Symbolic Combinatorics

In this talk:
▶ Lattice Walk Counting (∈ Enumerative Combinatorics)
▶ Creative Telescoping (∈ Symbolic Computation)
▶ And what one has to do with the other

In this session:
▶ Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
Symbolic Combinatorics
Symbolic Combinatorics
Symbolic Combinatorics
Symbolic Combinatorics

Symbolic

Enumerative Combinatorics

In this talk:
▶ Lattice Walk Counting ∈ Enumerative Combinatorics
▶ Creative Telescoping ∈ Symbolic Computation
▶ And what one has to do with the other

In this session:
▶ Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
Symbolic Combinatorics
Symbolic Computation
Enumerative Combinatorics
Symbolic Combinatorics

= Symbolic Computation

∪ Enumerative Combinatorics

In this talk:
▶ Lattice Walk Counting ∈ Enumerative Combinatorics
▶ Creative Telescoping ∈ Symbolic Computation
▶ And what one has to do with the other

In this session:
▶ Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
Symbolic Combinatorics
= Symbolic Computation
∩ Enumerative Combinatorics
Symbolic Combinatorics

= Symbolic Computation

+ Enumerative Combinatorics
Symbolic Combinatorics

= Symbolic Computation

+ Enumerative Combinatorics

In this talk:
Symbolic Combinatorics
= Symbolic Computation
+ Enumerative Combinatorics

In this talk:
▶ Lattice Walk Counting
Symbolic Combinatorics
= Symbolic Computation
+ Enumerative Combinatorics

In this talk:
- Lattice Walk Counting ∈ Enumerative Combinatorics

In this session:
- Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
Symbolic Combinatorics
= Symbolic Computation
+ Enumerative Combinatorics

In this talk:
- Lattice Walk Counting ∈ Enumerative Combinatorics
- Creative Telescoping
Symbolic Combinatorics

= Symbolic Computation

+ Enumerative Combinatorics

In this talk:

▶ Lattice Walk Counting ∈ Enumerative Combinatorics
▶ Creative Telescoping ∈ Symbolic Computation

In this session:

▶ Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
Symbolic Combinatorics
= Symbolic Computation
+ Enumerative Combinatorics

In this talk:
- Lattice Walk Counting ∈ Enumerative Combinatorics
- Creative Telescoping ∈ Symbolic Computation
- And what one has to do with the other
Symbolic Combinatorics
\[ = \text{Symbolic Computation} \]
\[ + \text{Enumerative Combinatorics} \]

In this talk:
- Lattice Walk Counting $\in$ Enumerative Combinatorics
- Creative Telescoping $\in$ Symbolic Computation
- And what one has to do with the other

In this session:
Symbolic Combinatorics
\[ = \text{Symbolic Computation} + \text{Enumerative Combinatorics} \]

*In this talk:*
- Lattice Walk Counting ∈ Enumerative Combinatorics
- Creative Telescoping ∈ Symbolic Computation
- And what one has to do with the other

*In this session:*
- Hopefully many other stories on how symbolic computation and enumerative combinatorics fertilize each other.
1. The Combinatorics Part.

Enumeration of Restricted Lattice Walks
Let $a_{n,i,j}$ be the number of walks starting at $(0,0)$ ending at $(i,j)$ consisting of $n$ steps never stepping out of the quarter plane.

Example: $a_{5,0,2} = 200$.

Let $a(t,x,y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$ be the generating function of $a_{n,i,j}$.

Question: What is $a(t,x,y)$?
Let $a_{n,i,j}$ be the number of walks

- starting at $(0,0)$
Let $a_{n,i,j}$ be the number of walks

- starting at $(0,0)$
- ending at $(i,j)$

Example: $a_{5,3,2} = 200$.

Let $a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$ be the generating function of $a_{n,i,j}$. 

