Algebraicity and transcendence of power series: combinatorial and computational aspects

Alin Bostan

Algorithmic and Enumerative Combinatorics
RISC, Hagenberg, August 1–5, 2016
Overview

1. Monday: Context and Examples
2. Tuesday: Properties and Criteria (1)
3. Wednesday: Properties and Criteria (2)
4. Thursday: Algorithmic Proofs of Algebraicity
5. Friday: Transcendence in Lattice Path Combinatorics
Part IV: Algorithmic Proofs of Algebraicity

Alin Bostan

Algebraicity and transcendence of power series
Overview

- **Gessel walks**: walks in \( \mathbb{N}^2 \) using only steps in \( \mathcal{S} = \{\uparrow, \downarrow, \leftarrow, \rightarrow\} \)
- \( g(n; i, j) \) = number of walks from \((0,0)\) to \((i,j)\) with \(n\) steps in \( \mathcal{S} \)

**Question**: Find the nature of the generating function

\[
G(t; x, y) = \sum_{i,j,n=0}^{\infty} g(n; i, j) x^i y^j t^n \in \mathbb{Q}[[x, y, t]]
\]
Overview

- **Gessel walks**: walks in \( \mathbb{N}^2 \) using only steps in \( \mathcal{S} = \{ \rightarrow, \leftarrow, \uparrow, \downarrow \} \)
- \( g(n; i, j) = \) number of walks from \((0,0)\) to \((i, j)\) with \(n\) steps in \(\mathcal{S}\)

**Question**: Find the nature of the generating function

\[
G(t; x, y) = \sum_{i,j,n=0}^{\infty} g(n; i, j) x^i y^j t^n \in \mathbb{Q}[[x, y, t]]
\]

**Theorem** (B.-Kauers 2010) \( G(t; x, y) \) is an algebraic function\(^\dagger\).

→ Effective, computer-driven discovery and proof

\(^\dagger\) Minimal polynomial \( P(x, y, t, G(t; x, y)) = 0 \) has \( > 10^{11} \) terms; \( \approx 30 \text{ Gb} \) (!)
What is “scientific method”? Philosophers and non-philosophers have discussed this question and have not yet finished discussing it. Yet as a first introduction it can be described in three syllables:

Guess and test.

Mathematicians too follow this advice in their research although they sometimes refuse to confess it. They have, however, something which the other scientists cannot really have. For mathematicians the advice is

First guess, then prove.
Personal bias: Experimental Mathematics using Computer Algebra

Alin Bostan

Algebraicity and transcendence of power series
Classification of univariate power series

- **Algebraic**: \( S(t) \in \mathbb{Q}[[t]] \) root of a polynomial \( P \in \mathbb{Q}[t] \), i.e., \( P(t, S(t)) = 0 \).
- **D-finite**: \( S(t) \in \mathbb{Q}[[t]] \) satisfying a linear differential equation with polynomial coefficients \( c_r(t) S^{(r)}(t) + \cdots + c_0(t) S(t) = 0 \).
- **Hypergeometric**: \( S(t) = \sum_{n=0}^{\infty} s_n t^n \) such that \( s_{n+1} s_n \in \mathbb{Q}(n) \). E.g., \( (a)_n = a(a+1) \cdots (a+n-1) \).
Classification of univariate power series

- **Algebraic**: $S(t) \in \mathbb{Q}[[t]]$ root of a polynomial $P \in \mathbb{Q}[t, T]$, i.e., $P(t, S(t)) = 0$. 
- **Hypergeometric**: $S(t) = \sum_{n=0}^{\infty} s_n t^n$ such that $s_{n+1}/s_n \in \mathbb{Q}(n)$. E.g., $(a)_n = a(a+1)\cdots(a+n-1)$. 
- **D-finite**: $S(t) \in \mathbb{Q}[[t]]$ satisfying a linear differential equation with polynomial coefficients.
Classification of univariate power series

▷ **Algebraic:** $S(t) \in \mathbb{Q}[[t]]$ root of a polynomial $P \in \mathbb{Q}[t, T]$, i.e., $P(t, S(t)) = 0$.

▷ **D-finite:** $S(t) \in \mathbb{Q}[[t]]$ satisfying a linear differential equation with polynomial coefficients $c_r(t)S^{(r)}(t) + \cdots + c_0(t)S(t) = 0$. 

▷ **Hypergeometric:** $S(t) = \sum_{n=0}^{\infty} s_n t^n$ such that $s_n + 1 s_n \in \mathbb{Q}(n)$.
Classification of univariate power series

- **Algebraic**: $S(t) \in \mathbb{Q}[[t]]$ root of a polynomial $P \in \mathbb{Q}[t, T]$, i.e., $P(t, S(t)) = 0$.

- **D-finite**: $S(t) \in \mathbb{Q}[[t]]$ satisfying a linear differential equation with polynomial coefficients $c_r(t)S^{(r)}(t) + \cdots + c_0(t)S(t) = 0$.

- **Hypergeometric**: $S(t) = \sum_{n=0}^{\infty} s_nt^n$ such that $\frac{s_{n+1}}{s_n} \in \mathbb{Q}(n)$. E.g.,

  $$2F_1 \left( \begin{array}{c} a \ b \\ c \end{array} \bigg| t \right) = \sum_{n=0}^{\infty} \frac{(a)_n(b)_n}{(c)_n} \frac{t^n}{n!}, \quad (a)_n = a(a+1)\cdots(a+n-1).$$
Classification of univariate power series

- **Algebraic**: $S(t) \in \mathbb{Q}[[t]]$ root of a polynomial $P \in \mathbb{Q}[t, T]$, i.e., $P(t, S(t)) = 0$.

