it is of interest to identify tree structures that can be solved in sub-exponential time. Caterpillars are one such structure. Here we will sketch the argument for the polynomial running time of TreeNash on both polymatrix and graphical caterpillar games. Full analysis is in Appendix G. To simplify the presentation here, all of the following analysis assumes the generic scenario, meaning no player will ever receive multiple messages with shared indifference points.

7.1 POLYMATRIX CATERPILLAR GAMES

We present the key idea in the context of a simple caterpillar polymatrix game. Here, Δ_{V_j} depends on v_{j-1}, v_{j+1}, and u_j and $\Delta_V = 0$ gives us the following.

$$v_{j+1} = -\frac{av_{j-1} + bu_j + d}{c} = \alpha v_{j-1} + \beta u_j + \gamma \ .$$

Conditioned on $u_j = 0$, we get a linear indifference curve $v_{j+1} = \alpha v_{j-1} + \gamma$ similar to the path case shown in Figure 5. We use it to map $B_{V_j,V_{j-1}}$ to B_{V_{j+1},V_j}. Conditioned on $u_j = 1$, the linear indifference curve $v_{j+1} = \alpha v_{j-1} + \beta + \gamma$ is parallel to that for $u_j = 0$. We redo the above mapping with respect to it. Suppose v_j^* is the indifference point of U_j, meaning U_j's best response is 0 or 1 on either side of v_j^*.

We now slice each of the above two B_{V_{j+1},V_j} messages using a horizontal line at v_j^*. We take one slice from each as described next and paste them into a collage. From B_{V_{j+1},V_j} conditioned on $u_j = 0$, we choose the slice where the v_j values correspond to U_j's best response of 0 and likewise for B_{V_{j+1},V_j} conditioned on $u_j = 1$. We will see gaps in the collage due to a *shifting effect* at v_j^* (or due to two indifference lines used for mapping). These gaps are connected by horizontal segments corresponding to the intersection points between a vertical line at v_j^* and the horizontal segments in $B_{V_j,V_{j-1}}$. The end result is B_{V_{j+1},V_j}.

Mapping $B_{V_j,V_{j-1}}$ to B_{V_{j+1},V_j}, each vertical segment leads to a horizontal segment, and each horizontal segment leads to either one or three vertical segments. We get the latter when the horizontal segment in $B_{V_j,V_{j-1}}$ is split by the vertical line at v_j^*, resulting in two vertical segments in B_{V_{j+1},V_j} that are shifted, plus one horizontal segment connecting the gap due to shifting. Using this property, we show that the number of segments in the j-th message B_{V_{j+1},V_j} is $m_j = m_{j-1} + 4j$. Therefore, the number of segments in the j-th message is quadratic in j, and the total number of segments across the downstream pass is cubic in the number of players. Generalizing the above reasoning to the s-spoke case gives us the following result.

Theorem 7. *The TreeNash algorithm runs in $O(n^3 s^2)$ time on polymatrix caterpillar games on length n, s-spoke caterpillars, whose input size is $O(ns)$.*

7.2 GRAPHICAL CATERPILLAR GAMES

The indifference curve in this case resembles that of the graphical path problem, but with additional terms for each spoke player. For example, an indifference curve in a 1-spoke caterpillar would take the following form.

$$v_{j+1} = -\frac{(bv_{j-1} + f)u_j + (ev_{j-1} + h)}{(av_{j-1} + d)u_j + (cv_{j-1} + g)} \ .$$

Similarly to the polymatrix case, $u_j = 0$ when $v_j < v_j^*$ and $u_j = 1$ when $v_j > v_j^*$, or vice versa. Also, similarly to the polymatrix case, when $v_j = v_j^*$, U_j is indifferent while v_{j-1} must take on one of a fixed number of values. But unlike the polymatrix case, the indifference curve is hyperbolic like the case shown in Figure 2. As a result, there are two cases to be considered for the horizontal segments of B_{V_{j+1},V_j}. For a given v_{j-1}, if it is the case that v_{j+1} is on the same side of the vertical asymptote for $u_j = 0$ and $u_j = 1$, B_{V_{j+1},V_j} has a horizontal segment joining the collage halves where they shifted at v_j^*. If instead $u_j = 0$ and $u_j = 1$ map to v_{j+1} values on opposite sides of the vertical asymptote, B_{V_{j+1},V_j} has two segments at v_j^* from each half of the collage to pure strategies of V_{j+1}. This leads to the following result.

Theorem 8. *On an s-spoke caterpillar with a spine of n nodes, the TreeNash algorithm runs in $O(n^2 2^s + n^3 s^2)$ time on graphical games. Since the input size is $O(n2^s)$, the algorithm is polynomial time.*

8 CONCLUSION

In this paper, we have introduced a new geometric approach to computing exact MSNE in special types of graphical games and polymatrix games. Revisiting the literature using this approach has led to the discovery of a tight upper bound for path-structured graphical games. In addition, we have given new computational results for path-structured polymatrix games, star-structured graphical and polymatrix games, and caterpillar-structured graphical and polymatrix games. The existing PPAD-completeness results suggest that slight extensions to the caterpillar structures that we considered in this paper may be provably hard. Despite that, the new geometric approach holds future promise for various other special types of graphical games. In particular, the counter-intuitive property of the negative effect of strategic symmetry demands further attention. Designing practically fast heuristics beyond those for star and caterpillar graphs could be another interesting future direction.

Author Contributions

The authors contributed equally to this research. Author names are in the order of seniority. EDL worked on this research as an undergraduate student at Bowdoin.

Acknowledgements

EDL and MTI were supported in part by NSF grant IIS-1910203. LEO was supported in part by NSF grant IIS-1907553.

We started this research in the summer of 2022. Since then, a number of undergraduate students at Bowdoin participated in it. Among them, we are especially thankful to Darien Gillespie and Luke Tingley for their early work on this research.

References

Xi Chen and Xiaotie Deng. Settling the complexity of 2-player Nash-equilibrium. Technical Report 140, Electronic Colloquium on Computational Complexity (ECCC), 2005. `http://eccc.hpi-web.de/eccc-reports/2005/TR05-140/index.html`.

Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a Nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009. doi: 10.1137/070699652. URL `http://link.aip.org/link/?SMJ/39/195/1`.

Argyrios Deligkas, John Fearnley, Rahul Savani, and Paul Spirakis. Computing approximate Nash equilibria in polymatrix games. In Tie-Yan Liu, Qi Qi, and Yinyu Ye, editors, *Web and Internet Economics: 10th International Conference, WINE 2014, Beijing, China, December 14-17, 2014. Proceedings*, pages 58–71, Cham, 2014. Springer International Publishing. ISBN 978-3-319-13129-0. doi: 10.1007/978-3-319-13129-0_5. URL `http://dx.doi.org/10.1007/978-3-319-13129-0_5`.

Argyrios Deligkas, John Fearnley, and Rahul Savani. Tree polymatrix games are PPAD-hard. In *ICALP*, volume 168, page 38. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

Edith Elkind, Leslie Ann Goldberg, and Paul Goldberg. Nash equilibria in graphical games on trees revisited. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, pages 100–109, 2006.

Paul W Goldberg. A survey of PPAD-completeness for computing Nash equilibria. *Surveys in Combinatorics*, 3: 1–32, 2011.

Georg Gottlob, Gianluigi Greco, and Francesco Scarcello. Pure Nash equilibria: Hard and easy games. *JAIR*, 24: 357–406, 2005.

Mohammad T. Irfan and Luis E. Ortiz. On influence, stable behavior, and the most influential individuals in networks: A game-theoretic approach. *Artificial Intelligence*, 215:79 – 119, 2014. ISSN 0004-3702. doi: http://dx.doi.org/10.1016/j.artint.2014.06.004. URL `http://www.sciencedirect.com/science/article/pii/S0004370214000812`.

E. B. Janovskaja. Equilibrium situations in multi-matrix games. *Litovsk. Mat. Sb.*, 8:381–384, 1968. ISSN 0132-2818.

Sham M. Kakade, Michael Kearns, and Luis E. Ortiz. Graphical economics. In *Proceedings of the 17th Annual Conference on Learning Theory (COLT)*, volume 3120/2004 of *Lecture Notes in Computer Science*, pages 17–32, Banff, Canada, July 2004. Springer Berlin / Heidelberg.

M. Kearns, M. Littman, and S. Singh. Graphical models for game theory. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 253–260, 2001.

Michael Kearns and Luis E. Ortiz. Algorithms for interdependet security games. In *Neural Information Processing Systems (NIPS)*, 2003.

Luis E Ortiz and Mohammad T Irfan. Tractable algorithms for approximate Nash equilibria in generalized graphical games with tree structure. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Aviad Rubinstein. Inapproximability of Nash equilibrium. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 409–418, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3536-2. doi: 10.1145/2746539.2746578. URL `http://doi.acm.org/10.1145/2746539.2746578`.

Haralampos Tsaknakis and Paul G. Spirakis. An optimization approach for approximate Nash equilibria. *Internet Math.*, 5(4):365–382, 2008. URL `http://projecteuclid.org/euclid.im/1265033170`.

Yuyang Yan, Jiahui Li, and Cristina Zaggia. The analysis of deep reinforcement learning for dynamic graphical games under artificial intelligence. *Scientific Reports*, 15 (1):23133, 2025.

Computing Exact Nash Equilibria in Graphical Games: A Geometric Approach to Paths, Stars, and Caterpillars (Supplementary Material)

Evan D. Lucca[1] **Mohammad T. Irfan**[2] **Luis E. Ortiz**[3]

[1]Department of Computer Science, Dartmouth College, Hanover, New Hampshire, USA
[2]Department of Computer Science, Bowdoin College, Brunswick, Maine, USA
[3]Department of Computer and Information Science, University of Michigan - Dearborn, Dearborn, Michigan, USA

A REVISITING PATHS THROUGH A GEOMETRIC LENS

A.1 DERIVATION OF EXPECTED PAYOFF

We show that for any arbitrary payoff function M, player U's expected payoff for playing a mixed strategy u when V plays v is: $M_U(u,v) = uM_U(1,v) + (1-u)M_U(0,v)$, where $u \equiv p_U(1)$ and $v \equiv p_V(1)$.

Basically, U's expected payoff when U plays a mixed strategy u (defined by the probability of playing 1) and V plays a mixed strategy v is the sum of two terms:

1. U's prob. of playing 1 $\times$ U's exp. payoff when U plays 1 and V plays v.
2. U's prob. of playing 0 $\times$ U's exp. payoff when U plays 0 and V plays v.

Using the definition of expected payoffs given in Section 2, we get the following.

$M_U(u,v) = \sum_{a_U \in \{0,1\}, a_V \in \{0,1\}} M_U(a_U, a_V)\, p_U(a_U)\, p_V(a_V)$

$= \sum_{a_V \in \{0,1\}} \Big[M_U(1, a_V)\, p_U(1)\, p_V(a_V) + M_U(0, a_V)\, p_U(0)\, p_V(a_V) \Big]$

$= \sum_{a_V \in \{0,1\}} \Big[M_U(1, a_V)\, u\, p_V(a_V) + M_U(0, a_V)\, (1-u)\, p_V(a_V) \Big]$

$= u \sum_{a_V \in \{0,1\}} M_U(1, a_V)\, p_V(a_V) + (1-u) \sum_{a_V \in \{0,1\}} M_U(0, a_V)\, p_V(a_V)$

$= uM_U(1,v) + (1-u)M_U(0,v)$.

The last line uses the definition of expected payoff given in Section 2 when player i plays an action $a_i \in A_i$ while i's neighbors play mixed strategies $p_{N_i - i}$.

A.2 DERIVATION OF LINEAR RATIONAL FUNCTION

In Section 3, we have shown that $M_V(u,v,w) = v\Delta_V(u,w) + M_V(u,0,w)$. Observe that V's mixed strategy v is not present in $M_V(u,0,w)$. Therefore V's decision to pick a strategy of 0, 1, or a mixed strategy follows logically from the value of Δ_V alone.

If we expand Δ_V, we find that it can be expressed in the following form.

$$\begin{aligned}\Delta_V(u,w) =& uw\left(M_V(1,1,1) - M_V(1,0,1) + M_V(0,1,0) - M_V(0,0,0) + \right.\\ & \left. -M_V(0,1,1) + M_V(0,0,1) - M_V(1,1,0) + M_V(1,0,0)\right) + \\ & u\left(M_V(1,1,0) - M_V(1,0,0) - M_V(0,1,0) + M_V(0,0,0)\right) + \\ & w\left(M_V(0,1,1) - M_V(0,0,1) - M_V(0,1,0) + M_V(0,0,0)\right) + \\ & \left(M_V(0,1,0) - M_V(0,0,0)\right) \\ =& c_3uw + c_2u + c_1w + c_0\end{aligned}$$

where the constants $c_0, ..., c_3$ are determined by V's payoffs for combinations of pure strategies:

$$\begin{aligned}c_0 \equiv& M_V(0,1,0) - M_V(0,0,0)\\ c_1 \equiv& M_V(0,1,1) - M_V(0,0,1) - c_0\\ c_2 \equiv& M_V(1,1,0) - M_V(1,0,0) - c_0\\ c_3 \equiv& M_V(1,1,1) - M_V(1,0,1) - c_2 - c_1 - c_0\ ,\end{aligned}$$

or equivalently,

$$\begin{aligned}M_V(0,1,0) - M_V(0,0,0) =& c_0\\ M_V(0,1,1) - M_V(0,0,1) =& c_1 + c_0\\ M_V(1,1,0) - M_V(1,0,0) =& c_2 + c_0\\ M_V(1,1,1) - M_V(1,0,1) =& c_3 + c_2 + c_1 + c_0\ .\end{aligned}$$

From this we can see that there are an uncountably infinite number of payoff functions, hence also games, consistent with any fixed values for $c_0, c_1, c_2, c_3 \in \mathbb{R}$.

If we set the $\Delta_V = 0$, we can determine the values of w for which V is indifferent as a function of u.

$$\begin{aligned}\Delta_V(u,w) = c_3uw + c_2u + c_1w + c_0 = 0\\ w = \frac{(-c_2)u + (-c_0)}{c_3u + c_1} = \frac{au+b}{cu+d}\end{aligned}$$

A.3 GEOMETRIC PROPERTIES OF LINEAR RATIONAL FUNCTION

Although in this paper, our illustrations show the hyperbolic indifference curves in the top-right and bottom-left quadrants, we explore the alternative scenario next. Notably, our results are preserved in the alternative scenario.

