Combinatorial Games

**Game 1** Start with \( n \) chips. Players A,B alternately take 1,2,3,4 chips until there are none left. The winner is the person who takes the last chip.

Example

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>A</th>
<th>B</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n = 10 )</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>( n = 11 )</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

B wins

B wins

What is the optimal strategy for this game?

**Game 2** Chip placed at point \((m,n)\). Players can move chip to \((m',n)\) or \((m,n')\) where \(0 \leq m' < m\) and \(0 \leq n' < n\). The player who makes the last move and puts the chip onto \((0,0)\) wins.

What is the optimal strategy for this game?

**Game 3** \( W \) is a set of words. A and B alternately remove words \( w_1,w_2,\ldots \) from \( W \). The rule is that the first letter of \( w_{i+1} \) must be the same as the last letter of \( w_i \). The player who makes the last legal move wins.

1 Abstraction

Represent each position by a vertex of a digraph \( D = (X,A) \). \((x,y)\) is an arc of \( D \) iff one can move from position \( x \) to position \( y \).

We assume that the digraph is finite and that it is **acyclic** i.e. there are no directed cycles.

The game starts with a chip on vertex \( x_0 \) say, and players alternately move the chip to \( x_1,x_2,\ldots \), where \( x_{i+1} \in \Gamma^+(x_i) \), the set of out-neighbours of \( x_i \). The game ends when the chip is on a **sink** i.e. a vertex of out-degree zero. The last player to move is the winner.

Example 1: \( D = (\{0,1,\ldots ,n\},A) \) where \((x,y) \in A \) iff \( x - y \in \{1,2,3,4\} \).

Example 2: \( D = (\{0,1,\ldots ,m\} \times \{0,1,\ldots ,n\},A) \) where \((x,y) \in \Gamma^+((x',y')) \) iff \( x = x' \) and \( y > y' \) or \( x > x' \) and \( y = y' \).

Example 3: \( D = (\{W,w\} : W' \subseteq W \setminus \{w\},A) \). \( w \) is the last word used and \( W' \) is the remaining set of unused words. \((A',w') \in \Gamma^+((A,w)) \) iff \( w' \in A \) and \( w' \) begins with the last letter of \( w \). Also, there is an arc from \((W,\cdot)\) to \((W \setminus \{w\},w)\) for all \( w \), corresponding to the games start.

We will first argue that such a game must eventually end. A **topological numbering** of digraph \( D \) is a map \( f : X \rightarrow [N], N = |X| \) which satisfies \((x,y) \in A\) implies \( f(x) < f(y) \).

**Theorem 1.** A finite digraph \( D = (X,A) \) is acyclic iff it admits at least one topological numbering.

**Proof** Suppose first that \( D \) has a topological numbering. We show that it is acyclic. Suppose that \( C = (x_1,x_2,\ldots ,x_k,x_1) \) is a directed cycle. Then \( f(x_1) < f(x_2) < \cdots < f(x_k) < f(x_1) \), contradiction.

Suppose now that \( D \) is acyclic. We first argue that \( D \) has at least one sink. Thus let \( P = (x_1,x_2,\ldots ,x_k) \) be a directed simple path in \( D \). We claim that \( x_k \) is a sink. If \( D \) contains an arc \((x_k,y)\) then either \( y = x_i, 1 \leq i \leq k-1 \) and this means that \( D \) contains the cycle \( x_i,x_{i+1},\ldots ,x_k,x_i \), contradiction or \( y \notin \{x_1,x_2,\ldots ,x_k\} \) and then \((P,y)\) is a longer simple path than \( P \), contradiction.
We can now prove by induction on $N$ that there is at least one topological numbering. If $N = 1$ and $X = \{x\}$ then $f(x) = 1$ defines a topological numbering.

Now assume that $N > 1$. Let $z$ be a sink of $D$ and define $f(z) = N$. The digraph $D' = D - z$ is acyclic and by the induction hypothesis it admits a topological numbering, $f : X \setminus \{z\} \to [N - 1]$. The function we have defined on $X$ is a topological numbering. If $(x, y) \in A$ then either $x, y \neq z$ and then $f(x) < f(y)$ by our assumption on $f$, or $y = z$ and then $f(x) < N = f(z)$ ($x \neq z$ because $z$ is a sink).

The fact that $D$ has a topological numbering implies that the game must end. Each move increases the $f$ value of the current position by at least one and so after at most $N$ moves a sink must be reached.

The positions of a game are partitioned into 2 sets:

- **P-positions**: The next player cannot win. The previous player can win regardless of the current player’s strategy.
- **N-positions**: The next player has a strategy for winning the game.

The main problem is to determine $N$ and $P$ and what the strategy is for winning from an N-position.

For $x \in X$ let $\Gamma^+(x) = \{y \in X : (x, y) \in A\}$ be the set of out-neighbours of $x$.

**Labelling procedure**

1. Label all sinks with $P$.
2. Label with $N$, every position $x$ for which there exists $y \in \Gamma^+(x)$ which is labelled with $P$.
3. Label with $P$, every position $x$ for which $\Gamma^+(x)$ is labelled with $N$.

A position $x$ is an N-position (winning) iff there is a move from $x$ to a losing position for the next player.

The labelling should be carried out in reverse topological order.

Thus there is a unique partition of $X$ into $N, P$ which satisfies the following:

**P1** All sinks are in $P$.

**P2** If $x \in N$ then $\Gamma^+(x) \cap P \neq \emptyset$.

**P3** If $x \in P$ then $\Gamma^+(x) \subseteq N$.

In Game 1, $P = \{5k : k \geq 0\}$ and in Game 2, $P = \{(x, x) : x \geq 0\}$.