Question: What is $a(t, x, y)$?
Let $a_{n,i,j}$ be the number of walks

- starting at $(0,0)$
- ending at $(i,j)$
- consisting of $n$ steps

Example: $a_{5,3,2} = 200$.
Let $a_{n,i,j}$ be the number of walks

- starting at $(0,0)$
- ending at $(i,j)$
- consisting of $n$ steps
- never stepping out of the quarter plane.
Let $a_{n,i,j}$ be the number of walks
  
  - starting at $(0,0)$
  - ending at $(i,j)$
  - consisting of $n$ steps
  - never stepping out of the quarter plane.

Example: $a_{5,3,2} = 200$. 

---

Let $a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$ be the generating function of $a_{n,i,j}$.

Question: What is $a(t, x, y)$?
Let $a_{n,i,j}$ be the number of walks
- starting at $(0, 0)$
- ending at $(i, j)$
- consisting of $n$ steps
- never stepping out of the quarter plane.

**Example:** $a_{5,3,2} = 200$.

Let

$$a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$$

be the *generating function* of $a_{n,i,j}$. 
Let $a_{n,i,j}$ be the number of walks

- starting at $(0, 0)$
- ending at $(i, j)$
- consisting of $n$ steps
- never stepping out of the quarter plane.

*Example:* $a_{5,3,2} = 200$.

Let

$$a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$$

be the *generating function* of $a_{n,i,j}$.
Let $a_{n,i,j}$ be the number of walks

- starting at $(0, 0)$
- ending at $(i, j)$
- consisting of $n$ steps
- never stepping out of the quarter plane.

**Example:** $a_{5,3,2} = 200$.

Let

$$a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$$

be the *generating function* of $a_{n,i,j}$. 
Let $a_{n,i,j}$ be the number of walks

- starting at $(0,0)$
- ending at $(i,j)$
- consisting of $n$ steps
- never stepping out of the quarter plane.

**Example:** $a_{5,3,2} = 200$.

Let

$$a(t, x, y) := \sum_{n=0}^{\infty} \sum_{i,j=0}^{\infty} a_{n,i,j} x^i y^j t^n$$

be the *generating function* of $a_{n,i,j}$.

**Question:** What is $a(t, x, y)$?
Starting point: The combinatorial definition.
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]

\[ + a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \]
\[ + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j}
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j}
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
\begin{align*}
a_{n+1,i,j} &= a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
&+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\end{align*}
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
Starting point: The combinatorial definition.

It immediately implies the *recurrence equation*

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]

which, together with the *boundary conditions*

\[
a_{n,i,-1} = 0 \quad a_{n,-1,j} = 0
\]
**Starting point:** The combinatorial definition.

It immediately implies the *recurrence equation*

\[
a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \\
+ a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1}
\]

which, together with the *boundary conditions*

\[
a_{n,i,-1} = 0 \\
\]

\[
a_{n,-1,j} = 0
\]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \]
\[ + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]

which, together with the boundary conditions

\[ a_{n,i,-1} = 0 \]
\[ a_{n,-1,j} = 0 \]
Starting point: The combinatorial definition.

It immediately implies the recurrence equation

\[ a_{n+1,i,j} = a_{n,i-1,j+1} + a_{n,i,j+1} + a_{n,i+1,j+1} + a_{n,i-1,j} \]
\[ + a_{n,i+1,j} + a_{n,i-1,j-1} + a_{n,i,j-1} + a_{n,i+1,j-1} \]

which, together with the boundary conditions

\[ a_{n,i,-1} = 0 \quad a_{n,-1,j} = 0 \]

and the initial value

\[ a_{0,0,0} = 1 \]

determines all the numbers \( a_{n,i,j} \).
THEN A MIRACLE OCCURS...
It follows for the *generating function* that

\[
a(t, x, y) = \frac{1}{xy} \left[ x^> \right] \left[ y^> \right] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)},
\]
It follows for the *generating function* that

\[ a(t, x, y) = \frac{1}{xy} [x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

where \([x^>][y^>]c(t, x, y)\) refers to the *positive part* of \(c(t, x, y)\):