We assume $c \neq 0$; otherwise, the function is linear. As noted in Figure 2, the hyperbolic curves of a linear rational function $w = \frac{au+b}{cu+d}$ can appear in either of the two diagonal quadrants. Here, the quadrants are defined by the vertical asymptote at $u = -\frac{d}{c}$ and the horizontal asymptote at $w = \frac{a}{c}$. It is well known how we can determine in which quadrants the curves appear. For this, we re-express the function as follows.

$$\begin{aligned}w &= \frac{au+b}{cu+d}\\ &= \frac{\frac{a}{c}(cu+d) - \frac{ad}{c} + b}{cu+d}\\ &= \frac{a}{c} + \frac{b - \frac{ad}{c}}{cu+d}\\ &= \frac{a}{c} + \frac{\frac{bc-ad}{c^2}}{u + \frac{d}{c}}\ .\end{aligned}$$

From the above expression, we see that whether w increases or decreases with increasing u depends on the sign of the expression $\frac{bc-ad}{c^2}$, which has the same the sign as $bc - ad$. Note that u appears in the denominator. Therefore, we obtain the following property.

Proposition 2. *Assuming $c \neq 0$, when $bc - ad > 0$, the hyperbolic curves of $w = \frac{au+b}{cu+d}$ appear in the top-right and bottom-left quadrants. When $bc - ad < 0$, these curves appear in top-left and bottom-right quadrants.*

We next consider the signs of the three regions in $[0,1]^2$ split by the two hyperbolic curves. This is used for deriving pure-strategy segments in Lemma 6. As described in Section 3, V's indifference function $w = \frac{au+b}{cu+d}$ is derived from $\Delta_V(u,w) = w(cu+d) - (au+b) = 0$. Therefore, for any (u,w) on the hyperbolic curves, $\Delta_V(u,w) = 0$. On the other hand, outside of the curves, the signs of the two outer regions and the central region between the curves correspond to the sign of Δ_V. Since the curves (where $\Delta_V = 0$) separate the outer regions from the central one, the two outer regions must have the same sign and the central region the opposite sign.

Although it is possible to precisely determine the signs of the outside and central regions for different quadrant configurations (using Proposition 2), the following question is more important for us.

Given a point (u^*, w^*) on the hyperbolic indifference curves, find the range of w values such that $\Delta_V(u^*, w) > 0$.

We can ask a similar question for $\Delta_V(u^*, w) < 0$, but for simplicity, let us focus on $\Delta_V(u^*, w) > 0$. Since $\Delta_V(u^*, w^*) = 0$, increasing and decreasing w^* while keeping u^* the same will lead to opposite signs for Δ_V. From the definition of $\Delta_V(u^*, w^*) = w^*(cu^* + d) - (au^* + b)$, we see that changing w^* only impacts the term $w^*(cu^* + d)$. Therefore, we get the following property.

Proposition 3. *Given a point (u^*, w^*) on the hyperbolic indifference curves, if $cu^* + d > 0$, then $\Delta_V(u^*, w) > 0$ for $w \in (w^*, 1]$. On the other hand, if $cu^* + d < 0$, then $\Delta_V(u^*, w) > 0$ for $w \in [0, w^*)$.*

Although we omit it here, we can derive a similar property for $\Delta_V(u^*, w) < 0$.

A.4 DERIVATION OF BEST RESPONSE POLICY

Horizontal segment at $u = 0$: $\{0\} \times [0, v^*]$. Figure 7 illustrates the generation process involving this segment, which indicates an interval of strategies of V for which U could play the fixed strategy of 0 as a best response. Player V can only play $v \in (0, v^*]$ if $\Delta_V = 0$. As $u = 0$ for this segment, we consult the indifference curve to determine the corresponding w, call it w_0, so that the point $(0, w_0)$ lies on the indifference curve. This coordinate pair's position on the curve indicates that $\Delta_V = 0$ when U plays 0 and W plays w_0. Because V can play 0 if $\Delta_V \leq 0$, V could also play $v = 0$ given this combination of U and W strategies. It is therefore possible that V could play any $v \in [0, v^*]$ if W plays w_0 based on this segment and the indifference curve. This results in the vertical segment $\{w_0\} \times [0, v^*]$ in $B_{W,V}$.

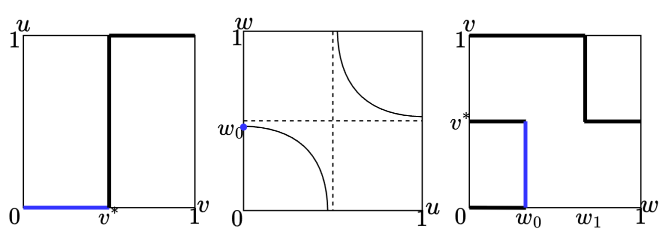

Figure 7: (Left) $B_{V,U}$ with the segment $I \equiv \{0\} \times [0, v^*]$ highlighted (in blue). (Center) The indifference curve with $(0, w_0)$ marked. (Right) $B_{W,V}$ with the segment resulting from I highlighted. (In this, and all subsequent diagrams illustrating the mapping of a segment of $B_{V,U}$ to its descendant(s) in $B_{W,V}$, we are including all the segments in the graph of $B_{W,V}$ for context, even though the source of some of the segments has yet to be explained.)

Horizontal segment at $u = 1$: $(u, v) \in \{1\} \times [v^*, 1]$. Figure 8 illustrates the generation process involving this segment, which indicates an interval of strategies of V for which U could play the fixed strategy of 1 as a best response. Player V can only play $v \in [v^*, 1)$ if $\Delta_V = 0$. As $u = 1$ for this segment, we consult the indifference curve to determine the corresponding w, call it w_1, so that the point $(1, w_1)$ lies on the indifference curve. This coordinate pair's position on the curve indicates that that $\Delta_V = 0$ when U plays 1 and W plays w_1. Because V can play 1 if $\Delta_V \geq 1$, V could also play $v = 1$ given this combination of U and W strategies. It is therefore possible that V could play any $v \in [v^*, 1]$ if W plays w_1 based on this segment and the indifference curve. This results in the vertical segment $\{w_1\} \times [v^*, 1]$ in $B_{W,V}$.

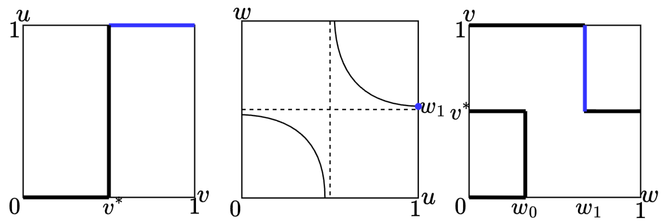

Figure 8: (Left) $B_{V,U}$ with the segment $I \equiv \{w_1\} \times [v^*, 1]$ highlighted (in blue). (Center) The indifference curve with $(1, w_1)$ marked. (Right) $B_{W,V}$ with the segment resulting from I highlighted.

Vertical segment at $v = v^*$**:** $(u, v) \in [0, 1] \times \{v^*\}$**.** Figure 9 illustrates the generation process involving this segment, which indicates a strategy of V, call it v^*, for which U can play any $u \in [0, 1]$ as a best response. Because $v^* \in (0, 1)$, i.e., it is a fully mixed strategy, we use the indifference curve to determine the w value for every u value so that the coordinate pair lies on the indifference curve. Player U can play any $u \in [0, 1]$, covering the full u range of the indifference curve, but the indifference curve is discontinuous in w, resulting in a discontinuous range of w values on the curve. Let u_0 and u_1 be the values of u where the indifference curve reaches $w = 0$ and $w = 1$, respectively, as it is approaching the vertical asymptote. (see Figure 9). For any $u \in [0, u_0]$, there is a corresponding $w \in [0, w_0]$ on the indifference curve. For $u \in (u_0, u_1)$, there is no value of w which lies on the indifference curve. For any $u \in [u_1, 1]$, there is a corresponding $w \in [w_1, 1]$ on the indifference curve. As such, $B_{W,V}$ has *two* horizontal segments at v^*, one over $w \in [0, u_0]$ and the other over $w \in [w_0, 1]$. In what follows, we also refer to the event just described in this case as the *splitting of a segment*, and call such a vertical segment a *split segment*, because it was the fact that the interval of u values in that vertical segment in $B_{V,U}$ "crossed" the vertical asymptote of the indifference curve of V that led to two (non-floating) horizontal segments in $B_{W,V}$.

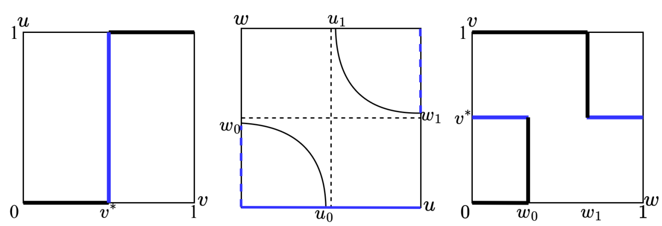

Figure 9: (Left) $B_{V,U}$ with the segment $I \equiv [0, 1] \times \{v^*\}$ highlighted (in blue). (Center) The indifference curve with the $u \in [0, 1]$ indicated by the solid highlight and the $w \in [0, w_0]$ and $w \in [w_1, 1]$ indicated by the dashes highlights. (Right) $B_{W,V}$ with the two split segments resulting from I highlighted.

Lastly, consider V's responses when $\Delta_V \neq 0$.

Case: $v = 0$**.** In this case, $(u, v) = (0, 0)$. Figure 10 illustrates the generation process involving this point, which indicates the best response of U when V plays a strategy of 0. Player V can only play $v = 0$ if $\Delta_V \leq 0$. As $u = 0$ at this point, we consult the indifference curve to determine the range of w values where $(0, w)$ is on the indifference curve or on the side of the indifference curve where $\Delta_V(u, w) < 0$.

We see in Figure 10 that U's only response to V playing 0 is to play 0 as well. The structure of the indifference curve function is such that for any indifference curve, either $\Delta_V < 0$ for all combinations of u and w located in the region between the two curves and $\Delta_V > 0$ for all combinations of u and w located in one of the two outer regions, or vice versa. The respective signs can be determined by evaluating Δ_V at the intersection of the asymptotes, as this point will always be in the middle region. For the purpose of this example, we assume the central region of the indifference curve to be positive and the outer regions to be negative. We thus see that at $u = 0$, the negative and indifferent regions of the indifference curve correspond to $w \in [0, w_0]$. This results in the horizontal segment $[0, w_0] \times \{0\}$ in $B_{W,V}$.

Case: $v = 1$**.** In this case, $(u, v) = (1, 1)$. Figure 11 illustrates the generation process involving this point, which indicates the best response of U when V plays a strategy of 1. Player V can only play $v = 0$ if $\Delta_V \leq 0$. As $u = 1$ at this point, we consult the indifference curve to determine the range of w values where $(1, w)$ is on the indifference curve or on the side of

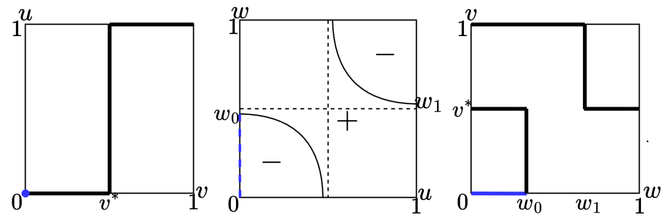

Figure 10: (Left) $B_{V,U}$ with $(0, 0)$ marked. (Center) The indifference curve with the $w \in [0, w_0]$ indicated by the highlight. (Right) $B_{W,V}$ with the resulting pure strategy segment at $v = 0$ highlighted.

the indifference curve where $\Delta_V(u, w) < 0$.

We see in Figure 11 that U's only response to V playing 1 is to play 1 as well. Following the same sign assumptions as the previous case, the central region of the indifference curve is positive. We thus see that at $u = 1$, the positive and indifferent regions of the indifference curve correspond to $w \in [0, w_1]$. This results in the horizontal segment $[0, w_1] \times \{0\}$ in $B_{W,V}$.

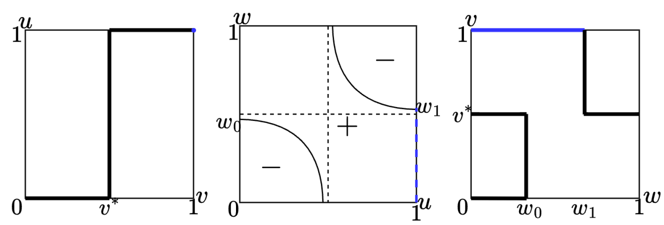

Figure 11: (Left) $B_{V,U}$ with $(1, 1)$ marked. (Center) The indifference curve with $w \in [0, w_1]$ indicated by the highlight. (Right) $B_{W,V}$ with the resulting pure strategy segments at $v = 1$ highlighted.

We thus see that the completed worst-case second best response policy contains 6 segments, 2 w-indifference points at w_0 and w_1, and 3 v-indifference points at 0, v^*, and 1. Observe that this is true regardless of the specific worst-case configuration.

B GRAPHICAL GAMES ON PATHS

B.1 WORST CASE ANALYSIS

Lemma 1. *All horizontal segments at the same u-indifference point in $B_{V,U}$ result in vertical segments at the same w-indifference point in $B_{W,V}$. In the worst case, each of these horizontal segments results in a distinct vertical segment.*

Proof. See Figure 12 for an illustration of the proof. To obtain the segment(s)[1] in $B_{W,V}$ resulting from a horizontal segment in $B_{V,U}$, first find the w value on the hyperbolic indifference curves corresponding to the u-indifference point of the horizontal segment. Here, we use the indifference curves to find w because the range of v values for the horizontal segment implies that V is indifferent. That w value is the w-indifference point for the resulting vertical segment in $B_{W,V}$. Because the hyperbolic indifference curves do not overlap themselves at any u value, there will be at most one w value on the curve for any u value. This implies that multiple horizontal segments at the same u-indifference point must result in vertical segments at the same w-indifference point.

Additionally, the range of v values for a vertical segment corresponds to the range of v values for the horizontal segment. This implies that a horizontal segment cannot result in more than one vertical segment. This is shown in Figure 12 using two segments colored pink and cyan. □

[1]We show in the next paragraph that there can be at most one segment.

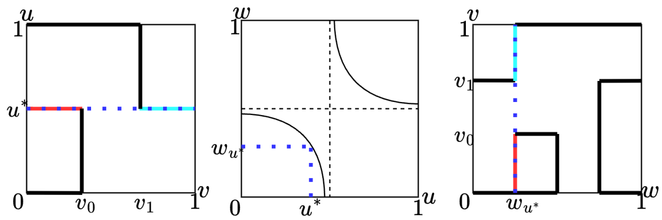

Figure 12: (Left) $B_{V,U}$ with two horizontal segments at the $u = u^*$. (Center) The indifference curve with u^* and the corresponding w_{u^*} which lies on the curve. (Right) $B_{W,V}$ with the two resulting vertical segments at $w = w^*$.

