

Automatic proofs in combinatorial game theory

Antoine Renard

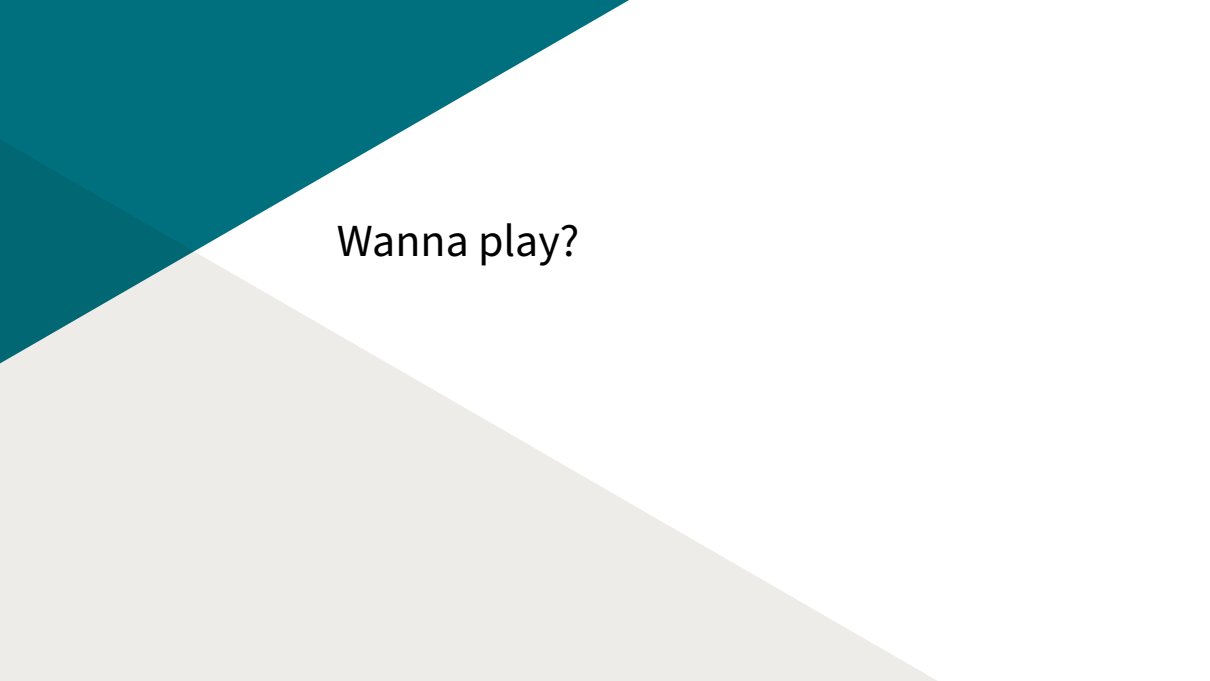
Joint work with Bastien Mignoty, Michel Rigo
& Markus A. Whiteland

Numeration 2025, Tsukuba, 8th September 2025

Overview



1. Wanna play?
2. Wythoff's game
3. Using Walnut for Wythoff's game
4. Some generalisations

The background is composed of three geometric sections: a dark teal triangle in the top-left corner, a light gray triangle in the bottom-left corner, and a white triangle in the top-right and center. The text "Wanna play?" is centered in the white section.

Wanna play?

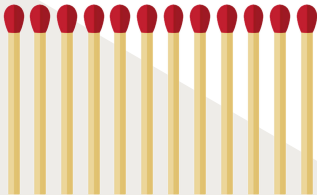
Wanna play?

Game of Nim



► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.





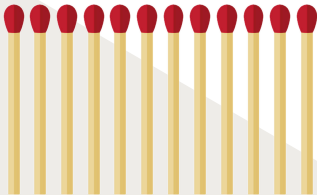
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)





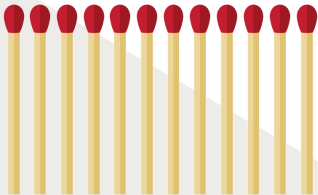
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



*"Never be so kind, you
forget to be clever"*

(T. Swift, 2020)



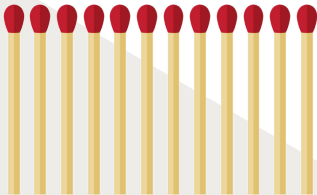
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



12 matches



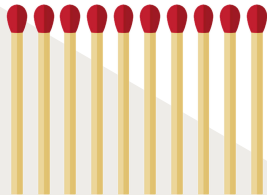
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



10 matches



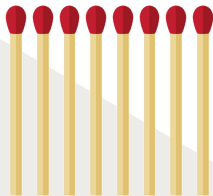
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



8 matches



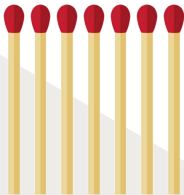
Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



7 matches



Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

► My goal: win the game (because who doesn't like winning?)



