Multidimensional Zeckendorf Decompositions

Jiarui Cheng[1] Sebastian Rodriguez Labastida[2]
Tianyu Shen[3] Alan Sun[4]

Mentors: Steven J. Miller[5] Garrett Tresch[6]

[1]Northeastern University [2]Universidad Panamericana [3]Shanghai University
[4]University of Michigan [5]Williams College [6]Texas A&M University

The 22nd International Conference on Fibonacci Numbers and Their Applications

Emails: 1790299332@qq.com (Shen), sjm1@williams.edu (Miller)

Fibonacci and Zeckendorf

Fibonacci Numbers: $F_{n+1} = F_n + F_{n-1}$ with $F_1 = 1, F_2 = 2$.

Fibonacci and Zeckendorf

Fibonacci Numbers: $F_{n+1} = F_n + F_{n-1}$ with $F_1 = 1, F_2 = 2$.

Zeckendorf's Theorem (1972)

Every positive integer can be written uniquely as a sum of non-consecutive Fibonacci numbers.

Fibonacci and Zeckendorf

Fibonacci Numbers: $F_{n+1} = F_n + F_{n-1}$ with $F_1 = 1, F_2 = 2$.

Zeckendorf's Theorem (1972)

Every positive integer can be written uniquely as a sum of non-consecutive Fibonacci numbers.

Example: $51 = 34 + 13 + 3 + 1 = F_8 + F_6 + F_3 + F_1$.

Generalized Recurrences

We can generalize the Fibonacci sequence to **Positive Linear Recurrence Sequences (PLRS).**

Generalized Recurrences

We can generalize the Fibonacci sequence to **Positive Linear Recurrence Sequences (PLRS).**

Definition (M. Kologlu, G.S. Kopp, S. J. Miller, Y. Wang, 2011)

For a positive coefficient vector $\vec{\mathbf{c}} = (c_1, \ldots, c_k) \in \mathbb{Z}^k$, we define a sequence $(X_n) \subseteq \mathbb{Z}$ such that:

1. $X_1 = 1$;
2. $X_{n+1} = c_1 X_n + \cdots + c_n X_1 + 1$, for $1 \leq n < k$;
3. $X_{n+1} = c_1 X_n + \cdots + c_k X_{n-k+1}$, for $n \geq k$.

Example

Let $\vec{\mathbf{c}} = (2, 1, 1)$.

Example

Let $\vec{\mathbf{c}} = (2, 1, 1)$.

- **Initial terms:**

$$
\begin{aligned}
X_1 &= 1, \\
X_2 &= 2 \cdot 1 + 1 = 3, \\
X_3 &= 2 \cdot 3 + 1 \cdot 1 + 1 = 8.
\end{aligned}
$$

Example

Let $\vec{\mathbf{c}} = (2, 1, 1)$.

- **Initial terms:**

$$
\begin{aligned}
X_1 &= 1, \\
X_2 &= 2 \cdot 1 + 1 = 3, \\
X_3 &= 2 \cdot 3 + 1 \cdot 1 + 1 = 8.
\end{aligned}
$$

- **Recurrence for $n > 3$:**

$$X_n = 2X_{n-1} + X_{n-2} + X_{n-3}.$$

Example

Let $\vec{\mathbf{c}} = (2, 1, 1)$.

- **Initial terms:**

$$\begin{aligned} X_1 &= 1, \\ X_2 &= 2 \cdot 1 + 1 = 3, \\ X_3 &= 2 \cdot 3 + 1 \cdot 1 + 1 = 8. \end{aligned}$$

- **Recurrence for $n > 3$:**

$$X_n = 2X_{n-1} + X_{n-2} + X_{n-3}.$$

- **Sequence:** $1, 3, 8, 20, 51, 130, \dots$.

Multidimensional Extension

Anderson and Bicknell-Johnson first extended this notion to a multidimensional setting, $\mathbb{Z}^{k-1}$.

Multidimensional Extension

Anderson and Bicknell-Johnson first extended this notion to a multidimensional setting, $\mathbb{Z}^{k-1}$.

