Efficient Algorithms in Computer Algebra
(2021–2022)

Frédéric Chyzak
frederic.chyzak@inria.fr

October 21, 2021

Chapters “Euclidean division, etc.” and “Linear recurrences with polynomial coefficients”
Part I

Last time’s exercises
Exercise 1

1. Describe a Newton iteration that directly computes a square root, without appealing to successive logarithm and exponential computations.

2. Estimate the complexity of this algorithm.
Exercise 1

1. Describe a Newton iteration that directly computes a square root, without appealing to successive logarithm and exponential computations.

2. Estimate the complexity of this algorithm.

\[ \Phi(F, G) := G^2 - F \] to get \( G = \sqrt{F} \) provides \( N(G) = (G + F/G)/2 \).

If \( G = \sqrt{F} + O(X^n) \), then \( \exists H, F = G^2(1 + X^n H) \), so \( N(G) = G + \frac{1}{2}GX^n H \).

Next, \( \sqrt{F} = G(1 + \frac{1}{2}X^n H + O(X^{2n})) \), so \( N(G) = \sqrt{F} + O(X^{2n}) \).

Computationally: given \( F = T + O(X^N) \), recursively compute \( \sqrt{F} = U + O(X^{N/2}) \), then return \( U + \text{rem}(T/U, X^N)/2 \).
Exercise 1

1. Describe a Newton iteration that directly computes a square root, without appealing to successive logarithm and exponential computations.

2. Estimate the complexity of this algorithm.

1. \( \Phi(F, G) := G^2 - F \) to get \( G = \sqrt{F} \) provides \( \mathcal{N}(G) = (G + F/G)/2 \).
   If \( G = \sqrt{F} + O(X^n) \), then \( \exists H, \ F = G^2(1 + X^n H) \), so \( \mathcal{N}(G) = G + \frac{1}{2} GX^n H \).
   Next, \( \sqrt{F} = G(1 + \frac{1}{2} X^n H + O(X^{2n})) \), so \( \mathcal{N}(G) = \sqrt{F} + O(X^{2n}) \).

   Computationally: given \( F = T + O(X^N) \), recursively compute \( \sqrt{F} = U + O(X^{\lceil N/2 \rceil}) \), then return \( U + \text{rem}(T/U, X^N)/2 \).

2. \( C(N) \leq C(N/2) + (3 + 2)M(N/2) + \lambda N \) leads to \( O(M(N)) \) for square rooting.
Exercise 2

1. Compute a differential equation satisfied by \( F := (\cos X)^2 + (\sin X)^2 \) by the algorithms for closures of D-finite series.

2. Deduce that \( F \) is equal to 1.
Exercise 2

1. Compute a differential equation satisfied by \( F := (\cos X)^2 + (\sin X)^2 \) by the algorithms for closures of D-finite series.

2. Deduce that \( F \) is equal to 1.

Proof:

\[
\begin{align*}
F &= (\cos X)^2 + (\sin X)^2 \\
F' &= 2 \cos X \cdot (-\sin X) + 2 \sin X \cdot \cos X \\
F'' &= -2 \sin X \cdot (-\sin X) + 2 \cos X \cdot \cos X + 2 \sin X \cdot \sin X \\
&= 2 \sin^2 X + 2 \cos^2 X \\
&= 2 \\
F''' &= 0 \\
F(0) &= 1, \quad F'(0) = F''(0) = 0
\end{align*}
\]
Exercise 2

1. Compute a differential equation satisfied by $F := (\cos X)^2 + (\sin X)^2$ by the algorithms for closures of D-finite series.