4 matches



Wanna play?

Game of Nim

► Rules:

- 2 players, taking turns, they may not pass;
- remove 1, 2 or 3 match(es);
- first player unable to play loses.

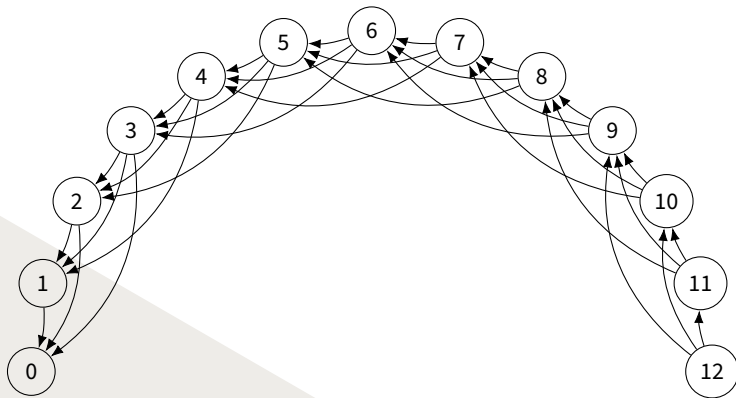
► My goal: win the game (because who doesn't like winning?)



1 match

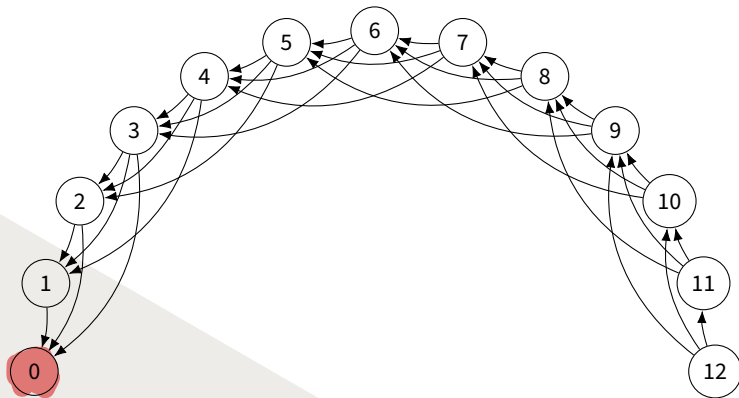
Wanna play?

Switching to graphs



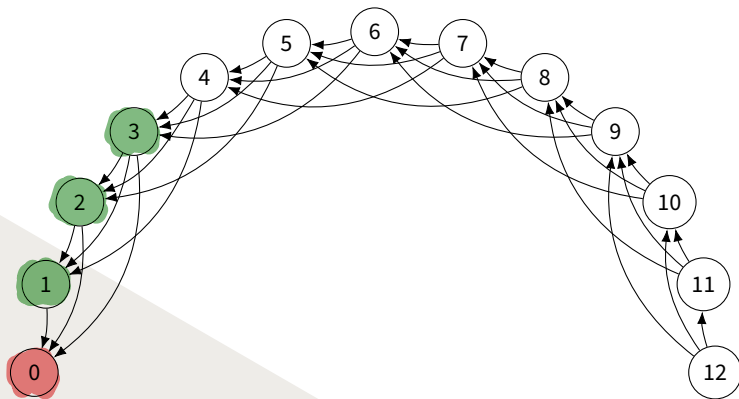
Wanna play?

Switching to graphs



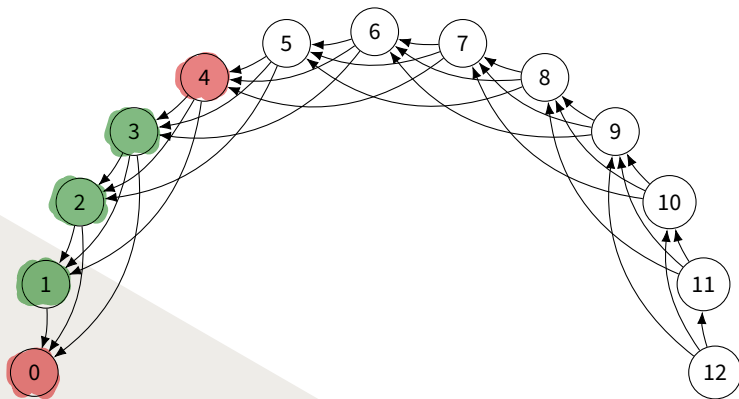
Wanna play?