Definition (AB-J, 2011)

- $\vec{\mathbf{X}}_0 := \vec{\mathbf{0}}$;
- $\vec{\mathbf{X}}_{-i} := \vec{\mathbf{e}}_i$, for $i < k$;
- $\vec{\mathbf{X}}_n := c_1\vec{\mathbf{X}}_{n-1} + \cdots + c_k\vec{\mathbf{X}}_{n-k}$, for $n \in \mathbb{Z}$.

Backward Recursion

Backward Recursion

While we usually move forward, the assumption $c_k = 1$ allows us to define new vectors backwards by the recursion:

Backward Recursion

Backward Recursion

While we usually move forward, the assumption $c_k = 1$ allows us to define new vectors backwards by the recursion:

$$\vec{\mathbf{X}}_n = \vec{\mathbf{X}}_{n+k} - \sum_{i=1}^{k-1} c_i \vec{\mathbf{X}}_{n+k-i}$$

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1, 1)$) $\rightarrow$ Non-adjacent

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1, 1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1,1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1,1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$**-satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1, 1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1,1)$) $\to$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and
- all coefficients are "bounded" appropriately.

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1, 1)$) $\to$ Non-adjacent

What is a **$\vec{\mathbf{c}}$-satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and
- all coefficients are "bounded" appropriately.

Example: Consider $\vec{\mathbf{c}} = (4, 2, 1)$, then

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1, 1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and
- all coefficients are "bounded" appropriately.

Example: Consider $\vec{\mathbf{c}} = (4, 2, 1)$, then

- $\vec{v_1} := 2\vec{\mathbf{X}}_{-1} + 4\vec{\mathbf{X}}_{-2} + 2\vec{\mathbf{X}}_{-3} + 1\vec{\mathbf{X}}_{-4} \longrightarrow$ 2, 4, 2, 1

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1,1)$) $\rightarrow$ Non-adjacent

What is a $\vec{\mathbf{c}}$-**satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and
- all coefficients are "bounded" appropriately.

Example: Consider $\vec{\mathbf{c}} = (4,2,1)$, then

- $\vec{v_1} := 2\vec{\mathbf{X}}_{-1} + 4\vec{\mathbf{X}}_{-2} + 2\vec{\mathbf{X}}_{-3} + 1\vec{\mathbf{X}}_{-4} \longrightarrow$ 2, 4, 2, 1
- $\vec{v_2} := 0\vec{\mathbf{X}}_{-1} + 1\vec{\mathbf{X}}_{-2} + 5\vec{\mathbf{X}}_{-3} + 1\vec{\mathbf{X}}_{-4} + 1\vec{\mathbf{X}}_{-5} \longrightarrow$ 0, 1, 5, 1, 1

Satisfying Representations

Fibonacci (PLRS with $\vec{\mathbf{c}} = (1,1)$) $\rightarrow$ Non-adjacent

What is a **$\vec{\mathbf{c}}$-satisfying** representation for $v \in \mathbb{Z}^{k-1}$?

- Sequence $(a_n)_{n=1}^{\infty} \subseteq \mathbb{Z}^+$;
- only consider negative terms, $v = \sum_{n=1}^{m} a_n \vec{\mathbf{X}}_{-n}$;
- cannot replace terms using the recursion; and
- all coefficients are "bounded" appropriately.

Example: Consider $\vec{\mathbf{c}} = (4,2,1)$, then

- $\vec{v_1} := 2\vec{\mathbf{X}}_{-1} + 4\vec{\mathbf{X}}_{-2} + 2\vec{\mathbf{X}}_{-3} + 1\vec{\mathbf{X}}_{-4} \longrightarrow$ 2, 4, 2, 1
- $\vec{v_2} := 0\vec{\mathbf{X}}_{-1} + 1\vec{\mathbf{X}}_{-2} + 5\vec{\mathbf{X}}_{-3} + 1\vec{\mathbf{X}}_{-4} + 1\vec{\mathbf{X}}_{-5} \longrightarrow$ 0, 1, 5, 1, 1
- $\vec{v_3} := 0\vec{\mathbf{X}}_{-1} + 2\vec{\mathbf{X}}_{-2} + 1\vec{\mathbf{X}}_{-3} + 0\vec{\mathbf{X}}_{-4} + 1\vec{\mathbf{X}}_{-5} \longrightarrow$ 0, 2, 1, 0, 1