2. Deduce that $F$ is equal to 1.

Proof:

$$z = c^2 + s^2$$
$$z' = 2cc' + 2ss'$$
$$z'' = 2c'^2 + 2cc'' + 2s'^2 + 2ss''$$
$$= 2c'^2 - 2c^2 + 2s'^2 - 2s^2$$
$$z''' = 4c'c'' - 4cc' + 4s's'' - 4ss'$$
$$= -8cc' - 8ss'$$
$$z''' + 4z' = 0,$$
$$z(0) = 1, z'(0) = z''(0) = 0$$

Remark: starting with $z = c^2 + (c')^2$ simplifies the calculation.
Define the formal power series

\[ A := \exp \left( \frac{1 - \sqrt{1 - 4X}}{2} \right). \]

1. Justify that this series is well defined.
2. By considering approximations to \(O(X^3)\), show that

\[ A = 1 + X + \frac{3}{2}X^2 + O(X^3). \]

3. Compute a linear differential equation with polynomial coefficients satisfied by \(A\), and check it at \(X = 0\).

4. Convert this linear differential equation to a linear recurrence on its coefficients sequence \((a_n)_{n \in \mathbb{N}}\) [hint: the quantities \(n_0\) and \(n_1\) defined on the slides of the lecture are \(n_0 = 0\) and \(n_1 = 2\)], and check it by verifying that \(a_3 = 19/6\).
Exercise 3 (solution)

First two points are maths that need to be done.
Next, let $R := \sqrt{1 - 4X}$, so that $A = \exp\left(\frac{1-R}{2}\right)$, $R' = \frac{-2R}{1-4X}$, and

$$A' = \frac{1}{1 - 4X} AR$$
$$A'' = \frac{4}{(1 - 4X)^2} AR + \frac{1}{(1 - 4X)^2} AR^2 + \frac{1}{1 - 4X} AR' = \frac{2}{(1 - 4X)^2} AR + \frac{1}{1 - 4X} A$$

$$0 = (1 - 4X)A'' - 2A' - A \quad [\text{check: } A''(0) - 2A'(0) - A(0) = 3 - 2 - 1 = 0]$$

$$0 = (n + 2)(n + 1)a_{n+2} - 4(n + 1)na_{n+1} - 2(n + 1)a_{n+1} - a_n$$

$$= (n + 2)(n + 1)a_{n+2} - 2(n + 1)(2n + 1)a_{n+1} - a_n \quad [\text{check: } n_0 = 0, \ n_1 = 2]$$

We know: $a_0 = 1, \ a_1 = 1, \ a_2 = 3/2$. We derive: $a_3 = (12 \times 3/2 + 1)/6 = 19/6$. 
Part II

Series Composition Again
Non modular: Brent and Kung’s 1978 algorithm in $O(\sqrt{N \log N} M(N))$.

Modular: new exponent in (Neiger, Salvy, Schost, Villard, 2021)

$$\kappa := 1 + ((\omega - 1)^{-1} + 2(\omega_2 - 2)^{-1})^{-1} < 1.43$$

- $\omega$ is the exponent for $n \times n$ by $n \times n$ matrix multiplication,
- $\omega_2$ is the exponent for $n \times n^2$ by $n^2 \times n$ matrix multiplication.
Part III

Linear recurrences
with constant coefficients
For a commutative ring $\mathbb{A}$:

**Euclidean division:** $(F, G) \mapsto (Q, R)$

$$\frac{F}{G} = Q + \frac{R}{G}$$

Can be computed in $O(M(n))$ ops by interpreting $Q$ as asymptotic expansion at $\infty$, when $n$ bounds the degrees of $F$ and $G$.

**Polynomial modular calculations**

Given a univariate $P \in \mathbb{A}[X]$ of degree $n$, define $\mathbb{B} = \mathbb{A}[X]/(P)$. Then:
- addition ($\mathbb{B} \times \mathbb{B} \to \mathbb{B}$) can be done in $O(n)$ ops,
- multiplication by a scalar ($\mathbb{A} \times \mathbb{B} \to \mathbb{B}$) can be done in $O(n)$ ops,
- multiplication ($\mathbb{B} \times \mathbb{B} \to \mathbb{B}$) can be done in $O(M(n))$ ops.
Linear recurrence sequences (LRS)

Definitions

A sequence $(a_n)_{n \in \mathbb{N}}$ with elements in some (general) ring $\mathbb{A}$ is a linear recurrence sequence of order $d$ if it satisfies a recurrence of the form

$$\forall n \geq 0, \quad a_{n+d} = p_{d-1}a_{n+d-1} + \cdots + p_0a_n,$$

where the $p_i$ are fixed elements of $\mathbb{A}$.