**Sprague-Grundy Numbering**

For $S \subseteq \{0, 1, 2, \ldots\}$ let

$$\text{mex}(S) = \min \{x \geq 0 : x \notin S\}.$$

Now given an acyclic digraph $D = X, A$ define $g$ recursively by

$$g(x) = \begin{cases} 0 & \text{if } x \text{ is a sink} \\ \text{mex}(\Gamma^+(x)) & \text{otherwise} \end{cases}$$

$g(x)$ can be computed in reverse topological order.
Lemma 1.

\[ x \in P \iff g(x) = 0. \]

Proof. Clearly P1 holds. We check P2 and P3.

P2: If \( g(x) > 0 \) there must be a \( y \in \Gamma^+(x) \) with \( g(y) = 0 \).

P3: If \( g(x) = 0 \) there cannot be a \( y \in \Gamma^+(x) \) with \( g(y) = 0 \).

Sums of games

Suppose that we have \( n \) games with digraphs \( D_i = (X_i, A_i) \), \( i = 1, 2, \ldots, n \). The sum of these games is played as follows. A position is a vector \( (x_1, x_2, \ldots, x_n) \in A_1 \times A_2 \times \cdots \times A_n \). To make a move, a player chooses \( i \) such that \( x_i \) is not a sink of \( D_i \) and then replaces \( x_i \) by \( y \in \Gamma_i^+(x_i) \). The game ends when each \( x_i \) is a sink of \( D_i \) for \( i = 1, 2, \ldots, n \).

Example Nim

In a one pile game, we start with \( a \geq 0 \) chips and while there is a positive number \( x \) of chips, a move consists of deleting \( y \leq x \) chips. In this game the N-positions are positive integers and the unique P-position is 0. The Sprague-Grundy numbering is defined by \( g(x) = x \).

In general, Nim consists of the sum of \( n \) single pile games starting with \( a_1, a_2, \ldots, a_n > 0 \). A move consists of deleting some chips from a non-empty pile.

We now show how to compute the Sprague-Grundy numbering for a sum of games.

For binary integers \( a = a_m a_{m-1} \cdots a_1 a_0 \) and \( b = b_m b_{m-1} \cdots b_1 b_0 \) we define \( a \oplus b = c_m c_{m-1} \cdots c_1 c_0 \) by \( c_i = 1 \) if \( a_i \neq b_i \) and \( c_i = 0 \) if \( a_i = b_i \) for \( i = 1, 2, \ldots, m \).

So for example \( 11 \oplus 5 = 14 \).

Theorem 2. If \( g_i \) is the Sprague-Grundy function for game \( i = 1, 2, \ldots, n \) then the Sprague-Grundy function \( g \) for the sum of the games is defined by

\[ g(x) = g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n) \]

where \( x = (x_1, x_2, \ldots, x_n) \).

Proof. It is enough to show

1. If \( x \in X \) is a sink of \( D \) then \( g(x) = 0 \).

2. If \( x \in X \) and \( g(x) = b > a \geq 0 \) then there exists \( x' \in \Gamma^+(x) \) such that \( g(x') = a \).

3. If \( x \in X \) and \( g(x) = b \) and \( x' \in \Gamma^+(x) \) then \( g(x') \neq g(x) \).

1. If \( x = (x_1, x_2, \ldots, x_n) \) is a sink then \( x_i \) is a sink of \( D_i \) for \( i = 1, 2, \ldots, n \). So

\[ g(x) = g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n) \]

\[ = 0 \oplus 0 \oplus \cdots \oplus 0 \]

\[ = 0. \]

2. Write \( d = a \oplus b \). Then

\[ a = d \oplus b \]

\[ = d \oplus g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n). \]

(1)
Now suppose that we can show there exists $i$ such that $d \oplus g_i(x_i) < g_i(x_i)$. Then since $g_i(x_i) = \text{mex}(\Gamma_i^+(x_i))$ there must exist $x_i' \in \Gamma_i^+(x_i)$ such that $g_i(x_i') = d \oplus g_i(x_i)$. Assume without loss of generality that $i = 1$. Then from (1) we have

$$a = g_1(x_1') \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n) = g(x_1', x_2, \ldots, x_n).$$

Furthermore, $(x_1', x_2, \ldots, x_n) \in \Gamma^+(x)$ and so we will have verified 2.

Let us prove that such an $i$ exists. Suppose that $2^{k-1} \leq d < 2^k$. Then $d$ has a 1 in position $k$ and no higher. Since $d_k = a_k \oplus b_k$ and $a < b$ we must have $a_k = 0$ and $b_k = 1$. Thus there is at least one $i$ such that $g_i(x_i)$ has a 1 in position $k$. But then $d \oplus g_i(x_i) < g_i(x_i)$ since $d$ “destroys” the $k$th bit of $g_i(x_i)$ and does not change any higher bit.

3. Suppose without loss of generality that $g(x_1', x_2, \ldots, x_n) = g(x_1, x_2, \ldots, x_n)$ where $x_1' \in \Gamma^+(x_1)$. Then $g_1(x_1') \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n) = g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_n(x_n)$ implies that $g_1(x_1') = g_1(x_1)$, contradiction.

If we apply this theorem to the game of Nim then if the position $x$ consists of piles of $x_i$ chips for $i = 1, 2, \ldots, n$ then $g(x) = x_1 \oplus x_2 \oplus \cdots \oplus x_n$.