Switching to graphs



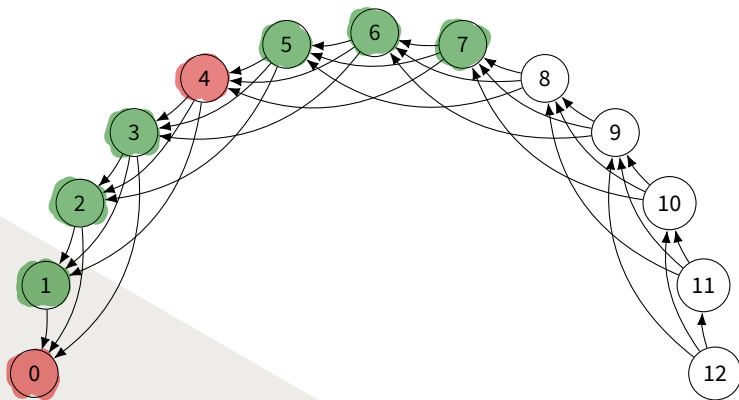
Wanna play?

Switching to graphs



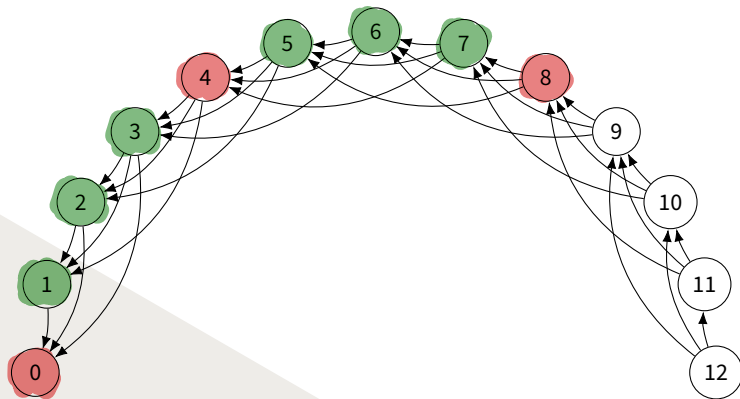
Wanna play?

Switching to graphs



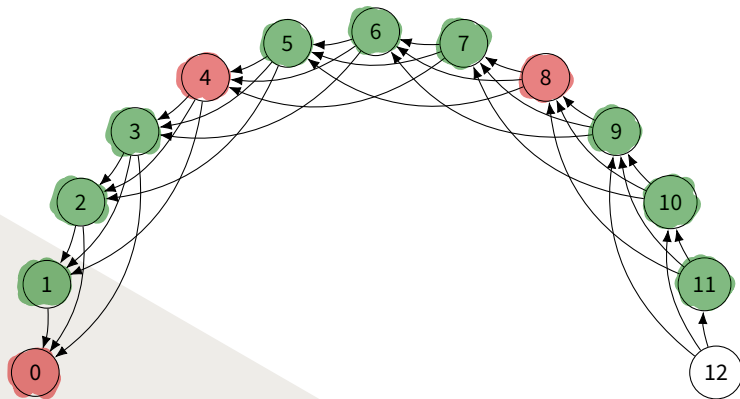
Wanna play?

Switching to graphs



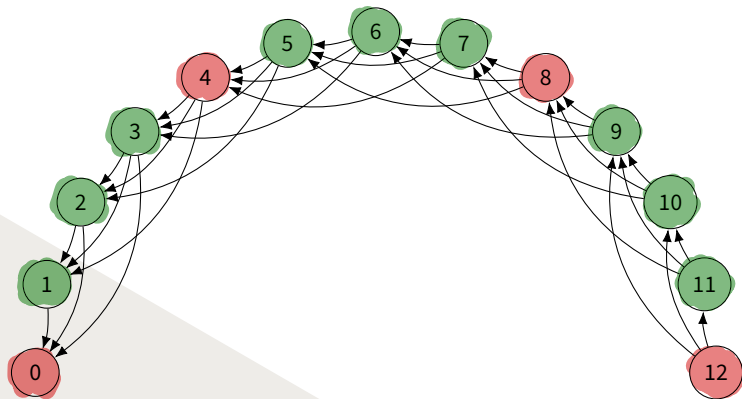
Wanna play?