Lemma 2. *The number of vertical segments C_j in $B_{W,V}$ is at most R_{j-1}.*

Proof. The graph of $B_{V,U}$ has R_{j-1} horizontal segments, each with a fixed u value corresponding to its u-indifference point. So, it will result in at most one segment at a fixed w-indifference point if there is a corresponding w value on the indifference curve. By Lemma 1, in the worst case, every segment in a u-indifference point of $B_{V,U}$ is mapped to another segment in the same w-indifference point of $B_{W,V}$. So, each horizontal segment of $B_{V,U}$ is mapped to one vertical segment of $B_{W,V}$ in the worst case. See Figure 13 for an illustration. □

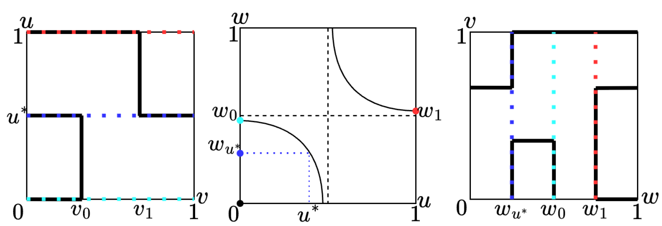

Figure 13: (Left) $B_{V,U}$ with the 3 horizontal indifference point values are highlighted with different colors. (Center) The indifference curves with the w value each u-indifference point maps to are highlighted using the same colors as before. (Right) $B_{W,V}$ with the mapped vertical indifference points.

Lemma 3. *A vertical segment from $B_{V,U}$ can result in at most two horizontal segments in $B_{W,V}$, and if it results in two horizontal segments, they will both be non-floating.*

Proof. Consider the range of u values for a vertical segment at a v-indifference point v_1 in $B_{V,U}$. The splitting of the corresponding segment in $B_{W,V}$ must be the result of the u range of the vertical segment crossing the vertical asymptote of the indifference curves and overlapping both curves. Assuming the non-degenerate worst case, one of these curves goes to $w \to +\infty$ as it approaches that asymptote, being cut off at $w = 1$, while the other goes to $w \to -\infty$, being cut off at $w = 0$. The two ranges of w values lead to two horizontal segments at v_1 in $B_{V,W}$, both of which are non-floating (because they end at $w = 0$ and $w = 1$). See Figure 14 for visualization. □

Lemma 4. *If the u-interval of a vertical segment at $v = v_1$ of $B_{V,U}$ does not overlap with both of V's hyperbolic indifference curves, then it will result in at most one horizontal segment s in $B_{W,V}$. Furthermore, if s is non-floating, then no other vertical segment at $v = v_1$ of $B_{V,U}$ can overlap with both curves of V's indifference hyperbola.*

Proof. First, the splitting of the horizontal segment in $B_{W,V}$ cannot happen if the u-interval of the corresponding vertical segment in $B_{V,U}$ does not overlap with both of V's hyperbolic indifference curves. Consider a vertical segment in $B_{V,U}$ that results in a horizontal segment s in $B_{W,V}$. In order for s to be non-floating, one of its end points must have a w value of 0 or 1. This requires the u-interval of the corresponding vertical segment in $B_{V,U}$ overlapping with either of the two indifference curves at the u point closest to the vertical asymptote. This prevents any other vertical u-interval at $v = v_1$ in

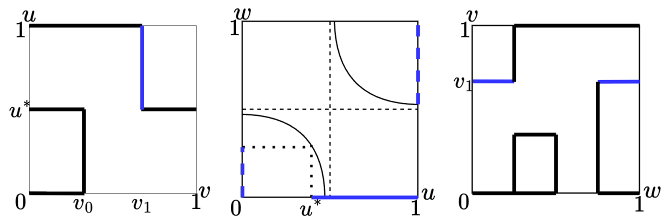

Figure 14: (Left) $B_{V,U}$ with the vertical segment that will be split highlighted. (Center) The indifference curve with that vertical segment's u interval highlighted on the u axis and the corresponding w intervals indicated by the two highlighted dashed lines. (Right) $B_{W,V}$ with the resulting two horizontal segments highlighted.

$B_{V,U}$ to overlap with both indifference curves because vertical segments at the same v-indifference point cannot overlap one another (otherwise, they would merge together into one vertical segment). See Figure 15 for an illustration. □

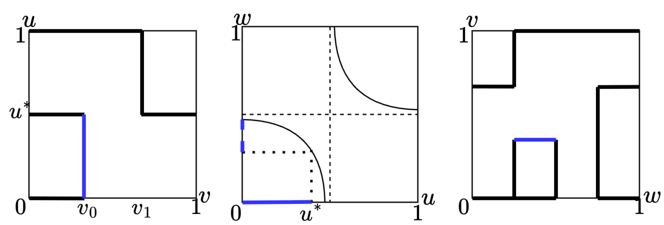

Figure 15: (Left) The vertical segment in $B_{V,U}$ that will not be split is highlighted in blue. (Center) The indifference curve with that vertical segment's u interval is highlighted on the u axis and the corresponding w interval is indicated by the dashed line. (Right) The resulting horizontal segment in $B_{W,V}$ is shown in blue.

Lemma 5. *All vertical segments at $v = v_1$ in $B_{V,U}$ result in horizontal segments at $v = v_1$ in $B_{W,V}$.*

Proof. For any vertical segment at $v = v_1$ in $B_{V,U}$, its u-interval and V's indifference curves determine the w-interval(s) of the corresponding horizontal segment(s) in $B_{W,V}$. Notably, the w-interval(s) remain at the same $v = v_1$ in $B_{W,V}$. Therefore, all vertical segments at a certain $v = v_1$ in $B_{V,U}$ lead to horizontal segments in $B_{W,V}$ at the same v-indifference point $v = v_1$. See Figure 16 for an illustration. □

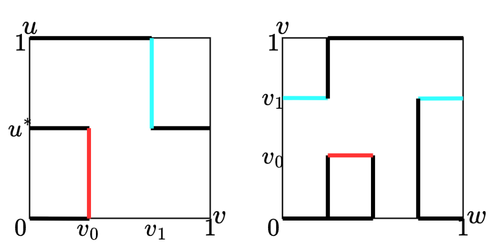

Figure 16: (Left) Two vertical segments at $v = v_0$ and $v = v_1$ in $B_{V,U}$ are highlighted in different colors. (Right) The horizontal segments resulting from the vertical segments are highlighted in the same color. See Figures 14 and 15 for the indifference curves used to determine the resultant segments.

Lemma 6. *$B_{W,V}$ can have at most three pure-strategy segments.*

Proof. As explained in Appendix A.4, the middle region and the two outer regions separated by V's hyperbolic indifference curves each indicate combinations of u and w which allow a pure strategy of either 0 or 1. For any worst-case construction in terms of the number of segments, each of the two pure strategies will correspond to one of the following two cases.

Case 1. *The pure strategy considered corresponds to the two outer regions of the indifference curve.*

For each best response of u to this pure strategy of v, if a corresponding w-interval exists, it must be within one of the two outer regions of the indifference curve. Since one of the outer regions contains $w = 0$ and the other $w = 1$, the w-interval must have either $w = 0$ or $w = 1$ as an endpoint. Multiple w-intervals having the same endpoint of $w = 0$ or 1 will merge into one horizontal pure-strategy segment in $B_{W,V}$. Since the two outer regions do not overlap, there can be at most two horizontal pure-strategy segments in $B_{W,V}$ for this pure strategy. See Figure 17 for visualization.

Case 2. *The pure strategy considered corresponds to the central region between the indifference curves.*

For each best response of U to this pure strategy, the corresponding w-interval will cross the horizontal asymptote and will have either $w = 0$ or $w = 1$ as an endpoint due to the geometric shape of the central region. Therefore, all corresponding w-intervals in this region will overlap at the horizontal asymptote, resulting in a single segment for this pure strategy. See Figure 18 for visualization. ☐

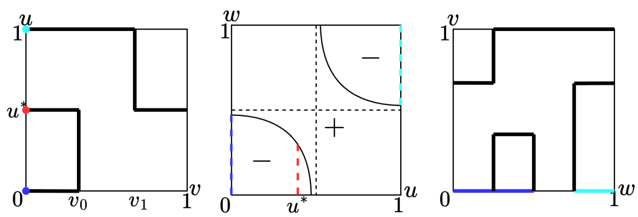

Figure 17: (Left) $B_{V,U}$ with all of U's best responses to V's pure strategy $v = 0$ marked with different colors. (Center) V's indifference curves split the $[0,1]^2$ region into $-$ and $+$, where the $-$ regions represent $\Delta_V(u,w) \leq 0$ and are the outer regions in this example. The w-intervals corresponding to U's best responses to $v = 0$ are drawn using the same color as in $B_{V,U}$. (Right) $B_{W,V}$ shows these w-intervals for $v = 0$. We call these V's pure strategy segments. There is no red pure strategy segment on the account that the w-interval of $u = u^*$ is a sub-interval of the w interval corresponding to $u = 0$, which is shown in dark blue.

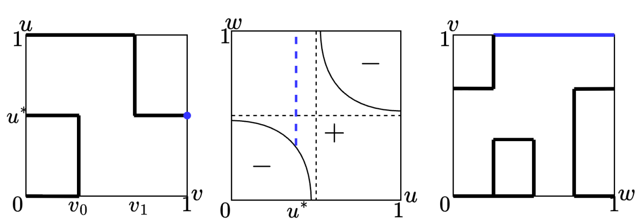

Figure 18: (Left) $B_{V,U}$ with U's best response to $v = 1$ marked in blue. (Center) The central region between the indifference curves represent $\Delta_V(u,w) \geq 0$. The w-interval corresponding to U's best response to $v = 1$ is shown with a blue dashed line. (Right) $B_{W,V}$ with the $v = 1$ pure-strategy segment drawn in blue.

Lemma 7. *In the worst case, the j-th message $B_{W,V}$ has j w-indifference points and $j+1$ v-indifference points.*

Proof. The proof is by induction. As shown for the base case ($j = 1$), the first message contains one horizontal axis indifference point and two vertical axis indifference points, so the base case of the lemma is true. For the induction step, assume that $B_{V,U}$ has j u-indifference points and $j-1$ v-indifference points. By Lemma 1, $B_{W,V}$ will have j w-indifference points. By Lemma 5, $B_{W,V}$ will have $j-1$ v-indifference points, resulting from the indifference points of $B_{V,U}$. Additionally, there will be a v-indifference point in $B_{W,V}$ at each of the two pure strategies of v as shown in Lemma 6. The total number of v indifference points will therefore be $j+1$. See Figure 19 for visualization. ☐

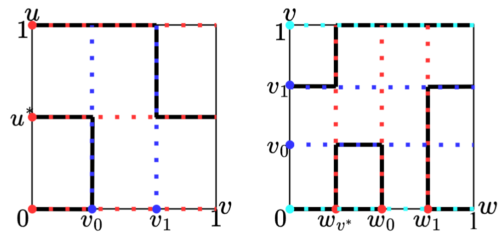

Figure 19: (Left) $B_{V,U}$ with the v-indifference points marked in blue and the u-indifference points marked in red. (Right) $B_{W,V}$ with the w- and v-indifference points in the same color as the indifference points from which they are derived. The pure strategy indifference points are shown in cyan.

Lemma 8. *If j is odd, in the worst case, a horizontal line at some $v = v^*$ crosses the interior of one vertical segment at each w-indifference point of $B_{W,V}$.*

Proof. The proof is by induction. For the base case ($j = 1$), as shown in Figure 1, there is only one indifference point with a vertical segment incident on it. Therefore, the base case of this lemma is trivially true.

For the induction step, assume that at an odd j, $B_{U,T}$ has a horizontal line at some t that crosses the interior of one vertical segment at each u-indifference point. We show that the lemma holds for $j + 2$ represented by $B_{W,V}$. First, derive the t-interval of the intersection of all these vertical segments in $B_{U,T}$. Using Lemma 3, in the worst case, this t-interval can cross the vertical asymptote of U's indifference curves, thereby resulting in a split of the horizontal segments at each u-indifference point in $B_{V,U}$. By Lemmas 3 and 5, there will be two non-floating horizontal segments at each mixed-strategy u-indifference point of $B_{V,U}$.

Additionally, by Lemma 6 and Proposition 1, there are at most three non-floating pure-strategy segments of U in $B_{V,U}$. The proof of Lemma 6 shows that U's pure-strategy segments arise from the two outer regions and the central region created by U's hyperbolic indifference curves. Therefore, in the worst case, one of these pure-strategy segments (either at $u = 0$ or $u = 1$) will have the following properties:

1. If it is at $u = 0$, it has an overlapping v-value with a pure-strategy segment at $u = 1$, and vice versa.
2. It must overlap with the whole v-interval of one of U's two hyperbolic indifference curves (see Figures 17 and 18 but note that those figures are in the context of V's indifference curves, not U's).

Combining the above two properties with the splitting of horizontal segments in $B_{V,U}$ described above, in the worst case, there is a v value in $B_{V,U}$ that overlaps with a horizontal segment at each u-indifference point. See Figure 20 for an illustration.

By Lemma 1, in the worst case, horizontal segments at different u-indifference points in $B_{V,U}$ result in vertical segments at different w-indifference points in $B_{W,V}$ (recall that $B_{W,V}$ represents the next odd number after j). As the proof of Lemma 1 lays out, the v-intervals of the horizontal segments of $B_{V,U}$ are the same as the v-intervals of the corresponding vertical segments of $B_{W,V}$. Therefore, the statement holds. □

Lemma 9. *If j is even, in the worst case, a horizontal line at some $v = v^*$ crosses the interior of a vertical segment at all but one w-indifference point of $B_{W,V}$, and no horizontal line crosses a vertical segment at each w-indifference point.*

Proof. The proof is by induction, the base case being the second message, where the first message is S-shaped. As shown in Figure 3, there are two indifference points on the horizontal axis. Each has a vertical segment, and the interiors of these two vertical segments do not have any common vertical-axis points. Therefore, the base case is trivially true.

Before working on the induction step, we present the intuition here. In the worst case, one of the two vertical segments of the base case will be split into two non-floating horizontal segments in the next message, and the other will lead to a floating horizontal segment. The interiors of the three resulting horizontal segments do not overlap (see Figures 14 and 15). Next, these horizontal segments will lead to vertical segments whose interiors do not overlap either. In fact, as shown in red in

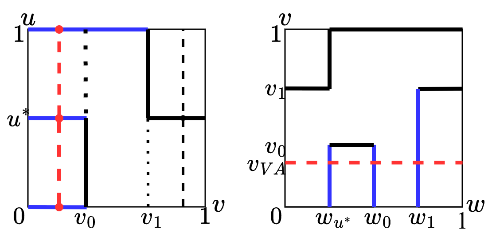

Figure 20: (Left) $B_{V,U}$ with a vertical line that intersects with a horizontal segment at each u-indifference point. (Right) $B_{W,V}$ with the mapping of the line and the corresponding vertical segments.