Sums of other subtraction games:

In our first example, $g(x) = x \mod 5$ and so for the sum of $n$ such games we have

$$g(x_1, x_2, \ldots, x_n) = (x_1 \mod 5) \oplus (x_2 \mod 5) \oplus \cdots \oplus (x_n \mod 5).$$

Another subtraction game.

One pile:

- A player can remove any even number of chips, but not the whole pile.
- A player can remove the whole pile if it is odd.

The terminal positions are 0 or 2.

**Lemma 2.** $g(0) = 0$, $g(2k) = k-1$ and $g(2k-1) = k$ for $k \geq 1$.

**Proof** 0, 2 are terminal positions and so $g(0) = g(2) = 0$. $g(1) = 1$ because the only position one can move to from 1 is 0. We prove remainder by induction on $k$.

Assume that $k > 1$.

$$g(2k) = \text{mex}\{g(2k-2), g(2k-4), \ldots, g(2)\} = \text{mex}\{k-2, k-3, \ldots, 0\} = k-1.$$

$$g(2k-1) = \text{mex}\{g(2k-3), g(2k-5), \ldots, g(1), g(0)\} = \text{mex}\{k-1, k-2, \ldots, 0\} = k.$$

A more complicated one pile game

Start with $n$ chips. First player can remove up to $n-1$ chips.

In general, if the previous player took $x$ chips, then the next player can take $y \leq x$ chips.

Thus a game’s position can be represented by $(n, x)$ where $n$ is the current size of the pile and $x$ is the maximum number of chips that can be removed in this round.
Theorem 3. Suppose that the position is \((n, x)\) where \(n = m2^k\) and \(m\) is odd. Then,

(a) This is an \(N\)-position if \(x \geq 2^k\).

(b) This is a \(P\)-position if \(m = 1\) and \(x < n\).

Proof For a non-negative integer \(n = m2^k\), let \(\langle n \rangle\) denote the number of bits in the binary expansion of \(n\) and let \(k = \rho(n)\) determine the position of the right-most one in this expansion. We claim that the following strategy is a win for the player in a position described in (a): Remove \(y = 2^k\) chips. Suppose this player is \(A\).

If \(m = 1\) then \(x \geq n\) and \(A\) wins. Otherwise, after such a move the position is \((n', y)\) where \(\rho(n') > \rho(n)\). Note first that \(\langle n' \rangle = \langle n \rangle - 1 > 0\). Thus \(B\) cannot win at this point. Second, \(B\) cannot remove more than \(2^k\) chips and so if \(B\) moves the position to \((n'', x'')\) then \(\langle n'' \rangle \geq \langle n' \rangle\) and furthermore, \(x'' \geq 2^{\rho(n'')}\), since \(x''\) must have a \(1\) in position \(\rho(n'')\). Thus, by induction, \(A\) is in an \(N\)-position and wins the game.

To prove (b), note that after the first move, the position satisfies the conditions of (a). \(\square\)

Let us next consider a generalisation of this game. There are 2 players \(A\) and \(B\) and \(A\) goes first. We have a non-decreasing function \(f\) from \(\mathbb{N} \to \mathbb{N}\) where \(\mathbb{N} = \{1, 2, \ldots\}\) which satisfies \(f(x) \geq x\).

At the first move \(A\) takes any number less than \(h\) from the pile, where \(h\) is the size of the initial pile. Then on a subsequent move, if a player takes \(x\) chips then the next player is constrained to take at most \(f(x)\) chips. Thus the above considered the cases \(f(x) = x\).

There is a set \(\mathcal{H} = \{H_1 = 1 < H_2 < \ldots\}\) of initial pile sizes for which the first player will lose, assuming that the second player plays optimally. Also, if the initial pile size \(h \notin \mathcal{H}\) then the first player has a winning strategy. It will turn out that the sequence satisfies the recurrence:

$$H_{j+1} = H_j + H_\ell \text{ where } H_\ell = \min_{i \leq j} \{H_i \mid f(H_i) \geq H_j\}, \quad \text{for } j \geq 0. \quad (2)$$

Note that

$$H_{j+1} \leq 2H_j. \quad (3)$$

[The reader should check that if \(f(x) = x\) then \(H_1 = 2^x\). Another case to check is \(f(x) = 2x\). This gives \(\mathcal{H} = \{1, 2, 3, 5, 8, \ldots\}\) i.e. the Fibonacci sequence.]

The key to the game is the following result.

**Theorem 4.** Every positive integer \(n\) can be uniquely written as the sum

$$n = H_{j_1} + H_{j_2} + \cdots + H_{j_p} \tag{4}$$

where \(f(H_{j_i}) < H_{j_{i+1}}, \text{ for } 1 \leq i < p\).

**Proof** We prove this by induction on \(n\). If \(n = 1\) then \(n = H_1\) is the unique decomposition.

**Existence**
Assume that any \(n < H_k\) can be represented as a sum of distinct \(H_{j_i}\)'s with \(f(H_{j_i}) < H_{j_{i+1}}\), and suppose that \(H_k \leq n < H_{k+1}\). Inequality (3) implies that \(n - H_k < H_k\).

It follows by induction that

$$n - H_k = H_{j_1} + \cdots + H_{j_p}, \quad (5)$$
where $f(H_{j_i}) < H_{j_i+1}$ for $i = 1, 2, ..., p - 1$. To establish existence we need only show that $f(H_{j_p}) < H_k$. Assume to the contrary that $f(H_{j_p}) \geq H_k$. But then for some $m \leq j_p$ we have

$$H_{k+1} = H_k + H_m \leq H_k + H_{j_p} \leq n,$$

contradicting the choice of $n$.