(This miracle was performed by the combinatorial wizards M. Bousquet-Melou and M. Mishna.)
It follows for the *generating function* that

\[ a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)} , \]

where \([x^>] [y^>] c(t, x, y)\) refers to the *positive part* of \(c(t, x, y)\):

\[ [x^>] [y^>] \sum_{n=0}^{\infty} \left( \sum_{i,j=n}^{n} c_{i,j,n} x^i y^j \right) t^n := \]
It follows for the **generating function** that

\[
a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1)x^{-1})y^{-1}+(x+1x^{-1})+(x+1+1x^{-1})y},
\]

where \([x^>] [y^>] \) refers to the **positive part** of \(c(t, x, y)\):

\[
[x^>] [y^>] \sum_{n=0}^{\infty} \left( \sum_{i,j=-n}^{n} c_{i,j,n} x^i y^j \right) t^n := \sum_{n=0}^{\infty} \left( \sum_{i,j=0}^{n} c_{i,j,n} x^i y^j \right) t^n
\]

(This miracle was performed by the combinatorial wizards M. Bousquet-Melou and M. Mishna.)
It follows for the generating function that

\[
a(t, x, y) = \frac{1}{xy} [x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)},
\]

where \([x^>][y^>]\) refers to the positive part of \(c(t, x, y)\):

\[
[x^>][y^>] \sum_{n=0}^{\infty} \left( \sum_{i,j=-n}^{n} c_{i,j,n} x^i y^j \right) t^n := \sum_{n=0}^{\infty} \left( \sum_{i,j=0}^{n} c_{i,j,n} x^i y^j \right) t^n
\]

This miracle was performed by the combinatorial wizards M. Bousquet-Melou and M. Mishna.
It follows for the \textit{generating function} that

\[
a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)} ,
\]

where \([x^>] [y^>] c(t, x, y)\) refers to the \textit{positive part} of \(c(t, x, y)\):

\[
[x^>] [y^>] \sum_{n=0}^{\infty} \left( \sum_{i,j=-n}^{n} c_{i,j,n} x^i y^j \right) t^n := \sum_{n=0}^{\infty} \left( \sum_{i,j=0}^{n} c_{i,j,n} x^i y^j \right) t^n \in \mathbb{Q}[x,y]
\]
It follows for the *generating function* that

\[
a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1)x^{-1})y^{-1}+(x+x^{-1})+(x+1)x^{-1})y},
\]

where \([x^>] [y^>] c(t, x, y)\) refers to the *positive part* of \(c(t, x, y)\):

\[
[x^>] [y^>] \sum_{n=0}^{\infty} \left( \sum_{i,j=-n}^{n} c_{i,j,n} x^i y^j \right) t^n := \sum_{n=0}^{\infty} \left( \sum_{i,j=0}^{n} c_{i,j,n} x^i y^j \right) t^n
\]

∈ ℚ(x,y) ∈ ℚ[x,y]

(This miracle was performed by the combinatorial wizards M. Bousquet-Melou and M. Mishna.)
It follows for the generating function that

\[ a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

It follows from here that \( a(t, x, y) \) is also equal to the formal residue

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)}, \]

where
It follows for the generating function that

\[ a(t, x, y) = \frac{1}{xy}[x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

It follows from here that \( a(t, x, y) \) is also equal to the formal residue

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)}, \]

where

\[ \text{res}_{u,v} \sum_{a,b,i,j,n} c_{a,b,i,j,n} u^a v^b x^i y^j t^n := \]
It follows for the generating function that

\[ a(t, x, y) = \frac{1}{xy} [x^>] [y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)} , \]

It follows from here that \( a(t, x, y) \) is also equal to the formal residue

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)} , \]

where

\[ \text{res}_{u,v} \sum_{a,b,i,j,n} c_{a,b,i,j,n} u^a v^b x^i y^j t^n := \sum_{i,j,n} c_{-1,-1,i,j,n} x^i y^j t^n \]
It follows for the \textit{generating function} that

\[ a(t, x, y) = \frac{1}{xy}[x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