- **D-finite**: $S(t) \in \mathbb{Q}[[t]]$ satisfying a linear differential equation with polynomial coefficients $c_r(t)S^{(r)}(t) + \cdots + c_0(t)S(t) = 0$.

- **Hypergeometric**: $S(t) = \sum_{n=0}^{\infty} s_n t^n$ such that $\frac{s_{n+1}}{s_n} \in \mathbb{Q}(n)$. E.g.,

$$\sum_{n=0}^{\infty} \frac{(a)_n (b)_n (c)_n}{(d)_n (e)_n} \frac{t^n}{n!}, \quad (a)_n = a(a+1) \cdots (a+n-1).$$
S ∈ \mathbb{Q}[[x, y, t]] is \textit{algebraic} if it is the root of a polynomial \( P \in \mathbb{Q}[x, y, t, T] \).
Classification of multivariate power series

\[ S \in \mathbb{Q}[[x, y, t]] \] is **algebraic** if it is the root of a polynomial \( P \in \mathbb{Q}[x, y, t, T] \).

\[ S \in \mathbb{Q}[[x, y, t]] \] is **D-finite** if it satisfies a system of linear partial differential equations with polynomial coefficients

\[
\sum_i a_i(t, x, y) \frac{\partial^i S}{\partial x^i} = 0, \quad \sum_i b_i(t, x, y) \frac{\partial^i S}{\partial y^i} = 0, \quad \sum_i c_i(t, x, y) \frac{\partial^i S}{\partial t^i} = 0.
\]
Gessel’s walks

\[ \mathcal{G} = \{\uparrow, \downarrow, \leftarrow, \rightarrow\} \]
**Conjecture 1** The generating function of Gessel excursions is equal to

\[
G(t; 0, 0) = \binom{5/6}{1/2} \binom{1/2}{2} \binom{1}{16t^2}
\]

\[
= \sum_{n=0}^{\infty} \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} (4t)^{2n}
\]

\[
= 1 + 2t^2 + 11t^4 + 85t^6 + 782t^7 + \cdots
\]

**Conjecture 2**

The full generating function \(G(t; x, y)\) is not D-finite.
Genesis of Gessel’s questions – the “simple walk” in different cones

The simple walk in the plane

[Pólya, 1921]:
▷ Formula $\binom{2n}{n}^2$ for $2n$-excursions
▷ Rational generating function

The simple walk in the half-plane and in the quarter-plane

▷ Formulas $\binom{2n+1}{n}C_n$, resp. $C_nC_{n+1}$, for $2n$-excursions [Arquès, 1986]
The simple walk in the cone with angle $45^\circ$

- Formula $C_nC_{n+2} - C_{n+1}^2$ for $2n$-excursions [Gouyou-Beauchamps, 1986]
- D-finite generating function [Gessel & Zeilberger, 1992]

What about the simple walk in the cone with angle $135^\circ$?
A relative of Gessel walks: Kreweras walks

\[ \mathcal{G} = \{ \downarrow, \leftarrow, \uparrow \} \quad F_{\mathcal{G}}(t; x, y) \equiv K(t; x, y) \]

\[ \mathcal{G} = \{ \uparrow, \swarrow, \leftarrow, \rightarrow \} \quad F_{\mathcal{G}}(t; x, y) \equiv G(t; x, y) \]

Example: A Kreweras excursion.

Alin Bostan
Algebraicity and transcendence of power series
Experimental mathematics –**Guess’n’Prove**– approach:

(S1) Generate data

(S2) Conjecture

(S3) Prove
Methodology for proving algebraicity

Experimental mathematics –Guess’n’Prove– approach:

(S1) **Generate data**
compute a high order expansion of the series $F_{\mathcal{S}}(t; x, y)$;

(S2) **Conjecture**
guess a candidate for the minimal polynomial of $F_{\mathcal{S}}(t; x, y)$, using Hermite-Padé approximation;

(S3) **Prove**
rigorously certify the minimal polynomials, using (exact) polynomial computations.
Methodology for proving algebraicity

Experimental mathematics –*Guess’n’Prove*– approach:

(S1) **Generate data**
   compute a high order expansion of the series $F_{\mathbb{S}}(t; x, y)$;

(S2) **Conjecture**
   guess candidates for minimal polynomials of $F_{\mathbb{S}}(t; x, 0)$ and $F_{\mathbb{S}}(t; 0, y)$, using Hermite-Padé approximation;

(S3) **Prove**
   rigorously certify the minimal polynomials, using (exact) polynomial computations.
Methodology for proving algebraicity

Experimental mathematics –Guess’n’Prove– approach:

(S1) **Generate data**
   compute a **high order expansion** of the series $F_{\mathcal{S}}(t; x, y)$;

(S2) **Conjecture**
   guess candidates for minimal polynomials of $F_{\mathcal{S}}(t; x, 0)$ and $F_{\mathcal{S}}(t; 0, y)$,
   using Hermite-Padé approximation;

(S3) **Prove**
   rigorously certify the minimal polynomials, using (exact) polynomial computations.

