Subtraction Games on Graphs

Antoine Dailly
(LIMOS, Clermont-Ferrand)

With Laurent Beaudou (LIMOS), Kyle Burke (Plymouth State University),
Pierre Coupechoux (LAAS), Sylvain Gravier (Institut Fourier),
Julien Moncel (LAAS), Aline Parreau (LIRIS), Éric Sopena (LaBRI).

Most of this work was realized in the ANR project GAG.

Games And Graphs

CGTCIV, January 25, 2023

Subtraction games

Subtraction game SUB(S)

Played on a heap of counters.
Removing k counters $\Leftrightarrow k \in S$.

Subtraction games

Subtraction game SUB(S)

Played on a heap of counters.
Removing k counters $\Leftrightarrow$ $k \in S$.

Theorem (Folklore)

If S is finite, then, the sequence of SUB(S) is ultimately periodic.

Theorem (Albert, Nowakowski, Wolfe, 2007)

If S is finite, then, the sequence of SUB($\mathbb{N} \setminus S$) is ultimately arithmetic periodic.

Subtraction games

Subtraction game SUB(S)

Played on a heap of counters.
Removing k counters $\Leftrightarrow k \in S$.

Theorem (Folklore)

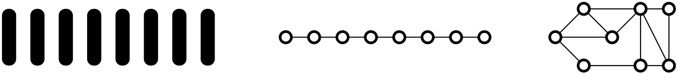
If S is finite, then, the sequence of SUB(S) is ultimately periodic.

Theorem (Albert, Nowakowski, Wolfe, 2007)

If S is finite, then, the sequence of SUB($\mathbb{N} \setminus S$) is ultimately arithmetic periodic.

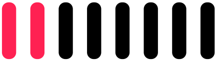
How to define those games on other structures?

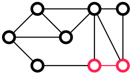
From heaps of counters to graphs



Playing on a heap $\sim$ Playing on a path $\sim$ Playing on a graph

From heaps of counters to graphs

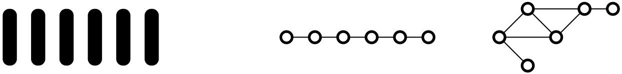




Playing on a heap $\sim$ Playing on a path $\sim$ Playing on a graph

- Removing counters → Removing a connected subgraph

From heaps of counters to graphs



Playing on a heap $\sim$ Playing on a path $\sim$ Playing on a graph

- Removing counters $\rightarrow$ Removing a connected subgraph

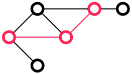
From heaps of counters to graphs

Playing on a heap

$\sim$

Playing on a path

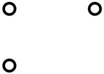
$\sim$



Playing on a graph

- Removing counters → Removing a connected subgraph
- Dividing a heap → Disconnecting a graph

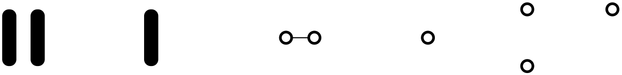
From heaps of counters to graphs



Playing on a heap $\sim$ Playing on a path $\sim$ Playing on a graph

- Removing counters → Removing a connected subgraph
- Dividing a heap → Disconnecting a graph

From heaps of counters to graphs



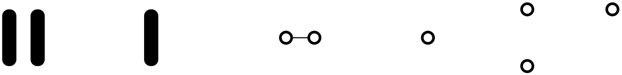
Playing on a heap $\sim$ Playing on a path $\sim$ Playing on a graph

- Removing counters → Removing a connected subgraph
- Dividing a heap → Disconnecting a graph

Connected subtraction game CSG(S)

Removing a connected subgraph of order k without disconnecting the graph $\Leftrightarrow k \in S$

From heaps of counters to graphs



Playing on a heap ∼ Playing on a path ∼ Playing on a graph

- Removing counters → Removing a connected subgraph
- Dividing a heap → Disconnecting a graph

Connected subtraction game CSG(S)

Removing a connected subgraph of order k without disconnecting the graph $\Leftrightarrow k \in S$

First introduced for octal games on graphs [BCDGMPS, 2018]

Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then $\mathsf{CSG}(S)$ is PSPACE-complete.