Switching to graphs



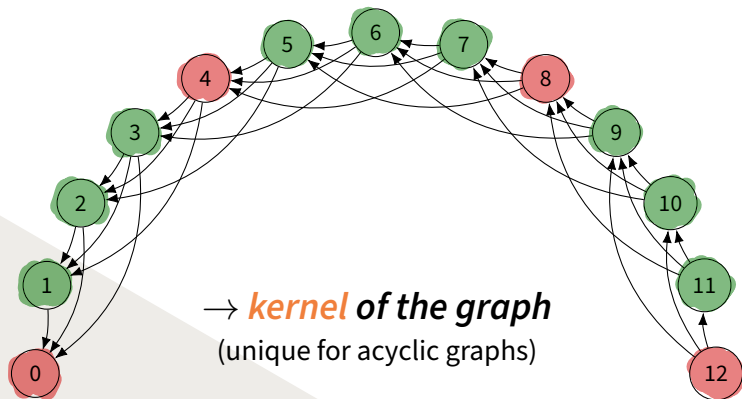
Wanna play?

Switching to graphs



Wanna play?

Switching to graphs





Wanna play?

\mathcal{P} - and \mathcal{N} -positions

A *position* of a game describes the actual configuration of the game before one of the two players makes a *move*.

For instance, in the previous game of Nim, the *position i* depicts the situation with i remaining sticks on the table.



Wanna play?

\mathcal{P} - and \mathcal{N} -positions

A **position** of a game describes the actual configuration of the game before one of the two players makes a **move**.

For instance, in the previous game of Nim, the **position i** depicts the situation with i remaining sticks on the table.

Two types of positions:

- ▶ **\mathcal{P} -position** = position of the game where the **previous** player is ensured to win; it is thus a **losing** position for the person about to play.
- ▶ **\mathcal{N} -position** = position of the game where the **next** player is ensured to win; it is thus a **winning** position for the person about to play.



Wanna play?

\mathcal{P} -positions and kernel

Proposition (folklore)

The sets of \mathcal{P} - and \mathcal{N} -positions of an impartial acyclic game are uniquely determined by the following two properties:

- ▶ Every move from a \mathcal{P} -position leads to an \mathcal{N} -position; equivalently there is no move between two \mathcal{P} -positions (*stability property* of $\mathcal{P}(G)$).
- ▶ From every \mathcal{N} -position, there exists a move leading to a \mathcal{P} -position (*absorbing property* of $\mathcal{P}(G)$).

Wanna play?



\mathcal{P} -positions and kernel

Proposition (folklore)

The sets of \mathcal{P} - and \mathcal{N} -positions of an impartial acyclic game are uniquely determined by the following two properties:

- ▶ Every move from a \mathcal{P} -position leads to an \mathcal{N} -position; equivalently there is no move between two \mathcal{P} -positions (*stability property* of $\mathcal{P}(G)$).
- ▶ From every \mathcal{N} -position, there exists a move leading to a \mathcal{P} -position (*absorbing property* of $\mathcal{P}(G)$).

→ *For an impartial acyclic game,
set of \mathcal{P} -position = kernel of the associated graph*

The background of the slide is composed of two large, overlapping geometric shapes. A teal-colored shape occupies the top-left corner, while a light gray shape occupies the bottom-left corner. The rest of the slide is white. The text 'Wythoff's game' is centered in the white area.

Wythoff's game

Wythoff's game

The rules

Wythoff's game is a 2-player impartial game which is a variation of the game of Nim.

- ▶ 2 piles of tokens,
- ▶ the players play one after another, they may not pass,
- ▶ the first player unable to play loses.

Allowed moves:

- ▶ removing a positive number of tokens from one pile,
- ▶ removing the same number of tokens from both piles.