$P = X^d - p_{d-1}X^{n+d-1} - \cdots - p_0$ is the characteristic polynomial of $(a_n)_{n \in \mathbb{N}}$. The right gcd of all characteristic polynomials is the minimal polynomial.

Examples

- the Fibonacci sequence (order 2): 1, 1, 2, 3, 5, 8, 13, 21, 34, \ldots,
- any sequence $(uM^nv)_{n \in \mathbb{N}}$ for fixed row $u$, matrix $M$, and column $v$,
- any sequence $(P(n))_{n \in \mathbb{N}}$ for a fixed polynomial $P$,
- the sequence of lengths in the Look-and-Say sequence (order 72):

$$|1| = 1, \quad |11| = 2, \quad |21| = 2, \quad |1211| = 4, \quad |111221| = 6,$$

$$|312211| = 6, \quad |1311221| = 8, \quad |1113213211| = 10, \ldots$$
Computing the $n$th term in a LRS (general A)

(Arithmetic complexity.)

Unrolling the recurrence $O(dn)$
Thus implicitly computing all first $n$ terms.

Binary powering $O(MM(d) \ln n)$

\[
\begin{pmatrix}
  a_{n+1} \\
  \vdots \\
  a_{n+d}
\end{pmatrix}
= C
\begin{pmatrix}
  a_n \\
  \vdots \\
  a_{n+d-1}
\end{pmatrix}
= C^{n+1}
\begin{pmatrix}
  a_0 \\
  \vdots \\
  a_{d-1}
\end{pmatrix}
\quad \text{for} \quad C =
\begin{pmatrix}
  0 & 1 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & 1 \\
  p_0 & p_1 & \cdots & p_{d-1}
\end{pmatrix}.
\]

Use $C^n = (C^{n/2})^2$ for even $n$ and $C^n = C \left(C^{(n-1)/2}\right)^2$ for odd $n$.

Can we do better w.r.t. $d$ for the $n$th term? for the first $n$ terms?
Properties of companion matrices (commutative $\mathbb{A}$)

Lemma

Given $(a_n)_{n \in \mathbb{N}}$ with characteristic polynomial $P$, define $\mathbb{B} = \mathbb{A}[X]/(P)$, $x = X + (P) \in \mathbb{B}$, and the companion matrix $\tilde{C} := C^T$ of $P$. Then:

1. $\tilde{C}$ satisfies $\forall n \geq 0, (a_{n+1}, \ldots, a_{n+d}) = (a_n, \ldots, a_{n+d-1}) \tilde{C}$,

2. $\tilde{C}$ is the matrix of the $\mathbb{A}$-linear multiplication by $x$ in $\mathbb{B}$ with respect to the basis $\mathcal{B} := (1, x, \ldots, x^{d-1})$.

Proof: (1): by definition of $P$. (2): the $j$th column of the wanted matrix contains the coordinates of $x \cdot x^{j-1}$. For $j < d$, use $x \cdot x^j = x^{j+1}$. For $j = d$, use $P(x) = 0$ in the form $x \cdot x^{d-1} = p_0 + \cdots + p_{d-1}x^{d-1}$.

Corollary

For any $k \in \mathbb{N}$, let $V_k$ denote the column vector of the coefficients of $x^k = X^k + (P)$ with respect to the basis $\mathcal{B}$. Then,

$$\forall n \geq 0, a_{k+n} = (a_n, \ldots, a_{n+d-1}) V_k.$$

Proof: From $x^k = x^k \cdot 1$ and iterating the lemma, $V_k$ is the first column of $\tilde{C}^k$. 
Computing the $n$th term in a LRS (commutative $\mathbb{A}$)

**Input** The coefficients $(p_0, \ldots, p_{d-1})$ of a linear recurrence; initial conditions $(a_0, \ldots, a_{d-1})$; an integer $n$.

**Output** The term $a_n$ of the sequence defined by $a_{i+d} = p_{d-1}a_{i+d-1} + \cdots + p_0a_i$.

1. Set $P = X^d - p_{d-1}X^{d-1} - \cdots - p_0$ and $x = X + (P)$.
2. Recursively compute $x^n = q_0 + q_1x + \cdots + q_{d-1}x^{d-1}$ by the modular formulas $x^k = (x^{k/2})^2$ for even $k$ and $x^k = x \cdot (x^{k-1}/2)^2$ for odd $k$.
3. Return $q_0a_0 + \cdots + q_{d-1}a_{d-1}$.