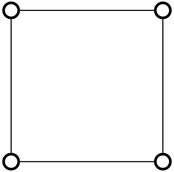
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



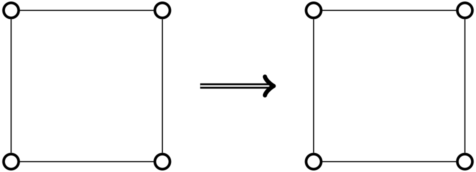
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



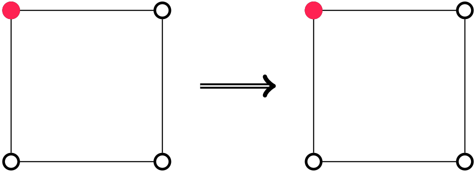
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



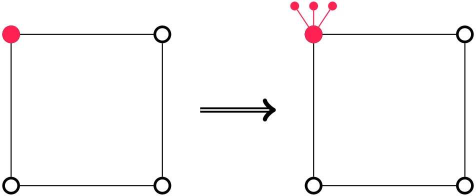
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



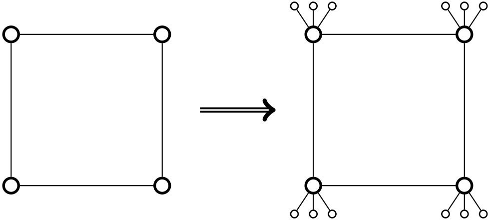
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then $\mathrm{CSG}(S)$ is PSPACE-complete.

Reduction from Node-Kayles

$M = max(S)$, here $M = 4$



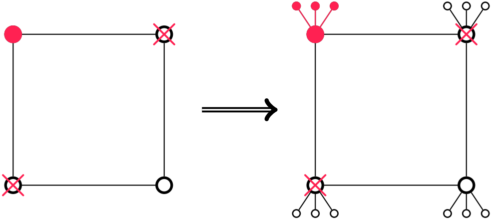
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from Node-Kayles

$M = max(S)$, here $M = 4$



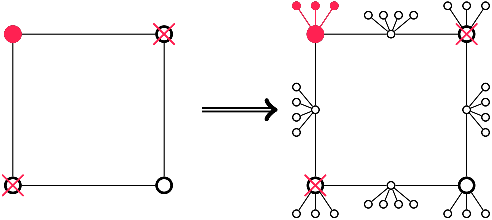
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



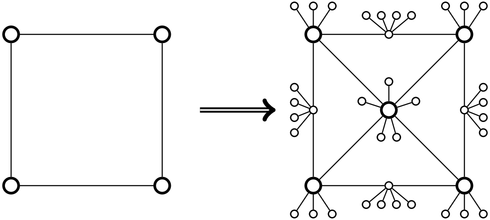
Complexity of subtraction games

Theorem [Burke and D., 2023+]

If S is finite and $1 \notin S$, then CSG(S) is PSPACE-complete.

Reduction from NODE-KAYLES

$M = max(S)$, here $M = 4$



Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

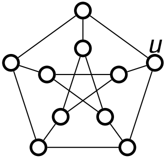
Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

Idea

Study the evolution of the Grundy values when appending a path to a given vertex.

$$G \underset{u}{\bullet\!\!-\!\!\bullet} P_k =$$



$(k = 0)$

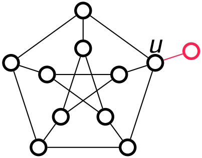
Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

Idea

Study the evolution of the Grundy values when appending a path to a given vertex.

$$G \underset{u}{\bullet\!\!-\!\!\bullet} P_k =$$



$(k = 1)$

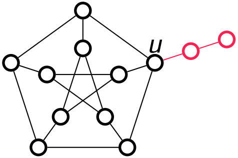
Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

Idea

Study the evolution of the Grundy values when appending a path to a given vertex.

$G \underset{u}{\bullet}\!-\!\bullet\, P_k =$



$(k = 2)$

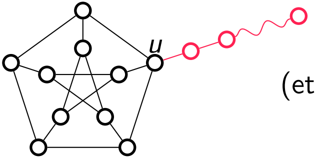
Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

Idea

Study the evolution of the Grundy values when appending a path to a given vertex.