Figure 4, none of their respective descendant segments will have overlapping interiors in future messages. Therefore, for any even j, no horizontal line can cross the interior of a vertical segment from each of the two descendant sets.

For the induction step, assume that at an even j, $B_{U,T}$ satisfies the statement. Consider the vertical segments (at all but one u-indifference points) whose interiors are crossed by a horizontal line at some t. Compute the t-interval t_I as the intersection of these vertical segments. Let u_0 denote that special u-indifference point not represented by these vertical segments. In the worst case, U's indifference curves are such that the vertical asymptote lies on t_I and each segment that crosses the asymptote overlaps with the curve on either side of it. $B_{V,U}$ will therefore contain two non-floating horizontal segments at each u-indifference point other than u_0. Segments at both pure strategies of U in $B_{V,U}$ will again be non-floating.

For the next even-numbered messages $B_{W,V}$, we will prove the following claims:

1. There is no horizontal line in $B_{W,V}$ that crosses the interior of a vertical segment at each w-indifference point.
2. There is a horizontal line that crosses the interior of a vertical segment at all but one w-indifference point in $B_{W,V}$.

For the first claim, if there is a horizontal line at $v = v'$ in $B_{W,V}$ that crosses the interior of a vertical segment at each w-indifference point, then there is a vertical line at $v = v'$ in $B_{V,U}$ that crosses the interior of a horizontal segment at each u-indifference point, including $u = 0$ and $u = 1$. This is because by Lemma 1, the v-range of a vertical segment in $B_{W,V}$ is the same as the v-range of the corresponding horizontal segment in $B_{V,U}$. Now, consider the previous message $B_{U,T}$. Since there is a one-to-one mapping between t and v values on U's hyperbolic indifference curves, suppose (t', v') is on U's indifference curve. Then, there is a horizontal line at $t = t'$ in $B_{U,T}$ that crosses a vertical segment at each u-indifference point. This contradicts our assumption about $u = u_0$ in $B_{U,T}$.

We prove the second claim by construction. We start with $B_{U,T}$. We have a horizontal line at $t = t'$ that intersects a vertical segment at each u-indifference point except u_0. Let $s \equiv \{u_0\} \times t_s$ be any vertical segment at u_0 in $B_{U,T}$ with its vertical range being t_s. First, there is no overlap between the interiors of t_s and t_I (which is the t-interval computed earlier as the intersection of the vertical segments in question). Otherwise, our assumption about u_0 would be violated. Therefore, s can only lead to a floating horizontal segment in $B_{V,U}$. Using the geometry of U's indifference curves, there is a one-to-one mapping between t and v values on U's indifference curves. Therefore, the v-intervals for t_s and t_I do not have overlapping interiors. Using the same mapping reasoning, there is a vertical line at $v = v'$ in $B_{V,U}$ that crosses the interior of a horizontal segment at each u-indifference point except u_0 (see Figure 21). Using Lemma 1, these horizontal segments lead to vertical segments in $B_{W,V}$ keeping their v-ranges the same. Therefore, the horizontal line at $v = v'$ in $B_{W,V}$ crosses a vertical segment at each w-indifference point except one that corresponds to u_0 in $B_{V,U}$. As in the proof of Lemma 8, we can incorporate pure-strategy segments in the above reasoning. $\square$

Lemma 10. *For $j > 2$, the maximum number of horizontal segments in $B_{W,V}$ is $C_{j-1} + j + 2 - j\%2$.*

Proof. We divide the induction into two cases.

Case 1. *j is odd.*

At the even $j - 1$ step, we have $B_{V,U}$. By Lemma 9, in the worst case, there is a horizontal line at some u that crosses a vertical segment at all but one v-indifference points. As in the proof of Lemma 9, each of these vertical segments is split into two horizontal segments in $B_{W,V}$. Therefore, in the worst case, the number of such vertical segments of $B_{V,U}$ that are

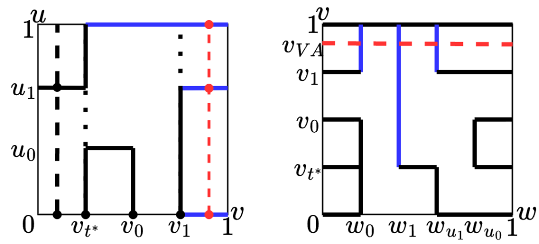

Figure 21: (Left) $B_{V,U}$ with a vertical line crossing a horizontal segments at each u-indifference point except u_0. (Right) $B_{W,V}$ with the mapping of the line and the vertical segments corresponding to those horizontal segments.

split into two horizontal segments in $B_{W,V}$ will be the number of v indifference points in $B_{V,U}$ minus one, which is $j-2$ according to Lemma 7. Each remaining vertical segment of $B_{V,U}$ leads to a horizontal segment in $B_{W,V}$. Since there are a total of C_{j-1} vertical segments in $B_{V,U}$ by definition, $B_{W,V}$ contains $C_{j-1}+j-2$ mixed-strategy horizontal segments.

Case 2. *j is even.*

Now, $B_{V,U}$ corresponds to an odd $j-1$. By Lemma 8, in the worst case, there is a horizontal line that crosses the interior of a vertical segment at each v-indifference point. The total number of v-indifference points in $B_{V,U}$ is $j-1$ by Lemma 7, and the total number of vertical segments in $B_{V,U}$ is C_{j-1}. Using the same reasoning as the previous case, we arrive at $C_{j-1}+j-1$ mixed-strategy horizontal segments in $B_{W,V}$.

These two cases can be combined in the general statement that the number of mixed-strategy horizontal segments in $B_{W,V}$ is $C_{j-1}+j-1-j\%2$. There will be at most three pure-strategy horizontal segments by Lemma 6, resulting in a total of $C_{j-1}+j+2-j\%2$ horizontal segments in the worst case. $\square$

B.2 TIGHT EXAMPLES

Following the main idea presented in Section 4.2, given $B_{V,U}$, we first build a set E of the endpoints of the vertical segments in it. We then sort E with respect to the u-coordinates. We find two successive endpoints E_1^* and E_2^* in E such that a horizontal line at their midpoint intersects with the maximum number of vertical segments. We use the midpoint's u-coordinate m^* as the location of the vertical asymptote for V's hyperbolic indifference curves. The intention is to maximize the number of splits when we convert the vertical segments in $B_{V,U}$ to horizontal segments in $B_{W,V}$. For that, we also make sure that the u-coordinates of E_1^* and E_2^* are on the hyperbola so that no vertical segment of $B_{V,U}$ is unmapped.

Next, we iterate through all the segments (horizontal and vertical) of $B_{V,U}$ to map the u values of their endpoints to w values using the generated indifference curves. If the endpoints are on opposite sides of the vertical asymptote, we create two non-floating segments from these endpoints using Lemma 3. If the endpoints are on the same side of the vertical asymptote, we create a single segment connecting these endpoints in accordance with Lemmas 4 and 5. We then check for pure-strategy segments in the exact manner described in Lemma 6. For dealing with the signs of the outer and central regions, we use Proposition 3. The list containing all of the newly-generated segments constitutes the next best-response policy.

We present the detailed pseudocode in Algorithm 1.

C DEGENERATE CASES OF INDIFFERENCE FUNCTION

Recall that the delta function of a player on a path takes the form:

$$\Delta_V(u,w) = c_3uw + c_2u + c_1w + c_0. \tag{1}$$

The above results on bounding the growth of segments operate under the generic case that any u-interval in $B_{V,U}$ can be mapped to a potential w-interval in $B_{W,V}$ using the following linear rational function.

Algorithm 1 Construct Worst-Case Instance

Input: List S of all segments in current worst case instance $B_{V,U}$.
Output: List S' of all segments in next worst case instance $B_{W,V}$.

(Hyperbola Generation)

Iterate through the elements of S, adding the endpoints of each vertical segment to new list E.
Sort E with respect to u-coordinates.
Iterate through E. Find the adjacent elements E_1^*, E_2^* for which the average u-value m^* lies on the maximum number of vertical segments.
Choose a, b, c, and d for the indifference curve hyperbola so that (1) the vertical asymptote $-d/c$ occurs at m^* and (2) the indifference curve is present at the u-coordinates of E_1^* and E_2^*.

(Mapped Segment Generation)

Initialize $S' = \emptyset$.
for each segment $s \in S$ **do**
    Input the u-coordinates of s into the indifference curve hyperbola to get the corresponding w-coordinates in $B_{W,V}$.
    If the w-coordinates are on opposite sides of $-d/c$, create two segments with w-intervals from those coordinates to the nearest pure-strategy of w at the same v-coordinate as s. Add these segments to S'.
    If the w-coordinates are on the same side of $-d/c$, create a single segment with these two coordinates as the endpoints of its w-interval over the same v-interval as the originating segment. Add this segment to S'.
end for

(Pure-Strategy Segment Generation)

Create an empty list L_s.
for $s \in S$ with an endpoint at $v = 1$ **do**
    If $cu + d > 0$ at this endpoint, create a segment at $v = 1$ with w-endpoints at the indifference curve output and 1 (using Proposition 3). Add this segment to L_s.
    If $cu + d < 0$ at this endpoint, create a segment at $v = 1$ with w-endpoints at the indifference curve output and 0. Add this segment to L_s.
end for

for $s \in S$ with an endpoint at $v = 0$ **do**
    If $cu + d < 0$ at this endpoint, create a segment at $v = 0$ with w-endpoints at the indifference output and 1. Add this segment to L_s.
    If $cu + d > 0$ at this endpoint, create a segment at $v = 0$ with w-endpoints at the indifference output and 0. Add this segment to L_s.
end for

Combine any overlapping segments in L_s, then add all elements of L_s to S'.
return S'

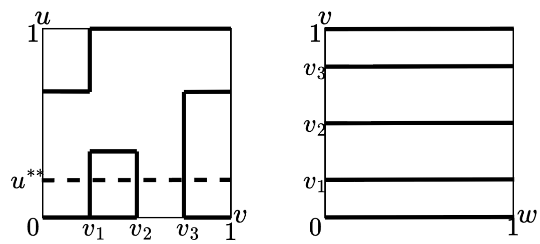

Figure 22: (Left) $B_{V,U}$ with the vertical asymptote of the indifference curve superimposed. (Right) the resulting $B_{W,V}$ if $c_3 = c_1 = 0$ in the indifference curve $\Delta_V(u, w) = 0$.

$$w = -\frac{c_2 u + c_0}{c_3 u + c_1} \tag{2}$$

We next address the cases in which V is no longer interdependent on one or both of its neighbors, thereby preventing the use of the hyperbolic indifference curves.

Case 1. $c_3 = c_1 = 0$

In this case, V is indifferent to the strategy of W when choosing its own strategy. The delta function of V is a linear function only in terms of u, and $\Delta_V = 0$ at $u^{**} \equiv -\frac{c_0}{c_2}$. As a result, $B_{W,V}$ will have a horizontal segments over all $w \in [0, 1]$ at every $\{v_i \mid B_{V,U}(v_i, u^{**}) = 1\}$. The size of this set of v_i, and by extension the number of mixed strategy horizontal segments in $B_{W,V}$, is bounded by the number of vertical indifference points in $B_{V,U}$.

To account for the best response of V playing a pure strategy of 1, check each best response u_i of player U to player V playing 1 from $B_{V,U}$ in Δ_v. If there is any best response u_i such that $\Delta_v(u_i) \geq 0$, then it is possible for V to play the pure strategy of 1, so $B_{W,V}$ contains a segment over $[0, 1] \times \{1\}$. The presence of a pure-strategy segment at $v = 0$ is derived in the same manner.

An example of this case is shown in Figure 22.

Case 2. $\frac{c_2}{c_3} = \frac{c_0}{c_1} = \alpha$*,* $c_1 \neq 0$*, and* $c_3 \neq 0$

In this case, we can re-express Δ_V as follows.

$$\Delta_V(u, w) = (w + \alpha)(c_3 u + c_1) \; .$$

$$\Delta_V(u, w) = 0 \iff w = -\alpha \text{ or } u = u^{**} \equiv -\frac{c_1}{c_3} = -\frac{c_0}{c_2} \; . \tag{3}$$

$$\Delta_V(u, w) > 0 \iff (w > -\alpha \text{ and } u > u^{**}) \text{ or } (w < -\alpha \text{ and } u < u^{**}) \; . \tag{4}$$

Using Condition 3, V becomes indifferent with respect to any strategy of W at $u^{**} \equiv -\frac{c_1}{c_3} = -\frac{c_0}{c_2}$. This leads to a similar analysis as Case 1. As shown in Figure 23, for each v_i such that $B_{V,U}(v_i, u^{**}) = 1$, we have a $[0, 1]$ horizontal segment at the v_i-indifference point in $B_{W,V}$.

Condition 3 further states that V becomes indifferent with respect to both of its neighbors U and W when $w = -\alpha$. This leads to a vertical $[0, 1]$ segment at $w = -\alpha$ in $B_{W,V}$. Note that this vertical segment is consistent with $B_{V,U}$ because U has some best response to any v. This is shown in Figure 23.

Next, for V's pure-strategy segments in $B_{W,V}$, Condition 4 states that these segments depend on both u and w. Following are the two possibilities (not mutually exclusive) of V's pure-strategy segments at $v = 1$.

1. For some $u > u^{**}$, $B_{V,U}(u, 1) = 1$: This leads to a $(-\alpha, 1]$ horizontal-strategy segment at $v = 1$ in $B_{W,V}$.
2. For some $u < u^{**}$, $B_{V,U}(u, 1) = 1$: This leads to a $[0, -\alpha)$ horizontal segment at $v = 1$ in $B_{W,V}$.

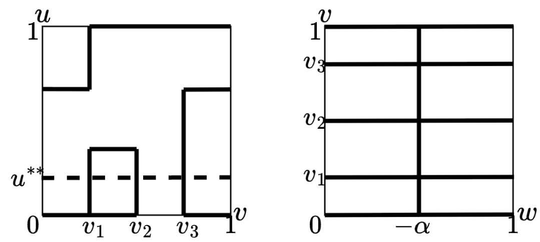

Figure 23: $B_{V,U}$ with the vertical asymptote of the indifference curve superimposed (left), and the resulting $B_{W,V}$ if $\frac{c_2}{c_3} = \frac{c_0}{c_1} = \alpha$ in the indifference curve (right).

Both of the above segments are present in the example shown in Figure 23. The pure-strategy segments at $v = 0$ can be derived in the same fashion.

The above analysis shows that the degenerate cases do not allow splitting a vertical segment into two horizontal segments which the hyperbolic indifference curves allow. This leads to the following property about degenerate cases.