**Uniqueness**

We will first prove by induction on $p$ that if $f(H_{j_i}) < H_{j_i+1}$ for $1 \leq i < p$ then

$$H_{j_1} + H_{j_2} + \cdots + H_{j_p} < H_{j_p+1}. \quad (6)$$

If $p = 2$ then we are saying that if $f(H_{j_1}) < H_{j_2}$ then $H_{j_1} + H_{j_2} < H_{j_2+1}$. But this follows directly from $H_{j_2+1} = H_{j_2} + H_m$ where $f(H_m) \geq H_{j_2}$ i.e. $H_m > H_{j_1}$.

So assume that (6) is true for $p \geq 2$. Now

$$H_{j_{p+1}+1} = H_{j_{p+1}} + H_m \text{ and } f(H_{j_p}) < H_{j_{p+1}};$$

implies that $m \geq j_p + 1$.

Thus

$$H_{j_p+1} \geq H_{j_{p+1}} + H_{j_{p+1}} > H_{j_{p+1}} + H_{j_p} + H_{j_{p-1}} + \cdots + H_{j_1},$$

after applying induction to get the second inequality.

This completes the induction for (6).

Now assume by induction on $k$ that $n < H_k$ has a unique decomposition (4). This is true for $k = 2$ and so now assume that $k \geq 2$ and $H_k \leq n < H_{k+1}$. Consider a decomposition

$$n = H_{j_1} + H_{j_2} + \cdots + H_{j_p}.$$

It follows from (6) that $j_p = k$. Indeed, $j_p \leq k$ since $n < H_{k+1}$ and if $j_p < k$ then $H_{j_1} + H_{j_2} + \cdots + H_{j_p} < H_{j_p+1} \leq H_k$, contradicting our choice of $n$. So $H_k$ appears in every decomposition of $n$.

Now (3) and $n < H_{k+1}$ implies $n - H_k < H_k$ and so, by induction, $n - H_k$ has a unique decomposition. But then if $n$ had two distinct decompositions, $H_k$ would appear in each, implying that $n - H_k$ also had two distinct decompositions, contradiction. \qed

One simple consequence of the uniqueness of the decomposition is that

$$H_k \neq H_{j_1} + H_{j_2} + \cdots + H_{j_p} \quad (7)$$

for all $k$ and sequences $j_1, j_2, ..., j_p$ where $f(H_{j_i}) < H_{j_i+1}$ for $i = 1, 2, ..., p - 1$.

It follows from the above Lemma that the integers $n$ can be given unique “binary” representations by representing $n = H_{j_1} + H_{j_2} + \cdots + H_{j_p}$ by the 0-1 string with a 1 in positions $j_1, j_2, ..., j_p$ and 0 everywhere else. We call this the $H$-representation of $n$. This then leads to the following

**Theorem 5.** Suppose that the position is $(n, *)$. Then,

(a) This is an $N$-position if $n \notin \mathcal{H} = \{H_1, H_2, \ldots, \}$.

(b) This is a $P$-position if $n \in \mathcal{H}$.

6
Proof

(a) The winning strategy is to delete a number of chips equal to $H_{j_i}$ where $j_i$ is the index of the rightmost 1 in the $H$-representation of $n$.

All we have to do is verify that this strategy is possible. Note that if A deletes $H_{j_i}$ chips, then B cannot respond by deleting $H_{j_2}$ chips, because $H_{j_2} > f(H_{j_i})$ and so it is only A that can reduce the number of 1’s in the $H$-representation of $n$.

The thing to check is that if A starts in $(n, \ast)$ then A can always delete $H_{j_i}$ chips i.e. the positions $(m, x)$ that A will face satisfy $f(x) \geq H_{j_i}$ where $m = H_{j_1} + H_{j_2} + \cdots + H_{j_n}$. We do this by induction on the number of plays in the game so far. It is true in the first move and suppose it is true for $(m, x)$ and A removes $H_{j_i}$ and B removes $y$ where $y \leq \min \{m - H_{j_i}, f(H_{j_i})\} < H_{j_2}$. Now

$$m - H_{j_i} - y = H_{j_2} - y + H_{j_3} \cdots + H_{j_n}$$

$$= H_{j_1} + H_{j_2} \cdots + H_{j_n} + H_{j_i} + H_{j_2} \cdots + H_{j_n}$$

and we need to argue that $H_{j_i} \leq f(y)$. But if $f(y) < H_{j_i}$, then we have

$$H_{j_2} = y + H_{j_1} + H_{j_2} \cdots + H_{j_n}$$

$$= H_{j_1} + \cdots + H_{j_n} + H_{j_2} + \cdots + H_{j_n}$$

where $f(H_{j_i}) \leq f(y) < H_{j_i}$, contradicting (7). Thus A can remove $H_{j_i}$ in the next round, as required.

(b) Assume that $n = H_k$. After A removes $x$ chips we have

$$H_k - x = H_{j_1} + H_{j_2} + \cdots + H_{j_n}$$

chips left.

All we have to show is that B can now remove $H_{j_i}$ chips i.e. $H_{j_i} \leq f(x)$. But if this is not the case then we argue as above that $H_k = H_{a_1} + \cdots + H_{a_n}$, where $x = H_{a_1} + \cdots + H_{a_n}$ and $f(H_{j_i}) \leq f(x) < H_{j_i}$, contradicting (7). \hfill \Box

Geography

Start with a chip sitting on a vertex $v$ of a graph or digraph $G$.