Wythoff's game

Characterising the \mathcal{P} -positions

There exist several characterisations of the \mathcal{P} -positions of Wythoff's game: they are equivalently given by



Wythoff's game

Characterising the \mathcal{P} -positions

There exist several characterisations of the \mathcal{P} -positions of Wythoff's game: they are equivalently given by

- ▶ the pairs of integers $(\lfloor n\varphi \rfloor, \lfloor n\varphi^2 \rfloor)$, with $\varphi = \frac{1+\sqrt{5}}{2}$ the golden ratio,



Wythoff's game

Characterising the \mathcal{P} -positions

There exist several characterisations of the \mathcal{P} -positions of Wythoff's game: they are equivalently given by

- ▶ the pairs of integers $(\lfloor n\varphi \rfloor, \lfloor n\varphi^2 \rfloor)$, with $\varphi = \frac{1+\sqrt{5}}{2}$ the golden ratio,
- ▶ the pairs (a_n, b_n) , defined by $(a_0, b_0) = (0, 0)$ and

$$a_n = \text{mex}\{a_i, b_i : 0 \leq i < n\} \quad \text{and} \quad b_n = a_n + n,$$



Wythoff's game

Characterising the \mathcal{P} -positions

There exist several characterisations of the \mathcal{P} -positions of Wythoff's game: they are equivalently given by

- ▶ the pairs of integers $(\lfloor n\varphi \rfloor, \lfloor n\varphi^2 \rfloor)$, with $\varphi = \frac{1+\sqrt{5}}{2}$ the golden ratio,
- ▶ the pairs (a_n, b_n) , defined by $(a_0, b_0) = (0, 0)$ and

$$a_n = \text{mex}\{a_i, b_i : 0 \leq i < n\} \quad \text{and} \quad b_n = a_n + n,$$

- ▶ the positions of 0's and 1's in the *Fibonacci word* **f**:

$$\text{fixed point of } \begin{cases} 0 \mapsto 01 \\ 1 \mapsto 0 \end{cases} \rightsquigarrow \mathbf{f} = 0100101001001 \dots$$



Wythoff's game

Characterising the \mathcal{P} -positions

There exist several characterisations of the \mathcal{P} -positions of Wythoff's game: they are equivalently given by

- ▶ the pairs of integers $(\lfloor n\varphi \rfloor, \lfloor n\varphi^2 \rfloor)$, with $\varphi = \frac{1+\sqrt{5}}{2}$ the golden ratio,
- ▶ the pairs (a_n, b_n) , defined by $(a_0, b_0) = (0, 0)$ and

$$a_n = \text{mex}\{a_i, b_i : 0 \leq i < n\} \quad \text{and} \quad b_n = a_n + n,$$

- ▶ the positions of 0's and 1's in the **Fibonacci word \mathbf{f}** :

$$\text{fixed point of } \begin{cases} 0 \mapsto 01 \\ 1 \mapsto 0 \end{cases} \rightsquigarrow \mathbf{f} = 0100101001001 \dots$$

First \mathcal{P} -positions: $(0, 0), (1, 2), (3, 5), (4, 7), (6, 10)$, etc.



Wythoff's game

Characterising the \mathcal{P} -positions – A famous result

In this talk, we will focus on the following characterisation of the \mathcal{P} -positions:

Theorem (Fraenkel, 1982)

A pair (a, b) of integers such that $a \leq b$ is a \mathcal{P} -position of Wythoff's game if and only if

- ▶ $\text{rep}_F(a)$ ends with an even number of zeroes,
- ▶ $\text{rep}_F(b)$ is a left-shift of $\text{rep}_F(a)$, i.e. $\text{rep}_F(b) = \text{rep}_F(a)0$.



Wythoff's game

Characterising the \mathcal{P} -positions – A famous result

In this talk, we will focus on the following characterisation of the \mathcal{P} -positions:

Theorem (Fraenkel, 1982)

A pair (a, b) of integers such that $a \leq b$ is a \mathcal{P} -position of Wythoff's game if and only if

- ▶ $\text{rep}_F(a)$ ends with an even number of zeroes,
- ▶ $\text{rep}_F(b)$ is a left-shift of $\text{rep}_F(a)$, i.e. $\text{rep}_F(b) = \text{rep}_F(a)0$.

For instance,

$(6, 10)$ is a \mathcal{P} -position, since

$$\begin{aligned}\text{rep}_F(6) &= 1001 \\ \text{rep}_F(10) &= 10010\end{aligned}$$

$(131, 212)$ is a \mathcal{P} -position, since

$$\begin{aligned}\text{rep}_F(131) &= 1010010000 \\ \text{rep}_F(212) &= 10100100000\end{aligned}$$

The background of the slide is composed of two large, overlapping geometric shapes. A teal-colored shape occupies the top-left corner, while a light beige shape occupies the bottom-left corner. The rest of the slide is white. The text is centered in the white area.

Using Walnut for Wythoff's game



Using Walnut for Wythoff's game

What is Walnut?

