Patience of Matrix Games

Kristoffer Arnsfelt Hansen[1], Rasmus Ibsen-Jensen[1],
Vladimir V. Podolskii[2], and Elias Tsigaridas[3]

[1]Aarhus University*
[2]Steklov Mathematical Institute†
[3]INRIA Paris-Rocquencourt and Universite Pierre et Marie Curie (Paris 6)‡

June 8, 2012

Abstract

For matrix games we study how small nonzero probability must be used in optimal strategies. We show that for $n \times n$ win-lose-draw games (i.e. $(-1, 0, 1)$ matrix games) nonzero probabilities smaller than $n^{-O(n)}$ are never needed. We also construct an explicit $n \times n$ win-lose game such that the unique optimal strategy uses a nonzero probability as small as $n^{-\Omega(n)}$. This is done by constructing an explicit $(-1, 1)$ nonsingular $n \times n$ matrix, for which the inverse has only nonnegative entries and where some of the entries are of value $n^{\Omega(n)}$.

Keywords: Matrix Games, Ill-conditioned Matrices, Nonnegative Inverse.

1 Introduction

Given a matrix game A we are interested in the following question: What is the smallest nonzero probability that must be used in optimal strategies. This quantity, the smallest nonzero probability of a strategy, was first considered in the context of recursive games (stochastic games where payoffs are only accumulated in absorbing states) by Everett [4]. To be more precise, if p is the smallest nonzero probability of a probability vector σ, we say that the *patience* of σ is $1/p$. Note that this is the

*Hansen and Ibsen-Jensen acknowledge support from the Danish National Research Foundation and The National Science Foundation of China (under the grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation, within which this work was performed. They also acknowledge support from the Center for Research in Foundations of Electronic Markets (CFEM), supported by the Danish Strategic Research Council.

†Part of the research was done during a visit to Aarhus University. The research is partially supported by the Russian Foundation for Basic Research and the programme "Leading Scientific Schools".

‡Partially supported from the EXACTA grant of the National Science Foundation of China (NSFC 60911130369) and the French National Research Agency (ANR-09-BLAN-0371-01).

precisely the expected number of times that σ must be sampled in order to observe the least likely outcome. Also $\lceil \log_2(1/p) \rceil$ is a lower bound on the number of random bits required in order to sample from σ using a source of uniform random bits.

In this paper we study the patience required for playing optimal strategies in matrix games. Our focus is how this quantity depends on the dimensions of the matrix game, rather than on the individual payoffs. In particular we consider win-lose and win-lose-draw matrix games. We model win-lose games as $(0, 1)$ matrices and win-lose-draw matrices as $(-1, 0, 1)$ matrices. Note that for win-lose games this choice of matrices have no consequence: the set of optimal strategies is invariant under addition by a number and multiplication by a positive number, applied simultaneously to every entry of the matrix. In particular, we can equivalently model win-lose matrix games as $(-1, 1)$ matrices.

We prove both upper and lower bounds on the patience required for playing optimal strategies for these two classes of matrix games. Our lower bounds build on previous constructions of *ill-conditioned* matrices [8, 1]. In particular we show that from any ill-conditioned matrix A, a matrix game can be derived with patience at least the size of the largest entry of the inverse of A. As such our question can be seen as yet another application of ill-conditioned matrices. A downside of this connection is that it is not *explicit* - namely, we do not know of a polynomial time algorithm for computing this derived matrix game, given the ill-conditioned matrix A as input. We address this unsatisfactory situation by constructing a variant of the ill-conditioned matrix constructed by Alon and Vũ [1], and study in detail the structure of the inverse matrix. We use this to construct an ill-conditioned $(-1, 1)$ matrix with a *non-negative inverse*, and from this we directly obtain an explicit construction of a win-lose matrix of high patience. This construction is in fact what we will call *fully-explicit*, meaning that each entry of the matrix can be computed in time polynomial in the bitlength of the dimension of the matrix.

Patience behaves very differently in matrix games compared to its original setting of recursive games [11, 9, 10]. First of all in recursive games optimal strategies are not guaranteed to exist; On the other hand, for every $\epsilon > 0$ both players have stationary strategies guaranteeing an expected payoff within ϵ of the value of the game from any starting position [4]. However, there exist recursive games with N positions, each with $m \geq 2$ actions, and every payoff is 0 or 1, where any $(1 - 4m^{-N/2})$-optimal strategy must have patience at least $2^{m^{N/3}}$. On the other hand, patience $(1/\epsilon)^{m^{O(N)}}$ is always sufficient for having an ϵ-optimal strategy in recursive games where every payoff is 0 or 1.

Consider now again the setting of matrix games. Lipton and Young proved that in a zero-sum $n \times n$ matrix game where all payoffs belongs to the interval $[0, 1]$, each player has a *simple* strategy guaranteeing an expected payoff within ϵ of the value of the game, where a simple strategy is a strategy that mixes uniformly on a multiset of $\lceil \ln n/\epsilon^2 \rceil$ actions [15]. Thus the patience of such strategies is also at most $\lceil \ln n/\epsilon^2 \rceil$.

In other words, comparing with our results below, if one is willing to give up ϵ payoff, one can play with patience that is smaller by an exponential magnitude than required for playing truly optimally.

1.1 Our Results

For stating our results we use standard matrix game terminology. We refer the reader to Section 2 for explanations of this terminology.

We define the *patience* of a strategy to be $1/p$, where p is the smallest nonzero probability the strategy x assigns to one of the actions. That is, the patience of a strategy x is $(\min\{x_i \mid x_i > 0\})^{-1}$.

Given a matrix game A, we define the *patience* $\tau^1(A)$ required for Player 1 to play A optimally to be the minimum patience of an optimal strategy for A. In a similar way we can define $\tau^2(A)$ for Player 2.

We are interested in the largest patience required for optimal strategies of win-lose and win-lose-draw games, as a function of the size of the matrix game. Thus, define $\tau_{\mathrm{wl}}(n)$ as the maximum of $\tau^1(A)$ taken over all $(0,1)$ $n \times n$ matrix games A, and similarly $\tau_{\mathrm{wld}}(n)$ as the maximum of $\tau^1(A)$ taken over all $(-1,0,1)$ $n \times n$ matrix games A.

Clearly the definition of $\tau_{\mathrm{wl}}(n)$ and $\tau_{\mathrm{wld}}(n)$ would be unchanged by considering $\tau^2(A)$ rather than $\tau^1(A)$. However we shall also consider the patience required by both players for optimal strategies. Thus, we define also $\widehat{\tau_{\mathrm{wl}}}(n)$ as the maximum of $\min(\tau^1(A), \tau^2(A))$ taken over all $(0,1)$ $n \times n$ matrix games A, and similarly $\widehat{\tau_{\mathrm{wld}}}(n)$ as the maximum of $\min(\tau^1(A), \tau^2(A))$ taken over all $(-1,0,1)$ $n \times n$ matrix games A. All these measures of patience are in fact closely related (cf. Section 2.3).

We are now able to state our results. First using a Theorem of Shapley and Snow [20] and standard estimates of the magnitude determinants we obtain the following basic upper bound on patience.

Proposition 1.

$$\tau_{wl}(n) \leq (n+2)^{\frac{n+2}{2}}/2^{n+1} \quad , \quad \tau_{wld}(n) \leq (n+1)^{\frac{n+1}{2}} \quad .$$

Next, using previous results on ill-conditioned matrices by Alon and Vũ [1] we obtain the following (non-explicit) lower bound on patience.

Theorem 2.

$$\widehat{\tau_{wld}}(n) \geq n^{\frac{n}{2}}/2^{n(2+o(1))} \quad .$$

By Corollary 15 from Section 2.3 we obtain a result for win-lose matrix games as well.

Corollary 3.

$$\widehat{\tau_{wl}}(n) \geq n^{\frac{n}{4}}/2^{n(5/4+o(1))} \quad .$$

Our main contribution is an explicit construction of a matrix game satisfying a similar patience lower bound.

Theorem 4. *Let $n = 2^m$ be a power of two. Then*

$$\tau_{wl}(n) \geq n^{\frac{n}{2}}/2^{n(1+o(1))} \quad .$$

Furthermore there is an algorithm that for each n and given indices i and j computes the entry (i,j) of the matrix witnessing the lower bound, in time polynomial in m.

1.2 Organization of the paper

In Section 2 we briefly introduce sign patterns of matrices and matrix games, followed by a more extensive coverage of patience of matrix games in Section 2.3. In particular this section provides a proof of the upper bound on patience of matrix games, Proposition 1. In Section 3 we consider the relationship between ill-conditioned matrices and patience of matrix games. In Section 4 we consider three easy examples of explicit ill-conditioned matrices and show how they give matrix games of large patience. Finally, in Section 5 we present our main contribution, an explicit construction of a win-lose matrix game of almost worst case patience.

2 Preliminaries

We shall denote by $\mathbf{1}$ a vector of appropriate dimension where every entry is 1. All vectors we consider are column vectors.

2.1 Sign Patterns of Matrices

A *full sign pattern* is a matrix with entries from $\{-1, 1\}$. A pair of vectors $\sigma^{(1)}, \sigma^{(2)}$ with entries from $\{-1, 1\}$ gives rise to a full sign pattern $\sigma^{(1)}(\sigma^{(2)})^{\mathsf{T}}$ We shall call a full sign pattern of this form a *block checkerboard sign pattern.*

Let $A = (a_{ij})$ be a $n \times n$ matrix with real valued entries. We say that A weakly obeys a block checkerboard sign pattern if there is a block checkerboard sign pattern $\Sigma = (\sigma_{ij})$ such that $\sigma_{ij} = 1$ implies $a_{ij} \geq 0$ and $\sigma_{ij} = -1$ implies $a_{ij} \leq 0$. Note that given A, Σ is not necessarily unique, depending upon the entries of A that are 0.

Lemma 5. *Let $A = (a_{ij})$ be a $n \times n$ nonsingular matrix with real valued entries, such that the inverse A^{-1} weakly obeys a block checkerboard sign pattern $\Sigma = (\sigma_{ij})$. Define the $n \times n$ matrix $B = A \circ \Sigma^{\mathsf{T}}$ to be the Hadamard product of A and the transpose of Σ (That is, $B = (b_{ij})$ is given by $b_{ij} = a_{ij}\sigma_{ji}$). Then the entries of the inverse B^{-1} are non-negative.*

Proof. This follows immediately by considering the identity $AA^{-1} = I$. □

2.2 Matrix Games

A *matrix game* is given by a $m \times n$ real matrix $A = (a_{ij})$. The entries a_{ij} are *payoffs.* The game is played by Player 1 selecting an action $i \in \{1, \ldots, m\}$ and Player 2 simultaneously selecting an action $j \in \{1, \ldots, n\}$, after which Player 1 receives a payoff of a_{ij} from Player 2. A *strategy* of a player is a probability distribution over the actions of the player. We shall view these as stochastic vectors. A strategy is *totally mixed* if it assign non-zero probability to each action.

Given a strategy x for Player 1 and a strategy y for Player 2, the expected payoff to Player 1 when the two players use the pair (x, y) of strategies is $x^{\mathsf{T}}Ay$. The celebrated minimax theorem of von Neumann [23] states that every matrix game has a *value.*

Theorem 6 (von Neumann)**.** *For any matrix game A, there is a number v such that*

$$v = \max_x \min_y x^\mathsf{T} A y = \min_y \max_x x^\mathsf{T} A y \ ,$$

where x and y range over strategies for the two players. The number v is called the value, $\mathrm{val}(A)$*, of A.*

A strategy x is a *maximin* strategy for Player 1, if $\min_y x^\mathsf{T} A y = \mathrm{val}(A)$. Similarly, a strategy y is a *minimax* strategy for Player 2, if $\max_x x^\mathsf{T} A y = \mathrm{val}(A)$. We will call both a maximin strategy for Player 1 and a minimax strategy for Player 2 for *optimal* strategies.

Shapley and Snow [20] characterized the set of optimal strategies as the convex hulls of *basic solutions.*

Theorem 7 (Shapley and Snow)**.** *Let X and Y be the sets of optimal strategies for Player 1 and Player 2 in a matrix game. Then X and Y are the convex hulls of the sets of* basic solutions *X^* and Y^*, where every pair of basic solutions $x \in X^*$ and $y \in Y^*$ correspond exactly to a square submatrix B of A, which satisfies:*

$$\begin{aligned} \mathrm{val}(A) &= \det(B)/\mathbf{1}^\mathsf{T} \operatorname{adj}(B)\mathbf{1} \ , \\ x_B^\mathsf{T} &= \mathbf{1}^\mathsf{T} \operatorname{adj}(B)/\mathbf{1}^\mathsf{T} \operatorname{adj}(B)\mathbf{1} \ , \\ y_B &= \operatorname{adj}(B)\mathbf{1}/\mathbf{1}^\mathsf{T} \operatorname{adj}(B)\mathbf{1} \ , \end{aligned} \tag{1}$$

where x_B and y_B are obtained from of x and y by restricting to the rows and columns of B, respectively.