A move consists of moving the chip to a neighbouring vertex. In edge geography, moving the chip from $x$ to $y$ deletes the edge $(x, y)$. In vertex geography, moving the chip from $x$ to $y$ deletes the vertex $x$.

The problem is given a position $(G, v)$, to determine whether this is a P or N position.

Complexity Both edge and vertex geography are Pspace-hard on digraphs. Edge geography is Pspace-hard on an undirected graph. Only vertex geography on a graph is polynomial time solvable.

2 Undirected Vertex Geography – UVG

Theorem 6. $(G, v)$ is an N-position in UVG iff every maximum matching of $G$ covers $v$.

Proof (i) Suppose that $M$ is a maximum matching of $G$ which covers $v$. Player 1’s strategy is now: Move along M-edge that contains current vertex.
If Player 1 were to lose, then there would exist a sequence of edges $e_1, f_1, \ldots, e_k, f_k$ such that $v \in e_1, e_1, e_2, \ldots, e_k \in M$, $f_1, f_2, \ldots, f_k \notin M$ and $f_k = (x, y)$ where $y$ is the current vertex for Player 1 and $y$ is not covered by $M$. But then if $A = \{e_1, e_2, \ldots, e_k\}$ and $B = \{f_1, f_2, \ldots, f_k\}$ then $(M \setminus A) \cup B$ is a maximum matching (same size as $M$) which does not cover $v$, contradiction.

(ii) Suppose now that there is some maximum matching $M$ which does not cover $v$. Then if $(v, w)$ is Player 1’s move, $w$ must be covered by $M$, else $M$ is not a maximum matching. Player 2’s strategy is now: Move along $M$-edge that contains current vertex. If Player 2 were to lose then there exists $e_1 = (v, w), e_1, \ldots, e_k, e_k, e_{k+1} = (x, y)$ where $y$ is the current vertex for Player 2 and $y$ is not covered by $M$. But then we have defined an augmenting path from $v$ to $y$ and so $M$ is not a maximum matching, contradiction.

Note that we can determine whether or not $v$ is covered by all maximum matchings as follows: Find the size $\sigma$ of the maximum matching $G$. This can be done in $O(n^2)$ time on an $n$-vertex graph. Then find the size $\sigma'$ of a maximum matching in $G - v$. Then $v$ is covered by all maximum matchings of $G$ if $\sigma \neq \sigma'$.

3 Undirected Edge Geography – UEG on a bipartite graph

An even kernel of $G$ is a non-empty set $S \subseteq V$ such that (i) $S$ is an independent set and (ii) $v \notin S$ implies that $deg_S(v)$ is even, (possibly zero). $deg_S(v)$ is the number of neighbours of $v$ in $S$.

Lemma 3. If $S$ is an even kernel and $v \in S$ then $(G, v)$ is a P-position in UEG.

Proof Any move at a vertex in $S$ takes the chip outside $S$ and then Player 2 can immediately put the chip back in $S$. After a move from $x \in S$ to $y \notin S$, $deg_S(y)$ will become odd and so there is an edge back to $S$. making this move, makes $deg_S(y)$ even again. Eventually, there will be no $S : S$ edges and Player 1 will be stuck in $S$.

We now discuss Bipartite UEG i.e. we assume that $G$ is bipartite, $G$ has bipartition consisting of a copy of $[m]$ and a disjoint copy of $[n]$ and edges set $E$. Now consider the $m \times n$ 0-1 matrix $A$ with $A(i, j) = 1$ iff $(i, j) \in E$.

We can play our game on this matrix: We are either positioned at row $i$ or we are positioned at column $j$. If we are positioned at row $i$, then we choose a $j$ such that $A(i, j) = 1$ and (i) make $A(i, j) = 0$ and (ii) move the position to column $j$. An analogous move is taken when we positioned at column $j$.

Lemma 4. Suppose the current position is row $i$. This is a P-position iff row $i$ is in the span of the remaining rows (is the sum (mod 2) of a subset of the other rows) or row $i$ is a zero row. A similar statement can be made if the position is column $j$.

Proof If row $i$ is a zero row then vertex $i$ is isolated and this is clearly a P-position. Otherwise, assume the position is row 1 and there exists $I \subseteq [m]$ such that $1 \in I$ and

$$r_1 = \sum_{i \in I \setminus \{1\}} r_i \pmod{2} \quad \text{or} \quad \sum_{i \in I} r_i = 0 \pmod{2} \quad (8)$$

where $r_i$ denotes row $i$.

$I$ is an even kernel: If $x \notin I$ then either (i) $x$ corresponds to a row and there are no $x, I$ edges or (ii) $x$ corresponds to a column and then $\sum_{i \in I} A(i, x) = 0 \pmod{2}$ from (8) and then $x$ has an even number of neighbours in $I$. 

8
Now suppose that (8) does not hold for any \( I \). We show that there exists a \( \ell \) such that \( A(1,\ell) = 1 \) and putting \( A(1,\ell) = 0 \) makes column \( \ell \) dependent on the remaining columns. Then we will be in a P-position, by the first part.

Let \( e_1 \) be the \( n \)-vector with a 1 in row 1 and a 0 everywhere else. Let \( A^* \) be obtained by adding \( e_1 \) to \( A \) as an \((n+1)\)th column. Now the row-rank of \( A^* \) is the same as the row-rank of \( A \) (here we are doing all arithmetic modulo 2). Suppose not, then if \( r_i \) is the \( i \)th row of \( A^* \) then there exists a set \( J \) such that

\[
\sum_{i \in J} r_i \equiv 0 (\text{mod } 2) \neq \sum_{i \in J} r_i^* (\text{mod } 2).
\]

Now \( 1 \notin J \) because \( r_1 \) is independent of the remaining rows of \( A \), but then \( \sum_{i \in J} r_i = 0 (\text{mod } 2) \) implies \( \sum_{i \in J} r_i^* = 0 (\text{mod } 2) \) since the last column has all zeros, except in row 1.