$G \underset{u}{\bullet}\!\!-\!\!\bullet P_k =$



(etc)

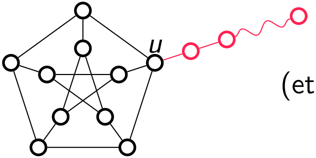
Defining regularity for octal games on graphs

- Subtraction games → Grundy sequence
- Subtraction games on graphs → ?

Idea

Study the evolution of the Grundy values when appending a path to a given vertex.

$G \underset{u}{\bullet}\!\!-\!\!\bullet P_k =$



(etc)

→ Already used for Node-Kayles [Fleischer and Trippen, 2004] and Arc-Kayles [Huggan and Stevens, 2016]

A periodicity result for connected subtraction games

Theorem [D., Moncel, Parreau, 2019]

If S is finite, , then, for every graph G and vertex u, the sequence of $\mathcal{G}(G \underset{u}{\bullet}\!\!-\!\!\bullet P_k)$ for CSG(S) is ultimately periodic.

A periodicity result for connected subtraction games

Theorem [D., Moncel, Parreau, 2019]

If S is finite, , then, for every graph G and vertex u, the sequence of $\mathcal{G}(G \underset{u}{\bullet}\!\!-\!\!\bullet P_k)$ for CSG(S) is ultimately periodic.

Proof idea

Induction on $|G|$.

1. $|G| \in \{0, 1\}$: paths

A periodicity result for connected subtraction games

Theorem [D., Moncel, Parreau, 2019]

If S is finite, , then, for every graph G and vertex u, the sequence of $\mathcal{G}(G \underset{u}{\bullet\!\!-\!\!\bullet} P_k)$ for CSG(S) is ultimately periodic.

Proof idea

Induction on $|G|$.

1. $|G| \in \{0, 1\}$: paths
2. Otherwise, three possible kinds of moves:
   - 2.1 Playing on $P_k \rightarrow |S|$ different moves
   - 2.2 Playing on G without removing $u \rightarrow$ at most $2^{|G|-1}$ different moves
   - 2.3 Emptying $G \rightarrow$ at most $|S|$ different moves

A periodicity result for connected subtraction games

Theorem [D., Moncel, Parreau, 2019]

If S is finite, , then, for every graph G and vertex u, the sequence of $\mathcal{G}(G \underset{u}{\bullet\!\!-\!\!\bullet} P_k)$ for CSG(S) is ultimately periodic.

Proof idea

Induction on $|G|$.

1. $|G| \in \{0, 1\}$: paths
2. Otherwise, three possible kinds of moves:
   - 2.1 Playing on $P_k \rightarrow |S|$ different moves
   - 2.2 Playing on G without removing $u \rightarrow$ at most $2^{|G|-1}$ different moves
   - 2.3 Emptying $G \rightarrow$ at most $|S|$ different moves

   $\Rightarrow \mathcal{G}(G) \leq C$

A periodicity result for connected subtraction games

Theorem [D., Moncel, Parreau, 2019]

If S is finite, , then, for every graph G and vertex u, the sequence of $\mathcal{G}(G \underset{u}{\bullet}\!\!-\!\!\bullet P_k)$ for CSG(S) is ultimately periodic.

Proof idea

Induction on $|G|$.

1. $|G| \in \{0, 1\}$: paths
2. Otherwise, three possible kinds of moves:
   - 2.1 Playing on $P_k \rightarrow |S|$ different moves
   - 2.2 Playing on G without removing $u \rightarrow$ at most $2^{|G|-1}$ different moves
   - 2.3 Emptying $G \rightarrow$ at most $|S|$ different moves

   $\Rightarrow \mathcal{G}(G) \leq C$

   Every move leads to a periodic sequence, by mex computation, we have the result.