Walnut is a free software system originally created by Hamoon Mousavi.

- ▶ extensively used for proving results in combinatorics on words and additive number theory;
- ▶ relies on *Büchi's theorem*: transform first-order logical formulas into finite automata for which decision procedures can be applied.

Hence, if a problem of interest can be expressed in a convenient extension of Presburger arithmetic $\langle \mathbb{N}, + \rangle$, it can then receive an automatic treatment.

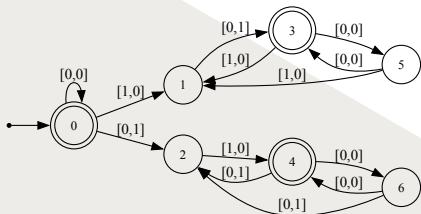


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$$\text{rep}_F(6) = 01001$$

$$\text{rep}_F(10) = 10010$$

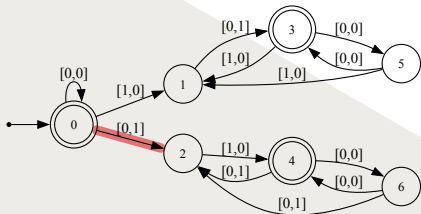


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$rep_F(6) = 01001$

$rep_F(10) = 10010$

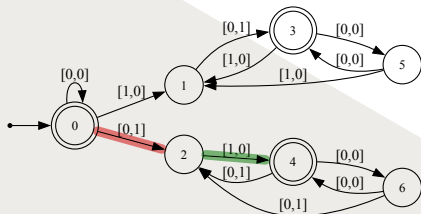


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$\text{rep}_F(6) = 01001$

$\text{rep}_F(10) = 10010$

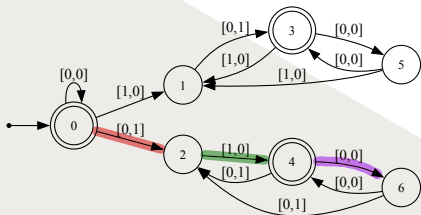


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$rep_F(6) = 01001$

$rep_F(10) = 10010$

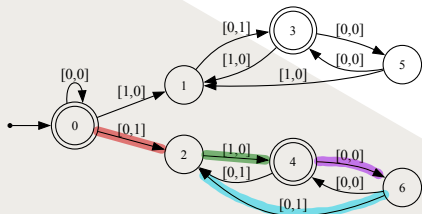


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$rep_F(6) = 01001$
 $rep_F(10) = 10010$

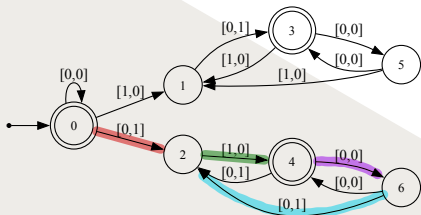


Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 1: Defining the candidate set for \mathcal{P} -positions in Walnut

```
reg end_even_zeros msd_fib "0*(00|0*1)*":  
reg left_shift {0,1} {0,1} "([0,0]|([0,1][1,1]*[1,0]))*":  
def ppos_asym "?msd_fib $end_even_zeros(a) & $left_shift(a,b)":  
def ppos "?msd_fib $ppos_asym(a,b) | $ppos_asym(b,a)":
```



Ex: we can evaluate $\$ppos(6, 10)$:

$rep_F(6) = 01001$
 $rep_F(10) = 10010$



Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Step 2: Verifying that this set is exactly the set of \mathcal{P} -positions

Recall that the sets of \mathcal{P} - and \mathcal{N} -positions of an impartial acyclic game are uniquely determined by the two following properties:

Stability, i.e. there is no move between two \mathcal{P} -positions

```
eval w_stable "?msd_fib Ap,q,r,s
((($ppos(p,q) & $ppos(r,s) & p >= r & q >= s)
=> ((p=r & q=s) | (p>r & q>s & p+s!=q+r))) ":
```

Absorbing, i.e. for each \mathcal{N} -position, there exists a move leading to a \mathcal{P} -position

```
eval w_absorbing "?msd_fib Ap,q (~$ppos(p,q) => Ex,y
( x<=p & y<=q & $ppos(x,y) & (p+y=q+x | p=x | q=y) )) ":
```



Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Both commands evaluate to TRUE, which proves Fraenkel's theorem!

We were able to use Walnut because:

- ▶ the rules of the game can be expressed using *first-order logic*,
- ▶ we have a "*regular*" candidate for the set of \mathcal{P} -positions,
- ▶ the Fibonacci numeration system is *addable*.