Thus rank \( A^* \) = rank \( A \) and so there exists \( K \subseteq [n] \) such that

\[
e_1 = \sum_{k \in K} c_k (\text{mod } 2) \quad \text{or} \quad e_1 + \sum_{k \in K} c_k = 0 (\text{mod } 2) \quad (9)
\]

where \( c_k \) denotes column \( k \) of \( A \). Thus there exists \( \ell \in K \) such that \( A(1,\ell) = 1 \). Now let \( c_j^* = c_j \) for \( j \neq \ell \) and \( c_\ell^* \) be obtained from \( c_\ell \) by putting \( A(1,\ell) = 0 \) i.e. \( c_\ell^* = c_\ell + e_1 \). But then (9) implies that \( \sum_{k \in K} c_k^* = 0 (\text{mod } 2) \) (\( K = \{k\} \) is a possibility here).

\[\square\]

**Tic Tac Toe and extensions**

We consider the following multi-dimensional version of Tic Tac Toe (Noughts and Crosses to the English). The board consists of \( [n]^d \). A point on the board is therefore a vector \((x_1, x_2, \ldots, x_d)\) where \( 1 \leq x_i \leq n \) for \( 1 \leq i \leq d \).

A line is a set points \((x^{(1)}_j, x^{(2)}_j, \ldots, x^{(d)}_j)\), \( j = 1, 2, \ldots, n \) where each sequence \( x^{(i)} \) is either (i) of the form \( k, k, \ldots, k \) for some \( k \in [n] \) or is (ii) \( 1, 2, \ldots, n \) or is (iii) \( n, n - 1, \ldots, 1 \). Finally, we cannot have Case (i) for all \( i \).

Thus in the (familiar) \( 3 \times 3 \) case, the top row is defined by \( x^{(1)} = 1, 1, 1 \) and \( x^{(2)} = 1, 2, 3 \) and the diagonal from the bottom left to the top right is defined by \( x^{(1)} = 3, 2, 1 \) and \( x^{(2)} = 1, 2, 3 \)

**Lemma 5.** The number of winning lines in the \((n, d)\) game is \( \frac{(n+2)^d - n^d}{2} \).

**Proof** In the definition of a line there are \( n \) choices for \( k \) in (i) and then (ii), (iii) make it up to \( n + 2 \). There are \( d \) independent choices for each \( i \) making \((n + 2)^d\). Now delete \( n^d \) choices where only Case (i) is used. Then divide by 2 because replacing (ii) by (iii) and vice-versa whenever Case (i) does not hold produces the same set of points (traversing the line in the other direction).

The game is played by 2 players. The Red player (X player) goes first and colours a point red. Then the Blue player (0 player) colours a different point blue and so on. A player wins if there is a line, all of whose points are that players colour. If neither player wins then the game is a draw. The second player does not have a winning strategy:

**Lemma 6.** Player 1 can always get at least a draw.

**Proof** We prove this by considering strategy stealing. Suppose that Player 2 did have a winning strategy. Then Player 1 can make an arbitrary first move \( x_1 \). Player 2 will then move with \( y_1 \). Player 1 will now win playing the winning strategy for Player 2 against a first move of \( y_1 \). This can be carried out until the strategy calls for move \( x_1 \) (if at all). But then Player 1 can make an arbitrary move and continue, since \( x_1 \) has already been made.

\[\square\]
3.1 Pairing Strategy

\[
\begin{bmatrix}
11 & 1 & 8 & 1 & 12 \\
6 & 2 & 2 & 9 & 10 \\
3 & 7 & * & 9 & 3 \\
6 & 7 & 4 & 4 & 10 \\
12 & 5 & 8 & 5 & 11
\end{bmatrix}
\]

The above array gives a strategy for Player 2 the 5 × 5 game \((d = 2, n = 5)\). For each of the 12 lines there is an associated pair of positions. If Player 1 chooses a position with a number \(i\), then Player 2 responds by choosing the other cell with the number \(i\). This ensures that Player 1 cannot take line \(i\). If Player 1 chooses the * then Player 2 can choose any cell with an unused number. So, later in the game if Player 1 chooses a cell with \(j\) and Player 2 already has the other \(j\), then Player 1 can choose an arbitrary cell. Player 2's strategy is to ensure that after all cells have been chosen, he/she will have chosen one of the numbered cells associated with each line. This prevents Player 1 from taking a whole line. This is called a pairing strategy.

We now generalise the game to the following: We have a family \(\mathcal{F} = A_1, A_2, \ldots, A_N \subseteq A\). A move consists of one player, taking an uncoloured member of \(A\) and giving it his colour. A player wins if one of the sets \(A_i\) is completely coloured with his colour.

A pairing strategy is a collection of distinct elements \(X = \{x_1, x_2, \ldots, x_{2N-1}, x_{2N}\}\) such that \(x_{2i-1}, x_{2i} \in A_i\) for \(i \geq 1\). This is called a draw forcing pairing. Player 2 responds to Player 1's choice of \(x_{2i+\delta}, \delta = 0, 1\) by choosing \(x_{2i+3-\delta}\). If Player 1 does not choose from \(X\), then Player 2 can choose any uncoloured element of \(X\). In this way, Player 2 avoids defeat, because at the end of the game Player 2 will have coloured at least one of each of the pairs \(x_{2i-1}, x_{2i}\) and so Player 1 cannot have completely coloured \(A_i\) for \(i = 1, 2, \ldots, N\).