It follows from here that \( a(t, x, y) \) is also equal to the \textit{formal residue}

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)} , \]

where

\[ \text{res}_{u,v} \sum_{a,b,i,j,n} c_{a,b,i,j,n} u^a v^b x^i y^j t^n := \sum_{i,j,n} c_{-1,-1,i,j,n} x^i y^j t^n \]

\[ \in \mathbb{Q}[u,v,x,y,\frac{1}{u},\frac{1}{v},\frac{1}{x},\frac{1}{y}][[t]] \]
It follows for the *generating function* that

\[
a(t, x, y) = \frac{1}{xy} [x^\geq][y^\geq] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)} ,
\]

It follows from here that \(a(t, x, y)\) is also equal to the *formal residue*

\[
\text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)} ,
\]

where

\[
\text{res}_{u,v} \sum_{a,b,i,j,n} c_{a,b,i,j,n} u^a v^b x^i y^j t^n := \sum_{i,j,n} c_{-1,-1,i,j,n} x^i y^j t^n
\]

\(\in \mathbb{Q}[u,v,x,y, \frac{1}{u}, \frac{1}{v}, \frac{1}{x}, \frac{1}{y}][[t]]\)

\(\in \mathbb{Q}[x,y][[t]]\)
It follows for the \textit{generating function} that

\[
a(t, x, y) = \frac{1}{xy}[x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}.
\]

It follows from here that \( a(t, x, y) \) is also equal to the \textit{formal residue}

\[
\text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)}.
\]

It follows from here that \( a(t, x, y) \) is \textit{D-finite}. 
It follows for the generating function that

\[ a(t, x, y) = \frac{1}{xy} [x^>][y^>] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

It follows from here that \( a(t, x, y) \) is also equal to the formal residue

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)}. \]

It follows from here that \( a(t, x, y) \) is D-finite.

A differential equation can be computed with creative telescoping.
It follows for the *generating function* that

\[ a(t, x, y) = \frac{1}{xy} [x^\rightarrow][y^\rightarrow] \frac{(x-x^{-1})(y-y^{-1})}{1-t((x+1+x^{-1})y^{-1}+(x+x^{-1})+(x+1+x^{-1})y)}, \]

It follows from here that \( a(t, x, y) \) is also equal to the *formal residue*

\[ \text{res}_{u,v} \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)}. \]

It follows from here that \( a(t, x, y) \) is *D-finite*.

A *differential equation* can be computed with *creative telescoping*.

Write

\[ R = \frac{1}{(1-xu)(1-yv)} \frac{(u-u^{-1})(v-v^{-1})}{1-t((u+1+u^{-1})v^{-1}+(u+u^{-1})+(u+1+u^{-1})v)} \]

so that \( a(t, x, y) = \text{res}_{u,v} R. \)
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).
**Observe:** \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

**Therefore:** if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t)D_t + p_2(t, x, y)D_t^2 + \cdots + p_r(t, x, y)D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \)
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

Therefore: if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t) D_t + p_2(t, x, y) D_t^2 + \cdots + p_r(t, x, y) D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]
Observe: $\text{res}_u D_u c(u) = 0$ for every series $c(u)$.

Therefore: if we can find a differential operator

$$P = p_0(t, x, y) + p_1(x, y, t) D_t + p_2(t, x, y) D_t^2 + \cdots + p_r(t, x, y) D_t^r$$

and two rational functions $Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y)$ with

$$P R + D_u Q_1 + D_v Q_2 = 0$$

then

$$\text{res}_{u, v}(P R + D_u Q_1 + D_v Q_2) = 0$$
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

Therefore: if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t) D_t + p_2(t, x, y) D_t^2 + \cdots + p_r(t, x, y) D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]

then

\[
\text{res}_{u,v}(P R) + \text{res}_{u,v}(D_u Q_1) + \text{res}_{u,v}(D_v Q_2) = 0
\]
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