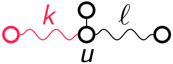
Regularity results

Game	Graph and vertex u	Regularity	Reference

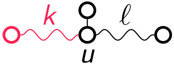
Regularity results

Game	Graph and vertex u	Regularity	Reference
Every CSG (S finite)	Every graph G, every vertex u	Ultimate periodicity	D., Moncel, Parreau (2019)

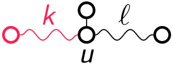
Regularity results

Game	Graph and vertex u	Regularity	Reference
Every CSG (S finite)	Every graph G, every vertex u	Ultimate periodicity	D., Moncel, Parreau (2019)
$CSG(S)$, $S = \{1, \ldots, N\}$	Star $K_{1,n}$, u central vertex	Period $N+1$	D., Moncel, Parreau (2019)
		Period $N+1$ Preperiod 0 or $N+1$	

Regularity results

Game	Graph and vertex u	Regularity	Reference
Every CSG (S finite)	Every graph G, every vertex u	Ultimate periodicity	D., Moncel, Parreau (2019)
$CSG(S)$, $S = \{1, ..., N\}$	Star $K_{1,n}$, u central vertex	Period $N+1$	D., Moncel, Parreau (2019)
		Period $N+1$ Preperiod 0 or $N+1$	
CSG($\{1, 2, 3\}$)	Any subdivided star, u central vertex or leaf	Period $N+1=4$	D., Moncel, Parreau (2019)
CSG($\{1, 2\}$)		Period $N+1=3$	BCDGMPS (2018)

Regularity results

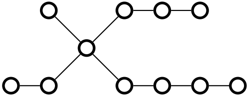
Game	Graph and vertex u	Regularity	Reference
Every CSG (S finite)	Every graph G, every vertex u	Ultimate periodicity	D., Moncel, Parreau (2019)
$CSG(S)$, $S = \{1, \ldots, N\}$	Star $K_{1,n}$, u central vertex	Period $N+1$	D., Moncel, Parreau (2019)
		Period $N+1$ Preperiod 0 or $N+1$	
CSG($\{1,2,3\}$)	Any subdivided star, u central vertex or leaf	Period $N+1=4$	D., Moncel, Parreau (2019)
CSG($\{1,2\}$)		Period $N+1=3$	Beaudou, Coupechoux D., Gravier, Moncel, Parreau, Sopena (2018)
	Any subdivided bistar, u central vertex or leaf		

CSG($\{1, 2\}$) on subdivided stars in polynomial-time

CSG($\{1,2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

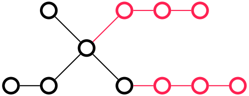
In CSG($\{1,2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



CSG($\{1,2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

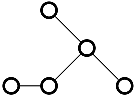
In CSG($\{1,2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



CSG($\{1, 2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

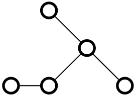
In CSG($\{1, 2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



CSG($\{1, 2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

In CSG($\{1, 2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



Number of paths of length 2

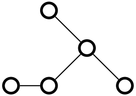
Number of paths

	0	1	2	3	4	5	$\cdots$	$2p$	$2p+1$
$\emptyset$	0								
0	1								
1	2	0							
2	0	1	2						
3	1	2	0	1					
4	0	3	1	2	0				
5	1	2	0	3	1	2			
$\vdots$									
$2p$	0	3	1	2	0	3	$(03)^*$	0	
$2p+1$	1	2	0	3	1	2	$(12)^*$	1	2

CSG($\{1, 2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

In CSG($\{1, 2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



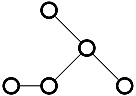
Number of paths of length 2

Number of paths	0	1	2	3	4	5	$\cdots$	$2p$	$2p+1$
$\emptyset$	0								
0	1								
1	2	0							
2	0	1	2						
3	1	$\boxed{2}$	0	1					
4	0	3	1	2	0				
5	1	2	0	3	1	2			
$\vdots$									
$2p$	0	3	1	2	0	3	$(03)^*$	0	
$2p+1$	1	2	0	3	1	2	$(12)^*$	1	2

CSG($\{1,2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

In CSG($\{1,2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.