**Modular exponentiation** $O(M(d) \ln n)$

Over a commutative ring $\mathbb{A}$, the $n$th term of a LRS can be computed using modular exponentiation in $O(M(d) \ln n)$ ops.

Proof: Modular multiplication in $\mathbb{B}$ is done in $O(M(d))$ ops.
Reciprocal polynomials

**Definition**

Given \( P = p_d X^d + \cdots + p_0 \in \mathbb{A}[X] \) of degree \( d \), define the *reciprocal polynomial* of \( P \) as

\[
\text{rec} \, P = X^{\deg P} P(1/X) = p_0 X^d + \cdots + p_d.
\]

**Properties**

- \( \text{rec}(PQ) = \text{rec}(P) \, \text{rec}(Q) \)
- \( \text{rec}(X^v P) = \text{rec} \, P \)
- \( \text{rec} \, \text{rec} \, P = P \iff P(0) \neq 0 \iff \deg \, \text{rec} \, P = \deg P \)
- \( P = X^{\text{val} \, P} \, \text{rec} \, \text{rec} \, P \)
- \( (\text{rec} \, P)(0) \neq 0 \)
Given $A = \sum_{n \in \mathbb{N}} a_n X^n \in \mathbb{A}[[X]]$, we define the notation $[X^n]A = a_n$.

From now on, $\mathbb{A}$ is a field $\mathbb{K}$, thus commutative.
Theorem

Given a monic polynomial $P$ of degree $d$, a sequence $(a_n)_{n \in \mathbb{N}}$, and the series $A = \sum_{n \in \mathbb{N}} a_n X^n$, both following assertions are equivalent:

1. $(a_n)_{n \in \mathbb{N}}$ is an LRS with characteristic polynomial $P$;
2. there exists $N \in \mathbb{K}[X]$ of degree $< d$ such that $A = N/\text{rec } P$ in $\mathbb{K}[[X]]$.

When these assertions hold, if moreover $P$ is the minimal polynomial of $(a_n)_{n \in \mathbb{N}}$, then

$$d = \max\{1 + \deg N, \deg \text{rec } P\} := m \quad \text{and} \quad \gcd(N, \text{rec } P) = 1.$$

Proof: Write $P = g_d X^d + \cdots + g_0$ with $g_d = 1$. For $n \in \mathbb{N}$, compute

$$[X^{d+n}] (A \text{ rec } P) = [X^{d+n}] \sum_{j=0}^{d} g_j X^{d-j} A = \sum_{j=0}^{d} g_j [X^{n+j}] A = \sum_{j=0}^{d} g_j a_{n+j}.$$

Then:

$$(1) \iff \forall n \in \mathbb{N}, \sum_{j=0}^{d} g_j a_{n+j} = 0 \iff \deg(A \text{ rec } P) < d \iff (2).$$

By construction, $d \geq m$. If $d > m$, $\deg P = d > \deg \text{rec } P$, so $g_0 = 0$, and $\text{rec}(P/X) = \text{rec } P$. Since $\deg N < d - 1 = \deg(P/X)$, by $(2) \Rightarrow (1)$, $P$ cannot be minimal. So $P$ minimal implies $d = m$. 
Theorem

Given a monic polynomial $P$ of degree $d$, a sequence $(a_n)_{n \in \mathbb{N}}$, and the series $A = \sum_{n \in \mathbb{N}} a_n X^n$, both following assertions are equivalent:

1. $(a_n)_{n \in \mathbb{N}}$ is an LRS with characteristic polynomial $P$;
2. there exists $N \in \mathbb{K}[X]$ of degree $< d$ such that $A = N / \text{rec } P$ in $\mathbb{K}[[X]]$.