If the value v of the matrix game A is nonzero this simplifies to

$$\mathrm{val}(A) = 1/\mathbf{1}^\mathsf{T} B^{-1}\mathbf{1} \ , \quad x_B^\mathsf{T} = v\mathbf{1}^\mathsf{T} B^{-1} \ , \quad y_B = vB^{-1}\mathbf{1} \ , \tag{2}$$

Conversely, we have the following result (see e.g. [6, Theorem 3.2]).

Theorem 8. *Let A be a $n \times n$ matrix game, where A is nonsingular and $\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1} \neq 0$. Define*

$$v = 1/\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1} \ , \quad x^\mathsf{T} = v\mathbf{1}^\mathsf{T} A^{-1} \ , \quad y = vA^{-1}\mathbf{1} \ .$$

If both $x \geq 0$ and $y \geq 0$ then $\mathrm{val}(A) = v$ and x and y are optimal strategies of A. If in fact both x and y are totally mixed, i.e. $x > 0$ and $y > 0$, then x and y are the unique optimal strategies.

2.3 Patience of matrix games

Basic Relations

Directly from the definitions we have:

Proposition 9.

$$\widehat{\tau_{wl}}(n) \leq \tau_{wl}(n) \ , \quad \widehat{\tau_{wld}}(n) \leq \tau_{wld}(n) \ .$$

Conversely we can from matrix games where one player must use a strategy of high patience construct a (larger) matrix game where this is the case for both players.

Proposition 10. *Let A be a $n \times n$ $(0,1)$ matrix game such that $0 < \mathrm{val}(A) < 1$. Then there exist a $2n \times 2n$ $(0,1)$ matrix game B such that $\widehat{\tau_{wl}}(B) \geq \max(\tau^1(A), \tau^2(A))$.*

Proof. Consider the $2n \times 2n$ matrix game B given by

$$B = \begin{bmatrix} A & 0 \\ 0 & \mathbf{1}\mathbf{1}^\mathsf{T} - A^\mathsf{T} \end{bmatrix} .$$

Note that $\mathrm{val}(\mathbf{1}\mathbf{1}^\mathsf{T} - A^\mathsf{T}) = 1 - \mathrm{val}(A) > 0$. It follows that the optimal strategies for Player 1 in B are of the form $\big((1 - \mathrm{val}(A))x^\mathsf{T}, \mathrm{val}(A)y^\mathsf{T}\big)$ and similarly the optimal strategies for Player 2 in B are of the form $\big((1 - \mathrm{val}(A))y^\mathsf{T}, \mathrm{val}(A)x^\mathsf{T}\big)$, where x and y are optimal strategies in A for Player 1 and Player 2, and the result follows. □

We have a similar statement for win-lose-draw matrix games.

Proposition 11. *Let A be a $n \times n$ $(-1,0,1)$ matrix game such that $-1 < \mathrm{val}(A) < 1$. Then there exist a $2n \times 2n$ $(-1,0,1)$ matrix game B such that $\widehat{\tau_{wl}}(B) \geq \max(\tau^1(A), \tau^2(A))$.*

Proof. The proof follows similarly to that of Proposition 10, by considering the $2n \times 2n$ matrix game B given by

$$B = \begin{bmatrix} A & -\mathbf{1}\mathbf{1}^\mathsf{T} \\ -\mathbf{1}\mathbf{1}^\mathsf{T} & -A^\mathsf{T} \end{bmatrix} .$$

and noting that this matrix game has the same optimal strategies as the matrix game obtained by adding 1 to each entry:

$$\begin{bmatrix} A + \mathbf{1}\mathbf{1}^\mathsf{T} & 0 \\ 0 & \mathbf{1}\mathbf{1}^\mathsf{T} - A^\mathsf{T} \end{bmatrix} ,$$

$\mathrm{val}(A + \mathbf{1}\mathbf{1}^\mathsf{T}) = \mathrm{val}(A) + 1 > 0$, and $\mathrm{val}(\mathbf{1}\mathbf{1}^\mathsf{T} - A^\mathsf{T}) = 1 - \mathrm{val}(A) > 0$. □

Since a win-lose matrix game of value 0 or 1 as well as a win-lose-draw matrix game of value -1 or 1 has trivial patience 1 for both players we have the following relations, complementing Proposition 9.

Corollary 12.

$$\widehat{\tau_{wl}}(2n) \geq \tau_{wl}(n) \ , \quad \widehat{\tau_{wld}}(2n) \geq \tau_{wld}(n) \ .$$

Next we consider the relationship between win-lose and win-lose-draw games. Immediately from the definition we have.

Proposition 13.

$$\tau_{wl}(n) \leq \tau_{wld}(n) \ , \quad \widehat{\tau_{wl}}(n) \leq \widehat{\tau_{wld}}(n) \ .$$

We next show how to convert win-lose-draw matrix games into win-lose matrix games.

Proposition 14. *Let A be a $n \times n$ $(-1,0,1)$ matrix game. Define the $2n \times 2n$ $(0,1)$ matrix game B obtained from A by replacing (-1)-entries by the 2×2 all-zero matrix, 0-entries by the 2×2 identity matrix, and 1-entries by the 2×2 all-ones matrix. Then $\tau^1(A) \leq \tau^1(B)$ and $\tau^2(A) \leq \tau^2(B)$*

Proof. Let x' be an optimal strategy for Player 1 in B. Define the strategy x for Player 1 in A by $x_i = x'_{2i-1} + x'_{2i}$. By definition we have $(x'^{\mathsf{T}}(2B - \mathbf{1}\mathbf{1}^{\mathsf{T}}))_j = 2(x'^{\mathsf{T}}B)_j - 1 \geq 2\,\mathrm{val}(B) - 1$ for all j. We then get

$$\begin{aligned}(x^{\mathsf{T}}A)_j &= \frac{1}{2}(x'^{\mathsf{T}}(2B - \mathbf{1}\mathbf{1}^{\mathsf{T}}))_{2j-1} + \frac{1}{2}(x'^{\mathsf{T}}(2B - \mathbf{1}\mathbf{1}^{\mathsf{T}}))_{2j} \\ &\geq \frac{1}{2}(2\,\mathrm{val}(B) - 1) + \frac{1}{2}(2\,\mathrm{val}(B) - 1) = 2\,\mathrm{val}(B) - 1\end{aligned}$$

Similarly, for an optimal strategy y' for Player 2 in B we define the strategy y for Player 2 in A by $y_i = y'_{2i-1} + y'_{2i}$ and obtain $(Ay)_i \leq 2\,\mathrm{val}(B) - 1$ for all i. It follows that $\mathrm{val}(A) = 2\,\mathrm{val}(B) - 1$ and x and y are optimal strategies in A. Since the patience of x is at most the patience of x' and the patience of y is at most the patience of y' the result follows. □

We then immediately have the following converse to Proposition 13.

Corollary 15.

$$\tau_{wld}(n) \leq \tau_{wl}(2n) \ , \quad \widehat{\tau_{wld}}(n) \leq \widehat{\tau_{wl}}(2n) \ .$$

Patience Upper Bound

Let A be a $n \times n$ matrix game with integer entries. We shall make use of Equation (1)[1]. Let B be a $m \times m$ submatrix of A corresponding to an optimal strategy x of Player 1. Define the auxiliary $(m+1) \times (m+1)$ matrix $M = \begin{bmatrix} 0 & \mathbf{1}^{\mathsf{T}} \\ \mathbf{1} & B \end{bmatrix}$. Computing the determinant of M by expanding along first column and then the first row we find that $\det(M) = -\mathbf{1}^{\mathsf{T}}\,\mathrm{adj}(B)\mathbf{1}$. Since the entries of $\mathrm{adj}(B)$ are integers, by Equation (1) we have that either $x_i = 0$ or $x_i \geq 1/|\det(M)|$. We may thus bound the patience of x by $|\det(M)|$.

Now, in case A is a $(0,1)$ matrix game, the matrix M is a $(0,1)$ matrix as well, of dimension at most $(n+1) \times (n+1)$. A bound of Faddeev and Sominskii [5] then gives $|\det(M)| \leq (n+2)^{\frac{n+2}{2}}/2^{n+1}$.

Similarly, in case A is a $(-1,0,1)$ matrix game, the matrix M is a $(-1,0,1)$ matrix as well, of dimension at most $(n+1) \times (n+1)$, and using the Hadamard bound we get $|\det(B)| \leq (n+1)^{\frac{n+1}{2}}$. Combining these, the proof of Proposition 1 follows.

[1] Alternatively one could do essentially the same derivation using the standard formulation of matrix games as linear programs.

3 Patience and Ill-conditioned Matrices

From Theorem 8 we see that a nonsingular $n \times n$ matrix A with $\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1} \neq 0$ defines a matrix game of patience at least $\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1}$, provided that *both* $\mathbf{1}^\mathsf{T} A^{-1}/\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1} > 0$ and $A^{-1}\mathbf{1}/\mathbf{1}^\mathsf{T} A^{-1}\mathbf{1} > 0$.

For a non-singular $n \times n$ matrix A, let $B = A^{-1} = (b_{ij})$ and define $\chi(A) = \max_{i,j} |b_{ij}|$. The problem of constructing $(0,1)$ or $(-1,1)$ matrices A for which $\chi(A)$ is large was considered first by Graham and Sloane [8], and later by Alon and Vũ [1]. Such matrices have besides the direct application of constructing ill-conditioned matrices, several applications such as flat simplices, coin weighing, indecomposable hypergraphs, and weights of Boolean threshold functions [8, 12, 1, 17, 2].

Define $\chi_1(n)$ as the maximum of $\chi(A)$ over all non-singular $n \times n$ $(0,1)$ matrices A. Define $\chi_2(n)$ to be the analogous quantity where $(-1,1)$ matrices are considered instead. Alon and Vũ [1], building on the techniques of Håstad, gave a near optimal construction of ill-conditioned matrices. More precisely they provide for every n an explicit $n \times n$ $(0,1)$ matrix A_1 and an explicit $n \times n$ $(-1,1)$ matrix A_2 such that $\chi(A_i) \geq n^{n/2}/2^{n(2+o(1))}$ for $i = 1,2$. When n is a power of 2 these lower bounds may be improved to $n^{n/2}/2^{n(1+o(1))}$. Upper bounds for $\chi_i(n)$ are derived from the Hadamard inequality.

Theorem 16 (Alon and Vũ)**.**

$$n^{\frac{n}{2}}/2^{n(2+o(1))} \leq \chi_i(n) \quad \textit{for } i = 1,2$$
$$\chi_1(n) \leq n^{\frac{n}{2}}/2^{n-1}$$
$$\chi_2(n) \leq (n-1)^{\frac{n-1}{2}}/2^{n-1}$$

In their application to indecomposable hypergraphs, Alon and Vũ construct a non-singular $(0,1)$ $n \times n$ matrix D such that $y = D^{-1}\mathbf{1} \geq 0$ and $|y_1/y_2| \geq n^{\frac{n}{2}}/2^{n(2+o(1))}$. Unfortunately this construction does not ensure that also $\mathbf{1}^\mathsf{T} D^{-1} \geq 0$, and hence we cannot use it to give a matrix game of large patience as described above.

It does however turn out that *any* matrix A with large $\chi(A)$ can be used to construct a matrix game with patience $\chi(A)$, as will be explained in the following section.

3.1 The Matrix Switching Game

Let B be any $n \times n$ matrix. We call the operation of flipping all the signs of an entire row a *row switch*, and similarly the operation of flipping all the signs of an entire column a *column switch*. We are interested in the sum of all entries, $\mathbf{1}^\mathsf{T} B'\mathbf{1}$, for matrices B' obtained from B using row and column switches. The matrix switching game for B is to find such a matrix B' maximizing $\mathbf{1}^\mathsf{T} B'\mathbf{1}$, the *value* of the switching game. Equivalently we may view the matrix switching game as the problem of maximizing the bilinear form $x^\mathsf{T} By$ over $x, y \in \{-1,1\}^n$. The special case of matrix switching game for $(-1,1)$ matrices is known as the Gale-Berlekamp switching game [22, Chapter 6] (or simply the Berlekamp switching game [21]).

Directly from the definition of the matrix switching game we have the following.

Lemma 17. *Let B be any $n \times n$ matrix, such that $\mathbf{1}^\mathsf{T} B \mathbf{1}$ can not be increased by a row or column switch. Then $\mathbf{1}^\mathsf{T} B \geq 0$ and $B\mathbf{1} \geq 0$.*

It is easy to see that the value of the matrix switching game is at least as large as the largest element of the matrix.

Lemma 18. *Let $B = (b_{ij})$ be any $n \times n$ matrix. Then there exist $x \in \{-1,1\}^n$ and $y \in \{-1,1\}^n$ such that $x^\mathsf{T} B y \geq \max_{ij} |b_{ij}|$.*

Proof. Let b_{ij} be the entry of largest absolute value in B. First perform column switches in B to make all entries of row i non-negative. Next perform a row switch in any row where the sum of the entries of the row is negative.