Using Walnut for Wythoff's game

Fraenkel's theorem using Walnut

Both commands evaluate to TRUE, which proves Fraenkel's theorem!

We were able to use Walnut because:

- ▶ the rules of the game can be expressed using *first-order logic*,
- ▶ we have a "*regular*" candidate for the set of \mathcal{P} -positions,
- ▶ the Fibonacci numeration system is *addable*.

Note also that we did have a candidate for the set of \mathcal{P} -positions.



Using Walnut for Wythoff's game

Other results about Wythoff's game

Thanks to Walnut, we also managed to:

- ▶ prove a 15-year-old conjecture describing the set of *forbidden moves* in Wythoff's game (i.e., the set of moves which would allow to play between two \mathcal{P} -positions),

E. Duchêne, A. Fraenkel, R. Nowakowski, M. Rigo (2010)

- ▶ show that all allowed moves in Wythoff's game are *non-redundant* (a move is said to be *redundant* if removing it from the rule-set does not affect the set of \mathcal{P} -positions),
- ▶ study some extensions/restrictions of Wythoff's game.



Using Walnut for Wythoff's game

Other results about Wythoff's game


Thanks to Walnut, we also managed to:

- ▶ prove a 15-year-old conjecture describing the set of *forbidden moves* in Wythoff's game (i.e., the set of moves which would allow to play between two \mathcal{P} -positions),

E. Duchêne, A. Fraenkel, R. Nowakowski, M. Rigo (2010)

- ▶ show that all allowed moves in Wythoff's game are *non-redundant* (a move is said to be *redundant* if removing it from the rule-set does not affect the set of \mathcal{P} -positions),
- ▶ study some extensions/restrictions of Wythoff's game.

What more can we do with this software?

The background of the slide is composed of two large, overlapping geometric shapes. A teal-colored shape, consisting of two overlapping triangles, is located in the upper-left corner. The rest of the slide is a light beige color.

Some generalisations



Some generalisations

Slightly changing the rules

Same rules as Wythoff's game, but a player may now remove $k > 0$ tokens from one heap and $\ell > 0$ from the other, provided that $|k - \ell| < m$ for a fixed integer $m \geq 1$.

Note that $m = 1 \rightsquigarrow$ Wythoff's game.



Some generalisations

Slightly changing the rules

Same rules as Wythoff's game, but a player may now remove $k > 0$ tokens from one heap and $\ell > 0$ from the other, provided that $|k - \ell| < m$ for a fixed integer $m \geq 1$.

Note that $m = 1 \rightsquigarrow$ Wythoff's game.

Here, we make use of *Ostrowski numeration systems*: consider the quadratic irrational

$$[1, \overline{m}] = \frac{2 - m + \sqrt{m^2 + 4}}{2}$$

→ the convergents give two numeration systems: the p -system (using numerators) and the q -system (using denominators).

Rem.: for $m = 1$, $[1, \overline{m}] = \varphi$ the golden ratio \rightsquigarrow Fibonacci numeration system.



Some generalisations

Slightly changing the rules

Theorem (Fraenkel, 1982)

A pair (a, b) of integers such that $a \leq b$ is a \mathcal{P} -position of m -Wythoff's game if and only if

- ▶ $\text{rep}_p(a)$ ends with an even number of zeroes,
- ▶ $\text{rep}_p(b)$ is a left-shift of $\text{rep}_p(a)$, i.e. $\text{rep}_p(b) = \text{rep}_p(a)0$,

where $\text{rep}_p(x)$ is the representation of x in the p -system associated to $[1, \overline{m}]$.

→ Walnut can handle such systems, and we can define the set of moves

```
ost ost2 [1] [2]:  
def ost2_move "?msd_ost2 (a+b>0) & (a=0 | b=0 |  
                (a>=b & a<b+2) | (a<b & b<a+2))";
```



Some generalisations

Using Walnut to find a new conjecture

Again here, we can ask what are the redundant moves of m -Wythoff's game. Using Walnut, we get the following conjecture:

Conjecture (Mignoty, R., Rigo, Whiteland, 2025)

Let $m \geq 2$. The set of redundant moves of the variation of Wythoff's game where one is allowed to remove $k > 0$ and $\ell > 0$ provided that $|k - \ell| < m$ is

$$\bigcup_{1 \leq i < m} \{(n, n + i), (n + i, n) \mid n \geq m - i + 2\}.$$

→ proved for $m = 2$ ($\sim 7\text{Gb}$), $m = 3, 4$ ($\sim 45\text{Gb}$, 20 minutes);

→ for $m = 5$, we quickly run out of memory \rightsquigarrow use Walnut differently.