When these assertions hold, if moreover $P$ is the minimal polynomial of $(a_n)_{n \in \mathbb{N}}$, then

$$d = \max\{1 + \deg N, \deg \text{rec } P\} := m \quad \text{and} \quad \gcd(N, \text{rec } P) = 1.$$

Proof: Write $U := \gcd(N, \text{rec } P)$, $F := \text{rec } (P) / U$. Then, $F(0) \neq 0 \neq U(0)$. For some $v \in \mathbb{N}$,

$$P = X^v \text{rec } \text{rec } P = X^v \text{rec } (F) \text{rec } (U).$$

Set $\tilde{P} := P / \text{rec } (U)$ and $\tilde{N} := N / U$, so that:

$$\begin{cases} A \text{rec } P = N \\ \deg N < \deg P \end{cases} \Rightarrow \begin{cases} AU \text{rec } \tilde{P} = U \tilde{N} \\ \deg \tilde{N} = \deg N - \deg U < \deg P - \deg U = \deg \tilde{P}. \end{cases}$$

So, $\tilde{P}$ is characteristic, and $P$ minimal implies $\deg P \leq \deg \tilde{P}$ and $U = 1$. 
Computing the $n$th coefficient of a rational series

Module exponentiation $O(M(d) \ln n)$

Given polynomials $F$ and $G$ of degree at most $d$, with coefficients in a field $K$, with $G(0) \neq 0$, the $n$th coefficient in the formal power series expansion of $F/G$ can be computed using $O(M(d) \ln n)$ ops in $K$.

Proof: Combine the observation that the coefficient sequence is a LRS with characteristic polynomial of degree $\deg G$ and the algorithm by modular exponentiation.
A recent algorithm by sections gains a factor of $3/2$

(Bostan, Mori, 2021)

Idea: compute recurrence for odd/even subsequence.

Input Polynomials $F$ and $G$ of degree at most $d$ with $G(0) \neq 0$; an integer $n$.

Output The $n$th coefficient of the expansion of $F/G$.

1. While $n > 0$, do:
   - Compute $U(X) := F(X)G(-X)$ and write it $U(X) = S_0(X^2) + XS_1(X^2)$.
   - Compute $V(X) := G(X)G(-X)$ and write it $V(X) = T(X^2)$.
   - Set
     
     $$(F, G, n) := \begin{cases} 
     (S_0, T, n/2) & \text{if } n \text{ is even}, \\
     (S_1, T, (n - 1)/2) & \text{if } n \text{ is odd}.
     \end{cases}$$

2. Return $F(0)/G(0)$.

Variations and generalizations also allow to compute a slice, and to perform modular exponentiation faster.
Computing the first $n$ coefficients of a rational series

Newton’s method \(O(M(n))\)
When \(d \leq n\).

Unrolling the recurrence \(O(dn)\)
Better than Newton for fixed \(d\).

By an interpretation as LRS \(O(nM(d)/d)\)
By computing successive slices of length \(d\).
(An algorithm will be given soon.)
Applications

First $n$ coefficients of a LRS

Given a LRS $(a_n)_{n \in \mathbb{N}}$, its characteristic polynomial of degree $d$, initial conditions $a_0, \ldots, a_{d-1}$, and an integer $n \geq d$, then $a_0, \ldots, a_n$ can be computed in $O(nM(d)/d)$ ops.

Proof: By immediate translation.

Multipoint evaluation of a polynomial at an arithmetic progression

Let $a \in \mathbb{K}$, $b \in \mathbb{K}$, $P \in \mathbb{K}[X]$ with $\deg P = d < n$. The multipoint evaluation $(P(b), P(a + b), P(2a + b), \ldots, P(na + b))$ can be computed in $O(nM(d)/d)$ ops.

Proof: The operator $\Delta_a : H \mapsto H(X + a) - H(X)$ decreases the degree,

$$0 = \Delta_a^{d+1}(P) = \sum_{i=0}^{d+1} (-1)^{d+1+i} \binom{d+1}{i} P(X + ia).$$

Setting $X = a$ makes $(P(na + b))_{n \in \mathbb{N}}$ be a LRS with characteristic polynomial $(X - 1)^{d+1}$.
Idea: generalize the formula \( a_{k+n} = (a_n, \ldots, a_{n+d-1}) V_k \).