Proposition 4. *For path graphical games with $n \geq 3$, considering the worst-case segment growth, the degenerate cases lead to strictly fewer segments than the generic case of hyperbolic indifference curves.*

D BREAKPOINT POLICIES

A breakpoint policy is a reduction of a best response policy which only contains a single connected component with endpoints at $(0, u_0)$ and $(1, u_1)$. It can be shown that if the upstream pass is performed with breakpoint policies in, the jth message will contain no more than $2j + 1$ segments by induction.

The base case is the same as that for generic best response policies.

The induction step is trivial for the indifference cases explained in the previous section; it should be clear from the explanations and Figures 22 and 23 that a breakpoint policy will contain at most 1 segment in the case of total w-indifference and 3 segments in the case of conditional indifference.

For the induction step on the generic case, assume that the jth breakpoint policy $BP_{V,U}$ has $2j + 1$ segments. We divide the mapping of the $j + 1$th breakpoint policy $BP_{W,V}$ into three cases based on the relation of the connected component to the indifference curve's vertical asymptote at u^*.

Case 1. *The connected component of $BP_{V,U}$ does not cross the vertical asymptote.*

As the connected component does not cross the vertical asymptote, none of the segments of $BP_{V,U}$ are split. This means that in the worst case, every segment of $BP_{W,U}$ results in 1 segment in $BP_{W,V}$. As the v values and intervals are not changed in this segment mapping, the mapped component in $BP_{W,V}$ will cover the entire $[0, 1]$ vertical interval. The endpoints of the mapped component will therefore be located at the pure strategies of V.

Because player U's responses to both pure strategies of V are on the same side of the indifference curve with respect to the vertical asymptote, the segments will be non-floating at opposite pure strategies of w, ensuring that the message contains a response of V to every possible strategy of W.

$BP_{W,V}$ will therefore have at most $2j + 1$ segments directly mapped from $BP_{V,U}$ as well as 2 pure strategy segments for a total of $2(j + 1) + 1$ segments.

An example mapping of a breakpoint policy through a player with this case can be found in Figure 24.

Case 2. *The connected component of $BP_{V,U}$ crosses the vertical asymptote, with u_0 and u_1 on the same side of the vertical asymptote.*

To generate a breakpoint policy in this case, one only needs to consider two subcomponent of $BP_{V,U}$: the component traced

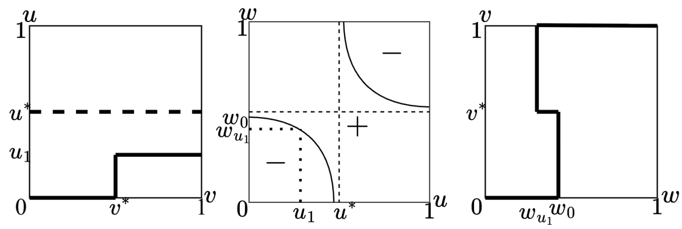

Figure 24: $B_{V,U}$ which does not cross the vertical asymptote in u (left), the indifference curve from $B_{V,U}$ to $B_{W,V}$ (center), and the resulting message $B_{W,V}$ (right).

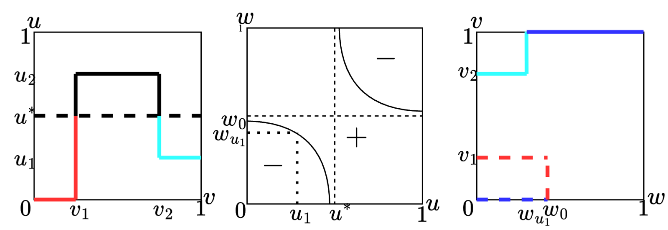

Figure 25: $B_{V,U}$ which crosses the vertical asymptote in u an even number of times (left), the indifference curve from $B_{V,U}$ to $B_{W,V}$ (center), and the resulting message $B_{W,V}$ (right).

from $(0, u_0)$ forward to the first intersection with $u = u^*$ and the component traced from $(1, u_1)$ to the last intersection with $u = u^*$.

Every segment or partial segment in each subcomponent would be mapped to at most one segment in $BP_{W,V}$ since the subcomponents do not cross the vertical asymptote. The mapping of the subcomponent with an endpoint at $v = 0$ in $BP_{V,U}$ would have an endpoint at $v = 0$, and the mapping of the subcomponent with an endpoint at $v = 1$ in $BP_{V,U}$ would have an endpoint at $v = 1$ in $BP_{W,V}$. Since both components approach $u = u^*$ from the same side, their other endpoints will either both be located at $w = 0$ or both be located at $w = 1$.

To determine which subcomponent we need for $BP_{W,V}$, we check the pure strategy segment of each. The mapping of each subcomponent from $BP_{V,U}$ extends to a different pure strategy of V. As we observed in the previous case, as both components are on the same side of u^*, the pure strategy segments would be non-floating and extend to opposite pure strategies of w. As both subcomponent mappings already extend to one pure strategy of w, the union of one of the mappings and its pure strategy segment will be a complete breakpoint policy.

A subcomponent of $B_{V,U}$ can contain at most $2j$ segments, as there must be at least one horizontal segment exclusive to each subcomponent in the generic case. With the additional pure strategy segment, $BP_{W,V}$ can contain at most $2j + 1$ segments.

An example mapping of a breakpoint policy through a player with this case can be found in Figure 25.

Case 3. *The connected component of $B_{V,U}$ crosses the vertical asymptote, with u_0 and u_1 on opposite sides of the vertical asymptote.*

In this case, one only needs to consider the same two subcomponents of $BP_{V,U}$ described in the previous case to generate $BP_{W,V}$. Once again, the mapping each traced component from $BP_{V,U}$ will result in a mapped component in $BP_{W,V}$ with an endpoint at an opposite pure strategy of V. Since the subcomponents approach u^* from opposite sides, the mappings of the two subcomponents into $BP_{W,V}$ have their other endpoint at opposite pure strategies of W.

Again like the previous case, the subcomponent whose mapping will be included in $BP_{W,V}$ is determined by checked the pure strategy segments. Because the mappings each include opposite pure strategies of V on opposite sides of the vertical asymptote, both would include non-floating pure strategy segments which extend to the same pure strategy of W. This ensures that the union of one subcomponent mapping and its pure strategy segment is constitutes a breakpoint policy.

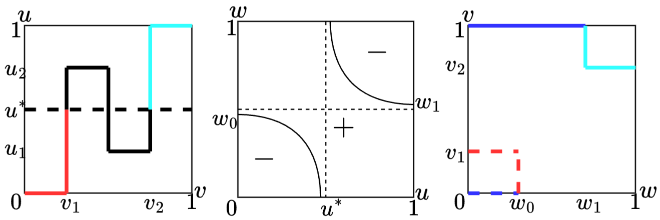

Figure 26: $B_{V,U}$ which crosses the vertical asymptote in u an odd number of times (left), the indifference curve from $B_{V,U}$ to $B_{W,V}$ (center), and the resulting message $B_{W,V}$ (right).

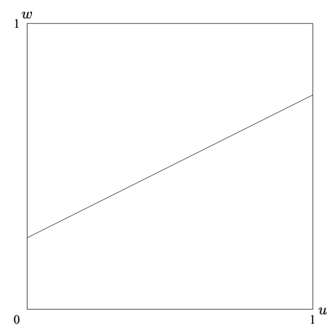

Figure 27: An example worst case indifference curve for a player in a polymatrix path game.

The segment bound of $BP_{W,V}$ is $2j + 1$ by the same argument as in the previous case.

An example mapping of a breakpoint policy through a player with this case can be found in Figure 26.

E POLYMATRIX GAMES ON PATHS

E.1 SEGMENT GROWTH

For polymatrix games on a path, we show that the j-th best response policy contains at most $2j + 1$ segments by induction.

Since a player in a polymatrix game has a 2×2 matrix for each player it shares an edge with and a graphical game player has a single size 2^{d+1} matrix, where d is its number of neighbors, the base case is the same as the base case for graphical games. Note that in the worst case, the base case message contains one horizontal segment at each pure strategy of U.

For the induction step, consider a player V who has received a message from player U and is sending a message to player W. Player V's expected payoff is the sum of the expected payoffs from its combinations of strategies with players U and W. The expected payoff can be rearranged to extract the delta function similarly to the graphical case, yielding:

$$\Delta_V(u, w) = c_2 w + c_1 u + c_0 = 0 \; .$$
$$w = \frac{(-c_1)u + (-c_0)}{c_2} \equiv au + b \; .$$

The full derivation of this result is detailed in the next subsection.

Observe that the polymatrix indifference curve takes the form of a linear function. Figure 27 is an example of such a linear indifference curve in the worst-case configuration.

To perform the induction step, assume that the j-th message $B_{V,U}$ contains $2j+1$ segments, including at most one horizontal segment at each pure strategy of U. As the indifference curve does not have a vertical asymptote, each segment of $B_{V,U}$ can

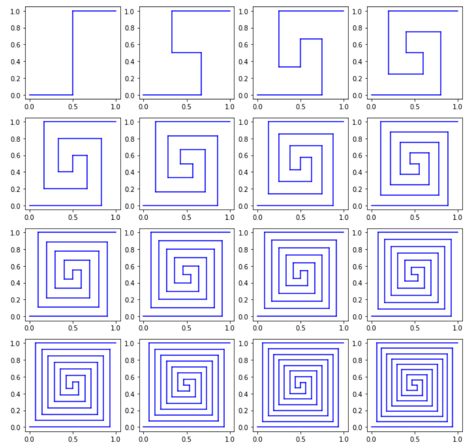

Figure 28: A worst-case tight example of segment growth along a polymatrix path.

be mapped to at most one segment of $B_{W,V}$. Despite the absence of hyperbolic indifference curves, the mapping procedure is the same as shown in Appendix A.4. Since $B_{V,U}$ contains horizontal segments at both $u = 0$ and $u = 1$ and we assume the generic case, a worst-case indifference must have a $w \in (0, 1)$ for every $u \in [0, 1]$, as shown in Figure 27. Otherwise, the positive or negative side of the indifference curve (i.e., a region of $\Delta_V(u, w) \neq 0$), may not have any w-interval at a particular u value, thereby reducing the number of pure-strategy segments.

To generate the horizontal segments at a pure strategy of $v = 1$, $B_{V,U}$ is checked to determine every possible best response of U at $v = 1$. Each one of these U strategies can be referred to V's indifference curve to determine a corresponding interval of W strategies for which $\Delta_v > 0$. Because of the linear construction of the indifference curve, the region where $\Delta_V > 0$ is continuous and extends to at least one pure strategy $w \in \{0, 1\}$ in the worst case. As a result, the union the w intervals corresponding to every U response to $v = 1$ will be one continuous segment. The same argument can be made to show that there will be at most one horizontal segment in $B_{W,V}$ at $v = 0$, leading to a total of two horizontal pure-strategy segments in $B_{W,V}$. Therefore, in the generic case, $B_{W,V}$ has at most $(2j + 1) + 2 = 2(j + 1) + 1$ segments. An example mapping from $B_{V,U}$ to $B_{W,V}$ in the worst case is shown in Figure 5. A tight example of the upper bound can be found in Figure 28.

E.2 POLYMATRIX INDIFFERENCE CURVE DERIVATION

Following Definition 2, let $M_{V,U}(v, u)$ denote V's expected local payoff from a neighbor U. In a polymatrix game, V's total payoff is the sum of local payoffs from its neighbors. We use the term $M_{V,U|u}(1)$, or simply $M_{V|u}(1)$ when clear from context, to denote V's expected local payoff from U when V plays $v = 1$ and U adopts a mixed strategy u. That is, $M_{V|u}(1) = uM_{V,U}(1, 1) + (1 - u)M_{V,U}(1, 0)$. Likewise, $M_{V|u}(0) = uM_{V,U}(0, 1) + (1 - u)M_{V,U}(0, 0)$ means V's expected local payoff from U for $v = 0$ under U's mixed strategy u. We get the following.

$$\begin{aligned}
M_V(u, v, w) \equiv & M_{V,U}(v, u) + M_{V,W}(v, w) \\
\equiv & uvM_{V,U}(1, 1) + u(1 - v)M_{V,U}(0, 1) + (1 - u)vM_{V,U}(1, 0) + (1 - u)(1 - v)M_{V,U}(0, 0) \\
& + vwM_{V,W}(1, 1) + v(1 - w)M_{V,W}(1, 0) + (1 - v)wM_{V,W}(0, 1) + (1 - v)(1 - w)M_{V,W}(0, 0) \\
= & v[M_{V|u}(1) + M_{V|w}(1)] + (1 - v)[M_{V|u}(0) + M_{V|w}(0)] \\
= & v[M_{V|u}(1) + M_{V|w}(1) - M_{V|u}(0) - M_{V|w}(0)] + M_{V|u}(0) + M_{V|w}(0) \\
= & v\Delta_V + M_{V|u}(0) + M_{V|w}(0) \ .
\end{aligned}$$

As in the graphical case, player V has no control over the second and third terms. So, V's best strategy to maximize the expected payoff depends solely on the Δ_V function.

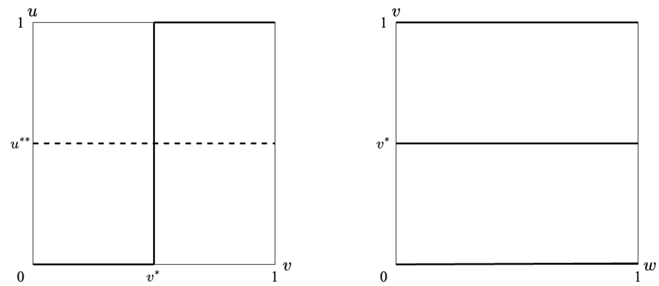

Figure 29: (Left) $B_{V,U}$ with $u = u^{**} = -\frac{c_0}{c_1}$ where $\Delta_V(u^{**}, w) = 0$, irrespective of w due to $c_2 = 0$. (Right) The resulting next message $B_{W,V}$ in this degenerate case.

Let us expand Δ_V and rearrange it into the following form.

$$\begin{aligned}
\Delta_V(u, w) &= M_{V|u}(1) + M_{V|w}(1) - M_{V|u}(0) - M_{V|w}(0) \\
&= u[M_{V,U}(1,1) - M_{V,U}(1,0) - M_{V,U}(0,1) + M_{V,U}(0,0)] \\
&\quad + w[M_{V,W}(1,1) - M_{V,W}(1,0) - M_{V,W}(0,1) + M_{V,W}(0,0)] \\
&\quad + M_{V,U}(1,0) + M_{V,W}(1,0) - M_{V,U}(0,0) - M_{V,W}(0,0) \\
&= c_2 w + c_1 u + c_0 \ .
\end{aligned}$$

Here, the constants c_0, c_1, and c_2 are appropriately defined as follows.

$$\begin{aligned}
c_0 &\equiv M_{V,U}(1,0) + M_{V,W}(1,0) - M_{V,U}(0,0) - M_{V,W}(0,0) \ . \\
c_1 &\equiv M_{V,U}(1,1) - M_{V,U}(1,0) - M_{V,U}(0,1) + M_{V,U}(0,0) \ . \\
c_2 &\equiv M_{V,W}(1,1) - M_{V,W}(1,0) - M_{V,W}(0,1) + M_{V,W}(0,0) \ .
\end{aligned}$$

As in the graphical case, we can set $\Delta_V = 0$ and solve for w in terms of u in order to generate V's indifference curve.

$$\begin{aligned}
\Delta_V(u, w) = c_2 w + c_1 u + c_0 = 0 \\
w = \frac{(-c_1)u + (-c_0)}{c_2} = au + b
\end{aligned}$$

where a and b are appropriately defined.