Anderson and Bicknell-Johnson

Theorem (AB-J)

Every $\vec{\mathbf{v}} \in \mathbb{Z}^{k-1}$ *has a unique* $\vec{\mathbf{c}} = (1, \ldots, 1)$*-satisfying representation.*

Weakly Decreasing Coefficients

Definition (Weakly decreasing)

Vector $\vec{\mathbf{c}} = (c_1, \ldots, c_k)$ is weakly decreasing if $c_n \geq c_{n+1}$.

Weakly Decreasing Coefficients

Definition (Weakly decreasing)

Vector $\vec{\mathbf{c}} = (c_1, \ldots, c_k)$ is weakly decreasing if $c_n \geq c_{n+1}$.

Theorem (Main Result)

If $\vec{\mathbf{c}} = (c_1, c_2, \ \ldots, \ c_k)$ is weakly decreasing and $c_k = 1$, every $\vec{\mathbf{v}} \in \mathbb{Z}^{k-1}$ has a **unique $\vec{\mathbf{c}}$-satisfying representation**.

Weakly Decreasing Coefficients

Definition (Weakly decreasing)

Vector $\vec{\mathbf{c}} = (c_1, \ldots, c_k)$ is weakly decreasing if $c_n \geq c_{n+1}$.

Theorem (Main Result)

If $\vec{\mathbf{c}} = (c_1, c_2, \ \ldots, \ c_k)$ is weakly decreasing and $c_k = 1$, every $\vec{\mathbf{v}} \in \mathbb{Z}^{k-1}$ has a **unique $\vec{c}$-satisfying representation***.*

- Prove uniqueness.
- Prove existence.

The Carrying and Borrowing Game

How to turn a *non-satisfying representation* into a satisfying one?

The Carrying and Borrowing Game

How to turn a *non-satisfying representation* into a satisfying one?

- **Borrowing:** If a coefficient is too high, decrease it by expanding the recursion.

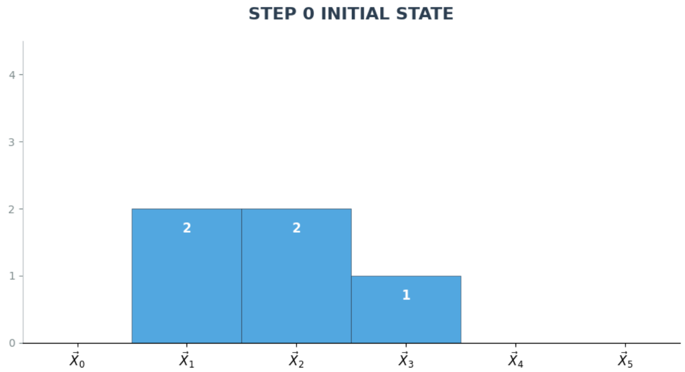
The Carrying and Borrowing Game

How to turn a *non-satisfying representation* into a satisfying one?

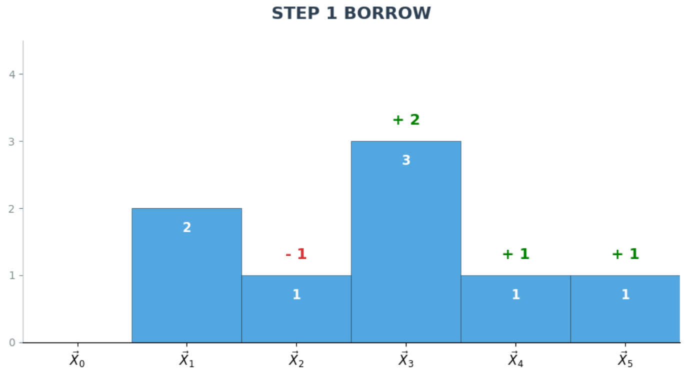
- **Borrowing:** If a coefficient is too high, decrease it by expanding the recursion.
- **Carrying:** If we have a copy of the recursion we can absorb or "condense" it into the previous term.