**Theorem 7.** If

\[
\left| \bigcup_{A \in \mathcal{G}} A \right| \geq 2|\mathcal{G}| \quad \forall \mathcal{G} \subseteq \mathcal{F}
\]

then there is a draw forcing pairing.

**Proof** We define a bipartite graph \(\Gamma\). \(A\) will be one side of the bipartition and \(B = \{b_1, b_2, \ldots, b_{2N}\}\). Here \(b_{2i-1}\) and \(b_{2i}\) both represent \(A_i\) in the sense that if \(a \in A_i\) then there is an edge \((a, b_{2i-1})\) and an edge \((a, b_{2i})\). A draw forcing pairing corresponds to a complete matching of \(B\) into \(A\) and the condition (10) implies that Hall's condition is satisfied. 

**Corollary 8.** If \(|A_i| \geq n\) for \(i = 1, 2, \ldots, n\) and every \(x \in A\) is contained in at most \(n/2\) sets of \(\mathcal{F}\) then there is a draw forcing pairing.

**Proof** The degree of \(a \in A\) is at most \(2(n/2)\) in \(\Gamma\) and the degree of each \(b \in B\) is at least \(n\). This implies (via Hall's condition) that there is a complete matching of \(B\) into \(A\).

Consider Tic tac Toe when case \(d = 2\). If \(n\) is even then every array element is in at most 3 lines (one row, one column and at most one diagonal) and if \(n\) is odd then every array element is in at most 4 lines (one row, one column and at most two diagonals). Thus there is a draw forcing pairing if \(n \geq 6\), \(n\) even and if \(n \geq 9\), \(n\) odd. (The cases \(n = 4, 7\) have been settled as draws. \(n = 7\) required the use of a computer to examine all possible strategies.

In general we have

**Lemma 7.** If \(n \geq 3^d - 1\) and \(n\) is odd or if \(n \geq 2^d - 1\) and \(n\) is even, then there is a draw forcing pairing of \((n, d)\) Tic tac Toe.  

10
Proof We only have to estimate the number of lines through a fixed point \( c = (c_1, c_2, \ldots, c_d) \). If \( n \) is odd then to choose a line \( L \) through \( c \) we specify, for each index \( i \), whether \( L \) is (i) constant on \( i \), (ii) increasing on \( i \) or (iii) decreasing on \( i \). This gives \( 3^d \) choices. Subtract 1 to avoid the all constant case and divide by 2 because each line gets counted twice this way.

When \( n \) is even, we observe that once we have chosen in which positions \( L \) is constant, \( L \) is determined. Suppose \( c_1 = x \) and 1 is not a fixed position. Then every other non-fixed position is \( x \) or \( n - x + 1 \). Assuming w.l.o.g. that \( x \leq n/2 \) we see that \( x < n - x = 1 \) and the positions with \( x \) increase together at the same time as the positions with \( n - x + 1 \) decrease together. Thus the number of lines through \( c \) in this case is bounded by \( \sum_{i=0}^{d-1} \binom{d}{i} = 2^d - 1 \). \( \square \)

3.2 Quasi-probabilistic method

We now prove a theorem of Erdős and Selfridge.

Theorem 9. If \( |A_i| \geq n \) for \( i \in [N] \) and \( N < 2^n - 1 \), then Player 2 can get a draw in the game defined by \( \mathcal{F} \).

Proof At any point in the game, let \( C_j \) denote the set of elements in \( A \) which have been coloured with Player \( j \)'s colour, \( j = 1, 2 \) and \( U = A \setminus C_1 \cup C_2 \). Let

\[
\Phi = \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|},
\]

Suppose that the players choices are \( x_1, y_1, x_2, y_2, \ldots \). Then we observe that immediately after Player 1's first move, \( \Phi < N2^{-(n-1)} < 1 \).

We will show that Player 2 can keep \( \Phi < 1 \) through out. Then at the end, when \( U = \emptyset \), \( \Phi = \sum_{i : A_i \cap C_2 = \emptyset} 1 \) implies that \( A_i \cap C_2 \neq \emptyset \) for all \( i \in [N] \).

So, now let \( \Phi_j \) be the value of \( \Phi \) after the choice of \( x_1, y_1, \ldots, x_j \). Then if \( U, C_1, C_2 \) are defined at precisely this time,

\[
\Phi_{j+1} - \Phi_j = - \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|} + \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|}
\]

\[
\leq - \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|} + \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|}
\]

We deduce that \( \Phi_{j+1} - \Phi_j \leq 0 \) if Player 2 chooses \( y_j \) to maximise over \( y \), \( \sum_{i : A_i \cap C_2 = \emptyset} 2^{-|A_i \cap U|} \).

In this way, Player 2 keeps \( \Phi < 1 \) and obtains a draw. \( \square \)

In the case of \((n, d)\) Tic Tac Toe, we see that Player 2 can force a draw if (see Lemma 5)

\[
\frac{(n+2)^d - n^d}{2} < 2^{n-1}
\]

which is implied, for \( n \) large, by

\[
n \geq (1 + \epsilon) d \log d
\]

where \( \epsilon > 0 \) is a small positive constant.
Shannon Switching Game Start with a connected multi-graph $G = (V, E)$.
Two players: Player A goes first and deletes edges and player B fortifies edges making them invulnerable to deletion by B. Player B wins iff the fortified edges contain a spanning tree of $G$.