E.3 DEGENERATE CASES IN POLYMATRIX GAMES

In the case of $c_2 = 0$, Δ_V has no dependence on the strategy of W, similar to the case of $c_3 = c_1 = 0$ for graphical games. We treat this case analogously. $B_{W,V}$ will have a horizontal segment over all $w \in [0, 1]$ at each v-indifference point from the set $\{v_i \mid B_{V,U}(v_i, -\frac{c_0}{c_1}) = 1\}$. The size of this set of v_i, and by extension the number of mixed-strategy horizontal segments in $B_{W,V}$, is bounded by the number of vertical indifference points in the j-th message $B_{V,U}$, which is j.

By the same process as in the analogous graphical case, every U response to each pure strategy of V is checked until one is found for which Δ_V returns an output with the sign that permits the corresponding pure strategy of V. This results in at most 2 additional segments of the form $[0, 1] \times \{0\}$ and $[0, 1] \times \{1\}$. This results in a maximum of $j + 2$ segments in $B_{W,V}$. An example mapping for this case can be found in Figure 29.

Observe that the upper bound for segments in the best response policies of polymatrix path games is the same as the upper bound for segments in breakpoint policies on graphical games. There are two interesting conclusions that we can draw from this. The first is that a worst-case polymatrix path game is irreducile under the breakpoint policy scheme. The second is that in the same $O(n^2)$ time for which a single Nash equilibrium can be guaranteed for a path graphical game, a representation of all Nash equilibria can be found for a path polymatrix game.

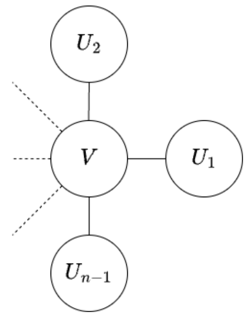

Figure 30: An n-player star, where V is the hub node and each U_i is a spoke node.

F STAR-STRUCTURED GAMES

In this section, we will apply the TreeNash framework of message passing to find a Nash equilibrium in a star-structured game. Although the TreeNash algorithm could be applied to a star rooted at any of its nodes, we treat the hub node as the root. This will help us extend the analysis to caterpillar graphs in the next section.

Every leaf player has a 2×2 payoff matrix and no preceding messages to receive, which is the same situation as the leaf player of a path game. Therefore, every leaf generates and sends a message to the root player in the same manner described in Section 3 and Appendix A.4 and illustrated in in Figure 1. As each message can be generated in constant time, the message propagation process is $O(n)$, linear in the number of players.

Every leaf player U_i thus has 3 distinct responses to a given strategy of V. If $v = v_i^*$, then player U_i can respond with any $u_i \in [0, 1]$. If $v \in [0, v_i^*)$, U_i must respond with a singular pure strategy $u \in \{0, 1\}$ which is determined by its payoff matrix. If $v \in (v_i^*, 1]$, then U_i must respond with the other pure strategy.

Let V^* be set of all strategies v in response to which one or more players are indifferent. For any continuous interval of strategies $v \notin V^*$, every player U_i has a single pure-strategy best response, and therefore, V has a singular payoff and Δ_V. At each $v \in V^*$, one or more U_i is indifferent, creating a segment in the best response policy which connects the pure strategy responses. This is reflected in a Δ_V function with fixed outputs over every continuous non-v_i^* interval, with transitions between these discrete outputs at the v_i^* strategies, as shown in Figure 6. For a Nash equilibrium to occur, it must be the case that $v = 1$ and $\Delta_V \geq 0$, or $v = 0$ and $\Delta_V \leq 0$, or $v \in (0, 1)$ and $\Delta_V = 0$. Any possible value of Δ_V can be found by at some $v \in V^*$. Therefore, to find a Nash equilibrium we only need to consider the leaf players' responses to V playing any v_i^* or a pure strategy.

To treat the root player's processing of these strategies in order to compute a Nash equilibrium, we will consider two cases: the leaves have distinct indifference points v_i^*, and two or more leaves have the same indifference point. For example, the former case can be found in almost all randomly generated payoff matrices, while the latter can occur in games where there are *symmetric* players having the same best response function. As with games on paths, we will consider graphical games and polymatrix games on stars.

F.1 GRAPHICAL GAMES ON STARS

In the case of a graphical game on a star, every leaf player will have a 2×2 payoff matrix, and the root player will have a payoff matrix of size 2^n, making the game's input or representation size exponential in the number of players.

Case 1. *The leaves have distinct indifference points.*

In this case, we compute an MSNE in $O(n)$ time, which is logarithmic in input size, by carrying out the following three steps. Each step requires $O(n)$ computation.

Step 1: Compute each leaf's indifference point

Similar to the first node U of the path case described in Section 3, we can derive each leaf node U_i's indifference point with respect to the root note V as follows.

$$\begin{aligned}\Delta_{U_i}(v) &\equiv M_{U_i}(1,v) - M_{U_i}(0,v)\\ &= vM_{U_i}(1,1) + (1-v)M_{U_i}(1,0) - vM_{U_i}(0,1) - (1-v)M_{U_i}(0,0)\\ &= v(M_{U_i}(1,1) - M_{U_i}(1,0) - M_{U_i}(0,1) + M_{U_i}(0,0)) + M_{U_i}(1,0) - M_{U_i}(0,0)\ .\end{aligned}$$

Setting $\Delta_{U_i}(v_i^*) = 0$, we get the following.

$$v_i^* = \frac{-M_{U_i}(1,0) + M_{U_i}(0,0)}{M_{U_i}(1,1) - M_{U_i}(1,0) - M_{U_i}(0,1) + M_{U_i}(0,0)}$$

For each leaf node i, we can compute the above in constant time, leading to $O(n)$ time for all leaves. Furthermore, v_i^* is a rational number and will be amenable to radix sorting in the next step.

Step 2: Compute the Δ_V vs. v map

First, a straightforward computation of the Δ_V vs. v plot shown in Figure 6 will cost $O(n^2)$ because at each v_i^* indifference point we will have to calculate the pure-strategy best responses of all leaves except U_i to V playing v_i^*. We can avoid repeated computation and achieve an $O(n)$ time as follows.

We begin by sorting the v_i^* indifference points for all U_i in increasing order (recall that each v_i^* is distinct). Since the v_i^* values are rational, we can apply the standard radix sorting by denominators followed by numerators to achieve $O(n)$-time sorting. Without loss of generality, suppose the sorted indifference points are $v_1^*, v_2^*, \ldots, v_{n-1}^*$.

We now consider v_1^*, where U_1 becomes indifferent. We compute the pure-strategy best responses of $U_2, \ldots, U_{n-1}$ to V playing v_1^* in a total of $O(n)$ time. We then compute the lowest and highest values of Δ_V by considering $u_1 \in \{0,1\}$, together with the pure-strategy best responses of $U_2, \ldots, U_{n-1}$. This gives us the range of Δ_V values at v_1^*.

We next compute the range of Δ_V values at v_2^*. For this, we only compute U_1's pure-strategy best response to v_2^* in $O(1)$ and reuse the previously computed pure-strategy best responses of $U_3, \ldots, U_{n-1}$ because v_2^* (just like v_1^*) is below their indifference points. We therefore obtain the range of Δ_V values at v_2^* in $O(1)$ time by considering $u_2 \in \{0,1\}$ in conjunction with the pure-strategy best responses of other leaves.

In this fashion, we compute the range of Δ_V at $v_3^*, \ldots, v_{n-1}^*$, each in $O(1)$ time. We also compute Δ_V at $v \in \{0,1\}$ in $O(n)$ time. The special cases of $v_1^* = 0$ and $v_{n-1}^* = 1$ can be handled appropriately. In between any successive indifference points, Δ_V is fixed due to each leaf node playing a pure strategy.

The total computation time for Step 2 is, therefore, $O(n)$.

Step 3: Compute an MSNE

We first examine the cases of $v \in \{0,1\}$. When V plays 0, we can look up the Δ_V vs. v plot at $v = 0$ to check if $\Delta_V \leq 0$, in which case we have found a Nash equilibrium. Similar is the case of $v = 1$.

Failing the above, we next examine each v_i^* in any order. If the range of Δ_V values cross 0, we have found an MSNE. This is because V will only play $v_i^* \in (0,1)$ if $\Delta_V = 0$ at that point. We also know the pure-strategy best responses of all leaf nodes other than U_i at v_i^*. To complete the computation of an MSNE, we need to compute u_i for which $\Delta_V = 0$. In this scenario, Δ_V becomes a function of u_i alone because the best responses of the other leaf nodes are fixed. Therefore, we obtain an equation $au_i + b = 0$ for some constants a and b, which can be solved for u_i.

Since we do $O(1)$ work at each v_i^*, Step 3 takes $O(n)$.

Case 2. *Two or more leaves have the same indifference point.*

If m leaf players are indifferent in response to the same strategy v_i^* of V, then the sub-matrix containing the payoffs needed to generate the Δ_V function will be of size 2^{m+1}. Accessing these matrix entries to generate Δ_V will therefore take $O(2^m)$, exponential in the number of indifferent leaf players, which is still linear in the game size.

The worst case Δ_V equation will have terms dependent on every possible selection of strategies. For example, a v-indifference point shared by three leaf players would have an equation of the form:

$$\Delta_V = c_7 u_j u_k u_l + c_6 u_j u_k + c_5 u_j u_l + c_4 u_k u_l + c_3 u_j + c_2 u_k + c_1 u_l + c_0 \ .$$

Finding an MSNE can be done by exploiting the monotonicity and continuity of the equation. We will divide this process into two cases, based on the location of v_i^*.

First, suppose v_i^* is located at a pure strategy of V. Because Δ_V is monotonic, its most extreme outputs will occur at pure-strategy combinations of the leaf players. The Δ_V function can be checked for each pure-strategy combination of the indifferent players until one outputs a value which enables V's pure strategy. If there is no combination with a satisfactory output, then there is no Nash equilibrium where V plays the given pure strategy. In the worst case, all 2^m pure-strategy combinations are checked.

Second, suppose $v_i \in (0, 1)$. In this case, we need a more precise combination of strategies to satisfy the equality $\Delta_V = 0$. Start by checking the pure-strategy combinations. In the event any pure strategy combination outputs 0 it will be a Nash equilibrium. If there is no such pure-strategy combination, a Nash equilibrium can still be found if there are two pure-strategy combinations $\vec{u}_{\min}$ and $\vec{u}_{\max}$ with oppositely signed outputs. Failing that, there is no Nash equilibrium at v_i^* by monotonicity. Assuming these two oppositely-signed pure-strategy combinations are found, we find an MSNE by exploiting the continuity of the Δ_V function. Let $\vec{u}_{\min}$ be the selected vector of leaf player strategies such that $\Delta_V(\vec{u}_{\min}) < 0$ and let $\vec{u}_{\max}$ be the selected vector of leaf player strategies such that $\Delta_V(\vec{u}_{\max}) > 0$. By continuity, there is guaranteed to be a vector of strategies $\vec{u}_0$ such that $\Delta_V(\vec{u}_0) = 0$ "between" $\vec{u}_{\min}$ and $\vec{u}_{\max}$.

We use binary search to locate this MSNE in sub-exponential time once pure-strategy combinations on opposite sides of $\Delta_V = 0$ have been located. The search time becomes negligible in the overall runtime because the process of generating Δ_V from V's payoff matrix in the first place and iterating through all possible pure-strategy combinations takes $O(2^m)$ time. Thus, the worst case would be when all leaves are indifferent in response to a singular strategy of V, in which case finding an MSNE would take $O(2^n)$, which is exponential in the number of players but linear in the size of the game.

F.2 POLYMATRIX GAMES ON STARS

In the case of a polymatrix game on a star, every leaf player will have one 2×2 payoff matrix, and the root player will have a 2×2 payoff matrix on each of its $n - 1$ edges, making the game size linear in the number of players. We consider two cases.

Case 1. *The leaves have distinct indifference points.*

As we show below, solving the case of distinct indifference points is similar to solving that for a graphical star. The computation is also linear in the number of players, which makes it linear in the size of the game.

The additive decomposition of payoffs in a graphical polymatrix game means that $\Delta_V(\vec{u})$ for a vector $\vec{u}$ of the mixed strategies of the leaves can be expressed in terms of $\Delta_V^{U_i}(u_i)$, the difference in V's local payoffs between $v = 1$ and $v = 0$ on each edge (V, U_i).

$$\begin{aligned}
\Delta_V(\vec{u}) &= M_{V|\vec{u}}(1) - M_{V|\vec{u}}(0) \\
&= M_{V,U_1}(1, u_1) + M_{V,U_2}(1, u_2) + \ldots + M_{V,U_{n-1}}(1, u_{n-1}) \\
&\quad - M_{V,U_1}(0, u_1) - M_{V,U_2}(0, u_2) - \ldots - M_{V,U_{n-1}}(0, u_{n-1}) \\
&= \sum_{i=1}^{n-1} \left(M_{V,U_i}(1, u_i) - M_{V,U_i}(0, u_i) \right) \\
&= \sum_{i=1}^{n-1} \Delta_V^{U_i}(u_i) \ .
\end{aligned}$$

Consider U_j's indifference point v_j^*. Every player $U_{i \neq j}$ has a single pure-strategy best response a_i to v_j^*. Therefore, $\Delta_V^{U_i}(a_i)$

becomes a constant for any $i \neq j$. We get the following at $v = v_j^*$ for appropriately defined constants c, c_0, and c_1.

$$\begin{aligned}
\Delta_{V|v_j^*}(\vec{u}) &= \Delta_V^{U_j}(u_j) + \sum_{i \neq j} \Delta_V^{U_i}(u_i) \\
&= \Delta_V^{U_j}(u_j) + c \\
&= M_{V,U_j}(1, u_j) - M_{V,U_j}(0, u_j) + c \\
&= u_j M_{V,U_j}(1,1) + (1-u_j) M_{V,U_j}(1,0) - u_j M_{V,U_j}(0,1) - (1-u_j) M_{V,U_j}(0,0) + c \\
&= c_1 u_j + c_0 \ .
\end{aligned}$$

Calculating Δ_V only requires accessing the two payoff-matrix entries where each $U_{i \neq j}$ plays its pure-strategy best response to v_j^*. All four entries of the payoff matrix associated with U_j are used because of U_j's strategic indifference. This means that $2(n-2)+4 = 2n$ entries are accessed to calculate Δ_V. Similar to the graphical star case, if $v_j^* \in (0,1)$, then V will play v_j^* only if $\Delta_V = 0$. Solving it for u_j, we get the following.

$$u_j = -\frac{c_0}{c_1}$$

If $-\frac{c_0}{c_1} \in [0,1]$, then an MSNE is found. Otherwise, no MSNE exists at $v = v_j^*$.