Thus the value of the game is at least $\max_{ij} |b_{ij}|$. □

Proposition 19. *Let A be a nonsingular $(-1,1)$ $n \times n$ matrix. Then there exist a block checkerboard sign pattern Σ such that the $(n+1) \times (n+1)$ $(-1,0,1)$ matrix game B given by*

$$B = \begin{bmatrix} 1 & 0 \\ 0 & A \circ \Sigma^\mathsf{T} \end{bmatrix}$$

satisfies $\tau^1(B) \geq \chi(A)$ and $\tau^2(B) \geq \chi(A)$.

Proof. Let $x, y \in \{-1,1\}^n$ maximize the bilinear form $x^\mathsf{T} A^{-1} y$, and define $\Sigma = xy^\mathsf{T}$. Let $C = A \circ \Sigma^\mathsf{T}$. Then $C^{-1} = A^{-1} \circ \Sigma$. By Lemma 17 we have $\mathbf{1}^\mathsf{T} C^{-1} \geq 0$ and $C^{-1}\mathbf{1} \geq 0$, and by Lemma 18 we have $\mathbf{1}^\mathsf{T} C^{-1} \mathbf{1} \geq \chi(A) > 0$. Hence Theorem 8 gives $v := \mathrm{val}(C) = 1/\mathbf{1}^\mathsf{T} C^{-1} \mathbf{1} > 0$.

Optimal strategies in B for Player 1 must then be of the form $(v/(1+v), x'^\mathsf{T}/(1+v))$, and similarly optimal strategies in B for Player 2 must be of the form $(v/(1+v), y'^\mathsf{T}/(1+v))$, where x and y are optimal strategies in C for Player 1 and Player 2. It follows these are of patience at least $(1+v)/v \geq 1/v = \mathbf{1}^\mathsf{T} C^{-1} \mathbf{1} \geq \chi(A)$. □

Combining this with Theorem 16 gives the proof of Theorem 2 and Corollary 3.

While these bounds matches the patience upper bound of Proposition 1 up to the constant in the exponent, and in the case of win-lose-draw games in fact match up to an exponential factor, a drawback is that the matrices giving these lower bounds are not very *explicit*. We can formalize such a statement using computational complexity theory, by looking at the complexity of the computational task to compute the $n \times n$ matrix of the family, given as input the number n.

The matrices constructed by Alon and Vũ from Theorem 16 are in fact very explicit in this sense as will be detailed in Section 5.8. Given such a matrix A, the proof of Proposition 19 proceeds to invert A, and then solve the matrix switching game for A^{-1}. While inverting A is a polynomial time computation (in n) it turns out that solving the Matrix Switching game is NP-hard [18] (this is true even in the case of the Gale-Berlekamp game). This NP-hardness result is not a real obstacle though; all that is needed for the proof of Proposition 19 to go through is that the vectors $x, y \in \{-1,1\}^n$ describe a *local maximum* of the matrix switching game, in the sense

that the bilinear form $x^\mathsf{T} A^{-1} y$ cannot be increased by changing a single coordinate of x or y, and furthermore that the value of this local maximum is at least $\chi(A)$. We will discuss this issue of explicitness in more detail in the next section.

3.2 Local Search and Bipartite Maximum Cut

In this section we will consider local search algorithms from the perspective of computational complexity in order to properly discuss the explicitness of the matrix games of high patience constructed in the previous section.

Johnson et.al. [13] formalized the notion of polynomial time local search problems by the complexity class PLS. A *local search problem* P is specified by the following:

- A set $\mathscr{I}$ of *instances*, given by a polynomial time algorithm that decides if a given string represents an instance of P.
- For each instance x, a set $\mathscr{F}(x)$ of *feasible solutions*, whose elements are strings bounded polynomially in the length of x.
- A specification of P as either a maximization problem or a minimization problem, together with a cost measure $c(x, y)$, for $x \in \mathscr{I}$ and $y \in \mathscr{F}(x)$, to be maximized or minimized, respectively.
- For each solution $y \in \mathscr{F}(x)$, a set $\mathscr{N}(x, y)$ of *neighboring solutions*.

We then say that P is in PLS if there exist polynomial time algorithms A, M, and C as follows:

- A, on input $x \in \mathscr{I}$, produces a solution $y \in \mathscr{F}(x)$.
- M, on input $x \in \mathscr{I}$ and $y \in \mathscr{F}(x)$, computes $c(x, y)$.
- C, on input $x \in \mathscr{I}$ and $y \in \mathscr{F}(x)$, either reports that y is the best solution in $\mathscr{N}(x, y)$, or produces a better solution $y' \in \mathscr{N}(x, y)$.

The *standard algorithm* to solve the problem P is to first run algorithm A, and then repeatedly run algorithm C until a locally optimum is found.

We can cast the matrix switching game of the previous section in this framework, showing that the problem is in PLS: Instances are integer $n \times n$ matrices A. A solution is given by a pair of vectors $x, y \in \{-1, 1\}^n$. The cost of a solution is $x^\mathsf{T} A y$, and neighbors of (x, y) are those obtained by flipping the sign of an entry in either x or y. Algorithms A, M, and C are immediate.

We note that to implement the proof of Proposition 19 one would need not only a locally optimum, but a locally optimum of a certain quality. We will suggest that it might be hard to even just find a locally optimum, or in other words solve the matrix switching game, without using specific knowledge of the input matrix A.

Using the notions of *reductions* and *completeness* one may in a similar way to the theory of NP-completeness argue that certain problem in PLS are unlikely to be solvable in polynomial time.

A PLS-problem P_1 is PLS-reducible [13] to another PLS-problem P_2, if there are polynomial time computable functions f and g, such that f maps instances of P_1 to instances of P_2, g maps pairs (y', x), where y' is a solution of $f(x)$, to a solution $g(y', x)$ of x, and in the case when y' is a local optimum of P_2, then $g(y, x)$ is a local optimum of P_1. With the notion of reduction in place, we say that a PLS-problem P is PLS-complete, if every other PLS-problem reduces to P by a PLS-reduction.

Schäffer and Yannakakis [19] found a number of natural local search problems to be PLS-complete. In particular they showed that the MaxCut problem is PLS-complete under the Flip neighborhood. Here the MaxCut problem is given as follows: Instances are graphs $G = (V, E)$, $V = \{1, \ldots, n\}$, with integer weights on the edges, w_{ij}. Solutions are cuts $(S, \overline{S})$, $S \subseteq V$, of the vertices, and the cost of a solution is the sum of the weights of edges connecting vertices across the cut, $\sum_{ij \in (S,\overline{S})} w_{ij}$. The Flip neighborhood is defined by the action of moving a single vertex across the cut.

Letting $x_i \in \{-1, 1\}$ be given by $x_i = 1$ if and only if $i \in S$, we see the MaxCut problem is equivalent to maximizing the quadratic form $\sum_{i<j} w_{ij}(1 - x_i x_j)/2$ over $x \in \{-1, 1\}^n$. Conversely, the problem of maximizing a quadratic form $x^\mathsf{T} A x$ over $x \in \{-1, 1\}^n$ can be formulated as a MaxCut instance.

Since, as we have seen, the matrix switching game is equivalent to maximizing a *bilinear* form over $\{-1, 1\}^n$, it is not surprising that we can also reformulate the matrix switching game as a maximum cut problem.

Indeed, let A be any $n \times n$ matrix A, and $x, y \in \{-1, 1\}^n$. Then

$$(x^\mathsf{T} A y - \mathbf{1}^\mathsf{T} A \mathbf{1})/2 = \sum_{i,j} -a_{ij}(1 - x_i y_j)/2 \ . \tag{3}$$

Define now the bipartite graph $G_A = (V_1, V_2, E)$, with $V_1 = \{1, \ldots, n\}$ and $V_2 = \{1', \ldots, n'\}$, with an edge (i, j) whenever $a_{ij} \neq 0$ of weight $w_{ij} = -a_{ij}$. This forms an instance of the *bipartite* MaxCut problem, which is the restriction of the MaxCut problem to bipartite graphs. That is, solutions are cuts $(S, \overline{S})$, $S \subseteq V_1 \cup V_2$, of the vertices, and the cost of a solution is again the sum of the weights of edges connecting vertices across the cut, $\sum_{ij \in (S,\overline{S})} w_{ij}$. Letting $x_i = 1$ if and only if $i \in S$ and $y_i = 1$ if and only if $i' \in S$, this is exactly the quantity expressed in Equation (3).

The bipartite MaxCut problem is NP-hard [16, 18]. But as argued, the question relevant for us should be if the bipartite MaxCut problem is PLS-complete under the Flip neighborhood as well (observe that the Flip neighborhood corresponds to row and column switches in the corresponding matrix switching game).

Despite our efforts, we have not been able to resolve this question. On the other hand we have not been able to solve the problem in polynomial time either. Failing to prove the problem to be PLS-complete does not *a priori* suggest that the local search problem might be easy. Actually, unlike the theory of NP-completeness, it is not rare to find local search problems that are neither known to be polynomial time solvable, nor PLS-complete. An important such example is the Travelling Salesman problem. Krentel showed the problem to be PLS-complete under the k-Opt neighborhood (which is defined by the process of removing from a tour arbitrary k edges and reconnecting the k resulting pieces into a new tour), for sufficiently large k [14]. It is still an open

problem if the same is true for the simple 2-Opt and 3-Opt neighborhoods [19].

We are able to show a weaker statement about the bipartite MaxCut problem. Define the 2-Flip neighborhood by the action of moving up to 2 vertices across the cut. We then have the following hardness result.

Proposition 20. *The bipartite* MaxCut *problem is* PLS*-complete under the 2-*Flip *neighborhood.*

Proof. The result is proved by a simple reduction from the ordinary MaxCut problem. Let $G = (V, E)$ with $V = \{1, \ldots, n\}$ be a graph with weight w forming a MaxCut instance. Let $M = 1 + \sum_{ij} |w_{ij}|$. Define a bipartite graph $G' = (V_1, V_2, E')$ with weights $w'_{ij'}$ as follows. We let $V_1 = \{1, \ldots, n\}$ and $V_2 = \{1', \ldots, n'\}$. Vertices i and i' are joined by an edge of "huge" weight M. Whenever $ij \in E$ we join i and j' as well as i' and j by an edge of weight $-w_{ij}$. We thus let $f(G, w) = (G', w')$ be the first function of the reduction.

First, observe if a given cut $(S', \overline{S'})$ of the vertices of G' does not satisfy $i \in S'$ if and only if $i' \in \overline{S'}$, then the weight of the cut can be improved by moving either of i or i' to the other partition of the cut. We will thus in the following calculation assume that this is not the case for any i. Now a cut $(S', \overline{S'})$ of G' induces a cut $(S, \overline{S})$ of G defined by $i \in S$ if and only if $i \in S'$. We then have

$$\begin{aligned} w'(S', \overline{S'}) &= nM + \sum_{i \in S', j' \in \overline{S'}} w'_{ij'} + \sum_{i' \in S', j \in \overline{S'}} w'_{i'j} = nM + 2 \sum_{i \in S', j' \in \overline{S'}} w'_{ij'} \\ &= nM - 2 \sum_{i \in S, j \in S} w_{ij} = (nM - 2 \sum_{ij} w_{ij}) + 2 \sum_{i \in S, j \in \overline{S}} w_{ij} \end{aligned}$$

From this we see that $w(S, \overline{S}) = w'(S', \overline{S'})/2 + (\sum_{ij} w_{ij} - nM/2)$.

We can thus simply define the last function g of the reduction as $g(S', \overline{S'}) = (S, \overline{S})$. □

Remark. One can observe that the reduction above in fact satisfies the notion of being *tight* as defined by Schäffer and Yannakakis [19]. We will not define the notion here, but just remark that it implies that the reduction gives a number of additional results besides showing PLS-completeness. For instance it implies that the standard algorithm must take exponential time in the worst case. And this is true no matter how the neighbors are chosen in each step of the local search procedure.

4 Explicit Examples of Exponential Patience

In this section we give several examples of matrix games that requires exponential patience. All of the examples are special *Toeplitz matrices*, that were constructed earlier for the purpose of studying ill-conditioned matrices [8] and extremal matrices with respect to the determinant [3]. Here a $n \times n$ matrix $A = (a_{ij})$ is called a Toeplitz matrix, if every left-to-right descending diagonal of A is constant, i.e. $a_{ij} = a_{i+1,j+1}$

for all i and j. We may thus specify A by the $2n-1$ numbers $a_{n,1},\dots,a_{1,1},\dots,a_{1,n}$. We shall use the notation

$$A = \mathbf{T}(a_{n,1}\dots a_{2,1}\underline{a_{1,1}}a_{1,2}\dots a_{1,n})$$

with the upper left element of A underlined.