+ Efficient Computer Algebra
Step (S1): high order series expansions

\[ f_S(n; i, j) \] satisfies the recurrence with constant coefficients

\[ f_S(n + 1; i, j) = \sum_{(u,v) \in S} f_S(n; i – u, j – v) \quad \text{for} \quad n, i, j \geq 0 \]

+ initial conditions \( f_S(0; i, j) = \delta_{0,i,j} \) and \( f_S(n; -1, j) = f_S(n; i, -1) = 0 \).
Step (S1): high order series expansions

\[ f_{\mathcal{G}}(n; i, j) \] satisfies the recurrence with constant coefficients

\[ f_{\mathcal{G}}(n + 1; i, j) = \sum_{(u, v) \in \mathcal{G}} f_{\mathcal{G}}(n; i - u, j - v) \quad \text{for} \quad n, i, j \geq 0 \]

+ initial conditions \( f_{\mathcal{G}}(0; i, j) = \delta_{0,i,j} \) and \( f_{\mathcal{G}}(n; -1, j) = f_{\mathcal{G}}(n; i, -1) = 0 \).

Example: for the Kreweras model,

\[ k(n + 1; i, j) = k(n; i + 1, j) \]
\[ + k(n; i, j + 1) \]
\[ + k(n; i - 1, j - 1) \]
Step (S1): high order series expansions

\( f_{\mathcal{G}}(n; i, j) \) satisfies the recurrence with constant coefficients

\[
f_{\mathcal{G}}(n+1; i, j) = \sum_{(u,v) \in \mathcal{G}} f_{\mathcal{G}}(n; i - u, j - v) \quad \text{for} \quad n, i, j \geq 0
\]

+ initial conditions \( f_{\mathcal{G}}(0; i, j) = \delta_{0,i,j} \) and \( f_{\mathcal{G}}(n; -1, j) = f_{\mathcal{G}}(n; i, -1) = 0 \).

**Example:** for the Kreweras model,

\[
k(n + 1; i, j) = k(n; i + 1, j) + k(n; i, j + 1) + k(n; i - 1, j - 1)
\]

▶ Recurrence is used to compute \( F_{\mathcal{G}}(t; x, y) \mod t^N \) for large \( N \).

\[
K(t; x, y) = 1 + xyt + (x^2y^2 + y + x)t^2 + (x^3y^3 + 2xy^2 + 2x^2y + 2)t^3
\]
\[
+ (x^4y^4 + 3x^2y^3 + 3x^3y^2 + 2y^2 + 6xy + 2x^2)t^4
\]
\[
+ (x^5y^5 + 4x^3y^4 + 4x^4y^3 + 5xy^3 + 12x^2y^2 + 5x^3y + 8y + 8x)t^5 + \cdots
\]
Step (S2): guessing equations for $F_{\mathcal{G}}(t; x, y)$, a first idea

In terms of generating series, the recurrence on $k(n; i, j)$ reads

\[
(xy - (x + y + x^2 y^2) t) K(t; x, y) \\
= xy - xt K(t; x, 0) - yt K(t; 0, y)
\]  \hspace{1cm} \text{(KerEq)}

▷ A similar kernel equation holds for $F_{\mathcal{G}}(t; x, y)$, for any $\mathcal{G}$-walk.

Corollary. $F_{\mathcal{G}}(t; x, y)$ is algebraic (resp. D-finite) if and only if $F_{\mathcal{G}}(t; x, 0)$ and $F_{\mathcal{G}}(t; 0, y)$ are both algebraic (resp. D-finite).

▷ Crucial simplification: equations for $G(t; x, y)$ are huge ($\approx 30$ Gb)
Step (S2): guessing equations for $F_S(t; x, 0)$ & $F_S(t; 0, y)$

Task 1: Given the first $N$ terms of $S = F_S(t; x, 0) \in \mathbb{Q}[x][[t]]$, search for a differential equation satisfied by $S$ at precision $N$:

$$c_r(x, t) \cdot \frac{\partial^r S}{\partial t^r} + \cdots + c_1(x, t) \cdot \frac{\partial S}{\partial t} + c_0(x, t) \cdot S = 0 \mod t^N.$$
Step (S2): guessing equations for $F_{\mathcal{G}}(t; x, 0)$ & $F_{\mathcal{G}}(t; 0, y)$

**Task 1:** Given the first $N$ terms of $S = F_{\mathcal{G}}(t; x, 0) \in \mathbb{Q}[x][[t]]$, search for a differential equation satisfied by $S$ at precision $N$:

$$c_r(x, t) \cdot \frac{\partial^r S}{\partial t^r} + \cdots + c_1(x, t) \cdot \frac{\partial S}{\partial t} + c_0(x, t) \cdot S = 0 \mod t^N.$$ 

**Task 2:** Search for an algebraic equation $\mathcal{P}_{x,0}(S) = 0 \mod t^N$. 

In practice, we use modular Hermite-Padé approximation (Beckermann-Labahn algorithm) combined with (rational) evaluation-interpolation and rational number reconstruction. Fast (FFT-based) arithmetic in $\mathbb{F}_p[[t]]$. 