Theorem 10. Player B wins iff $G$ contains two edge disjoint spanning trees.

Proof (a) Here we assume that $G$ has two edge disjoint spanning trees $T_1, T_2$. We prove this by induction on $|V|$. If $|V| = 2$ then $G$ must contain at least two parallel edges joining the two vertices and so B can win. Suppose next that $|V| > 2$. Suppose that A deletes an edge $e = (x, y)$ of $T_2$ red. This breaks $T_2$ into two sub-trees $T'_2, T''_2$. B will choose an edge $f = (u, v) \in T_1$ with one end in $V(T'_2)$ and the other end in $V(T''_2)$. Now contract the edge $f$. In the new graph $G'$, both $T_1$ and $T_2$ become spanning trees $T'_1$ and $T'_2$ and they are edge disjoint. It follows by induction that B can win the game on $G'$ and then wins the game on $G$ by uncontracting the edge $f$. Of course $f$ is chosen first of all still!

If $A$ chooses an edge $x$ in neither of the trees then $B$ can choose an arbitrary edge $f$ of $T_1$. Now let $e$ be any edge of the unique cycle contained in $T_2 + e$. $B$ can continue playing on $G - x$ as though $e$ was the deleted edge. We can contract $f$ as before and apply the above inductive argument.

(b) For this part we use a Theorem due to Nash-Williams:

Theorem 11. Let $k$ be a positive integer. Then $G$ contains $k$ edge disjoint spanning trees iff for every partition $P = (V_1, V_2, \ldots, V_t)$ of $V$ we have

$$e(P) = |E(P)| = \sum_{1 \leq i < j \leq t} e(V_i, V_j) \geq k(\ell - 1).$$

Here $E(P)$ is the set of edges joining different parts of the partition and $e(V_i, V_j)$ is the number of edges joining $V_i$ and $V_j$.

Let us apply Theorem 11 with $k = 2$. If $G$ does not contain two edge disjoint spanning trees, then it contains a partition $P = (V_1, V_2, \ldots, V_t)$ with $e(P) \leq 2\ell - 3$. Player A starts by deleting an edge $e \in E(P)$. B will fortify an edge $f = (u, v)$. If $u, v$ join different sets in the partition $P$ we can merge them and consider $P'$ which has one less part and satisfies $e(P') \leq e(P) - 2$ (edges $e, f$ have gone from the count). Otherwise B chooses an edge entirely inside a part of $P$ and the number of parts does not change, but $e(P)$ goes down by one. Eventually, we come to a point where one part is joined to the rest of the graph by a single edge ($2\ell - 3 = 1$ when $\ell = 2$) and $A$ wins by deleting this edge.

Sketch of proof of Theorem 11

If $P = (V_1, V_2, \ldots, V_t)$ is a partition and $T$ is a spanning tree then $T$ contains at least $\ell - 1$ edges of $E(P)$ and the only if part is straightforward.

Suppose now that (11) holds for all partitions. Let $F$ be the set of edge disjoint forests containing the maximum number of edges. If $F = (F_1, F_2, \ldots, F_k) \in F$ and $e \in E \setminus E[F]$ then every $F_i + e$ contains a cycle. If $e'$ belongs to this cycle then $F_i' \in F$ where $F_i' = F_j$ for $j \neq i$ and $F_i' = F_i + e'$. We say that $F_i'$ is obtained from $F_i$ by a replacement.

Consider now a fixed $F_0 = (F_1^0, F_2^0, \ldots, F_k^0) \in F$ and let $F^0$ be the set of $k$-tuples in $F$ that can be obtained from $F^0$ by a sequence of replacements. Then let

$$E^0 = \bigcup_{F \in F^0} (E \setminus E([F])).$$

Claim 1. For every $e^0 \in E \setminus E([F^0])$ there exists a set $U \subseteq V$ that contains the endpoints of $e^0$ and induces a connected tree in $F_i^0$ for $1 \leq i \leq k$. 

12
Assume the claim for the moment. Suppose that not every $F_i^0$ is a spanning tree. Then $G$ contains at least $k(|V| - 1)$ edges (from (11) applied to the partition of $V$ into singletons) and so there exists $e^0 \in E \setminus E[F^0]$. Shrink the vertices of the set $U$ in the claim to a single vertex $v_U$ to obtain a graph $G'$. Apply induction to $G'$ to get a set of $k$ disjoint spanning trees $T_1', T_2', \ldots, T_k'$ of $G'$. Now expand $v_U$ back to $U$. Each $T_i'$ expands to a spanning tree of $G$. In this way we get $k$ edge-disjoint spanning trees of $G$.

**Proof of Claim 1**

Let $G^0 = (V, E^0)$ and let $C_0$ be the component of $G^0$ that contains $e^0$. Let $U = V(C^0)$. First verify that if $F = (F_1, F_2, \ldots, F_k) \in \mathcal{F}_0$ and $F'$ is obtained from $F$ by a replacement and $x, y$ are the ends of a path in $F_i' \cap U$ then $x, y$ are joined by a path $xF_i'y \subseteq U$. (Exercise).

We now show that $F_i^0 \cap U$ is connected. Let $(x, y)$ be an edge of $C^0$. Since $C^0$ is connected, we only have to show that $F_i^0$ contains a path from $x$ to $y$, all of whose vertices belong to $U$. But this follows by using the exercise and backwards induction starting from some $F \in \mathcal{F}_0$ for which $F_i$ contains the edge $(x, y)$. 

\(\square\)