Therefore: if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t) D_t + p_2(t, x, y) D^2_t + \cdots + p_r(t, x, y) D^r_t
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]

then

\[
\text{res}_{u,v}(P R) = 0
\]
**Observe:** \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

**Therefore:** if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t)D_t + p_2(t, x, y)D_t^2 + \cdots + p_r(t, x, y)D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]

then

\[
P \left( \text{res}_{u,v} R \right) = 0
\]
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

Therefore: if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t)D_t + p_2(t, x, y)D_t^2 + \cdots + p_r(t, x, y)D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]

then

\[
P a(t, x, y) = 0
\]
Observe: \( \text{res}_u D_u c(u) = 0 \) for every series \( c(u) \).

Therefore: if we can find a differential operator

\[
P = p_0(t, x, y) + p_1(x, y, t) D_t + p_2(t, x, y) D_t^2 + \cdots + p_r(t, x, y) D_t^r
\]

and two rational functions \( Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y) \) with

\[
P R + D_u Q_1 + D_v Q_2 = 0
\]

then

\[
P a(t, x, y) = 0
\]

Note: Knowing \( P \), we can compute a \textit{closed form} for \( a(t, x, y) \).
Observe: $\text{res}_u D_u c(u) = 0$ for every series $c(u)$.

Therefore: if we can find a differential operator

$$P = p_0(t, x, y) + p_1(x, y, t)D_t + p_2(t, x, y)D_t^2 + \cdots + p_r(t, x, y)D_t^r$$

and two rational functions $Q_1, Q_2 \in \mathbb{Q}(t, u, v, x, y)$ with

$$P R + D_u Q_1 + D_v Q_2 = 0$$

then

$$P a(t, x, y) = 0$$

Note: Knowing $P$, we can compute a closed form for $a(t, x, y)$.

But: Computing $P, Q_1, Q_2$ is quite costly.
Simplify the problem by setting \( x = y = 1 \).
Simplify the problem by setting $x = y = 1$.

Note: The coefficients of $t^n$ in $a(t, 1, 1)$ count the number of walks with $n$ steps and arbitrary endpoint.
Simplify the problem by setting $x = y = 1$.

Note: The coefficients of $t^n$ in $a(t, 1, 1)$ count the number of walks with $n$ steps and arbitrary endpoint.

Integration software (e.g., by C. Koutschan) finds the equation

$$
(t + 1)(2t - 1)(4t + 1)(8t - 1)t^2 D_t^3 a(t, 1, 1)
$$

$$
+ (576t^4 + 200t^3 - 252t^2 - 33t + 5)D_t^2 a(t, 1, 1)
$$

$$
+ (288t^4 + 22t^3 - 117t^2 - 12t + 1)D_t a(t, 1, 1)
$$

$$
+ 12(32t^3 - 6t^2 - 12t - 1)a(t, 1, 1) = 0.
$$
*Simplify* the problem by setting $x = y = 1$.

**Note:** The coefficients of $t^n$ in $a(t, 1, 1)$ count the number of walks with $n$ steps and arbitrary endpoint.

**Integration software** (e.g., by C. Koutschan) finds the equation

$$
(t + 1)(2t - 1)(4t + 1)(8t - 1)t^2 D_t^3 a(t, 1, 1) \\
+ (576t^4 + 200t^3 - 252t^2 - 33t + 5)D_t^2 a(t, 1, 1) \\
+ (288t^4 + 22t^3 - 117t^2 - 12t + 1)D_t a(t, 1, 1) \\
+ 12(32t^3 - 6t^2 - 12t - 1)a(t, 1, 1) = 0.
$$

From here follows the *final result*

$$
a(t, 1, 1) = -\frac{1}{t} \int_t \frac{16t^2 + 24t - 1}{(1 + 4x)^5} \binom{5/4}{2} \binom{5/4}{2} \left| \frac{-2t(t+1)(t-1/8)}{(t+1/4)^4} \right|.
$$
Variation: What happens if we forbid steps into certain directions?
Variation: What happens if we forbid steps into certain directions?