STEP 0 INITIAL STATE

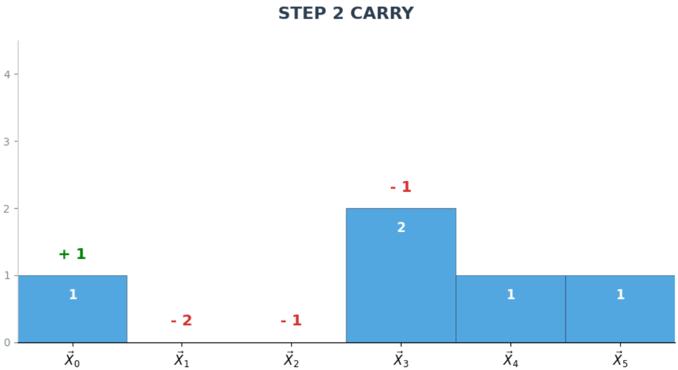
Let $\vec{\mathbf{c}} = (2, 1, 1)$.



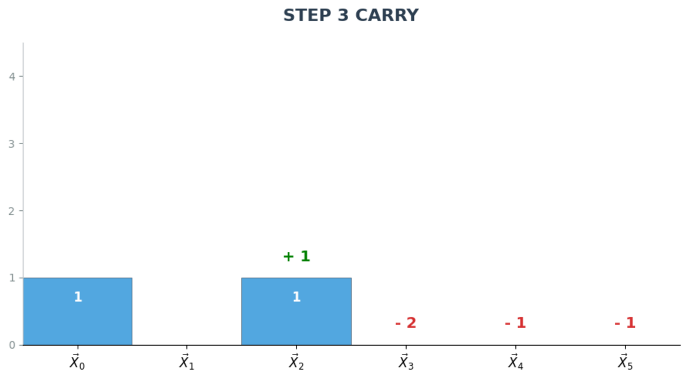
STEP 1 BORROW



STEP 2 CARRY


STEP 2 CARRY
4
3
2
1
0
+ 1
1
- 2
- 1
- 1
2
1
1
$\vec{X}_0$
$\vec{X}_1$
$\vec{X}_2$
$\vec{X}_3$
$\vec{X}_4$
$\vec{X}_5$

STEP 3 CARRY


STEP 3 CARRY
4
3
2
1
0
1
+ 1
1
- 2
- 1
- 1
$\vec{X}_0$
$\vec{X}_1$
$\vec{X}_2$
$\vec{X}_3$
$\vec{X}_4$
$\vec{X}_5$

Algorithm

In general, how do we go from non-satisfying to satisfying?

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

1. Start reading from left to right;

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

1. Start reading from left to right;
2. if you complete a "copy" of the sequence, carry it up;

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

1. Start reading from left to right;
2. if you complete a "copy" of the sequence, carry it up;
3. else, take the first "overfilled" element and borrow from it;

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

1. Start reading from left to right;
2. if you complete a "copy" of the sequence, carry it up;
3. else, take the first "overfilled" element and borrow from it;
4. if the result is still non-satisfying, repeat.

Algorithm

In general, how do we go from non-satisfying to satisfying?
Suppose we have a non-satisfying representation.

1. Start reading from left to right;
2. if you complete a "copy" of the sequence, carry it up;
3. else, take the first "overfilled" element and borrow from it;
4. if the result is still non-satisfying, repeat.

If the algorithm terminates, we get a $\vec{\mathbf{c}}$-satisfying representation.

Algorithm Termination

Does the algorithm always terminate?

Algorithm Termination

Does the algorithm always terminate? NO.

Algorithm Termination

Does the algorithm always terminate? NO.