Number of paths of length 2

Number of paths

	0	1	2	3	4	5	$\cdots$	$2p$	$2p+1$
$\emptyset$	0								
0	1								
1	2	0							
2	0	1	2						
3	1	$\boxed{2}$	0	1					
4	0	3	1	2	0				
5	1	2	0	3	1	2			
$\vdots$									
$2p$	0	3	1	2	0	3	$(03)^*$	0	
$2p+1$	1	2	0	3	1	2	$(12)^*$	1	2

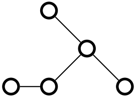
Theorem (BCDGMPS18)

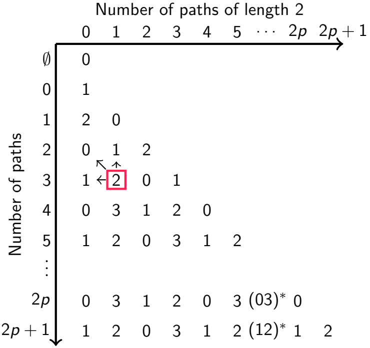
There is a polynomial-time algorithm computing the Grundy value of a subdivided star in CSG($\{1,2\}$).

CSG($\{1, 2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

In CSG($\{1, 2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.





Number of paths of length 2

Number of paths

	0	1	2	3	4	5	⋯	$2p$	$2p+1$
∅	0								
0	1								
1	2	0							
2	0	1	2						
3	1	$\boxed{2}$	0	1					
4	0	3	1	2	0				
5	1	2	0	3	1	2			
⋮									
$2p$	0	3	1	2	0	3	$(03)^*$	0	
$2p+1$	1	2	0	3	1	2	$(12)^*$	1	2

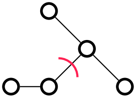
Theorem (BCDGMPS18)

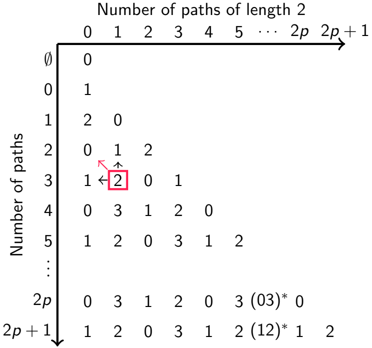
There is a polynomial-time algorithm computing the Grundy value of a subdivided star in CSG($\{1, 2\}$).

CSG($\{1,2\}$) on subdivided stars in polynomial-time

Lemma (BCDGMPS18)

In CSG($\{1,2\}$), we can reduce the paths of a subdivided stars to their length modulo 3.





Number of paths of length 2

Number of paths	0	1	2	3	4	5	$\cdots$	$2p$	$2p+1$
$\emptyset$	0								
0	1								
1	2	0							
2	0	1	2						
3	1	$\boxed{2}$	0	1					
4	0	3	1	2	0				
5	1	2	0	3	1	2			
$\vdots$									
$2p$	0	3	1	2	0	3	$(03)^*$	0	
$2p+1$	1	2	0	3	1	2	$(12)^*$	1	2

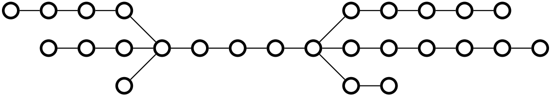
Theorem (BCDGMPS18)

There is a polynomial-time algorithm computing the Grundy value of a subdivided star in CSG($\{1,2\}$).

CSG($\{1,2\}$) on subdivided bistars

Paths reduction in bistars

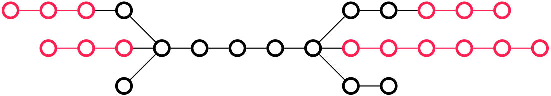
Periodicity ⇒ reduction of stars' paths.



CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

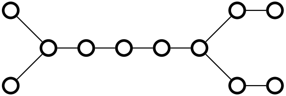
Periodicity ⇒ reduction of stars' paths.



CSG($\{1,2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

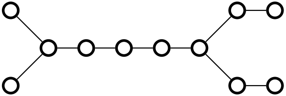


CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

Reduction of the central path.