Alin Bostan
Algebraicity and transcendence of power series
Step (S2): guessing equations for $F_S(t; x, 0) & F_S(t; 0, y)$

**Task 1:** Given the first $N$ terms of $S = F_S(t; x, 0) \in \mathbb{Q}[x][[t]]$, search for a differential equation satisfied by $S$ at precision $N$: 

$$c_r(x, t) \cdot \frac{\partial^r S}{\partial t^r} + \cdots + c_1(x, t) \cdot \frac{\partial S}{\partial t} + c_0(x, t) \cdot S = 0 \mod t^N.$$ 

**Task 2:** Search for an algebraic equation $\mathcal{P}_{x,0}(S) = 0 \mod t^N$.

- Both tasks amount to linear algebra in size $N$ over $\mathbb{Q}(x)$.
- In practice, we use modular Hermite-Padé approximation (Beckermann-Labahn algorithm) combined with (rational) evaluation-interpolation and rational number reconstruction.
- Fast (FFT-based) arithmetic in $\mathbb{F}_p[t]$. 

Alin Bostan
Algebraicity and transcendence of power series
Step (S2): guessing equations for $K(t; x, 0)$

Using $N = 80$ terms of $K(t; x, 0)$, one can guess

▷ a linear differential equation of order 4, degrees $(14, 11)$ in $(t, x)$, such that

$$
t^3 \cdot (3t - 1) \cdot (9t^2 + 3t + 1) \cdot (3t^2 + 24t^2 x^3 - 3xt - 2x^2) \cdot \\
(16t^2 x^5 + 4x^4 - 72t^4 x^3 - 18x^3 t + 5t^2 x^2 + 18xt^3 - 9t^4) \cdot \\
(4t^2 x^3 - t^2 + 2xt - x^2) \cdot \frac{\partial^4 K(t; x, 0)}{\partial t^4} + \cdots
$$

$$= 0 \text{ mod } t^{80}$$

▷ a polynomial of tridegree $(6, 10, 6)$ in $(T, t, x)$

$$\mathcal{P}_{x, 0} = x^6 t^{10} T^6 - 3x^4 t^8 (x - 2t) T^5 + \\
+ x^2 t^6 \left(12t^2 + 3t^2 x^3 - 12xt + \frac{7}{2} x^2\right) T^4 + \cdots$$

such that $\mathcal{P}_{x, 0}(K(t; x, 0), t, x) = 0 \text{ mod } t^{80}$.
Step (S2): guessing equations for $G(t; x, 0)$ and $G(t; 0, y)$

Using $N = 1200$ terms of $G(t; x, y)$, our guesser found candidates

- $P_{x,0}$ in $\mathbb{Z}[x, t, T]$ of degree $(32, 43, 24)$, coefficients of 21 digits
- $P_{0,y}$ in $\mathbb{Z}[y, t, T]$ of degree $(40, 44, 24)$, coefficients of 23 digits

such that

$$P_{x,0}(x, t, G(t; x, 0)) = P_{0,y}(y, t, G(t; 0, y)) = 0 \mod t^{1200}.$$
Using $N = 1200$ terms of $G(t; x, y)$, our guesser found candidates

- $\mathcal{P}_{x,0}$ in $\mathbb{Z}[x, t, T]$ of degree $(32, 43, 24)$, coefficients of 21 digits
- $\mathcal{P}_{0,y}$ in $\mathbb{Z}[y, t, T]$ of degree $(40, 44, 24)$, coefficients of 23 digits

such that

$$\mathcal{P}_{x,0}(x, t, G(t; x, 0)) = \mathcal{P}_{0,y}(y, t, G(t; 0, y)) = 0 \mod t^{1200}.$$ 

▶ Guessing $\mathcal{P}_{x,0}$ by undetermined coefficients would have required to solve a dense linear system of size $\approx 100\,000$, and $\approx 1000$ digits entries!
Step (S2): guessing equations for $G(t; x, 0)$ and $G(t; 0, y)$

Using $N = 1200$ terms of $G(t; x, y)$, our guesser found candidates
- $\mathcal{P}_{x,0}$ in $\mathbb{Z}[x, t, T]$ of degree $(32, 43, 24)$, coefficients of 21 digits
- $\mathcal{P}_{0,y}$ in $\mathbb{Z}[y, t, T]$ of degree $(40, 44, 24)$, coefficients of 23 digits

such that

$$\mathcal{P}_{x,0}(x, t, G(t; x, 0)) = \mathcal{P}_{0,y}(y, t, G(t; 0, y)) = 0 \mod t^{1200}.$$

Guessing $\mathcal{P}_{x,0}$ by undetermined coefficients would have required to solve a dense linear system of size $\approx 100\,000$, and $\approx 1000$ digits entries!

We actually first guessed differential equations†, then computed their $p$-curvatures to empirically certify them. This led us suspect the algebraicity of $G(t; x, 0)$ and $G(t; 0, y)$, using Grothendieck’s conjecture as an oracle.

† of order 11, and bidegree $(96, 78)$ for $G(t; x, 0)$, and $(68, 28)$ for $G(t; 0, y)$
Guessing is good, proving is better [Pólya, 1957]

George Pólya

Guessing and Proving

Guessing is good, proving is better.
Step (S3): warm-up – **Gessel excursions** are algebraic

Theorem. \( g(t) := G(\sqrt{t}; 0, 0) = \sum_{n=0}^{\infty} \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} (16t)^n \) is algebraic.
Step (S3): warm-up – Gessel excursions are algebraic

Theorem. $g(t) := G(\sqrt{t}; 0, 0) = \sum_{n=0}^{\infty} \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} (16t)^n$ is algebraic.