Example (Not weakly decreasing)

Let $\vec{\mathbf{c}} = (1, 3, 1)$ and consider the representation $0, 2$.

Algorithm Termination

Does the algorithm always terminate? NO.

Example (Not weakly decreasing)

Let $\vec{\mathbf{c}} = (1, 3, 1)$ and consider the representation $0, 2$.

B. $0, 1, 1, 3, 1;$

B. $0, 1, 1, 1, 3, 6, 2;$

C. $0, 1, 2, 0, 0, 5, 2;$

Algorithm Termination

Does the algorithm always terminate? NO.

Example (Not weakly decreasing)

Let $\vec{\mathbf{c}} = (1, 3, 1)$ and consider the representation $0, 2$.

B. $0, 1, 1, 3, 1;$

B. $0, 1, 1, 1, 3, 6, 2;$

C. $0, 1, 2, 0, 0, 5, 2;$

B. $0, 1, 2, 0, 0, 1, 6, 12, 4;$

C. $0, 1, 2, 0, 1, 0, 3, 11, 4;$

...eventually we get $0, 1, 2, 0, 1, 3, 0, 1, 3, 0, 0, 5, 2$.

Algorithm Termination

Does the algorithm always terminate? NO.

Example (Not weakly decreasing)

Let $\vec{\mathbf{c}} = (1, 3, 1)$ and consider the representation $0, 2$.

B. $0, 1, 1, 3, 1;$

B. $0, 1, 1, 1, 3, 6, 2;$

C. $0, 1, 2, 0, 0, 5, 2;$

B. $0, 1, 2, 0, 0, 1, 6, 12, 4;$

C. $0, 1, 2, 0, 1, 0, 3, 11, 4;$

...eventually we get $0, 1, 2, 0, 1, 3, 0, 1, 3, 0, 0, 5, 2$.

Observation: The prefix $0, 1, 2, 0$ and suffix $0, 5, 2$ remain fixed, while the middle gains an extra $1, 3, 0, 1, 3, 0$ block each cycle. The algorithm will therefore loop forever, inserting more and more terms in the middle without ever terminating.

Back to Main Theorem

Theorem (Main Theorem)

If $\vec{\mathbf{c}} = (c_1, c_2, \ \ldots, \ c_k)$ *is weakly decreasing and* $c_k = 1$*, then for every vector* $\vec{\mathbf{u}} \in \mathbb{Z}^{k-1}$ *there is always a representation for which the algorithm terminates.*

Special Properties

Central Limit Type Theorem (Lekkerkerker, 1952)

As $n \to \infty$, the distribution of the number of summands in the Zeckendorf decomposition for $m \in [F_n, F_{n+1})$ is **Gaussian**.

Special Properties

Central Limit Type Theorem (Lekkerkerker, 1952)

As $n \to \infty$, the distribution of the number of summands in the Zeckendorf decomposition for $m \in [F_n, F_{n+1})$ is **Gaussian**.

[Theorem 3.5]

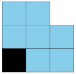
For weakly decreasing $\vec{\mathbf{c}}$ with $c_k = 1$: As $n \to \infty$ the distribution of number of summands in the general Zeckendorf representation for $\vec{\mathbf{v}} \in R_n$ (generalized regions) is **Gaussian**.

Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.

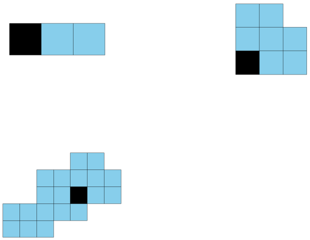
Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.



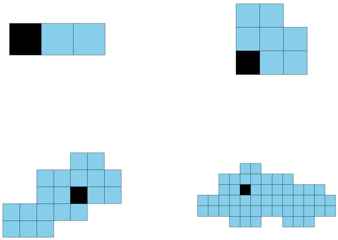
Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.