**Lemma**

Assume \( F \in \mathbb{K}[X] \), \( G \in \mathbb{K}[X] \), \( d = \deg G > \deg F \), \( G(0) = 1 \), \( k \in \mathbb{N} \). Upon setting \( a_n := [X^n](F/G) \), we get:

\[
0 \leq i < d \Rightarrow a_{k+i} = [X^{2d-2-i}] \left( (a_{2d-2} + \cdots + a_0 X^{2d-2}) \ \text{rem}(X^k, \ \text{rec} \ G) \right).
\]

Proof: \( F/G = N/\ \text{rec} \ P \) for \( N := F \) and \( P := \text{rec} \ G \). So \( P \) is a characteristic polynomial of \( (a_n)_{n \in \mathbb{N}} \). Recall the notation

\[
\mathbb{B} = \mathbb{A}[X]/(P), \ x = X + (P), \ \tilde{C} = C^T, \ V_k = \tilde{C} e_1^T.
\]

Then, \( \text{rem}(X^k, \ \text{rec} \ G) = (1, X, \ldots, X^{d-1}) \cdot V_k \). Compute:

\[
[X^{2d-2-i}] \sum_{j=0}^{2d-2} a_j X^{2d-2-j} \sum_{\ell=0}^{d-1} (V_k)_\ell X^\ell = \sum_{\ell=0}^{d-1} (V_k)_\ell \sum_{j=0}^{2d-2} [X^{-i}] a_j X^{\ell-j} = \sum_{\ell=0}^{d-1} a_{i+\ell} (V_k)_\ell.
\]

This is the product of \( A_i = (a_i, \ldots, a_{d+i-1}) \) by \( V_k \), which is \( A_i V_k = a_{k+i} \).
Rational series expansion by modular multiplications

**Input** Polynomials $F$ and $G$ with $\deg F < \deg G = d$, $G(0) \neq 0$, $n > d$.

**Output** Coefficients $a_0, \ldots, a_n$ in the expansion $F/G = \sum_{i \geq 0} a_i X^i$.

1. Compute $(a_0, \ldots, a_{2d-2})$ by Newton’s iteration.
2. Compute $Y_1 = \text{rem}(X^{d-1}, \text{rec } G)$ and $Y^* = \text{rem}(X^d, \text{rec } G)$.
3. For $i = 2, \ldots, \lceil n/d \rceil$:
   1. Compute $Y_i = \text{rem}(Y^* Y_{i-1}, \text{rec } G)$.  
   2. Compute $P = (a_{2d-2} + \cdots + a_0 X^{2d-2}) Y_i$.
   3. For $0 \leq j < d$, extract $a_{id-1+j} = [X^{2d-2-j}] P$.  

4. Return $(a_0, \ldots, a_n)$.

**Theorem**

The algorithm is correct. It computes in $O(nM(d)/d)$ ops.

Proof: Correctness follows from the lemma and a tedious verification of indices/exponents. Complexity: Newton in $O(M(d))$; $Y^*$ for free; each $Y_i$ involves a product in $\mathbb{B}$, in $O(M(d))$; each $P$ involves a product in degrees $2d - 2$ and $d - 1$, in $O(M(d))$. 

Frédéric Chyzak

AECF, MPRI C-2-22
Part IV

Linear recurrences with polynomial coefficients
Regular vs. singular linear recurrences

Reminder: P-recursive sequences

Def: A sequence \((a_n)_{n\geq 0} \in \mathbb{K}^\mathbb{N}\) is polynomially recursive if it satisfies a linear difference equation with coefficients \(p_j\) in \(\mathbb{K}[X]\):

\[
\forall n \in \mathbb{N}, \quad p_r(n)a_{n+r} + \cdots + p_0(n)a_n = 0.
\]

\(r = \text{order} \quad d = \max_i \deg p_i = \text{degree}\)

\[
\forall n \geq r, \quad a_n = -p_r(n-r)^{-1}\left(p_{r-1}(n-r)a_{n-1} + \cdots + p_0(n-r)a_{n-r}\right)
\]

Def: Regular and singular recurrences

A recurrence is regular if \(\forall n \geq r, \ p_r(n - r) \neq 0\), and singular otherwise.