▶ Different step sets lead to different generating functions.
▶ Different generating functions have different algebraic properties.

Kreweras walks: The generating function is algebraic.
Variation: What happens if we forbid steps into certain directions?
Variation: What happens if we forbid steps into certain directions?
**Variation:** What happens if we forbid steps into certain directions?

- Different step sets lead to different generating functions.

![Diagram showing a grid with constraints and directions](image)
**Variation:** What happens if we forbid steps into certain directions?

- Different step sets lead to different generating functions.
- Different generating functions have different algebraic properties.
Variation: What happens if we forbid steps into certain directions?

- Different step sets lead to different generating functions.
- Different generating functions have different algebraic properties.
- *Kreweras walks:* The generating function is algebraic.
**Variation:** What happens if we forbid steps into certain directions?

- Different step sets lead to different generating functions.
- Different generating functions have different algebraic properties.
- *Gessel walks:* The generating function is algebraic.
Variation: What happens if we forbid steps into certain directions?

- Different step sets lead to different generating functions.
- Different generating functions have different algebraic properties.
- **Mishna-Rechnitzer walks:** The generating function is not D-finite.
Variation: What happens if we forbid steps into certain directions?

- **Bousquet-Melou-Mishna classification:** We know for every step set whether the corresponding generating function is algebraic, D-finite transcendental, or not D-finite.
Variation: What happens if we forbid steps into certain directions?

- **Bousquet-Melou-Mishna classification**: We know for every step set whether the corresponding generating function is algebraic, D-finite transcendental, or not D-finite.

- **Our contribution**: For the cases where the generating function is D-finite transcendental, we find an explicit $\binom{2}{1} F_1$ representation.
2. The Computer Algebra Part.

*Fine Tuning Creative Telescoping*
Creative telescoping. (Differential case, one free variable)
**Creative telescoping.** (Differential case, one free variable)

**Given:** a rational function $R(x, y)$
Creative telescoping. (Differential case, one free variable)

**Given:** a rational function \( R(x,y) \)

**Find:** a differential operator \( P \), free of \( y \) and \( D_y \), and a rational function \( Q \) in \( x \) and \( y \) such that

\[
P R + D_y Q = 0.
\]
Creative telescoping. (Differential case, one free variable)

Given: a rational function \( R(x, y) \)
Find: a differential operator \( P \), free of \( y \) and \( D_y \), and a rational function \( Q \) in \( x \) and \( y \) such that

\[
P R + D_y Q = 0.
\]

▶ This is called a creative telescoping relation for \( R \).
Creative telescoping. (Differential case, one free variable)

**Given:** a rational function $R(x, y)$

**Find:** a differential operator $P$, free of $y$ and $D_y$, and a rational function $Q$ in $x$ and $y$ such that

$$P R + D_y Q = 0.$$ 

- This is called a *creative telescoping relation* for $R$.
- The operator $P$ is called its *telescoperc*.
Creative telescoping. (Differential case, one free variable)

**Given**: a rational function \( R(x, y) \)

**Find**: a differential operator \( P \), free of \( y \) and \( D_y \), and a rational function \( Q \) in \( x \) and \( y \) such that

\[ P R + D_y Q = 0. \]

- This is called a *creative telescoping relation* for \( R \).
- The operator \( P \) is called its *telescoper*.
- The rational function \( Q \) is called its *certificate*. 
Creative telescoping. (Differential case, one free variable)

Given: a rational function $R(x, y)$
Find: a differential operator $P$, free of $y$ and $D_y$, and a rational function $Q$ in $x$ and $y$ such that

$$PR + DyQ = 0.$$ 

This is called a creative telescoping relation for $R$.

The operator $P$ is called its telescoper.

The rational function $Q$ is called its certificate.