Proof: First guess a polynomial $P(t, T)$ in $\mathbb{Q}[t, T]$, then prove that $P$ admits the power series $g(t) = \sum_{n=0}^{\infty} g_n t^n$ as a root.
Step (S3): warm-up – Gessel excursions are algebraic

Theorem.  \( g(t) := G(\sqrt{t}; 0, 0) = \sum_{n=0}^{\infty} \frac{(5/6)_n(1/2)_n}{(5/3)_n(2)_n} (16t)^n \) is algebraic.

Proof: First guess a polynomial \( P(t, T) \) in \( \mathbb{Q}[t, T] \), then prove that \( P \) admits the power series \( g(t) = \sum_{n=0}^{\infty} g_n t^n \) as a root.

1. Such a \( P \) can be guessed from the first 100 terms of \( g(t) \).
Step (S3): warm-up – **Gessel excursions** are algebraic

**Theorem.** \( g(t) := G(\sqrt{t}; 0, 0) = \sum_{n=0}^{\infty} \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} (16t)^n \) is algebraic.

**Proof:** First guess a polynomial \( P(t, T) \) in \( \mathbb{Q}[t, T] \), then prove that \( P \) admits the power series \( g(t) = \sum_{n=0}^{\infty} g_n t^n \) as a root.

1. Such a \( P \) can be **guessed** from the first 100 terms of \( g(t) \).
2. **Implicit function theorem:** \( \exists! \) root \( r(t) \in \mathbb{Q}[[t]] \) of \( P \).
Theorem. \( g(t) := G(\sqrt{t}; 0, 0) = \sum_{n=0}^{\infty} \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} (16t)^n \) is algebraic.

Proof: First guess a polynomial \( P(t, T) \) in \( \mathbb{Q}[t, T] \), then prove that \( P \) admits the power series \( g(t) = \sum_{n=0}^{\infty} g_n t^n \) as a root.

1. Such a \( P \) can be guessed from the first 100 terms of \( g(t) \).

2. Implicit function theorem: \( \exists! \) root \( r(t) \in \mathbb{Q}[[t]] \) of \( P \).

3. \( r(t) = \sum_{n=0}^{\infty} r_n t^n \) being algebraic, it is D-finite, and so is \( (r_n) \):

   \[ (n + 2)(3n + 5)r_{n+1} - 4(6n + 5)(2n + 1)r_n = 0, \quad r_0 = 1 \]

   \[ \Rightarrow \text{solution } r_n = \frac{(5/6)_n (1/2)_n}{(5/3)_n (2)_n} 16^n = g_n, \text{ thus } g(t) = r(t) \text{ is algebraic.} \]
Step (S3): rigorous proof for Kreheras walks

1. Setting \( y_0 = \frac{x-t-\sqrt{x^2-2tx+t^2(1-4x^3)}}{2tx^2} = t + \frac{1}{x} t^2 + \frac{x^3+1}{x^2} t^3 + \cdots \) in the kernel equation

\[
(xy - (x + y + x^2y^2)t)K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; 0, y)
\]

\( \neq 0 \)
Step (S3): rigorous proof for Kreneras walks

1. Setting $y_0 = \frac{x - t - \sqrt{x^2 - 2tx + t^2(1 - 4x^3)}}{2tx^2} = t + \frac{1}{x}t^2 + \frac{x^3 + 1}{x^2}t^3 + \cdots$ in the kernel equation

\[
(xy - (x + y + x^2y^2)t)K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0)
\]

\[
! = 0
\]
Step (S3): rigorous proof for Kreweras walks

1. Setting 
   \[ y_0 = \frac{x-t-\sqrt{x^2-2tx+t^2(1+4x^3)}}{2tx^2} = t + \frac{1}{x}t^2 + \frac{x^3+1}{x^2}t^3 + \cdots \]  
   in the kernel equation

\[ (xy - (x + y + x^2y^2)t)K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0) \]

shows that \( U = K(t; x, 0) \) satisfies the reduced kernel equation

\[ 0 = x \cdot y_0 - x \cdot t \cdot U(t, x) - y_0 \cdot t \cdot U(t, y_0) \]  

(RKerEq)
Step (S3): rigorous proof for Kreheras walks

1. Setting \( y_0 = \frac{x - t - \sqrt{x^2 - 2tx + t^2(1 - 4x^3)}}{2tx^2} = t + \frac{1}{x} t^2 + \frac{x^3 + 1}{x^2} t^3 + \cdots \) in the kernel equation

\[
(xy - (x + y + x^2y^2)t)K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0)
\]

shows that \( U = K(t; x, 0) \) satisfies the reduced kernel equation

\[
0 = x \cdot y_0 - x \cdot t \cdot U(t, x) - y_0 \cdot t \cdot U(t, y_0)
\]  \hspace{1cm} (RKerEq)

2. \( U = K(t; x, 0) \) is the unique solution in \( \mathbb{Q}[[x, t]] \) of (RKerEq).
Step (S3): rigorous proof for Kreweras walks

1. Setting $y_0 = \frac{x-t-\sqrt{x^2-2tx+t^2(1-4x^3)}}{2tx^2} = t + \frac{1}{x} t^2 + \frac{x^3+1}{x^2} t^3 + \cdots$ in the kernel equation

\[
(xy - (x + y + x^2y^2)t) K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0)
\]

shows that $U = K(t; x, 0)$ satisfies the reduced kernel equation

\[
0 = x \cdot y_0 - x \cdot t \cdot U(t, x) - y_0 \cdot t \cdot U(t, y_0) \tag{R KerEq}
\]

2. $U = K(t; x, 0)$ is the unique solution in $\mathbb{Q}[[x,t]]$ of (R KerEq).