For all the examples we use Theorem 8 to compute the patience. For the first matrix we can do this directly, whereas for the last two examples we need to go via Lemma 5, using that the inverse of the matrices turn out to weakly obey block checkerboard sign patterns. This in turn gives rise to win-lose-draw matrix games, whereas the first matrix give rise to a win-lose matrix game. These $n\times n$ win-lose-draw matrix games can be converted to $2n\times 2n$ win-lose matrix games of the same patience using Proposition 14, but we can do better in these examples using the tight connection between $n\times n$ $(0,1)$ matrices and $(n+1)\times(n+1)$ $(-1,1)$ matrices, we describe next. Let A be a $n\times n$ $(0,1)$ matrix. Then define a $(n+1)\times(n+1)$ $(-1,1)$ matrix $\Phi(A)$ by

$$\Phi(A) = \begin{bmatrix} 1 & \mathbf{1}^\mathsf{T} \\ -\mathbf{1} & 2A-\mathbf{1}\mathbf{1}^\mathsf{T} \end{bmatrix} . \tag{4}$$

If A is invertible then $\Phi(A)$ is invertible as well and we have (cf. [1])

$$\Phi(A)^{-1} = \frac{1}{2}\begin{bmatrix} 2-\mathbf{1}^\mathsf{T}A^{-1}\mathbf{1} & -\mathbf{1}^\mathsf{T}A^{-1} \\ A^{-1}\mathbf{1} & A^{-1} \end{bmatrix} .$$

4.1 $(0,1)$ Hessenberg matrix

Ching [3] considered the following Hessenberg-Toeplitz matrices. For given n define the $n\times n$ matrix $D_n=(d_{ij})$ by $d_{i,i-k}=1$ if $k\in\{-1,0,2,4,\dots\}$ and $d_{i,i-k}=0$ otherwise. Alternatively, $D_n=\mathbf{T}(0101\dots10\underline{1}10000\dots)$.

It was shown by Ching that for any $n\times n$ $(0,1)$ Hessenberg matrix A_n (i.e. A_n is a triangular matrix except that the diagonals above and below the main diagonal may also be nonzero) with $n>2$ satisfies $|\det(A_n)|\le\det(D_n)$. We remark however that the matrix D_n is actually the transpose of the upper-right $n\times n$ submatrix of the matrix T_{n+1}, defined by Graham and Sloane, that we will consider in the next subsection. Also, in fact Graham and Sloane already obtained the result of Ching while showing properties of T_n (see [8], Lemma 9).

For us, however, the matrix D_n has the advantage over the other examples we consider, that it directly gives a matrix game where the optimal strategies are totally mixed.

Example.

$$D_5 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Let F_n denote the nth Fibonacci number, given by $F_n = F_{n-1} + F_{n-2}$ for $n > 2$, $F_1 = F_2 = 1$, and $F_0 = 0$. Alternatively

$$F_n = \left(\varphi^n - (1-\varphi)^n\right)/\sqrt{5} \ ,$$

where $\varphi = (1+\sqrt{5})/2 = 1.61803...$ is the golden ratio.

We shall not compute the inverse of D_n, but just determine the information needed to apply Theorem 8. We can determine a recurrence for the determinant of D_n by expanding along the first row, and obtain $\det(D_n) = \det(D_{n-1}) + \det(D_{n-2})$ for $n > 2$. Also $\det(D_1) = \det(D_2) = 1$, and hence $\det(D_n) = F_n$. Similarly, one may easily verify the following.

Lemma 21. *Let $\widetilde{x}, \widetilde{y} \in \mathbb{R}^n$ be defined by $\widetilde{x}_i = \widetilde{y}_{n-i+1} = F_i/F_n$ for $i < n$ and $\widetilde{x}_n = \widetilde{y}_1 = F_{n-2}/F_n$. Then $\widetilde{x}^\mathsf{T} D_n = \mathbf{1}^\mathsf{T}$ and $D_n \widetilde{y} = \mathbf{1}$. Also $\sum_{i=1}^n \widetilde{x}_i = \sum_{i=1}^n \widetilde{y}_i = (2F_n - 1)/F_n$, and hence $\mathbf{1}^\mathsf{T} D_n^{-1} \mathbf{1} = (2F_n - 1)/F_n$.*

From this and Theorem 8 we immediately obtain a statement about the matrix D_n viewed as a matrix game.

Proposition 22. *The matrix game D_n has value $v = F_n/(2F_n - 1)$ and unique optimal strategies x and y for the two players, where $x_i = y_{n-i+1} = F_i/F_n$ for $i < n$ and $x_n = y_1 = vF_{n-2}/F_n$. In particular we have $x_1 = y_n = 1/(2F_n - 1)$, and the patience of both x and y is precisely $2F_n - 1$, and asymptotically $\Omega(\varphi^n)$.*

4.2 Triangular matrix

Graham and Sloane [8] defined the following triangular matrices. For given n define the $n \times n$ matrix $t_n = (t_{ij})$ by $t_{i,i+k} = 1$ if $k \in \{0, 1, 3, 5, \dots\}$ and $t_{i,i-k} = 0$ otherwise. Alternatively $t_n = \mathbf{T}(0 \dots 0\underline{1}101010 \dots)$.

Example.

$$t_6 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

As pointed out by Graham and Sloane, the inverse of t_n is again a triangular Toeplitz matrix, namely

$$t_n^{-1} = \mathbf{T}(0, \dots, 0, \underline{1}, -F_1, F_2, -F_3, \dots, (-1)^{n-1} F_{n-1}) \ .$$

This means that t_n^{-1} weakly obeys the (block) checkerboard sign pattern $\Sigma_n = (\sigma_{ij})$ given by $\sigma_{ij} = (-1)^{i+j}$. We thus have that the matrix $\bar{t}_n = t_n \circ \Sigma = \mathbf{T}(0 \dots 0, \underline{1}, -1, 0, -1, 0, -1, 0 \dots)$ has inverse $\mathbf{T}(0, \dots, 0, \underline{1}, F_1, F_2, \dots, F_{n-1})$.

One may now easily verify the following.

Lemma 23. *Let $\widetilde{x}, \widetilde{y} \in \mathbb{R}^n$ be defined by $\widetilde{x}_i = \widetilde{y}_{n-i+1} = F_{i+1}$. Then $\widetilde{x}^\mathsf{T}\bar{t}_n = \mathbf{1}^\mathsf{T}$ and $\bar{t}_n\widetilde{y} = \mathbf{1}$. Also $\sum_{i=1}^n \widetilde{x}_i = \sum_{i=1}^n \widetilde{y}_i = \mathbf{1}^\mathsf{T}\bar{t}_n^{-1}\mathbf{1} = F_{n+3} - 2$.*

And from Theorem 8 we obtain the following.

Proposition 24. *The $(-1, 0, 1)$ matrix game $\bar{t}_n$ has value $v = 1/(F_{n+3}-2)$ and unique optimal strategies x and y for the two players, where $x_i = y_{n-i+1} = F_{i+1}/(F_{n+3} - 2)$. In particular we have $x_1 = y_n = 1/(F_{n+3} - 2)$, and the patience of both x and y is precisely $F_{n+3} - 2$, and asymptotically $\Omega(\varphi^n)$.*

We next derive a win-lose matrix game of similar patience using Equation (4). Define a vector f and its reverse f_R by

$$f = ((-1)^{n-2}F_{n-2}, \ldots, -F_1, F_0, -1)^\mathsf{T}$$
$$f_R = (-1, F_0, -F_1, \ldots, (-1)^{n-2}F_{n-2})^\mathsf{T} \ .$$

One may verify that the inverse of $\Phi(t_n)$ is then given by

$$\Phi(t_n)^{-1} = \frac{1}{2}\begin{bmatrix} (-1)^{n-2}F_{n-3} & f_R^\mathsf{T} \\ -f & t_n^{-1} \end{bmatrix} \ ,$$

and we see that $\Phi(t_n)^{-1}$ weakly obeys the similar (block) checkerboard sign pattern $-\Sigma_{n+1}$. Thus the matrix $\overline{t'}_n = \Phi(t_n)^{-1} \circ (-\Sigma_{n+1})$ has a non-negative inverse.

One may now easily verify the following.

Lemma 25. *Let $\widetilde{x}, \widetilde{y} \in \mathbb{R}^{n+1}$ be defined by $\widetilde{x}_1 = \widetilde{y}_{n+1} = F_{n-1}$, and $\widetilde{x}_i = \widetilde{y}_{n-i+2} = F_{i-1}$, for $i \geq 2$. Then $\widetilde{x}^\mathsf{T}\overline{t'}_n = \mathbf{1}^\mathsf{T}$ and $\overline{t'}_n\widetilde{y} = \mathbf{1}$. Also $\sum_{i=1}^{n+1} \widetilde{x}_i = \sum_{i=1}^{n+1} \widetilde{y}_i = \mathbf{1}^\mathsf{T}\overline{t'}_n^{-1}\mathbf{1} = F_{n-1} + F_{n+2} - 1$.*

And from Theorem 8 we obtain the following.

Proposition 26. *The $(-1, 1)$ matrix game $\overline{t'}_n$ has value $v = 1/(F_{n-1}+F_{n+2}-1)$ and unique optimal strategies x and y for the two players, where $x_1 = y_{n+1} = F_{n-1}/(F_{n-1}+F_{n+2} - 1)$ and $x_i = y_{n-i+2} = F_{i-1}/(F_{n-1} + F_{n+2} - 1)$, for $i \geq 2$*

In particular we have $x_2 = y_{n+1} = 1/(F_{n-1} + F_{n+2} - 1)$, and the patience of both x and y is precisely $F_{n-1} + F_{n+2} - 1$, and asymptotically $\Omega(\varphi^n)$.

4.3 Toeplitz matrix

Graham and Sloane [8] additionally defined the following Toeplitz matrices. For given n define the $n \times n$ matrix $T_n = (t_{ij})$ by $t_{i,i-k} = 1$ if $k \in \{-3, -1, 0, 3, 4, 6, 7, \ldots\}$ and $t_{i,i-k} = 0$ otherwise. Alternatively $T_n = \mathbf{T}(\ldots 11001100\underline{1}10100000\ldots)$. The inverse of T_n was computed by Graham and Sloane using Trench's algorithm, and is described below.

Define sequences $\{p_n\}$ and $\{q_n\}$ of integers by $q_0 = 0$, $p_0 = q_1 = q_2 = 1$, and inductively

$$p_n = p_{n-1} + q_{n-1} \ , \quad \text{for } n \geq 1 \ ,$$
$$q_n = q_{n-1} + p_{n-2} \ , \quad \text{for } n \geq 3 \ .$$

From these definitions one may determine their asymptotics as $p_n, q_n = \Omega(\rho_3^n)$, where $\rho_3 = 1.75488...$ is the largest root of $x^3 - 2x^2 + x - 1$.

It turns out that the inverse of T_n is symmetric about the top-right to bottom-left diagonal and of the following form.

$$T_n^{-1} = \left[\begin{array}{rrrrrr|rrr} -1 & -p_2 & -p_3 & \cdots & -p_{n-4} & -p_{n-3} & q_{n-2} & -q_{n-3}{}^{\dagger} & p_{n-2} \\ 1 & q_1 & q_2 & \cdots & q_{n-5} & q_{n-4} & -p_{n-4} & p_{n-5} & -q_{n-3}{}^{\dagger} \\ -1 & -q_2 & -q_3 & \cdots & -q_{n-4} & -q_{n-3} & p_{n-3} & -p_{n-4} & q_{n-2} \\ \hline 1 & p_1 & p_2 & \cdots & p_{n-5} & p_{n-4} & -q_{n-3} & q_{n-4} & -p_{n-3} \\ 0 & 1 & p_1 & \cdots & p_{n-6} & p_{n-5} & -q_{n-4} & q_{n-5} & -p_{n-4} \\ 0 & 0 & 1 & \cdots & p_{n-7} & p_{n-6} & -q_{n-5} & q_{n-6} & -p_{n-5} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \end{array}\right]$$

Note that T_n^{-1} weakly obeys the block checkerboard sign pattern

$$\Sigma = (-1, 1, -1, 1, \ldots, 1)(1, \ldots, 1, -1, 1, -1)^{\mathsf{T}} \ .$$

By Lemma 5 the matrix $\overline{T}_n = T_n \circ \Sigma^{\mathsf{T}}$ has as inverse a matrix with all entries being non-negative. We only give an asymptotic lower bound on the patience of $\overline{T}_n$.

Proposition 27. *The $(-1, 0, 1)$ matrix game $\overline{T}_n$ has unique optimal strategies x and y for the two players each of patience $\Omega(\rho_3^n)$.*

Proof. Note that $\mathbf{1}^{\mathsf{T}}\overline{T}_n^{-1}\mathbf{1} = \Omega(p_n) = \Omega(\rho_3^n)$, $(\mathbf{1}^{\mathsf{T}}\overline{T}_n^{-1})_1 = (\overline{T}_n^{-1}\mathbf{1})_n = 4$. Using Theorem 8 we have that $x_1 = y_n = 1/\Omega(\rho_3^n)$, and the patience of x and y is $\Omega(\rho_3^n)$. □

Again we may derive a win-lose matrix game of similar patience using Equation (4). Define a vector g and its reverse g_R by

$$g = (1, 0, 1, 0, \ldots, 0)^{\mathsf{T}}$$
$$g_R = (0, \ldots, 0, 1, 0, 1)^{\mathsf{T}} \ .$$

One may verify that the inverse of $\Phi(T_n)$ is then given by

$$\Phi(T_n)^{-1} = \frac{1}{2}\begin{bmatrix} 0 & -g_R^{\mathsf{T}} \\ g & T_n^{-1} \end{bmatrix} \ ,$$

and we see that $\Phi(T_n)^{-1}$ weakly obeys the similar (block) checkerboard sign pattern

$$\Sigma' = (1, -1, 1, -1, 1, \ldots, 1)(-1, 1, \ldots, 1, -1, 1, -1)^{\mathsf{T}} \ .$$

Thus the matrix $\overline{T'}_n = \Phi(T_n)^{-1} \circ \Sigma'$ has a non-negative inverse. We then have the following.