Some generalisations

Beyond Ostrowski systems

Same rules as Wythoff's game, but a player may now remove $k > 0$ tokens from one heap and $\ell > 0$ from the other, provided that $0 < k \leq \ell < sk + m$ for two positive integers m, s . For $s = 1 \rightsquigarrow m$ -Wythoff's game; for $m = s = 1 \rightsquigarrow$ Wythoff's game.



Some generalisations

Beyond Ostrowski systems

Same rules as Wythoff's game, but a player may now remove $k > 0$ tokens from one heap and $\ell > 0$ from the other, provided that $0 < k \leq \ell < sk + m$ for two positive integers m, s . For $s = 1 \rightsquigarrow m$ -Wythoff's game; for $m = s = 1 \rightsquigarrow$ Wythoff's game.

We do get the same result (Fraenkel, 1998) using the following numeration system: we define the linear recurrence sequence $(U_i)_{i \geq 0}$ by

$$U_0 = 1, \quad U_1 = s + m \quad \text{and} \quad U_i = (s + m - 1)U_{i-1} + sU_{i-2} \quad \forall i \geq 2.$$

We then have that each integer $n > 0$ has a unique representation $d_\ell \cdots d_0$ such that

$$n = \sum_{i=0}^{\ell} d_i U_i, \quad \text{with } d_\ell \neq 0.$$



Some generalisations

Beyond Ostrowski systems

We are lucky with this numeration system, for we are in the *Pisot* case, which means:

- ▶ we have a regular candidate,
- ▶ we have an addable system.

So, in principle, we can use Walnut!



Some generalisations

Beyond Ostrowski systems

We are lucky with this numeration system, for we are in the *Pisot* case, which means:

- ▶ we have a regular candidate,
- ▶ we have an addable system.

So, in principle, we can use Walnut!

Some more work here all the same: no "built-in" automata for this system, so we have to provide Walnut with

- ▶ an automaton recognising *U*-representations,
- ▶ another one computing the addition \rightsquigarrow using the work of C. Frougny and J. Sakarovitch to build the *zero-automaton*. We can then check in Walnut that the automaton does what it should (O. Carton, J.-M. Couvreur, M. Delacourt, and N. Ollinger, 2025).

The background of the slide is composed of two large, overlapping geometric shapes. A teal-colored shape, consisting of two overlapping triangles, is located in the upper-left corner. The rest of the slide is a light beige color.

Conclusion

Conclusion



Fraenkel's combinatorial games and Walnut:
It's a *match*!

Conclusion



Fraenkel's combinatorial games and Walnut:
It's a *match*!

- ▶ rules can be written using *first-order logic*,
- ▶ we have "nice" numeration systems (*addable*),
- ▶ the set of \mathcal{P} -position is *regular*.

Consequences: automatic proofs of old and new results, conjectures, building new games

Conclusion



Fraenkel's combinatorial games and Walnut: It's a *match*!

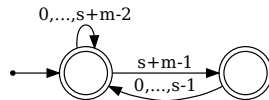
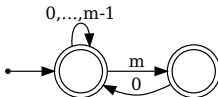
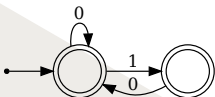
- ▶ rules can be written using *first-order logic*,
- ▶ we have "nice" numeration systems (*addable*),
- ▶ the set of \mathcal{P} -position is *regular*.

Consequences: automatic proofs of old and new results, conjectures, building new games
However...

- ▶ automatic proofs are obtained for *fixed* parameters,
- ▶ *state complexity* could be problematic,
- ▶ difficult to cope with *Tribonacci adder*.



Thank you for your attention!



Checking the adder



```
eval test1 "?msd_a3ba2 Ax,y Ez x+y=z":  
eval test2 "?msd_a3ba2 Ax,y,z,t (x+y=z & x+y=t) => z=t":  
eval test3 "?msd_a3ba2 Ax,z (x+0=z) <=> x=z":  
eval test4 "?msd_a3ba2 Ax,t (x+1=t) <=>  
                (x<t & (Ay x<y => t <= y))":  
eval test5 "?msd_a3ba2 Ax,y,z,u,t (u=y+1 & t=z+1) =>  
                (x+y=z <=> x+u=t)":
```