If $v_j^* \in \{0,1\}$, both pure strategies of u_j can be tested in the equation to see if either produces $\Delta_V(u_j) \geq 0$ for $v_j^* = 1$ or $\Delta_V(u_j) \leq 0$ for $v_j^* = 0$.

Similar to the graphical star case, it takes time linear in the number of players to perform each of the following three steps: computing the indifference points of the leaves, computing the Δ_V vs. v map, and finally, computing an MSNE. Therefore, the whole algorithm runs in $O(n)$.

Case 2. *Two or more leaves have the same indifference point.*

We now consider the more challenging case where two or more leaves can have the same indifference point. Suppose $U_1, \ldots, U_m$ (wlog) have the same indifference point. In this case, Δ_V is still computed by summing over the contributions from each leaf player. For each of the m leaf players having the same indifference point, all 2×2 entries in the associated payoff matrix must be accessed to generate Δ_V. Below is the form of Δ_V at the given indifference point as function of $u_1, \ldots, u_m$, which are the mixed strategies of the indifferent players. Here, other players with fixed pure-strategy best responses will have two payoff entries which reduce into a constant term c'.

$$\begin{aligned}
\Delta_V =& u_1 M_{V,U_1}(1,1) + (1-u_1) M_{V,U_1}(1,0) + \ldots + u_m M_{V,U_m}(1,1) + (1-u_m) M_{V,U_m}(1,0) \\
&- u_1 M_{V,U_1}(0,1) - (1-u_1) M_{V,U_1}(0,0) - \ldots - u_m M_{V,U_m}(0,1) - (1-u_m) M_{V,U_m}(0,0) \\
&+ c' \\
=& c_m u_m + \ldots + c_1 u_1 + c_0 \ .
\end{aligned}$$

The strategy of each u_i which maximizes or minimizes Δ_V can be determined by the sign of c_i. Let $\Delta_V^{\min}$ and $\Delta_V^{\max}$ denote the minimum and maximum possible Δ_V, respectively. In the event that the indifference point $v_j^* = 0$ or $v_j^* = 1$, it can be determined if a Nash equilibrium exists by checking if $\Delta_V^{\min} \leq 0$ or $\Delta_V^{\max} \geq 0$, respectively.

If $v_j^* \in (0,1)$, an MSNE can be computed by exploiting continuity as was done for graphical games. If $\Delta_V^{\min} > 0$ or $\Delta_V^{\max} < 0$, there is no MSNE at v_j^* because V will only play $v_j^* \in (0,1)$ if $\Delta_V = 0$. Otherwise, start with the pure-strategy combination for $\Delta_V^{\min}$. Let Δ_V^j be the output of Δ_V when first j players in an arbitrary ordering have been switched from their minimizing to maximizing pure strategies. Iterate through the leaf players and keep switching their strategies until $\Delta_V^j \geq 0$. If $\Delta_V^j = 0$, then $v = v_j^*$ and the combination of the leaf players' pure strategies is an MSNE. If $\Delta_V^j > 0$ instead, turn u_j into a variable while leaving all other leaf players playing their pure strategies from Δ_V^j, set $\Delta_V = 0$, and solve for u_j. There is guaranteed to be a $u_j \in (0,1)$ that solves the equation. This gives us an MSNE.

This process of checking for an MSNE with m simultaneously indifferent leaf players is $O(m)$. As such, the running time remains linear in the number of players and therefore, linear in the input size even for shared indifference points.

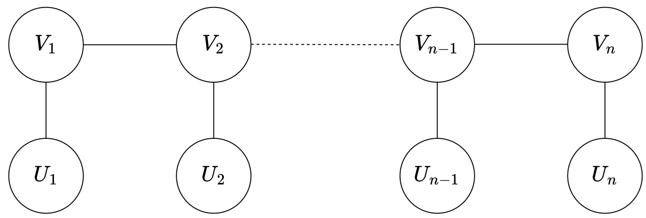

Figure 31: A diagram of a length n 1-spoke caterpillar.

G CATERPILLAR GAMES

Caterpillar graphs, as defined in the main body of the paper, combine the features of paths and stars. An example of a simple caterpillar graph is shown in in Figure 31. The following sections demonstrate the polynomial running time of both polymatrix and graphical caterpillar games.

G.1 POLYMATRIX CATERPILLAR GAMES

We start by considering a simple caterpillar in the form of Figure 31, which we root at V_n. The worst-case message sent from a U_i player will contain 3 segments in the S-shaped form shown in the left message in Figure 5. The growing complexity of messages will be reflected in the messages B_{V_{i+1},V_i} along the spine of the caterpillar.

For the first message along the spine, B_{V_2,V_1}, player V_1 is only receiving an S-shaped messages from U_1. As such, the worst case message will look the same as the worst case message along a polymatrix path, of the same form as the right message in Figure 5. The Δ_{V_j} function of each subsequent player V_j will have a separate linear term for the strategy of each of its three neighbors. Similar to polymatrix paths, one can then set $\Delta_{V_j} = 0$ and rearrange to find an indifference for v_{j+1} as a function of v_{j-1} and u_j for appropriately defined constants:

$$v_{j+1} = -\frac{av_{j-1} + bu_j + d}{c} = \alpha v_{j-1} + \beta u_j + \gamma \ .$$

Because of the limited set of strategies of U_j, the indifference curve will always take one of three more restricted forms depending on the strategy of V_j.

The cases where U_j has a pure-strategy best response is as follows.

Case 1. *U_j's best response to the given v_j is $u_j = 0$.*

U_j's strategy will always be 0, meaning that the u_j term can be removed from the the indifference curve formula to yield:

$$v_{j+1} = \alpha v_{j-1} + \gamma$$

This is a linear equation of the same form as an indifferent curve in a polymatrix game on path.

Case 2. *U_j's best response to the given v_j is $u_j = 1$*

U_j's strategy will always be 1, meaning that the u_j term can be combined with the constant term to yield:

$$v_{j+1} = \alpha v_{j-1} + \beta + \gamma$$

This is the same as the indifference curve from the previous case with a vertical translation of β.

This leaves one case of interest.

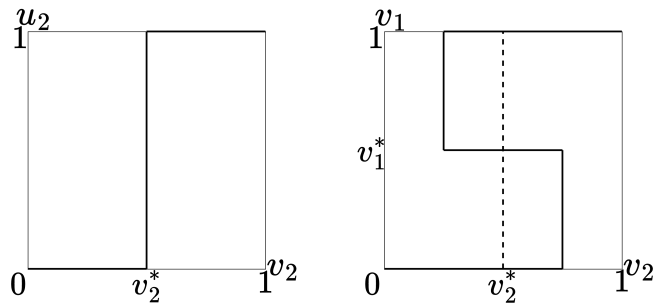

Figure 32: Example messages which the second spine node V_2 could receive during the downstream pass: An S-shaped message from its spoke node U_2 (left) and message from the previous spine node V_1, with U_2's indifference strategy superimposed (right).

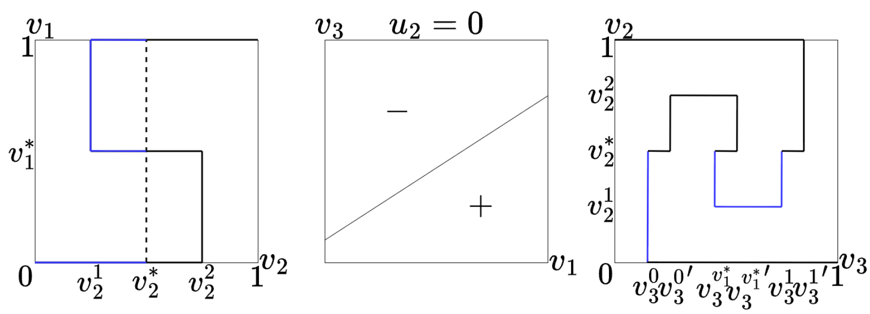

Figure 33: B_{V_2,V_1} with V_1's best responses when U_2 plays 0 highlighted in blue (left), a worst-case polymatrix indifference curve with a fixed strategy of $u_2 = 0$ (center), and the resulting message B_{V_3,V_2} with the mappings of the blue segments also highlighted (right).

Case 3. *U_j is indifferent in response to the given v_j*

From $B_{V_j,V_{j-1}}$, we can identify the best responses of V_{j-1} at v_j by computing the intersections of the vertical line at v_j with the horizontal segments of $B_{V_j,V_{j-1}}$. For each of these best-response strategies, v_{j-1} is fixed to some v^i_{j-1} and the indifference curve becomes:

$$v_{j+1} = \alpha v^i_{j-1} + \beta u_j + \gamma \,.$$

To demonstrate the process of mapping the messages player V_j receives to the message it sends to V_{j+1}, the following section will go through the process of generating the message from V_2 to V_3 using the example worst case messages shown in Figure 32.

Observe that the dashed line on B_{V_2,V_1} indicates the strategy of V_2 for which player U_2 is indifferent. All segments and partial segments of B_{V_2,V_1} to the left of the dashed line are mapped into B_{V_3,V_2} using the indifference curve with $u_2 = 0$ as shown in Figure 33. The vertical indifference points of B_{V_3,V_2} mapped using this indifference curve are those which are not primed.

All segments and partial segments of B_{V_2,V_1} to the right of the dashed line are mapped into B_{V_3,V_2} using the indifference curve with $u_2 = 1$ as shown in Figure 34. The vertical indifference points of B_{V_3,V_2} mapped using this indifference curve are indicated with a prime.

The intersections of segments of B_{V_2,V_1} with the dashed line at V_2^* indicate where U_2 can play any $u_2 \in [0,1]$. Each V_1

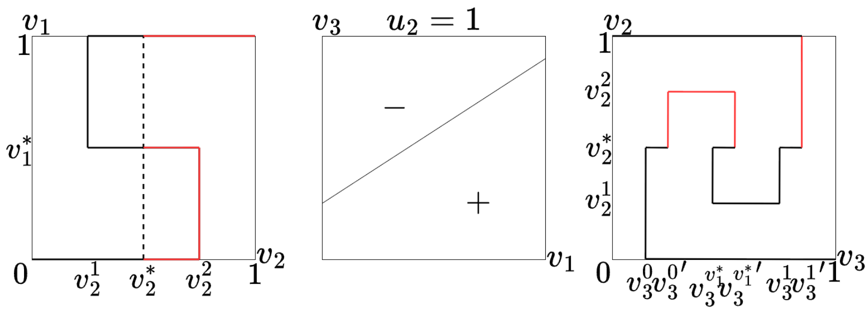

Figure 34: B_{V_2,V_1} with V_1's best responses when U_2 plays 1 highlighted in red (left), a worst case polymatrix indifference curve with a fixed strategy of $u_2 = 1$ (center), and the resulting message B_{V_3,V_2} with the mappings of the red segments also highlighted (right).

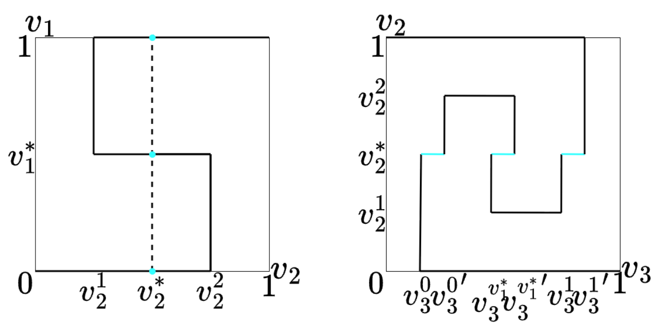

Figure 35: B_{V_2,V_1} with V_1's best responses when U_2 is indifferent highlighted in cyan (left) and the resulting B_{V_3,V_2} with resulting horizontal segments also highlighted (right). Each V_1 strategy (cyan point) where an intersection occurs is mapped to B_{V_3,V_2} as a horizontal segment at v_2^* over the v_3 interval $[v_3^i, {v_3^i}']$. Here, the endpoints v_3^i and ${v_3^i}'$ are the mappings for $u_2 = 0$ and $u_2 = 1$, respectively.

strategy v_1^i where an intersection occurs will be mapped to B_{V_3,V_2} as a horizontal segment at v_2^* over the v_3 interval $[v_3^i, {v_3^i}']$, as these endpoints are the mappings for $u_2 = 0$ and $u_2 = 1$ and the indifference curves are all continuous. A visualization can be found in Figure 35. Note that each of these segments will have a length of β for reasons previously explained.

Lastly, the pure strategy segments of B_{V_3,V_2} will be determined by checking the sign above and below the indifference curve like in a path game, as shown in Figure 36. Observe that the relative location of the positive and negative regions will be the same in both the $u_2 = 0$ and $u_2 = 1$ indifference curves.

We will now analyze the mapping process to place a bound on the rate of segment growth along the spine. Every segment in B_{V_{j+1},V_j} which does not cross the vertical indifference point of U_{j+1} is mapped with the same principles as a polymatrix path game. The growth rate from pure strategies is also the same. The only difference in segment growth comes from the horizontal segments of B_{V_{j+1},V_j} which cross the indifference point of U_{j+1}. As was previously explained and can be seen in Figures 33-35, each of these segments can map to at most 3 segments in $B_{V_{j+2},V_{j+1}}$. The partial segments on each side of the intersection each map to a vertical segment which extends to v_{j+1}^*, and the intersection itself maps to a horizontal segment at v_{j+1}^* connecting them.