3. The guessed candidate $\mathcal{P}_{x,0}$ has one solution $H(t, x)$ in $\mathbb{Q}[[x,t]]$. 
Step (S3): rigorous proof for Kremeras walks

1. Setting $y_0 = \frac{x-t-\sqrt{x^2-2tx+t^2(1-4x^3)}}{2tx^2} = t + \frac{1}{x} t^2 + \frac{x^3+1}{x^2} t^3 + \cdots$ in the kernel equation

$$\left( xy - (x + y + x^2y^2)t \right) K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0)$$

shows that $U = K(t; x, 0)$ satisfies the reduced kernel equation

$$0 = x \cdot y_0 - x \cdot t \cdot U(t, x) - y_0 \cdot t \cdot U(t, y_0)$$

(RKerEq)

2. $U = K(t; x, 0)$ is the unique solution in $\mathbb{Q}[[x, t]]$ of (RKerEq).

3. The guessed candidate $P_{x,0}$ has one solution $H(t, x)$ in $\mathbb{Q}[[x, t]]$.

4. Resultant computations + verification of initial terms

$$\implies U = H(t, x)$$

also satisfies (RKerEq).
Step (S3): rigorous proof for Kreheras walks

1. Setting $y_0 = \frac{x-t-\sqrt{x^2-2tx+t^2(1-4x^3)}}{2tx^2} = t + \frac{1}{x}t^2 + \frac{x^3+1}{x^2}t^3 + \cdots$ in the kernel equation

\[
(xy - (x + y + x^2y^2)t)K(t; x, y) = xy - xtK(t; x, 0) - ytK(t; y, 0) \neq 0
\]

shows that $U = K(t; x, 0)$ satisfies the reduced kernel equation

\[
0 = x \cdot y_0 - x \cdot t \cdot U(t, x) - y_0 \cdot t \cdot U(t, y_0)
\]  (RKerEq)

2. $U = K(t; x, 0)$ is the unique solution in $\mathbb{Q}[[x, t]]$ of (RKerEq).

3. The guessed candidate $P_{x,0}$ has one solution $H(t, x)$ in $\mathbb{Q}[[x, t]]$.

4. Resultant computations + verification of initial terms

\[
\Longrightarrow \quad U = H(t, x) \text{ also satisfies (RKerEq).}
\]

5. Uniqueness: $H(t, x) = K(t; x, 0) \implies K(t; x, 0)$ is algebraic!
Algebraicity of Kreweras walks: a computer proof in a nutshell

[bostan@inria ~]$ maple
 |\═|
 Maple 19 (APPLE UNIVERSAL OSX)
 .\_____/\ |\_\|\_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2014
 \ MAPLE / All rights reserved. Maple is a trademark of
 <____ ____> Waterloo Maple Inc.
 |
 Type ? for help.

# HIGH ORDER EXPANSION (S1)
> st, bu := time(), kernelopts(bytesused):
> f := proc(n, i, j)
  option remember;
  if i < 0 or j < 0 or n < 0 then 0
  elif n = 0 then if i = 0 and j = 0 then 1 else 0 fi
  else f(n-1, i-1, j-1) + f(n-1, i, j+1) + f(n-1, i+1, j) fi
end:
> S := series(add(add(f(k, i, 0)*x^i, i = 0..k)*t^k, k = 0..80), t, 80):

# GUESSING (S2)
> libname := ".", libname: gfun:-version();
     3.62
> gfun:-seriestoalgeq(S, Fx(t)):
> P := collect(numer(subs(Fx(t) = T, %[1])), T):

# RIGOROUS PROOF (S3)
> ker := (T, t, x) -> (x+T+x^2*T^2)*t-x*T:
> pol := unapply(P, T, t, x):
> p1 := resultant(pol(z-T, t, x), ker(t*z, t, x), z):
> p2 := subs(T=x*T, resultant(numer(pol(T/z, t, z)), ker(z, t, x), z)):
> normal(primpart(p1, T)/primpart(p2, T));

# time (in sec) and memory consumption (in Mb)
> trunc(time()-st), trunc((kernelopts(bytesused)-bu)/1000^2);

7, 617
Step (S3): rigorous proof for Gessel walks

Same strategy, but several complications:
- stepset diagonal symmetry is lost: \( G(t; x, y) \neq G(t; y, x) \);
- \( G(t; 0, 0) \) occurs in (KerEq) (because of the step \( \downarrow \));
- equations are \( \approx 5000 \) times bigger.

\[ \rightarrow \text{replace equation (RKerEq) by a system of 2 reduced kernel equations.} \]

\[ \rightarrow \text{fast algorithms needed (e.g., [B., Flajolet, Salvy \& Schost 2006] for computations with algebraic series).} \]
Guess’n’Prove is a powerful method, especially when combined with efficient computer algebra.

It is robust: it can be used to uniformly prove algebraicity.