Motivation

A regular recurrence of order \(r\) uniquely determines a P-recursive solution from initial conditions \((a_0, \ldots, a_{r-1})\).

In what follows, we consider regular recurrences only.
Different complexity models require different approaches

### Reminder: LRS with coefficients and initial conditions in some field $\mathbb{K}$

<table>
<thead>
<tr>
<th>problem</th>
<th>$n$th term</th>
<th>first $n$ terms</th>
<th>(naive unrolling)</th>
</tr>
</thead>
<tbody>
<tr>
<td>arithmetic</td>
<td>$O(M(r) \ln n)$</td>
<td>$O(nM(r)/r)$</td>
<td>$O(rn)$</td>
</tr>
</tbody>
</table>

$M(n)$ : multiplication function for polynomials of degree $< n$

$M_{\mathbb{Z}}(n)$ : multiplication function for integers of $< n$ bits

$MM(r)$ : complexity of multiplying $r \times r$ matrices

### To be obtained: P-rec. seq. with coeffs in $\mathbb{Z}[X]$ and initial conditions in $\mathbb{Z}$

<table>
<thead>
<tr>
<th>problem</th>
<th>$n$th term</th>
<th>first $n$ terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>arithmetic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>unrolling rec.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>baby-step giant-step</td>
<td>$O(MM(r)M(\sqrt{dn}) \ln(dn))$</td>
<td></td>
</tr>
<tr>
<td>binary</td>
<td></td>
<td></td>
</tr>
<tr>
<td>unrolling rec.</td>
<td>$O(rdn)$</td>
<td></td>
</tr>
<tr>
<td>binary splitting</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

More complex formulas take input bit size into account.
The prototype $n!$ and Stirling’s formula

Computing $(1!, 2!, \ldots, n!)$ by the naive recurrence $a_n = n \times a_{n-1}$

- $O(n)$ arithmetic ops, which is optimal.
- $O(n^2 \text{M}_\mathbb{Z}(\ln n))$ bit ops, which is quasi-optimal.

Proof:

\[
\text{Stirling's formula: } \ln n! = n \ln n - n + \frac{1}{2} \ln n + O(1), \text{ when } n \to \infty.
\]

And even: $\ln n! \leq n \ln n$ when $n \geq 1$. The multiplication $k \times (k - 1)!$ is in sizes $\lceil \ln k \rceil$ and $\lceil k \ln k \rceil$. This unbalanced product has cost $ck\text{M}_\mathbb{Z}(\ln k)$ for some fixed $c$. Total cost:

\[
\sum_{k=1}^{n} ck\text{M}_\mathbb{Z}(\ln k) \leq \left( \sum_{k=1}^{n} ck \right) \text{M}_\mathbb{Z}(\ln n) = O(n^2 \text{M}_\mathbb{Z}(\ln n)).
\]

Total size is $O(n^2 \ln n)$ by Stirling’s formula again.
Unrolling a general recurrence: arithmetic complexity

\[ a_{n+r} = -\frac{1}{p_r(n)}(p_{r-1}(n)a_{n+r-1} + \cdots + p_0(n)a_n) \]

Evaluating a polynomial of degree \( d \) takes \( O(d) \) ops, hence a total \( O(rdn) \).
Unrolling a general recurrence: matrix formulation

- Introduce $A_n = (a_n, \ldots, a_{n+r-1})^T$ to observe $A_{n+1} = \frac{1}{p_r(n)} M(n) A_n$ where

\[
M(n) = \begin{pmatrix}
0 & p_r(n) & & & \\
& 0 & p_r(n) & & \\
& & \ddots & \ddots & \\
& & & 0 & p_r(n) \\
-p_0(n) & \ldots & & & -p_{r-1}(n)
\end{pmatrix}.
\]

- Define $(b_n)_{n \in \mathbb{N}}$ by $b_0 = 1$ and $b_{n+1} = p_r(n) b_n$ and get a “matrix factorial”

\[A_n = \frac{1}{b_n} F_n \quad \text{where} \quad F_n = (M(n-1) \cdots M(0)) A_0.\]