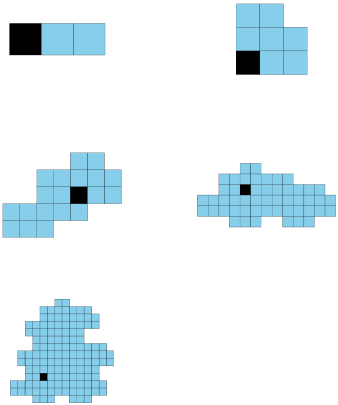
Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.



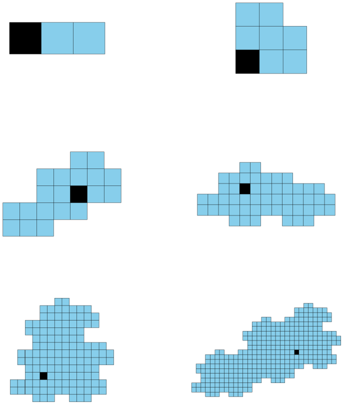
Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.



Illustrations

Illustrations for $\vec{\mathbf{c}} = (2, 1, 1)$.

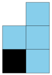


Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).

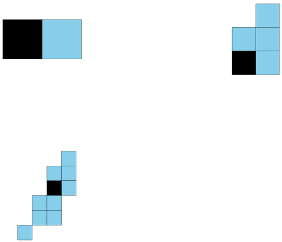
Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).



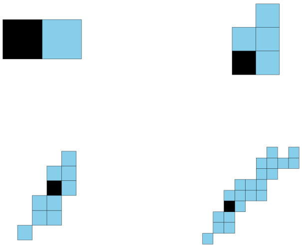
Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).



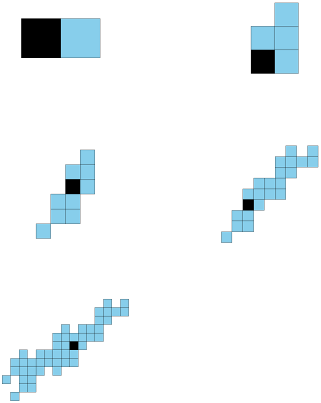
Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).



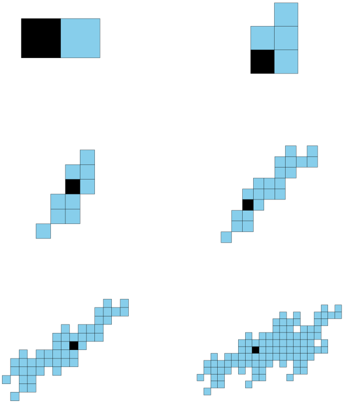
Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).



Illustrations

Illustrations for $\vec{\mathbf{c}} = (1, 2, 1)$ (Not weakly decreasing).



Further Research

1. Which conditions on the coefficient vector $\vec{\mathbf{c}}$ characterize when the "carrying and borrowing game" terminates?

Further Research

1. Which conditions on the coefficient vector $\vec{\mathbf{c}}$ characterize when the "carrying and borrowing game" terminates?

2. How "quickly" do these representations fill the entire space depending on the coefficient vector $\vec{\mathbf{c}}$?

Thank you!

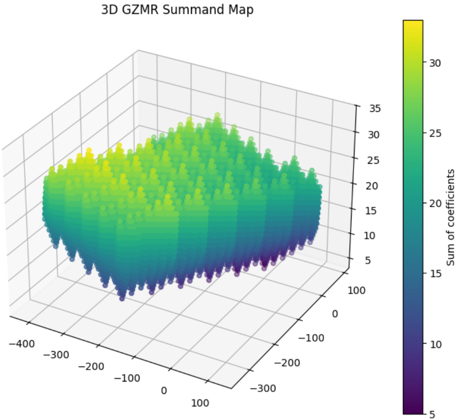
Emails: Tianyu Shen: `1790299332@qq.com`
Steven J. Miller: `sjm1@williams.edu`

Supported by NSF Grant DMS2341670 (Polymath Jr.)

3D illustrations

Let $\vec{\mathbf{c}} = (5, 1, 1)$.

3D illustrations



Let $\vec{\mathbf{c}} = (5, 1, 1)$.