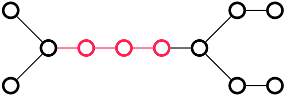


CSG($\{1,2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

Reduction of the central path.

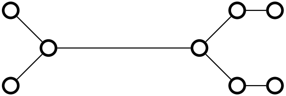


CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

Reduction of the central path.

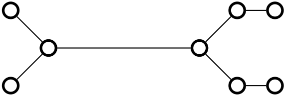


CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

Reduction of the central path.



Theorem [BCDGMPS, 2018]

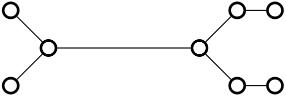
There is a polynomial-time algorithm computing the Grundy value of a subdivided bistar in CSG($\{1, 2\}$), using two refinements of the nim-sum.

CSG($\{1,2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

Reduction of the central path.



Theorem [BCDGMPS, 2018]

There is a polynomial-time algorithm computing the Grundy value of a subdivided bistar in CSG($\{1,2\}$), using two refinements of the nim-sum.

Reduction of paths for CSG($\{1,2\}$)

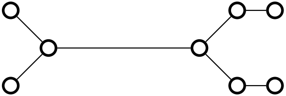
Paths

CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

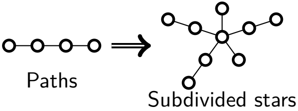
Reduction of the central path.



Theorem [BCDGMPS, 2018]

There is a polynomial-time algorithm computing the Grundy value of a subdivided bistar in CSG($\{1, 2\}$), using two refinements of the nim-sum.

Reduction of paths for CSG($\{1, 2\}$)

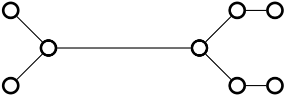


CSG($\{1,2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

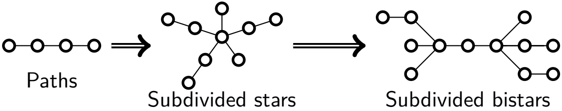
Reduction of the central path.



Theorem [BCDGMPS, 2018]

There is a polynomial-time algorithm computing the Grundy value of a subdivided bistar in CSG($\{1,2\}$), using two refinements of the nim-sum.

Reduction of paths for CSG($\{1,2\}$)

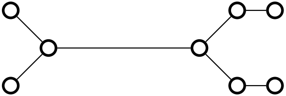


CSG($\{1, 2\}$) on subdivided bistars

Paths reduction in bistars

Periodicity ⇒ reduction of stars' paths.

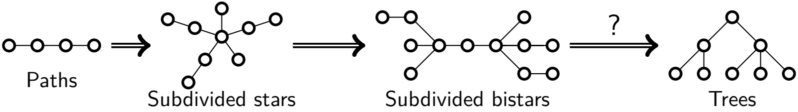
Reduction of the central path.



Theorem [BCDGMPS, 2018]

There is a polynomial-time algorithm computing the Grundy value of a subdivided bistar in CSG($\{1, 2\}$), using two refinements of the nim-sum.

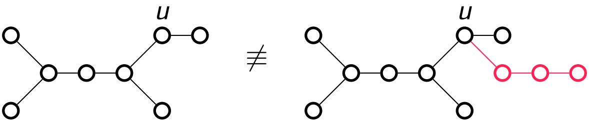
Reduction of paths for CSG($\{1, 2\}$)



CSG($\{1,2\}$) on trees?

Proposition

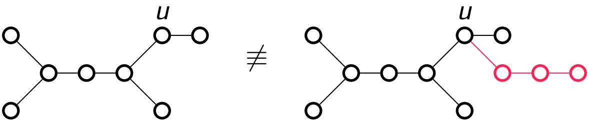
The reduction of paths does not work in trees:



CSG($\{1, 2\}$) on trees?

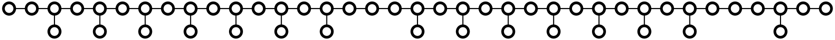
Proposition

The reduction of paths does not work in trees:



Unbounded values?

The following caterpillar has Grundy value 10:



A few other games

CSG({2})

Arc-Kayles without disconnecting the graph.