Let m_j be the total number of segments in the j-th message along the spine, B_{V_{j+1},V_j}. Let k_j be the maximum number of possible intersections between the segments of B_{V_{j+1},V_j} and a vertical line at some v_{j+1}^*. each vertical segment leads to a horizontal segment, and each horizontal segment leads to either one or three vertical segments. We get the latter when the horizontal segment in B_{V_{j+1},V_j} is split by the vertical line at v_{j+1}^*, resulting in two vertical segments in $B_{V_{j+2},V_{j+1}}$ that are shifted, plus one horizontal segment connecting the gap due to shifting. Therefore, in the worst case, the maximum number of segments in the $j+1$-st message $B_{V_{j+2},V_{j+1}}$ is the following:

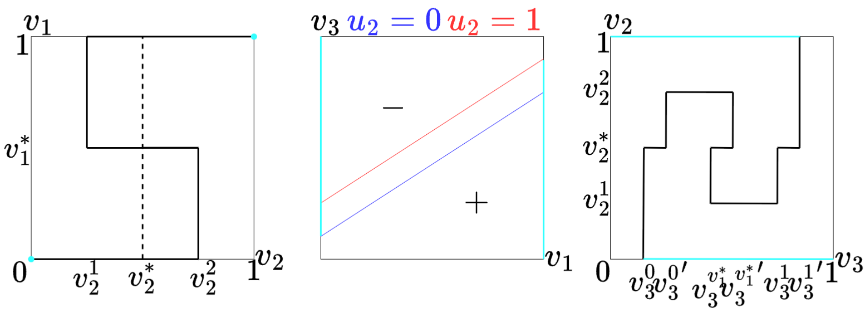

Figure 36: B_{V_2,V_1} with V_1's pure strategy responses highlighted in cyan (left), the indifference curves for each pure strategy of u_2 with the intervals enabling each pure strategy superimposed using their corresponding indifference curves (center), and B_{V_3,V_2} with the resulting pure strategy segments highlighted in cyan (right).

$$m_{j+1} = (m_j - k_j) + 3k_j + 2 = m_j + 2k_j + 2 \ .$$

Thus, all that remains to find a tight upper bound is to bound the maximum number of intersections k_j.

Lemma 11. *Let k_j be the maximum number of horizontal segments whose interiors can be intersected by a vertical line in the j-th message in a simple caterpillar polymatrix game. In the worst case, $k_j = 2j + 1$.*

Proof. For the base case of $j = 1$, the statement clearly holds.

For any $j > 1$, we first relate k_j to l_j, defined as the maximum number of intersections with a horizontal line. The vertical intervals of segments in the $j + 1$-st message are the same as the horizontal intervals of the j-th message, yielding:

$$l_{j+1} = k_j \ .$$

In the worst case, all vertical segments in the j-th message will map to horizontal segments in the $j + 1$-st message. In the $j + 1$-st message, there are three sources of intersections, which can all be observed in the right plot of Figure 37. There are the two horizontal pure strategy segments, which are capable of both overlapping the same horizontal interval as the maximum number of intersections. These can be seen as the dark blue segments in the right plot of Figure 37. Additionally, it is possible for one of the horizontal segments at the indifference point v_j^* of U_j to also lie on this overlapping interval. An example of this can be seen in the middle cyan segment of Figure 37.

Lastly, it is possible for an additional intersection to occur as a result of the relative displacement of the messages on either side of the aforementioned indifference point v_j^*. Consider that the previously explained method of mapping one message to the next along the spine can be thought of in geometric terms as mapping the previous message B_{V_{j+1},V_j} as if it was along a path, then shifting the part of the message $B_{V_{j+2},V_{j+1}}$ located on one side of the horizontal line at v_{j+1}^* (the indifference point of U_{j+1}) and then adding horizontal segments at the location of the shift to maintain continuity. Through this process, it is possible for vertical segments that did not overlap (or only overlapped at a point) in the previous message to be mapped to horizontal segments that overlap over an interval. An example of this process can be seen in the mapping of the two red segments in Figure 37.

The reason that this geometric shifting can create at most one additional intersection in the worst case is as follows. We assume that in the worst case, we have generated a construction where the biggest increase in the number of segments occurs between messages. As has been previously explained, this results from the vertical line indicating U_j's indifference point intersecting the largest number of horizontal segments. In the first message, this means placing the vertical line between the two horizontal indifference points. As shown in the second plot of Figure 37 in cyan color, the two sides of the first message are shifted in such a way that two vertical segments, which previously overlapped at a point, now overlap over an interval. Picking the next indifference point to be in this interval maximizes the number of overlapping horizontal segments in the

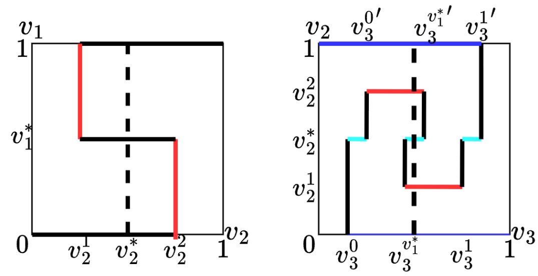

Figure 37: B_{V_2,V_1} with the maximally intersecting indifference strategy of U_2 superimposed (left) and B_{V_3,V_2} with the maximally intersecting indifference strategy of U_3 superimposed (right).

next message, which is 5 rather than 3. The result of this process is that the interval in the next message with the maximum number of intersections is produced by the middle horizontal segment at the strategy v_j^* of the previous message where U_j is indifferent. Notice that there is an odd number of these horizontal segments. Because these segments result from a segment whose two regions were shifted relative to each other, the vertical segments connected on the same side all extend in one direction, while those on the other side extend in the opposite direction.

Now, for the induction step, assume that we are starting with a message that has the previously described configuration. After this message has been mapped as if along a path, each of the horizontal segments at v_j^* are mapped to a vertical segment at some v_{j+2}^{**}, the mapping of v_j^* through one of the indifference curves with a fixed pure-strategy of U_{j+1} (because the message is being mapped as if along a path). The connecting vertical segments above them in the j-th message are mapped to segments either all to the left or all to the right of them in the $j+1$-st message. The vertical segments below them in the j-the message are all mapped on the opposite side. The shifting of the two sides of v_{j+1}^* in the $j+1$-st message can take place in one of two possible directions. In one direction, which is illustrated in Figure 37, the shift will result in an interval over which the horizontal segments above v_{j+1}^* and extending to v_{j+2}^{**} from the left will overlap with the horizontal segments below v_{j+1}^* and extending to v_{j+2}^{**} from the right. In the other direction, the shift will result in an interval over which the horizontal segments above v_{j+1}^* and extending to v_{j+2}^{**} from the right will overlap with the horizontal segments below v_{j+1}^* and extending to v_{j+2}^{**} from the left. If we were to perform the latter direction in Figure 37, no vertical line would have intersected both red segments because they would have shifted apart from each other.

Because of the odd number of horizontal segments at v_{j+2}^{**} and the already explained configuration of horizontal segments, there will either be one more horizontal segment extending to v_{j+2}^{**} from the right above V_{j+1} than there is below V_{j+1} and one more horizontal segment extending to v_{j+2}^{**} from the left below V_{j+1} than above V_{j+1}, or vice versa. This means that there is one direction where the shift can occur where there is a net gain of one intersection compared to no shifting. This is the direction that the shift will occur in the worst case.

By construction of the shifting process, this interval over which one new interval is created will also contain one of the new horizontal segments at v_{j+1}^* (cyan segment in Figure 37). This means that the previous odd number of maximum intersections must increase by two on account of shifting in the worst case and remain an odd number. Additionally, as the previous message had an odd number of intersections at v_{j+1}^*, this message will have an odd number of horizontal segments at v_{j+1}^* in the worst case.

These factors combined lead to an addition of at most 4 intersections of horizontal segments relative to those of vertical segments in the previous message. We can therefore make the following statement about the maximum number of horizontal segments whose interiors can be intersected by a vertical line:

$$k_{j+1} = l_j + 4 \ .$$

Using substitution, we can find independent equations for k_j and l_j:

$$l_{j+1} = l_{j-1} + 4 \ .$$
$$k_{j+1} = k_{j-1} + 4 \ .$$

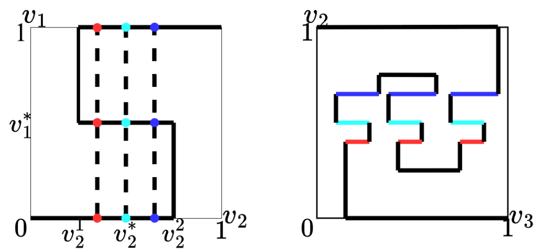

Figure 38: B_{V_2,V_1} of a 3-spoke caterpillar with the the indifference strategies of the spokes superimposed and the best responses of V_1 when each spoke player is indifference highlighted (left) and B_{V_3,V_2} with the resulting horizontal segments from these strategies highlighted in corresponding colors (right).

Additionally, we see from the base cases presented above that $k_1 = 3$ and $k_2 = 5$ in the worst case. Therefore, we can then express the above relation with the following closed form equation:

$$k_j = 2j + 1 \; .$$

□

Making this substitution back into the first formula for the total number of segments gives us the following:

$$m_{j+1} = m_j + 2(2j+1) + 2 = m_j + 4j + 4 \; .$$

Therefore, the growth of segments is linear, meaning that the number of segments in the j-th message is quadratic in j, and the total number of segments across the downstream pass is cubic in the number of players. This gives us the following result.

The TreeNash algorithm runs in $O(n^3)$ time on simple caterpillar polymatrix games.

We now generalize this result to caterpillar games with a maximum number s of spoke players. An example mapping of the first message along the spine of a 3-spoke caterpillar can be found in Figure 38.

In this case, each spoke sends a message which adds a strategy of V_j for which that spoke is indifferent and a shift occurs. For our purposes, we assume the generic case in which no two spokes become indifferent simultaneously.

Each spoke behaves in the worst case in the same way as the single spoke in the simple caterpillar case described above. Furthermore, as shown in Figure 38, the compounding effect is linear in the number of spokes in the worst case. In particular, for each spoke, we get two additional horizontal segments whose interiors are crossed by a vertical line in the worst case (compared to the path case). So, whereas we had $k_{j+1} = l_j + 2$ for the path case, we now have $k_{j+1} = l_j + 2s + 2$, where s is the maximum number of spokes connected to a spine node. As before, the $+2$ term comes from pure-strategy segments. We get the following relationships.

$$\begin{aligned} m_{j+1} &= m_j + 2sk_j + 2 \; . \\ l_{j+1} &= k_j \; . \\ k_{j+1} &= l_j + 2s + 2 \; . \\ k_{j+1} &= k_{j-1} + 2s + 2 \implies k_j = 2sj + O(1) \; . \end{aligned}$$

We see that for a maximum number of spoke players s on each spine player, k_j still grows linearly in j, meaning that the growth in m_j is still linear, and so our bounds are the same as those for the simple caterpillar. This leads to Theorem 7 in the main body of the paper.

G.2 GRAPHICAL CATERPILLAR GAMES

Having already analyzed the polymatrix caterpillar, determining the worst case for a graphical caterpillar is considerably simpler. The spoke players will each still have a 2×2 payoff matrix and send an S-shaped message to the associated spine player with at most a single strategy for which the spoke player is indifferent.

We again start by considering the simple caterpillar. For messages sent along the spine of a simple caterpillar, the delta function of a spine player receiving a message from the preceding spine and from its spoke player to send a message to the next player along the spine is as follows:

$$\Delta_V = av_{i-1}u_iv_{i+1} + bv_{i-1}u_i + cv_{i-1}v_{i+1} + du_iv_{i+1} + ev_{i-1} + fu_i + gv_{i+1} + h$$

which when isolated for v_{i+1} becomes:

$$v_{i+1} = -\frac{(bv_{i-1} + f)u_i + (ev_{i-1} + h)}{(av_{i-1} + d)u_i + (cv_{i-1} + g)}$$

where constant $a, .., h$ are derived from the payoffs of combinations of pure strategies.

Let v_i^* be the strategy of player V_i for which spoke player U_i is indifferent. For all strategies on one side of v_i^*, U_i will only have a best response of 0, yielding the indifference curve:

$$v_{i+1} = -\frac{ev_{i-1} + h}{cv_{i-1} + g}$$

Whereas on the other side of v_i^* U_i will have a fixed response of 1, yielding the indifference curve:

$$v_{i+1} = -\frac{(b + e)v_{i-1} + (f + h)}{(a + c)v_{i-1} + (d + g)}$$

When V_i plays the strategy v_i^*, u_i is free to mix from 0 to 1 while V_{i-1} has a fixed strategy in the generic case. While the notion of shifting one section of the message relative to another does not carry over from the polymatrix case, mapping one message to the next can be thought of as taking the section of the jth message on either side of v_j^*, mapping each half with its corresponding indifference curve, and then using the u_j-varying indifference curve to connect them.

The behavior which differentiates the messages propagated by graphical games from those of polymatrix games is splitting, which increases the total number of segments, but not the number of segments which can be intersected in the worst case. Therefore the worst case recursive relationship between the number of vertical and horizontal intersections is the same as in the polymatrix case:

$$\begin{aligned} l_{j+1} &= k_j \\ k_{j+1} &= l_j + 4 \\ l_{j+1} &= l_{j-1} + 4 \\ k_{j+1} &= k_{j-1} + 4 \end{aligned}$$

When it comes to segment growth itself, there are three differences from the polymatrix case.

First, the indifference curve when u_i ranges over $(0, 1)$ takes the form of an indifference curve for a graphical game on a path. This means that the mappings of a horizontal segment which crosses v_i^* could either be connected by a horizontal line segment between them, or that segment could be split, and each vertical segment could be connected to a separate horizontal segment extending to the nearest pure strategy, effectively splitting.

Second, it is possible for vertical segments on either side of v_i^* to be split if they happen to cross the vertical asymptote of the corresponding indifference curve. Since each side of v_i^* has its own indifference curve with a different vertical asymptote in the generic case, the worst possible number of splits would be twice the maximum number of vertical segments which can be intersected with a horizontal line.

Third, recall from the discussion of graphical games on paths that in the worst case an indifference curve can yield two horizontal segments at one pure strategy and one at the other. Since our message combines two different indifference curves, it is possible in the worst case that each allows 2 segments at the pure strategy it maps to for a total of 4 pure strategy segments.

We then formulate these observations into the following equation:

$$m_{j+1} = m_j + 3k_j + 2l_j + 4$$

This relation may be slightly loose since we have not produced a tight example, but it still shows that the graphical case grows by the same order as the polymatrix case.

To generalize to a maximum-degree s graphical caterpillar, we assume like in the polymatrix case that the indifference value of every spoke player can contribute the same segment and intersection growth. The result is the following:

$$\begin{aligned} k_{j+1} &= l_j + 2s + 2 \\ l_{j+1} &= k_j \\ m_{j+1} &= m_j + 3sk_j + (s+1)l_j + 4 \end{aligned}$$

Again, this result has the potential for being loose since we do not have a tight example.

The interesting result here is that maximum-degree s caterpillar games, despite having a higher degree than path games and being a combination of the path and star games structures, are still on the same order of time and space efficiency as path graphical games for finding a complete representation of all possible equilibria. This makes sense upon further reflection. The splitting process of graphical games on a path depends on the number of vertical segments which can be intersected. The segment growth caused by spoke players is based on the number of horizontal segments which can be intersected. It is therefore logical that the combination of these processes would be additive rather than compounding.

This analysis leads to Theorem 8 in the main body of the paper.