There are algorithms for computing $(P, Q)$ for given $R$. 
Question: How to make these algorithms faster?
Question: How to make these algorithms faster?
Where are the degrees of freedom?
**Question:** How to make these algorithms faster?
Where are the degrees of freedom?
The solution $(P, Q)$ is *not unique*. 
**Question:** How to make these algorithms faster?

Where are the degrees of freedom?

The solution \((P, Q)\) is *not unique*.

Are some solutions *cheaper* than others?
Question: How to make these algorithms faster?
Where are the degrees of freedom?
The solution \((P, Q)\) is not unique.
Are some solutions cheaper than others?
If so, what is the cheapest?
**Question:** How to make these algorithms faster?
Where are the degrees of freedom?
The solution \((P, Q)\) is *not unique*.
Are some solutions *cheaper* than others?
If so, what is the *cheapest*?
Every telescopeter \(P\) has a certain *order* \(r\) and *degree* \(d\).
**Question:** How to make these algorithms faster?

Where are the degrees of freedom?

The solution \((P, Q)\) is *not unique*.

Are some solutions *cheaper* than others?

If so, what is the *cheapest*?

Every telescopier \(P\) has a certain *order* \(r\) and *degree* \(d\).

**Example:** For

\[
P = (5x^4 - 6x^2 + 5x + 8)D_x^2 + (9x^4 - 10x^3 + 4x^2 + 8)D_x \\
+ (8x^4 + 10x^3 - 8x + 9)
\]

we have \(r = 2\) and \(d = 4\).
**Question:** How to make these algorithms faster?
Where are the degrees of freedom?
The solution \((P, Q)\) is *not unique*.
Are some solutions *cheaper* than others?
If so, what is the *cheapest*?

Every telescooper \(P\) has a certain *order* \(r\) and *degree* \(d\).

**Example:** For

\[
P = (5x^4 - 6x^2 + 5x + 8)D_x^2 + (9x^4 - 10x^3 + 4x^2 + 8)D_x \\
+ (8x^4 + 10x^3 - 8x + 9)
\]

we have \(r = 2\) and \(d = 4\).
**Question:** How to make these algorithms faster?

Where are the degrees of freedom?

The solution \((P, Q)\) is *not unique*.

Are some solutions *cheaper* than others?

If so, what is the *cheapest*?

Every telescoper \(P\) has a certain *order* \(r\) and *degree* \(d\).

**Example:** For

\[
P = (5x^4 - 6x^2 + 5x + 8)D_x^2 + (9x^4 - 10x^3 + 4x^2 + 8)D_x + (8x^4 + 10x^3 - 8x + 9)
\]

we have \(r = 2\) and \(d = 4\).
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescopener of order \(r\) and degree \(d\) exists?
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescoper of order \(r\) and degree \(d\) exists?
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescopener of order \(r\) and degree \(d\) exists?
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescoper of order \(r\) and degree \(d\) exists?
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescoper of order \(r\) and degree \(d\) exists?
For a fixed input, what are the points \((r, d) \in \mathbb{N}^2\) for which a creative telescoping relation with a telescoper of order \(r\) and degree \(d\) exists?
Question: What is the shape of the gray area?
**Question:** What is the shape of the gray area?

**Answer:** We can construct a univariate *hyperbola* which passes approximately along the boundary of the area.
**Question:** What is the shape of the gray area?

**Answer:** We can construct a univariate *hyperbola* which passes approximately along the boundary of the area.
Using this hyperbola, we can choose what we want to compute.

As for computational complexity:

- For small input, the minimal order operator is the cheapest.
- For "industrial size input", operators of [slightly] nonminimal order are cheaper.
- For astronomic input, it is most efficient to compute the operator of order $1 \cdot (1 + \sqrt{17})$ $r_{min}$, where $r_{min}$ is the size of the minimal operator.
Using this hyperbola, we can *choose* what we want to compute.

*Question:* Which point $(r, d)$ is *optimal*?
Using this hyperbola, we can choose what we want to compute.