Proposition 28. *The $(-1, 1)$ matrix game $\overline{T'}_n$ has unique optimal strategies x and y for the two players each of patience $\Omega(\rho_3^n)$.*

Proof. Note that $\mathbf{1}^{\mathsf{T}}\overline{T'}_n^{-1}\mathbf{1} = \Omega(p_n) = \Omega(\rho_3^n)$, $(\mathbf{1}^{\mathsf{T}}\overline{T}_n^{-1})_1 = (\overline{T}_n^{-1}\mathbf{1})_1 = 1$. Using Theorem 8 we have that $x_1 = y_1 = 1/\Omega(\rho_3^n)$, and the patience of x and y is $\Omega(\rho_3^n)$. □

† In [8] this entry is incorrectly written as $-q_{n-1}$.

5 Explicit matrices of almost worst case patience

In this section we present the proof of our main result, Theorem 4. The overall strategy for the proof is similar to the last examples of Section 4. Namely for any $n = 2^m$, we first construct a non-singular $n \times n$ $(-1, 1)$ matrix A for which $\chi(A) \geq n^{\frac{n}{2}}/2^{n(1+o(1))}$. This matrix is a specific instance of the ill-conditioned matrices constructed by Alon and Vũ [1]. This immediately means that the inverse of A has an entry of magnitude $n^{\frac{n}{2}}/2^{n(1+o(1))}$ by the analysis of Alon and Vũ (or alternatively it is easily derived from the more involved analysis of this section). But just as important for us, the specifics of our construction allows us to show that A^{-1} weakly obeys a block checkerboard sign pattern Σ. Using Lemma 5 this means that the $(-1, 1)$ matrix $B = (b_{ij}) = A \circ \Sigma^{\mathsf{T}}$ has a non-negative inverse. We can then apply the result of Shapley and Snow to analyze the patience of the matrix game B. Specifically by Theorem 8, the matrix game B has unique optimal strategies x and y. In particular the strategy x is given by

$$x^{\mathsf{T}} = \mathbf{1}^{\mathsf{T}} B^{-1} / \mathbf{1}^{\mathsf{T}} B^{-1} \mathbf{1} \ .$$

In Section 5.4.1 we compute the first column of A^{-1} and from this analysis we find

$$\sum_{i=1}^{n} b_{i1} = -(1 - m/2) + m/2 = m - 1 \ .$$

Since B^{-1} is non-negative we also have $\mathbf{1}^{\mathsf{T}} B^{-1} \mathbf{1} \geq \chi(A) \geq n^{\frac{n}{2}}/2^{n(1+o(1))}$. Thus the patience of B is at least $n^{\frac{n}{2}}/2^{n(1+o(1))}$ as well.

The rest of the section is organized as follows. In Section 5.1 we review the details of the construction of Alon and Vũ. In Section 5.2 we define the specific instance of this construction that we will use. In Sections 5.3 and the following three sections we show that the matrix weakly obeys a specific block checkerboard sign pattern. Finally in Section 5.8 we give a sketch of the proof that our construction is fully explicit.

5.1 The Alon-Vũ matrix

We review here the matrix construction of Alon and Vũ. Let $n = 2^m$ be a power of two. Let $\alpha_1, \ldots, \alpha_n$ be an ordering of the n subsets of $[m]$ that satisfies

- $|\alpha_i| \leq |\alpha_{i+1}|$
- $|\alpha_i \triangle \alpha_{i+1}| \leq 2$.

Such an ordering was shown to exist by Håstad [12]. In Section 5.2 we will construct a particular such ordering. Define for convenience $\alpha_0 = \emptyset$. Given the ordering we define a $n \times n$ $(-1, 1)$ matrix $A = (a_{ij})$ by the following rules (for intuition behind this construction see [12]).

1. If $\alpha_j \cap (\alpha_{i-1} \cup \alpha_i) = \alpha_{i-1} \triangle \alpha_i$ and $|\alpha_{i-1} \triangle \alpha_i| = 2$, then $a_{ij} = -1$.
2. $\alpha_j \cap (\alpha_{i-1} \cup \alpha_i) \neq \emptyset$, but case (1) does not occur, then $a_{ij} = (-1)^{|\alpha_{i-1} \cup \alpha_j| + 1}$.

3. If $\alpha_j \cap (\alpha_{i-1} \cup \alpha_i) = \emptyset$, then $a_{ij} = 1$.

For analyzing the matrix, Alon and Vũ essentially considered a LQ decomposition of A. Define the $(-1,1)$ matrix $Q = (q_{ij})$ by $q_{ij} = (-1)^{|\alpha_i \cap \alpha_j|}$. Then Q is a symmetric Hadamard matrix, $Q^2 = nI$.

For $i > 1$, define subsets A_i of $[n]$ by $A_i = \alpha_{i-1} \cup \alpha_i$, and from these, further define families F_i of subsets of $[n]$ by the following rules.

1. If $|\alpha_{i-1} \triangle \alpha_i| = 2$, then $F_i = \{\alpha_s \mid \alpha_s \subseteq A_i, |\alpha_s \cap (\alpha_{i-1} \triangle \alpha_i)| = 1\}$.
2. If $|\alpha_{i-1} \triangle \alpha_i| = 1$, then $F_i = \{\alpha_s \subseteq A_i\}$

Whenever $|\alpha_i| = k$ we have $|F_i| = 2^k$ for both cases. Next, define the lower triangular matrix $L = (l_{ij})$ as follows. Let $l_{11} = 1$ and $l_{1j} = 0$, for $j > 1$. For $i > 1$ we let

$$l_{ij} = \begin{cases} (\frac{1}{2})^{k-1} - 1 & \text{if } j = i-1 \\ (\frac{1}{2})^{k-1} & \text{if } \alpha_j \in F_i \setminus \{\alpha_{i-1}\} \\ 0 & \text{if } \alpha_j \notin F_i \end{cases} .$$

One can then verify the following.

Lemma 29 ([1], Lemma 2.1.2)**.**

$$A = LQ \ .$$

5.2 The ordering.

Here we construct a specific ordering of the subsets of $[m]$ satisfying the requirements given in Section 5.1. We first construct separate orderings for the subsets of size k for every k. These will have the property that the first set in the order is the lexicographically smallest set, i.e. $\{1, \ldots, k\}$, and the last set of the order is the lexicographically largest set, i.e. $\{m-k+1, \ldots, m\}$.

If $\beta \subseteq [m]$ denote by $(\beta + i)$ the subset of $[m+i]$ defined by $(\beta + i) = \{j + i \in [m+i] \mid j \in \beta\}$ (this definition makes sense also when $\beta = \emptyset$, in which case the result is also $\emptyset$). Let $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_\ell)$ be an ordering of the subsets $\beta_1, \ldots, \beta_\ell \subseteq [m]$. We denote by $\mathrm{rev}(\boldsymbol{\beta})$ the reverse ordering $\mathrm{rev}(\boldsymbol{\beta}) = (\beta_\ell, \ldots, \beta_1)$. By $(\boldsymbol{\beta} + i)$ we denote the ordering $(\boldsymbol{\beta} + i) = ((\beta_1 + i), \ldots, (\beta_\ell + i))$. By $(\{i\} \cup \boldsymbol{\beta})$ we denote the ordering $(\{i\} \cup \boldsymbol{\beta}) = ((\{i\} \cup \beta_1), \ldots, (\{i\} \cup \beta_\ell))$. (These definitions make sense even if $\boldsymbol{\beta}$ is the empty list, resulting in the empty list as well). If $\boldsymbol{\beta}' = (\beta'_1, \ldots, \beta'_{\ell'})$ is another ordering of different subsets, we denote by $\boldsymbol{\beta} \circ \boldsymbol{\beta}'$ the ordering $\boldsymbol{\beta} \circ \boldsymbol{\beta}' = (\beta_1, \ldots, \beta_\ell, \beta'_1, \ldots, \beta'_{\ell'})$.

The separate ordering for subsets of size k of $[m]$, is defined by induction on k and m. Denote this ordering by $\boldsymbol{\beta}_m^{(k)}$. For $k = 0$ we have just the empty set $\emptyset$, and hence $\boldsymbol{\beta}_m^{(0)} = (\emptyset)$. For convenience, define $\boldsymbol{\beta}_m^{(k)}$ as the empty order $\boldsymbol{\beta}_m^{(k)} = ()$, when $k > m$.

We now construct the ordering of subsets of size k of $[m]$ for $m \geq k > 0$. Assume by induction we have ordered the subsets of size $k-1$ of $[m']$ for all $m' \geq k-1$. Then we define

$$\boldsymbol{\beta}_m^{(k)} = (\{1\} \cup (\boldsymbol{\beta}_{m-1}^{(k-1)} + 1)) \circ (\{2\} \cup \mathrm{rev}(\boldsymbol{\beta}_{m-2}^{(k-1)} + 2)) \circ (\boldsymbol{\beta}_{m-2}^{(k)} + 2) \ .$$

We see the ordering begins with the first $\binom{m-1}{k-1}$ sets containing element 1, starting with the set $\{1,\dots,k\}$ and ending with the set $\{1,m-k+2,\dots,m\}$. Next follows the $\binom{m-2}{k-1}$ sets containing element 2 but not element 1, starting with the set $\{2,m-k+2,\dots,m\}$ and ending with the set $\{2,\dots,k+1\}$. Finally follows all the $\binom{m-2}{k}$ sets not containing the elements 1 and 2, starting with the set $\{3,\dots,k+2\}$ and ending with the set $\{m-k+1,\dots,m\}$. We note that between all these neighboring ending sets and starting sets the symmetric difference is exactly 2, and we have covered all $\binom{m}{k} = \binom{m-1}{k-1} + \binom{m-2}{k-1} + \binom{m-2}{k}$ sets. By induction the ordering thus satisfies the requirement of symmetric differences being at most 2. Note that the first set is the lexicographically smallest set, and the last set is the lexicographically largest set.

Example. For $m = 4$ we have the following orderings.

$$\begin{aligned}
\boldsymbol{\beta}_4^{(0)} &= (\emptyset)\\
\boldsymbol{\beta}_4^{(1)} &= (\{1\},\{2\},\{3\},\{4\})\\
\boldsymbol{\beta}_4^{(2)} &= (\{1,2\},\{1,3\},\{1,4\},\{2,4\},\{2,3\},\{3,4\})\\
\boldsymbol{\beta}_4^{(3)} &= (\{1,2,3\},\{1,2,4\},\{1,3,4\},\{2,3,4\})\\
\boldsymbol{\beta}_4^{(0)} &= (\{1,2,3,4\})
\end{aligned}$$

Next, we construct the full ordering $\boldsymbol{\beta}_m$ by combining all $\boldsymbol{\beta}_m^{(k)}$. First, for $k = 2$ we define a *shifted* version of the ordering. Let $\ell = \binom{m}{2}$. Let $\boldsymbol{\beta}_m^{(2)} = (\beta_{m,1}^{(2)},\dots,\beta_{m,\ell}^{(2)})$. Then define $\widehat{\boldsymbol{\beta}_m^{(2)}} = (\widehat{\beta_{m,1}^{(2)}},\dots,\widehat{\beta_{m,\ell}^{(2)}})$ by $\widehat{\beta_{m,i}^{(2)}} = \{(j-1) \mod m \mid j \in \beta_{m,i}^{(2)}\}$. Having this shifted version of sets of size 2 will be critically used in our proof. Now we can finally define

$$\boldsymbol{\beta}_m = \boldsymbol{\beta}_m^0 \circ \boldsymbol{\beta}_m^1 \circ \widehat{\boldsymbol{\beta}_m^{(2)}} \circ \mathrm{rev}(\boldsymbol{\beta}_m^3) \circ \boldsymbol{\beta}_m^4 \circ \mathrm{rev}(\boldsymbol{\beta}_m^5) \circ \boldsymbol{\beta}_m^6 \circ \cdots \circ \boldsymbol{\beta}_m^m \ .$$

In other words, $\boldsymbol{\beta}_m$ begins with the concatenation of $\boldsymbol{\beta}_m^0$, $\boldsymbol{\beta}_m^1$, and $\widehat{\boldsymbol{\beta}_m^{(2)}}$, after which the remaining orders $\boldsymbol{\beta}_m^k$ are concatenated with $\boldsymbol{\beta}_m^k$ reversed if k is odd.

We now verify that the two properties the order must satisfy holds. Clearly the sets are ordered in nondecreasing size. We have already established the requirement about symmetric differences within each order of sets of a given size. We next consider the pairs of ending sets and starting sets. The first 3 such pairs are $(\emptyset,\{1\})$, $(\{m\},\{1,m\})$, and $(\{m-2,m-1\},\{m-2,m-1,m\})$. In each case the symmetric difference is exactly 1. The general case follows by recalling that each order starts with the lexicographically smallest set and ends with the lexicographically largest set, and every second order is reversed.