Brute-force and/or use of naive algorithms = hopeless. E.g. size of algebraic equations for $G(t; x, y) \approx 30$ Gb.
INSIDE THE BOX

–Hermite-Padé approximants–
Definition: Given a column vector $\mathbf{F} = (f_1, \ldots, f_n)^T \in \mathbb{Q}[[x]]^n$ and an $n$-tuple $\mathbf{d} = (d_1, \ldots, d_n) \in \mathbb{N}^n$, a Hermite-Padé approximant of type $\mathbf{d}$ for $\mathbf{F}$ is a row vector $\mathbf{P} = (P_1, \ldots, P_n) \in \mathbb{Q}[x]^n$, ($\mathbf{P} \neq 0$), such that:

1. $\mathbf{P} \cdot \mathbf{F} = P_1 f_1 + \cdots + P_n f_n = O(x^{\sigma})$ with $\sigma = \sum_i (d_i + 1) - 1$,
2. $\deg(P_i) \leq d_i$ for all $i$.

$\sigma$ is called the order of the approximant $\mathbf{P}$.

▶ Very useful concept in number theory (irrationality/transcendence):

- [Hermite 1873]: $e$ is transcendent.
- [Lindemann 1882]: $\pi$ is transcendent; so does $e^\alpha$ for any $\alpha \in \overline{\mathbb{Q}} \setminus \{0\}$.
- [Apéry 1978, Beukers 1981]: $\zeta(3) = \sum_{n \geq 1} \frac{1}{n^3}$ is irrational.
- [Rivoal 2000]: there exist infinite values of $k$ such that $\zeta(2k + 1) \notin \mathbb{Q}$. 


Let us compute a Hermite-Padé approximant of type $(1, 1, 1)$ for $(1, C, C^2)$, where \( C(x) = 1 + x + 2x^2 + 5x^3 + 14x^4 + 42x^5 + O(x^6) \).

This boils down to finding \( \alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1 \) (not all zero) such that

\[
\alpha_0 + \alpha_1 x + (\beta_0 + \beta_1 x)(1 + x + 2x^2 + 5x^3 + 14x^4) + (\gamma_0 + \gamma_1 x)(1 + 2x + 5x^2 + 14x^3 + 42x^4) = O(x^5)
\]

Identifying coefficients, this is equivalent to a homogeneous linear system:

\[
\begin{bmatrix}
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 1 & 2 & 1 \\
0 & 0 & 2 & 1 & 5 & 2 \\
0 & 0 & 5 & 2 & 14 & 5 \\
0 & 0 & 14 & 5 & 42 & 14
\end{bmatrix}
\begin{bmatrix}
\alpha_0 \\
\alpha_1 \\
\beta_0 \\
\beta_1 \\
\gamma_0 \\
\gamma_1
\end{bmatrix}
= 0 \iff
\begin{bmatrix}
1 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 1 & 2 \\
0 & 0 & 2 & 1 & 5 \\
0 & 0 & 5 & 2 & 14 \\
0 & 0 & 14 & 5 & 42
\end{bmatrix}
\begin{bmatrix}
\alpha_0 \\
\alpha_1 \\
\beta_0 \\
\beta_1 \\
\gamma_0 \\
\gamma_1
\end{bmatrix}
= -\gamma_1
\begin{bmatrix}
1 \\
2 \\
5 \\
14
\end{bmatrix}.
\]

By homogeneity, one can choose \( \gamma_1 = 1 \).

Then, the violet minor shows that one can take
\( (\beta_0, \beta_1, \gamma_0) = (-1, 0, 0) \).

The other values are \( \alpha_0 = 1, \alpha_1 = 0 \).

Thus the approximant is \((1, -1, x)\), which corresponds to
\( P = 1 - y + xy^2 \)
such that \( P(x, C(x)) = 0 \mod x^5 \).
Hermite-Padé approximants of \( n = 2 \) power series are related to Padé approximants, i.e. to approximation of series by rational functions.

**Algebraic approximants** = Hermite-Padé approximants for \( f_\ell = A^{\ell-1} \), where \( A \in \mathbb{Q}[[x]] \)

**Differential approximants** = Hermite-Padé approximants for \( f_\ell = A^{(\ell-1)} \), where \( A \in \mathbb{Q}[[x]] \)

\[
\begin{align*}
> \text{listtoalgeq}([1,1,2,5,14,42,132,429],y(x)); \\
&\quad 2 \\
&\quad [1 - y(x) + x y(x) , \text{ogf}] \\
> \text{listtodiffeq}([1,1,2,5,14,42,132,429],y(x)); \\
&\quad / 2 \\
&\quad /d \quad \backslash \quad |d | \\
&\quad [{-2 y(x) + (2 - 4 x) |-- y(x)| + x |--- y(x)|} , y(0) = 1, D(y)(0) = 1] , \text{egf}] \\
&\quad \backslash dx \quad / \quad | 2 | \\
&\quad \backslash dx \quad / 
\end{align*}
\]
Existence and naive computation

**Theorem** For any vector \( F = (f_1, \ldots, f_n)^T \in \mathbb{Q}[[x]]^n \) and for any \( n \)-tuple \( d = (d_1, \ldots, d_n) \in \mathbb{N}^n \), there exists a Hermite-Padé approx. of type \( d \) for \( F \).