**Question:** Which point \((r, d)\) is optimal?

**Answer:** Depends on what you want to optimize...
Using this hyperbola, we can *choose* what we want to compute.

**Question:** Which point \((r, d)\) is *optimal*?

**Answer:** Depends on what you want to optimize...
Using this hyperbola, we can *choose* what we want to compute.

**Question:** Which point \((r, d)\) is *optimal*?

**Answer:** Depends on what you want to optimize...
Using this hyperbola, we can \textit{choose} what we want to compute.

\textbf{Question:} Which point \((r, d)\) is \textit{optimal}?

\textbf{Answer:} Depends on what you want to optimize.

As for computational complexity:

\begin{itemize}
\item For small input, the minimal order operator is the cheapest.
\item For "industrial size input", operators of [slightly] nonminimal order are cheaper.
\item For astronomic input, it is most efficient to compute the operator of order \(14 (1 + \sqrt{17}) r_{\min}\), where \(r_{\min}\) is the size of the minimal operator.
\end{itemize}
Using this hyperbola, we can *choose* what we want to compute.

**Question:** Which point \((r, d)\) is *optimal*?

**Answer:** Depends on what you want to optimize...
Using this hyperbola, we can *choose* what we want to compute.

*Question:* Which point \((r, d)\) is *optimal*?

*Answer:* Depends on what you want to optimize...
Using this hyperbola, we can *choose* what we want to compute.

**Question:** Which point \((r, d)\) is *optimal*?

**Answer:** Depends on what you want to optimize.

As for computational complexity:

- For *small input*, the minimal order operator is the cheapest.
Using this hyperbola, we can choose what we want to compute.

**Question:** Which point \((r, d)\) is optimal?

**Answer:** Depends on what you want to optimize.

As for computational complexity:

- For *small input*, the minimal order operator is the cheapest.
- For *“industrial size input”*, operators of [slightly] nonminimal order are cheaper.
Using this hyperbola, we can choose what we want to compute.

**Question:** Which point \((r, d)\) is optimal?

**Answer:** Depends on what you want to optimize...

As for computational complexity:

- For *small input*, the minimal order operator is the cheapest.
- For “*industrial size input*”, operators of [slightly] nonminimal order are cheaper.
- For *astronomic input*, it is most efficient to compute the operator of order \(\frac{1}{4}(1 + \sqrt{17})r_{\text{min}}\), where \(r_{\text{min}}\) is the size of the minimal operator.
3. **Conclusion.**

*Symbolic Computation*

* + *Enumerative Combinatorics*
Combinatorics pushes computer algebra:
Combinatorics pushes computer algebra:

- Complicated expressions arising in combinatorics generate a demand for algorithms for dealing with them.
**Combinatorics pushes computer algebra:**

- Complicated expressions arising in combinatorics generate a demand for algorithms for dealing with them.
- If you really want to compute something, these algorithms should better terminate before your NFS grant.
Combinatorics pushes computer algebra:

- Complicated expressions arising in combinatorics generate a demand for algorithms for dealing with them.
- If you really want to compute something, these algorithms should better terminate before your NFS grant.

Computer algebra pushes combinatorics:
Combinatorics pushes computer algebra:

- Complicated expressions arising in combinatorics generate a demand for algorithms for dealing with them.
- If you really want to compute something, these algorithms should better terminate before your NFS grant.

Computer algebra pushes combinatorics:

- The existence of powerful computational machinery suggests to rephrase a combinatorial problem as input for them.
**Combinatorics** pushes **computer algebra:**

- Complicated expressions arising in combinatorics generate a demand for algorithms for dealing with them.
- If you really want to compute something, these algorithms should better terminate before your NFS grant.

**Computer algebra** pushes **combinatorics:**

- The existence of powerful computational machinery suggests to rephrase a combinatorial problem as input for them.
- Unexpected output may lead to combinatorial insight or raise new questions.