Example.

$$\begin{aligned}\boldsymbol{\beta}_4 = (\emptyset,\\ &\{1\},\{2\},\{3\},\{4\}\\ &\{4,1\},\{4,2\},\{4,3\},\{1,3\},\{1,2\},\{2,3\}\\ &\{2,3,4\},\{1,3,4\},\{1,2,4\},\{1,2,3\}\\ &\{1,2,3,4\})\end{aligned}$$

5.3 Sign pattern of the inverse of A

In this section we let $\alpha_1, \ldots, \alpha_n$ denote the particular ordering defined in Section 5.2, and we consider the construction of the matrix A from Section 5.1 with respect to this ordering, together with the corresponding matrices L and Q, sets A_i, and families of subsets F_i.

Definition 30. For a subset $\alpha \subseteq \{1, \ldots, m\}$ we let $\mathrm{num}(\alpha)$ denote the unique $j \in \{1, \ldots, n\}$ such that $\alpha = \alpha_j$. Define also $i_k = \min\{j : |\alpha_j| = k\}$, for all k.

We remark that i_k does not depend on the particular order we consider, but is fully defined by the conditions of Section 5.1.

We prove that the matrix A, for $m \geq 2$ has an inverse that weakly obeys a block checkerboard sign pattern. Namely we show that A^{-1} weakly obeys a sign pattern Σ of the following kind.

$$\Sigma = \left[\begin{array}{c|c} -1 & 1 \\ \hline 1 & -1 \\ \hline -1 & 1 \\ \hline 1 & -1 \\ \hline \vdots & \vdots \end{array}\right],$$

The n rows are divided into $m+1$ blocks. Block k corresponds to the subsets of size $k-1$, for $k = 1, \ldots, m+1$. That is, block k consists of the rows i for which $|\alpha_i| = k-1$. The columns are divided into precisely two blocks. For $m \geq 6$, we in fact prove that the first block of columns is of size $2m-1$. Thus $\Sigma = \sigma^{(1)}(\sigma^{(2)})^{\mathsf{T}}$, where

$$\sigma_i^{(1)} = (-1)^{|\alpha_i|} \quad \text{and} \quad \sigma^{(2)} = (\overbrace{-1, \ldots, -1}^{2m-1}, \overbrace{1, \ldots, 1}^{n-2m+1}) \ .$$

One may verify by hand that the matrices for $m = 4$ and $m = 5$ also weakly obeys this sign pattern. The matrices for $m = 2$ and $m = 3$ do not weakly obey this sign pattern, but the similar sign pattern where the columns are divided into two blocks of equal size. The matrix for $m = 1$ does not weakly obey a block checkerboard sign pattern.

We prove this for the first column as a special case in Subsection 5.4. The remaining columns we handle as follows. By Lemma 29 we have $A = LQ$. Column j of A^{-1} is then the solution of the linear system

$$LQx = e_j \tag{5}$$

Defining $z = Qx$, we have the equivalent system

$$Lz = e_j \tag{6}$$

In Subsections 5.5, 5.6, and 5.7 we prove that for $j > 1$ we have $|z_n| > \sum_{i=1}^{n-1} |z_i|$. Then, since $x = \frac{1}{n} Qz$, by the definition of Q we have

$$\operatorname{sgn}(x_i) = (-1)^{|\alpha_i|} \operatorname{sgn}(z_n)$$

Furthermore, we prove that $z_n > 0$ for $i < 2m$, and $z_n < 0$ for $i \geq 2m$, thus establishing the claimed sign pattern. Note that this latter part is not necessary to claim that A^{-1} weakly obeys a block checkerboard sign pattern. But proving this allows us to argue that the construction is fully-explicit.

We will several times use the following simple facts.

Lemma 31. *Let $w_1 \geq 1$ and $w_{i+1} \geq (2+\epsilon) w_i$, for $i \geq 1$ and $\epsilon > 0$. Let $c > 0$. Then*

$$w_s > \sum_{\ell=1}^{s-1} w_\ell + c w_1 \ ,$$

for $s \geq \log_{2+\epsilon}(c/\epsilon) + 2$.

Proof. Clearly $w_i > \sum_{\ell=1}^{i-1} w_\ell$ for all i, and thus

$$w_s \geq (2+\epsilon) w_{s-1} > w_{s-1} + \sum_{\ell=1}^{s-2} w_\ell + \epsilon w_{s-1} \geq \sum_{\ell=1}^{s-1} w_\ell + \epsilon (2+\epsilon)^{s-2} w_1 \geq \sum_{\ell=1}^{s-1} w_\ell + c w_1 \ .$$

□

Lemma 32. *Let z be the solution of $Lz = e_i$, and let $s \geq i_3$ be such that $|z_s| > \sum_{\ell=1}^{s-1} |z_\ell|$. Then $|z_n| > \sum_{\ell=1}^{n-1} |z_\ell|$, and $\operatorname{sgn}(z_n) = \operatorname{sgn}(z_s)$.*

Proof. The proof is a simple induction argument. Let $j > s$, and $k = |\alpha_j|$. If $\ell < i$ then $z_\ell = 0$. Hence our assumption implies that $s \geq i$, and thus $j > i$. We then have the equation

$$\frac{1}{2^{k-1}} \left((1 - 2^{k-1}) z_{j-1} + \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}\}} z_{\alpha_\ell} \right) = 0 \ ,$$

which means

$$z_j = (2^{k-1} - 1) z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_{\alpha_\ell} \ .$$

We treat the case of $z_s > 0$; the case of $z_s < 0$ is analogous. By induction we can estimate

$$z_j \geq (2^{k-1} - 1) z_{j-1} - \sum_{\ell=1}^{j-2} |z_\ell| \geq (2^{k-1} - 2) z_{j-1} > \sum_{\ell=1}^{j-1} |z_\ell| \ ,$$

since $k \geq 3$. □

Below we state as an example the matrices L (with zero entries omitted), A, and A^{-1}, for $m = 4$.

$$L = \left[\begin{array}{c|cccc|cccccc|cccc|c}
1 & & & & & & & & & & & & & & & \\
\hline
 & 1 & & & & & & & & & & & & & & \\
 & & 1 & & & & & & & & & & & & & \\
 & & & 1 & & & & & & & & & & & & \\
 & & & & 1 & & & & & & & & & & & \\
\hline
1/2 & 1/2 & & & -1/2 & 1/2 & & & & & & & & & & \\
 & 1/2 & 1/2 & & & -1/2 & 1/2 & & & & & & & & & \\
 & & 1/2 & 1/2 & & & -1/2 & 1/2 & & & & & & & & \\
 & 1/2 & & & 1/2 & & & -1/2 & 1/2 & & & & & & & \\
 & & 1/2 & 1/2 & & & & & -1/2 & 1/2 & & & & & & \\
 & 1/2 & & 1/2 & & & & & & -1/2 & 1/2 & & & & & \\
\hline
1/4 & & 1/4 & 1/4 & 1/4 & & 1/4 & 1/4 & & & -3/4 & 1/4 & & & & \\
 & 1/4 & 1/4 & & & 1/4 & 1/4 & & 1/4 & & 1/4 & -3/4 & 1/4 & & & \\
 & & 1/4 & 1/4 & & & 1/4 & 1/4 & 1/4 & 1/4 & & & -3/4 & 1/4 & & \\
 & & & 1/4 & 1/4 & 1/4 & 1/4 & & 1/4 & & 1/4 & & & -3/4 & 1/4 & \\
\hline
1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & -7/8 & 1/8
\end{array}\right]$$

$$A = \left[\begin{array}{c|cccc|cccccc|cccc|c}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\
1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\
\hline
1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \\
1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 \\
1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 \\
1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\
\hline
1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 \\
1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
\hline
1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1
\end{array}\right]$$

$$A^{-1} = \frac{1}{2}\left[\begin{array}{c|cccccc||cccc|ccccc}
-2 & -95 & -117 & -195 & -108 & -50 & -30 & 50 & 110 & 136 & 142 & 108 & 36 & 12 & 4 & 1 \\
\hline
1 & 91 & 114 & 192 & 108 & 52 & 32 & -48 & -108 & -134 & -140 & -107 & -36 & -12 & -4 & -1 \\
1 & 90 & 112 & 189 & 106 & 51 & 31 & -47 & -106 & -132 & -138 & -105 & -35 & -12 & -4 & -1 \\
1 & 85 & 106 & 178 & 100 & 48 & 29 & -45 & -100 & -124 & -130 & -99 & -33 & -11 & -4 & -1 \\
1 & 69 & 86 & 145 & 81 & 39 & 24 & -36 & -81 & -101 & -106 & -81 & -27 & -9 & -3 & -1 \\
\hline
0 & -67 & -84 & -143 & -81 & -40 & -25 & 35 & 80 & 100 & 105 & 80 & 27 & 9 & 3 & 1 \\
0 & -65 & -82 & -139 & -79 & -39 & -24 & 34 & 78 & 97 & 102 & 78 & 26 & 9 & 3 & 1 \\
0 & -60 & -76 & -129 & -73 & -36 & -22 & 32 & 72 & 90 & 94 & 72 & 24 & 8 & 3 & 1 \\
0 & -82 & -103 & -175 & -99 & -49 & -30 & 43 & 98 & 122 & 128 & 98 & 33 & 11 & 4 & 1 \\
0 & -87 & -110 & -186 & -105 & -52 & -32 & 46 & 104 & 130 & 136 & 104 & 35 & 12 & 4 & 1 \\
0 & -80 & -101 & -172 & -97 & -48 & -30 & 42 & 96 & 120 & 126 & 96 & 32 & 11 & 4 & 1 \\
\hline
0 & 57 & 72 & 123 & 70 & 35 & 22 & -30 & -69 & -86 & -90 & -69 & -23 & -8 & -3 & -1 \\
0 & 59 & 75 & 127 & 72 & 36 & 22 & -31 & -71 & -89 & -93 & -71 & -24 & -8 & -3 & -1 \\
0 & 64 & 81 & 138 & 78 & 39 & 24 & -34 & -77 & -96 & -101 & -77 & -26 & -9 & -3 & -1 \\
0 & 78 & 99 & 169 & 96 & 48 & 30 & -41 & -94 & -118 & -124 & -95 & -32 & -11 & -4 & -1 \\
\hline
0 & -57 & -72 & -122 & -69 & -34 & -21 & 30 & 68 & 85 & 89 & 68 & 23 & 8 & 3 & 1
\end{array}\right]$$

5.4 First column

We first solve the equation $Lz = e_1$. By induction we show that $z_j = 1 - k$, whenever $|\alpha_j| = k$. For the base case, clearly $z_1 = 1$, since the first row of L is e_1^T. Next for the induction step we treat the cases of $k = 1$ and $k \geq 2$ separately.

Let $|\alpha_j| = 1$. When $j = i_1 = 2$, we have $A_2 = \{1\}$, and hence $F_2 = \{\alpha_1, \alpha_2\}$. When $j > i_1 = 2$, we have $A_2 = \{j-2, j-1\}$, and hence $F_j = \{\alpha_{j-1}, \alpha_j\}$. In both cases we actually have $F_j \setminus \{\alpha_{j-1}, \alpha_j\} = \emptyset$. Thus

$$z_j = (2^{1-1} - 1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_\ell = 0 \ .$$

Let $|\alpha_j| = k \geq 2$. Consider first $j = i_k$. We have $|A_j| = |\alpha_j| = k$, and F_j contains $\binom{k}{s}$ sets of size s. By the induction hypothesis we have

$$\begin{aligned}
z_j &= (2^{k-1} - 1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_\ell \\
&= (2^{k-1} - 1)(1 - (k-1)) - \sum_{s=0}^{k-2} \binom{k}{s}(1-s) - (k-1)(1-(k-1)) \\
&= (2^{k-1} - 1)(2-k) - \sum_{s=0}^{k} \binom{k}{s}(1-s) + (2-k) + (1-k) \\
&= (2^{k-1} - 1)(2-k) - (2-k)2^{k-1} + (2-k) + (1-k) = 1-k
\end{aligned}$$

Consider next $j > i_k$. We have $|A_j| = k+1$, and F_j contains $2\binom{k-1}{s-1}$ sets of size $s > 0$. In particular F_j contains only the sets α_j and α_{j-1} of size k. Again by the induction

hypothesis we have

$$\begin{aligned} z_j &= (2^{k-1}-1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1},\alpha_j\}} z_\ell \\ &= (2^{k-1}-1)(1-k) - \sum_{s=0}^{k-2} 2\binom{k-1}{s}(1-(s+1)) \\ &= (2^{k-1}-1)(1-k) + 2\sum_{s=0}^{k-1}\binom{k-1}{s}s - 2(k-1) \\ &= (2^{k-1}-1)(1-k) + 2(k-1)2^{k-2} - 2(k-1) = 1-k \end{aligned}$$