**Proof**: The undetermined coefficients of \( P_i = \sum_{j=0}^{d_i} p_{i,j} x^j \) satisfy a linear homogeneous system with \( \sigma = \sum_i (d_i + 1) - 1 \) eqs and \( \sigma + 1 \) unknowns.

**Corollary** Computation in \( O(\sigma^\omega) \), for \( 2 \leq \omega \leq 3 \) (linear algebra exponent)

▷ There are better algorithms (the linear system is structured, Sylvester-like):
- Derksen’s algorithm (Gaussian-like elimination) \( O(\sigma^2) \)
- Beckermann-Labahn’s algorithm (DAC) \( \tilde{O}(\sigma) = O(\sigma \log^2 \sigma) \)
Theorem [Beckermann, Labahn, 1994] One can compute a Hermite-Padé approximant of type \((d, \ldots, d)\) for \(F = (f_1, \ldots, f_n)\) in \(\tilde{O}(n^\omega d)\) ops. in \(Q\)

Ideas:
- Compute a whole matrix of approximants
- Exploit divide-and-conquer

Algorithm:
1. If \(\sigma = n(d + 1) - 1 \leq \text{threshold}\), call the naive algorithm
2. Else:
   1. recursively compute \(P_1 \in \mathbb{Q}[x]^{n \times n}\) s.t. \(P_1 \cdot F = O(x^{\sigma/2}), \deg(P_1) \approx \frac{d}{2}\)
   2. compute “residue” \(R\) such that \(P_1 \cdot F = x^{\sigma/2} \cdot (R + O(x^{\sigma/2}))\)
   3. recursively compute \(P_2 \in \mathbb{Q}[x]^{n \times n}\) s.t. \(P_2 \cdot R = O(x^{\sigma/2}), \deg(P_2) \approx \frac{d}{2}\)
   4. return \(P := P_2 \cdot P_1\)

▷ The precise choices of degrees is a delicate issue
▷ Corollary: Gcd, extended gcd, Padé approximants in \(\tilde{O}(d)\)
INSIDE THE BOX

–Special resultants–
Any polynomial \( F = x^n + a_1 x^{n-1} + \cdots + a_n \) in \( \mathbb{Q}[x] \) can be represented by its first \( n \) power sums \( S_i = \sum_{F(\alpha)=0} \alpha^i \)

Conversions coefficients ↔ power sums can be performed
- either in \( O(n^2) \) using Newton identities (naive way):
  \[ ia_i + S_1 a_{i-1} + \cdots + S_i = 0, \quad 1 \leq i \leq n \]
- or in \( \tilde{O}(n) \) using generating series
  \[ \frac{\text{rev}(F)'}{\text{rev}(F)} = - \sum_{i \geq 0} S_{i+1} x^i \iff \text{rev}(F) = \exp \left( - \sum_{i \geq 1} \frac{S_i}{i} x^i \right) \]
Special bivariate resultants \cite{BostanFlajoletSalvySchost2006}

Manipulation of algebraic numbers: composed products and sums

\[ F \otimes G = \prod_{F(\alpha)=0,G(\beta)=0} (x - \alpha \beta), \quad F \oplus G = \prod_{F(\alpha)=0,G(\beta)=0} (x - (\alpha + \beta)) \]

Output size:

\[ N = \deg(F) \deg(G) \]

Linear algebra:

\[ \chi_{xy}, \chi_{x+y} \text{ in } \mathbb{Q}[x,y]/(F(x), G(y)) \]

Resultants:

\[ \text{Res}_y \left( F(y), y^{\deg(G)} G(x/y) \right), \text{Res}_y \left( F(y), G(x-y) \right) \]

Better: \( \otimes \) and \( \oplus \) are easy in Newton representation

\[ \sum \alpha^s \sum \beta^s = \sum (\alpha \beta)^s \quad \text{and} \quad \sum \frac{\sum (\alpha + \beta)^s}{s!} x^s = \left( \sum \frac{\sum \alpha^s}{s!} x^s \right) \left( \sum \frac{\sum \beta^s}{s!} x^s \right) \]
Newton’s tangent method: real case [Newton, 1671]

\[ x_{\kappa+1} = N(x_\kappa) = x_\kappa - \frac{x_\kappa^2 - 2}{2x_\kappa}, \quad x_0 = 1 \]

- \( x_1 = 1.50000000000000000000000000000000 \)
- \( x_2 = 1.4166666666666666666666666666667 \)
- \( x_3 = 1.4142156862745098039215686274510 \)
- \( x_4 = 1.4142135623746899106262955788901 \)
- \( x_5 = 1.4142135623730950488016896235025 \)
Newton’s tangent method: power series case

In order to solve $\varphi(x, g) = 0$ in $\mathbb{Q}[[x]]$ iterate

$$g_{\kappa+1} = g_\kappa - \frac{\varphi(g_\kappa)}{\varphi_y(g_\kappa)} \mod x^{2\kappa+1}$$

▷ The number of correct coefficients doubles after each iteration
▷ Total cost \(= 2 \times \left(\text{the cost of the last iteration}\right)\)

Theorem [Cook 1966, Sieveking 1972 & Kung 1974, Brent 1975]
Division, logarithm and exponential of power series in $\mathbb{Q}[[x]]$ can be computed at precision $N$ using $\tilde{O}(N)$ operations in $\mathbb{Q}$
In practice

Alin Bostan

Algebraicity and transcendence of power series
Thanks for your attention!