A few other games

CSG({2})

ARC-KAYLES without disconnecting the graph.

- On trees and $2 \times n$ grids: every possible move will be played, so no strategy...

A few other games

CSG($\{2\}$)

ARC-KAYLES without disconnecting the graph.

- On trees and $2 \times n$ grids: every possible move will be played, so no strategy...
- On $3 \times n$ grids: always possible to empty the grid!

A few other games

CSG({2})

ARC-KAYLES without disconnecting the graph.

- On trees and $2 \times n$ grids: every possible move will be played, so no strategy...
- On $3 \times n$ grids: always possible to empty the grid!
- Newsflash: PSPACE-complete on bipartite graphs, split graphs and graphs of any given girth!

A few other games

CSG($\{2\}$)

ARC-KAYLES without disconnecting the graph.

- On trees and $2 \times n$ grids: every possible move will be played, so no strategy...
- On $3 \times n$ grids: always possible to empty the grid!
- Newsflash: PSPACE-complete on bipartite graphs, split graphs and graphs of any given girth!

Adjoining integers

We can adjoin M to S if $\mathcal{G}(G)$ is the same for CSG(S) and CSG($S \cup \{M\}$)

Theorem [D. Moncel, Parreau, 2019]

Let G be a subdivided star. Then, $\mathcal{G}(G)$ is the same for CSG($\{1, 2\}$) and CSG($\{1, 2, 4\}$).

A few other games

CSG({2})

ARC-KAYLES without disconnecting the graph.

- On trees and $2 \times n$ grids: every possible move will be played, so no strategy...
- On $3 \times n$ grids: always possible to empty the grid!
- Newsflash: PSPACE-complete on bipartite graphs, split graphs and graphs of any given girth!

Adjoining integers

We can adjoin M to S if $\mathcal{G}(G)$ is the same for CSG(S) and CSG($S \cup \{M\}$)

Theorem [D. Moncel, Parreau, 2019]

Let G be a subdivided star. Then, $\mathcal{G}(G)$ is the same for CSG($\{1,2\}$) and CSG($\{1,2,4\}$).

However, this is not always possible, even on subdivided stars!

Final words

Some interesting open problems

Final words

Some interesting open problems

1. Complexity of CSG(S) when $1 \in S$
2. Regularity of CSG($\mathbb{N} \setminus S$)
3. Adjoining integers to S: when is it possible?

Final words

Some interesting open problems

1. Complexity of CSG(S) when $1 \in S$
2. Regularity of CSG($\mathbb{N} \setminus S$)
3. Adjoining integers to S: when is it possible?
4. CSG($\{1, \ldots, N\}$) for $N \geq 4$ on subdivided stars: can we still reduce paths?
5. CSG($\{1, 2\}$) on trees: which regularity?
6. CSG($\{1, 2\}$) on trees: are Grundy values bounded?

Final words

Some interesting open problems

1. Complexity of CSG(S) when $1 \in S$
2. Regularity of CSG($\mathbb{N} \setminus S$)
3. Adjoining integers to S: when is it possible?
4. CSG($\{1, \ldots, N\}$) for $N \geq 4$ on subdivided stars: can we still reduce paths?
5. CSG($\{1, 2\}$) on trees: which regularity?
6. CSG($\{1, 2\}$) on trees: are Grundy values bounded?
7. CSG($\{2\}$) on larger grids: will they still be emptied?
8. CSG($\{N\}$) for $N \geq 3$ on trees: is there some strategy?

Final words

Some interesting open problems

1. Complexity of CSG(S) when $1 \in S$
2. Regularity of CSG($\mathbb{N} \setminus S$)
3. Adjoining integers to S: when is it possible?
4. CSG($\{1, \dots, N\}$) for $N \geq 4$ on subdivided stars: can we still reduce paths?
5. CSG($\{1, 2\}$) on trees: which regularity?
6. CSG($\{1, 2\}$) on trees: are Grundy values bounded?
7. CSG($\{2\}$) on larger grids: will they still be emptied?
8. CSG($\{N\}$) for $N \geq 3$ on trees: is there some strategy?