5.4.1 Sign pattern

Next we compute $x = \frac{1}{n}Qz$. We need the following well-known identity ([7, Equation 5.42])

Lemma 33. *Let $P(a) = c_0 + c_1 a + \cdots + c_k a^k$ be a polynomial, $k \geq 0$. Then*

$$\sum_{a=0}^{k}\binom{k}{a}(-1)^a P(a) = (-1)^k k!\, c_k \ .$$

□

Consider given j, and let $k = |\alpha_j|$. Then

$$x_j = \frac{1}{n}\sum_{\ell=0}^{n}(-1)^{|\alpha_j \cap \alpha_\ell|} z_\ell = \frac{1}{n}\sum_{\ell=0}^{n}(-1)^{|\alpha_j \cap \alpha_\ell|}(1-|\alpha_\ell|) \tag{7}$$

For given ℓ, let $a = |\alpha_j \cap \alpha_\ell|$ and $b = |\alpha_\ell| - a$. Then we may collect the terms in Equation (7) according to a and b and obtain

$$x_j = 2^{-m}\sum_{a=0}^{k}\binom{k}{a}(-1)^a \sum_{b=0}^{m-k}\binom{m-k}{b}(1-a-b)$$

We first evaluate the innermost summation

$$\begin{aligned} &\sum_{b=0}^{m-k}\binom{m-k}{b}(1-a-b) = (1-a)\sum_{b=0}^{m-k}\binom{m-k}{b} - \sum_{b=0}^{m-k}\binom{m-k}{b}b \\ &= (1-a)2^{m-k} - (m-k)2^{m-k-1} = 2^{m-k}(1-(m-k)/2-a) \end{aligned}$$

Thus,

$$x_j = 2^{-k}\sum_{a=0}^{k}\binom{k}{a}(-1)^a(1-(m-k)/2-a)$$

By Lemma 33 we have $x_1 = 1 - m/2$, since $|\alpha_1| = 0$, $x_j = 1/2$, for $k = |\alpha_j| = 1$, and $x_j = 0$ when $k = |\alpha_j| > 1$.

5.5 Second block

Here we consider the equation $Lz = e_i$, for $|\alpha_i| = 1$. The last column of the block is handled separately. In this and the following sections we shall use the notation z_α as a shorthand for the entry $z_{\mathrm{num}(\alpha)}$.

5.5.1 First $m-1$ columns

We have $z_1 = 0$, since the first row of L is e_1^T. Next, $z_j = 0$ for $j \in \{2, \dots, m+1\} \setminus \{i\}$, and $z_i = 1$, since row j of L is e_j^T for all $j \in \{2, \dots, m+1\}$.

For $j = i_2$ we have the equation

$$\frac{1}{2}(z_\emptyset + z_{\{1\}} + (1-2)z_{\{m\}} + z_{i_2}) = 0 \ ,$$

since $A_{i_2} = \{1, m\}$, and $F_{i_2} = \{\emptyset, \{1\}, \{m\}, \{1, m\}\}$. We assume here that $i < m+1 = \mathrm{num}(\{m\})$, and thus have $z_{i_2} = -z_{\{1\}}$. Thus $z_{i_2} = 0$ when $i > 2$, and $z_{i_2} = -1$ when $i = 2$. For $i_2 < j < i_3$ we have $|\alpha_j \triangle \alpha_{j-1}| = 2$. Let $\alpha_j \triangle \alpha_{j-1} = \{a, b\}$. Then $F_j = \{\{a\}, \{b\}, \alpha_{j-1}, \alpha_j\}$ and we have the equation

$$\frac{1}{2}(z_{\{a\}} + z_{\{b\}} + (1-2)z_{j-1} + z_j) = 0 \ .$$

Hence $z_j = z_{j-1} - 1$ if $i \in \alpha_j \triangle \alpha_{j-1}$, and $z_j = z_{j-1}$ otherwise.

Note that any element of $\{2, \dots, m-2\}$ appears 2 times in the symmetric differences $\alpha_{j-1} \triangle \alpha_j$ for $i_2 < j \le \mathrm{num}(\{m-1, m\})$, whereas the elements 1 and $m-1$ appear only 1 time. Hence it follows that $z_{\{m,m-2\}}$ and $z_{\{m,m-1\}}$ are both at least -2.

Note that when $m \ge 6$, any element of $\{2, \dots, m-1\}$ appears at least 4 times in the symmetric differences $\alpha_{j-1} \triangle \alpha_j$ for $i_2 < j < i_3$, whereas the element 1 appear exactly 3 times. In both cases this implies by the above that $z_{i_3-1} \le -4$.

Consider now $j = i_3$. Then $A_j = \{m-2, m-1, m\}$ and $F_j = \big\{\emptyset, \{m-2\}, \{m-1\}, \{m\}, \{m-2, m-1\}, \{m-2, m\}, \{m-1, m\}, \{m-2, m-1, m\}\big\}$, and we thus have the equation

$$\frac{1}{4}(z_\emptyset + z_{\{m-2\}} + z_{\{m-1\}} + z_{\{m\}} + z_{\{m-2,m\}} + z_{\{m-1,m\}} + (1-4)z_{i_3-1} + z_{i_3}) = 0 \ ,$$

and hence $z_{i_3} = 3z_{i_3-1} - z_{\{m,m-2\}} - z_{\{m,m-1\}}$, when $i \notin \{m-2, m-1\}$, and $z_{i_3} = 3z_{i_3-1} - z_{\{m,m-2\}} - z_{\{m,m-1\}} - 1$, otherwise. In both cases $z_{i_3} \le 3z_{i_3-1} - z_{\{m,m-2\}} - z_{\{m,m-1\}}$. We already have the estimates $z_{i_3-1} \le -4$, and $z_{\{m,m-2\}}, z_{\{m,m-1\}} \ge -2$. Hence $z_{\{m,m-2\}} + z_{\{m,m-1\}} \ge -4 \ge z_{i_3-1}$, and it follows

$$z_{i_3} \le 2z_{i_3-1} \le -8 \ .$$

Next consider $j = i_3 + 1$. Then $A_j = \{m-3, m-2, m-1, m\}$, but the set F_j depends on whether m is even or odd. Write $A_j = \{m-3, m-1, a, b\}$, where $\alpha_{i_3} \triangle \alpha_{i_3+1} = \{m-3, a\}$. Then $F_j = \big\{\{m-3\}, \{a\}, \{m-3, m-1\}, \{m-3, b\}, \{a, m-$

$1\}, \{m-2, m\}, \{m-3, m-1, b\}, \{m-2, m-1, m\}\}$. Let $\beta_1, \ldots, \beta_4$ denote the sets of size 2 of F_j. We then have the equation

$$\frac{1}{4}(z_{\{m-3\}} + z_{\{a\}} + z_{\beta_1} + z_{\beta_2} + z_{\beta_3} + z_{\beta_4} + (1-4)z_{i_3} + z_{i_3+1}) = 0$$

and hence $z_{i_3+1} = 3z_{i_3} - \sum_{s=1}^{4} z_{\beta_s}$, when $i \notin \{m-3, a\}$, and $z_{i_3+1} = 3z_{i_3} - \sum_{s=1}^{4} z_{\beta_s} - 1$, otherwise. In both cases $z_{i_3+1} \leq 3z_{i_3} - \sum_{s=1}^{4} z_{\beta_s}$. As seen $z_{\{m-2,m\}} \geq -2 \geq \frac{1}{2} z_{i_3-1}$, and for the other β_s we have $z_{\beta_s} \geq z_{i_3-1}$. Thus

$$z_{i_3+1} \leq 3z_{i_3} - (3 + \frac{1}{2})z_{i_3-1} \leq 3z_{i_3} - (3 + \frac{1}{2})\frac{1}{2}z_{i_3} = \frac{5}{4}z_{i_3} \ .$$

For the remaining $j = i_3+\ell+1$, $\ell > 0$, for which $|\alpha_j| = 3$ we have the similar inequality

$$z_{i_3+\ell+1} \leq 3z_{i_3+\ell} - \sum_{s=1}^{4} z_{\beta_s} \leq 3z_{i_3+\ell} - 4z_{i_3-1} \leq 3z_{i_3+\ell} - 2z_{i_3} \leq 3z_{i_3+\ell} - \frac{8}{5}z_{i_3+1} \ ,$$

for appropriate sets $\beta_1, \ldots, \beta_4$ of size 2.

Claim 34. *For $j = i_3 + \ell + 1$, $\ell > 0$, for which $|\alpha_j| = 3$ we have*

$$z_{i_3+\ell+1} \leq \frac{3^\ell + 4}{3^{\ell-1} + 4} z_{i_3+\ell}$$

Proof. The proof is by induction on ℓ. For $\ell = 1$ we have

$$z_{i_3+2} \leq 3z_{i_3+1} - \frac{8}{5}z_{i_3+1}z_{i_3+1} = \frac{3+4}{1+4}z_{i_3+1} \ .$$

Next, for the induction step

$$\begin{aligned} z_{i_3+\ell+1} &\leq 3z_{i_3+\ell} - \frac{8}{5}z_{i_3+1} \leq 3z_{i_3+\ell} - \frac{8}{5}\left(\prod_{s=1}^{\ell-1} \frac{3^{s-1}+4}{3^s+4}\right) z_{i_3+\ell} \\ &= (3 - \frac{8}{5}\frac{5}{3^{\ell-1}+4})z_{i_3+\ell} = \frac{3^\ell+4}{3^{\ell-1}+4}z_{i_3+\ell} \ . \end{aligned}$$

□

We next find s such that

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell| \ .$$

We estimate the first $i_3 + 3$ terms separately. We have at most $1 + m(m-1)/2 + 4$ nonzero terms, each of absolute value less than $|z_{i_3+4}|$. That is

$$\sum_{\ell=1}^{i_3+3} |z_\ell| < m^2 |z_{i_3+4}| \ ,$$

using $m \geq 6$.

Using Claim 34, observing that $\frac{3^\ell+4}{3^{\ell-1}+4}$ is increasing with ℓ, and $\frac{3^4+4}{3^{4-1}+4} \geq \frac{5}{2}$, we can apply Lemma 31 with $w_i = |z_{i_3+3+i}|$, $c = m^2$, $\epsilon = \frac{1}{2}$ to obtain

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell|$$

for $s = \log_{5/2}(2m^2) + 5$. Note that $i_3 + s < i_4$, since there are $\binom{m}{3}$ sets of size 3, and $m \geq 6$. Also by Claim 34 we have that $z_{i_3+s} < 0$. By Lemma 32 we thus have

$$z_n < -\sum_{\ell=1}^{n-1} |z_\ell| \ .$$

5.5.2 Last column

Here we consider the last column of the second block, corresponding to solving the equation $Lz = e_{i_2-1}$. As above we have $z_1 = 0$, and $z_j = 0$ for $j \in \{2, \ldots, m\}$, and $z_{i_2-1} = 1$. For $j = i_2$ we have the equation

$$\frac{1}{2}(z_\emptyset + z_{\{1\}} + (1-2)z_{\{m\}} + z_{i_2}) = 0 \ ,$$

since $A_{i_2} = \{1, m\}$, and $F_{i_2} = \{\emptyset, \{1\}, \{m\}, \{1, m\}\}$. It follows that $z_{i_2} = 1$. Also as above for $i_2 < j < i_3$, $z_j = z_{j-1}$ if $m \in \alpha_j \triangle \alpha_{j-1}$, and $z_j = z_{j-1}$ otherwise. All sets of size 2 containing the element m comes before all other sets of size 2 in the order, and hence the case of $m \in \alpha_j \triangle \alpha_{j-1}$ occurs only when $j = \mathrm{num}(\{m, m-1\}) + 1 = 2m+1$. Thus we have $z_j = 1$ when $i_2 < j \leq 2m$ and $z_j = 0$ when $2m < j < i_3$.

For $j = i_3$, we have the equation

$$\frac{1}{4}(z_\emptyset + z_{\{m-2\}} + z_{\{m-1\}} + z_{\{m\}} + z_{\{m-2,m\}} + z_{\{m-1,m\}} + (1-4)z_{i_3-1} + z_{i_3}) = 0 \ ,$$

since again $A_j = \{m-2, m-1, m\}$ and $F_j = \big\{\emptyset, \{m-2\}, \{m-1\}, \{m\}, \{m-2, m-1\}, \{m-2, m\}, \{m-1, m\}, \{m-2, m-1, m\}\big\}$. From this we see that $z_{i_3} = -3$.

For $i_3 < j < i_4$ we have

$$z_j = (2^{3-1} - 1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_\ell \leq 3z_{j-1} \ ,$$

and also $z_j < 0$. Again we find s such that

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell| \ .$$

Note that $\sum_{\ell=1}^{i_3-1} |z_\ell| = 1 + (m-1) = m$. We can then apply Lemma 31 with $w_i = |z_{i_3+i-1}|$, $c = m$, $\epsilon = 1$ to obtain

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell|$$

for $s = \log_3(m) + 2$, noting that $i_3 + s < i_4$. We also have $z_{i_3+s} < 0$. By Lemma 32 we then have

$$z_n < -\sum_{\ell=1}^{n-1} |z_\ell| \ .$$

5.6 Third block

Here we consider the equation $Lz = e_i$, for $|\alpha_i| = 2$. We have $z_j = 0$ for $j < i$, $z_i = 1$, and hence $z_j = (2^{2-1} - 1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_\ell = 1$, for $i < j < i_3$. We remark for future use that $\sum_{\ell=1}^{i_3-1} |z_\ell| \leq \binom{m}{2}$.

Consider now $j = i_3$. As usual we have the equation

$$\frac{1}{4}(z_\emptyset + z_{\{m-2\}} + z_{\{m-1\}} + z_{\{m\}} + z_{\{m-2,m\}} + z_{\{m-1,m\}} + (1-4)z_{i_3-1} + z_{i_3}) = 0 \ ,$$

and hence

$$z_{i_3} = 3z_{i_3-1} - z_{\{m-2,m\}} - z_{\{m-1,m\}} = 3 - z_{\{m-2,m\}} - z_{\{m-1,m\}}.$$

Next consider $j = i_3 + 1$. As above $A_j = \{m-3, m-2, m-1, m\}$, where the set F_j depends on whether m is even or odd. Write again $A_j = \{m-3, m-1, a, b\}$, where $\alpha_{i_3} \triangle \alpha_{i_3+1} = \{m-3, a\}$. Then $F_j = \big\{\{m-3\}, \{a\}, \{m-3, m-1\}, \{m-3, b\}, \{a, m-1\}, \{m-2, m\}, \{m-3, m-1, b\}, \{m-2, m-1, m\}\big\}$. Let $\beta_1, \ldots, \beta_4$ denote the sets of size 2 of F_j. We then have the equation

$$\frac{1}{4}(z_{\{m-3\}} + z_{\{a\}} + z_{\beta_1} + z_{\beta_2} + z_{\beta_3} + z_{\beta_4} + (1-4)z_{i_3} + z_{i_3+1}) = 0 \ ,$$

and hence $z_{i_3+1} = 3z_{i_3} - \sum_{s=1}^{4} z_{\beta_s}$. We consider below 3 cases depending on the relationship between i and $\mathrm{num}(\{m-2, m\}) = 2m - 1$.

- $i < \mathrm{num}(\{m-2, m\})$: Here $z_{i_3} = 1$. Also in our ordering, for $s \in \{1, \ldots, 4\}$ we have $\mathrm{num}(\beta_s) \geq \mathrm{num}(\{m-3, m\}) \geq i$. It follows that $z_{i_3+1} = -1$. For $i_3 + 1 < j < i_4$, we have $z_j = (2^{3-1} - 1)z_{j-1} - \sum_{\alpha_\ell \in F_j \setminus \{\alpha_{j-1}, \alpha_j\}} z_\ell \leq 3z_{j-1}$. We can thus apply Lemma 31 with $w_i = |z_{i_3+i}|$, $c = m^2/2$, $\epsilon = 1$ to obtain

  $$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell|$$

  for $s = \log_3(m^2/2) + 2$, noting that $i_3 + s < i_4$. We also have $z_{i_3+s} < 0$. By Lemma 32 we then have

  $$z_n < -\sum_{\ell=1}^{n-1} |z_\ell| \ .$$

- $i = \mathrm{num}(\{m-2, m\})$: Here $z_{i_3} = 1$ as well. In case $a = m$ we have $z_{i_3+1} = -1$. Also

$$z_{i_3+2} \leq (2^{3-1} - 1) z_{i_3+1} - \sum_{\alpha_\ell \in F_{i_3+2} \setminus \{\alpha_{i_3+1}, \alpha_{i_3+2}\}} z_\ell \leq 3 z_{i_3+1} = -3 \ .$$

However in case $a = m-2$ we have $z_{i_3+1} = 0$. Thus we consider $j = i_3+2$ as well. We have here that $\alpha_j = \{m-3, m-2, m\}$ and thus $\alpha_{i_3+1} \triangle \alpha_{i_3+2} = \{m-1, m\}$. Then

$$z_{i_3+2} = -z_{\{m-3,m-1\}} - z_{\{m-2,m-1\}} - z_{\{m-3,m\}} - z_{\{m-2,m\}} = -3$$

For $i_3 + 2 < j < i_4$, we have as before $z_j \leq 3 z_{j-1}$, and can thus apply Lemma 31 with $w_i = |z_{i_3+i+1}|$, $c = m^2/2$, $\epsilon = 1$ to obtain

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell|$$

for $s = \log_3(m^2/2) + 3$, noting that $i_3 + s < i_4$. We also have $z_{i_3+s} < 0$. By Lemma 32 we then have

$$z_n < -\sum_{\ell=1}^{n-1} |z_\ell| \ .$$

- $i > \mathrm{num}(\{m-2, m\})$: Here $z_{i_3} \geq 2$. Since $\mathrm{num}(\{m-2, m\}) < \mathrm{num}(\{m-1, m\}) \leq i$ we have $z_{i_3+1} = 3 z_{i_3} - \sum_{s=1}^4 z_{\beta_s} \geq 3 z_{i_3} - 3 \geq \frac{3}{2} z_{i_3}$. Let $\beta_1^{(2)}, \ldots, \beta_4^{(2)}$ denote the sets of size 2 of F_{i_3+2}. Then $z_{i_3+2} = 3 z_{i_3+1} - \sum_{s=1}^4 z_{\beta_s^{(2)}} \geq \frac{5}{3} z_{i_3+1} + 2 z_{i_3} - 4 \geq \frac{5}{3} z_{i_3+1}$. Again, let $\beta_1^{(3)}, \ldots, \beta_4^{(3)}$ denote the sets of size 2 of F_{i_3+3}. Then $z_{i_3+3} = 3 z_{i_3+2} - \sum_{s=1}^4 z_{\beta_s^{(3)}} \geq \frac{11}{5} z_{i_3+2} + \frac{4}{5} \cdot \frac{5}{3} \cdot \frac{3}{2} z_{i_3} - 4 \geq \frac{11}{5} z_{i_3+2}$. By induction it is now easy to derive $z_j \geq \frac{11}{5} z_{j-1}$ for $i_3 + 3 < j < i_4$. We can thus apply Lemma 31 with $w_i = z_{i_3+i+1}$, $c = m^2/2$, $\epsilon = \frac{1}{5}$ to obtain

$$|z_{i_3+s}| > \sum_{\ell=1}^{i_3+s-1} |z_\ell|$$

for $s = \log_{\frac{11}{5}}(m^2/2) + 3$, noting that $i_3 + s < i_4$. We also have $z_{i_3+s} > 0$. By Lemma 32 we then have

$$z_n > \sum_{\ell=1}^{n-1} |z_\ell| \ .$$

5.7 Remaining columns

Here we consider the equation $Lz = e_i$, for $|\alpha_i| = k \geq 3$. We then have that $z_j = 0$ for $j < i$, $z_i = 2^{k-1} \geq 4$. Clearly $z_i > \sum_{\ell=1}^{i-1} |z_\ell| = 0$. By Lemma 32 we then have

$$z_n > \sum_{\ell=1}^{n-1} |z_\ell| \ .$$

5.8 Explicitness

We discuss here in more detail the explicitness of our construction. We say that a family $\{A_n\}$ of matrices, where A_n is a $n \times n$ matrix, is *explicit*, if there is an algorithm that given as input n computes the matrix A_n in time polynomial in n. We say that the family is *fully-explicit*, if there is an algorithm that given as input i, j, and n, computes entry (i, j) of A_n in time polynomial in $\log(n)$. Clearly the latter definition is more restrictive than the former.

We next give a sketch of an argument that the matrices just constructed are fully-explicit. Let A be the $2^m \times 2^m$ matrix of Section 5.1 with the order of Section 5.2, and let B be the $2^m \times 2^m$ matrix obtained by the Hadamard product of A with the transpose of the sign pattern Σ of Section 5.3. In order to compute entry (i, j) of B we compute separately the entry (i, j) of A and the entry (j, i) of Σ.

To compute entry (j, i) of Σ we need to determine which block of rows that row j belongs to and to check whether $i > 2m - 1$. The former is determined by finding k such that

$$\sum_{\ell=0}^{k} \binom{m}{\ell} < j \leq \sum_{\ell=0}^{k+1} \binom{m}{\ell} \ .$$

This is easily done in time polynomial in m.

To compute entry (i, j) of A it is sufficient to observe that the following task can be computed in time polynomial in m: Given index i, compute the set of the order, α_i. To do this, first compute the k such that $|\alpha_i| = k$. This is the same task as just considered, and identifies the order β_m^k. Depending on k we may need to consider the reverse of this. By appropriately adjusting i, we may just consider consider finding the set i' of the order β_m^k (for $k = 2$ we also need to *shift* the set afterwards). This can be done by first identifying the smallest element a of the set $\alpha_{i'}$ and recursing. Specifically, a can be determined by the inequalities

$$\sum_{\ell=1}^{a-1} \binom{m-\ell}{k-1} < i' \leq \sum_{\ell=1}^{a} \binom{m-\ell}{k-1} \ .$$

Then i' is adjusted by subtracting the sum $\sum_{\ell=1}^{a-1} \binom{m-\ell}{k-1}$, and continuing with β_{m-a}^{k-1}, possibly adjusting i' again if the order is reversed, finding the next-smallest element and so on.

References

[1] N. Alon and V. H. Vũ. Anti-Hadamard matrices, coin weighing, threshold gates, and indecomposable hypergraphs. *Journal of Combinatorial Theory, Series A*, 79(1):133–160, 1997.

[2] L. Babai, K. A. Hansen, V. V. Podolskii, and X. Sun. Weights of exact threshold functions. In *Proceedings of the 35th International Symposium on Mathematical Foundations of Computer Science, MFCS 2010*, volume 6281 of *Lecture Notes in Computer Science*, pages 66–77. Springer, 2010.

[3] L. Ching. The maximum determinant of an $n \times n$ lower Hessenberg $(0, 1)$ matrix. *Linear Algebra and its Applications*, 183:147–153, 1993.

[4] H. Everett. Recursive games. In *Contributions to the Theory of Games Vol. III*, volume 39 of *Ann. Math. Studies*, pages 67–78. Princeton University Press, 1957.

[5] D. K. Faddeev and I. S. Sominskii. *Problems in Higher Algebra.* W. H. Freeman, 1965. (translated by J. L. Brenner).

[6] T. S. Ferguson. Game theory. `http://www.math.ucla.edu/~tom/Game_Theory/Contents.html`.

[7] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994.

[8] R. L. Graham and N. J. A. Sloane. Anti-Hadamard matrices. *Linear Algebra and its Applications*, 62:113–137, 1984.

[9] K. A. Hansen, R. Ibsen-Jensen, and P. B. Miltersen. The complexity of solving reachability games using value and strategy iteration. In *6th Int. Comp. Sci. Symp. in Russia, CSR*, LNCS. Springer, 2011.

[10] K. A. Hansen, M. Koucký, N. Lauritzen, P. B. Miltersen, and E. P. Tsigaridas. Exact algorithms for solving stochastic games: extended abstract. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011*, pages 205–214. ACM, 2011.

[11] K. A. Hansen, M. Koucký, and P. B. Miltersen. Winning concurrent reachability games requires doubly exponential patience. In *Proc. of IEEE Symp. on Logic in Comp. Sci., LICS*, pages 332–341, 2009.

[12] J. Håstad. On the size of weights for threshold gates. *SIAM Journal on Discrete Mathematics*, 7(3):484–492, 1994.

[13] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *Journal of Computer and System Sciences*, 37(1):79–100, 1988.

[14] M. W. Krentel. Structure in locally optimal solutions (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, FOCS'89*, pages 216–221. IEEE Computer Society, 1989.

[15] R. J. Lipton and N. E. Young. Simple strategies for large zero-sum games with applications to complexity theory. In *Proceedings of the 26th Annual Symposium on the Theory of Computing*, pages 734–740. ACM Press, 1994.

[16] S. T. McCormick, M. Rao, and G. Rinaldi. Easy and difficult objective functions for max cut. *Mathematical Programming*, 94:459–466, 2003.

[17] V. V. Podolskii. Perceptrons of large weight. *Problems of Information Transmission*, 45(1):46–53, 2009.

[18] R. M. Roth and K. Viswanathan. On the hardness of decoding the Gale-Berlekamp code. *IEEE Transactions on Information Theory*, 54(3):1050–1060, 2008.

[19] A. A. Schäffer and M. Yannakakis. Simple local search problems that are hard to solve. *SIAM Journal on Computing*, 20(1):56–87, 1991.

[20] L. S. Shapley and R. N. Snow. Basic solutions of discrete games. In *Contributions to the Theory of Games*, number 24 in Annals of Mathematics Studies, pages 27–35. Princeton University Press, 1950.

[21] N. J. A. Sloane. Unsolved problems related to the covering radius of codes. In T. M. Cover and B. Gopinath, editors, *Open problems in communication and computation*. Springer-Verlag, 1987.

[22] J. Spencer. *Ten Lectures on the Probabilistic Method*. SIAM, 1987.

[23] J. von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100:295–320, 1928.