- In particular,

\[a_n = \frac{e_1 F_n}{b_n}.\]
Unrolling a general recurrence: size bounds

**Norm of a vector $V$, subordinate norm of a matrice $M$**

\[
\|V\| = \sum_{j=1}^{r} |v_j|, \quad \|M\| = \max_{1 \leq j \leq r} \sum_{i=1}^{r} |m_{i,j}|, \quad \|MV\| \leq \|M\|\|V\|. 
\]

$h \geq$ bit size of coefficients of $p_i$ \quad $s \geq$ bit size of the $a_j$ for $0 \leq j < r$

Hypothesis: $h = O(1)$ and $s = O(1)$ when $r, d, n \to \infty$.

\[
|p_i(k)| \leq 2^h (d + 1) \max\{1, k\}^d \leq 2^h (d + 1)(k + 1)^d \\
\|M(k)\| \leq 2^{h+1} (d + 1)(k + 1)^d \\
\|A_0\| \leq r 2^s \\
|e_1F_n| \leq \|M(n - 1)\| \cdots \|M(0)\| \|A_0\| \leq r \ 2^{(h+1)n+s} (d + 1)^n (n!)^d
\]

\[
\ln |p_i(n)| = O(d \ln n), \quad \ln |e_1F_n| = O(dn \ln n + \ln r), \quad \ln |b_n| = O(dn \ln n).
\]
Unrolling a general recurrence: bit complexity

**Input** A regular recurrence $p_r(n)a_{n+r} + \cdots + p_0(n)a_n = 0$ with coefficients $p_i \in \mathbb{Z}[X]$; initial conditions $a_0, \ldots, a_{r-1}$ in $\mathbb{Z}$; some integer $n \geq r$.

**Output** The rational numbers $(a_0, \ldots, a_n)$.

1. Set $F_0 := (a_0, \ldots, a_{r-1})^T$ and $b_0 := 1$.
2. For $k$ from 0 to $n - r$:
   - Compute $F_{k+1} := M(k)F_k$, $b_{k+1} := p_r(k)b_k$, $a_{k+r} := (e_rF_{k+1})/b_{k+1}$.
3. Return $(a_0, \ldots, a_n)$.

The dominant cost at step $k$ is the matrix-vector product:

- **Sizes**: $\ln |e_i M(k) e_j^T| \leq O(d \ln k)$ and $\ln |e_i F_k| \leq O(dk \ln k + \ln r)$.
- **Cost**: $O(rkM_\mathbb{Z}(d \ln k))$.

Hence the announced $O(rn^2M_\mathbb{Z}(d \ln n))$ after summing over $k$.

**Output size**: $O(dn^2 \ln n)$. 

Frédéric Chyzak  
AECF, MPRI C-2-22
Binary splitting for $n!$: an algorithm to balance products

By Stirling’s formula, observe:

$$\ln\left((n/2)!\right) \sim \frac{1}{2} \ln(n!) \sim \ln \frac{n!}{(n/2)!}. $$

**Algorithm**

For $a \leq b$, define a recursive calculation of $P(a, b) := (a + 1)(a + 2) \cdots b$ by:

$$P(a, b) = P(a, m)P(m, b) \quad \text{where} \quad m = \left\lfloor \frac{a + b}{2} \right\rfloor.$$

$M_Z(s)$ increases with $s$, so $P(a, m)$ is computed faster than $P(m, b)$.

The binary cost $C(a, b)$ of computing $P(a, b)$ this way satisfies:

$$C(a, b) \leq C(a, m) + C(m, b) + M_Z\left(\max(\ln P(a, m), \ln P(m, b))\right),$$

$$C(a, b) \leq 2C(m, b) + M_Z\left(\ln P(m, b)\right).$$
Binary splitting for \( n! \): complexity analysis

For \( n = 2^\ell \) and \( 1 \leq k \leq \ell \):

\[
C(0, n) \leq 2C(n/2, n) + M_Z\left(\ln P(n/2, n)\right) \leq 2C(n/2, n) + M_Z\left(\frac{n}{2} \ln n\right)
\]

\[
\leq 4C(3n/4, n) + 2M_Z\left(\frac{n}{4} \ln n\right) + M_Z\left(\frac{n}{2} \ln n\right) \leq 4C(3n/4, n) + 2M_Z\left(\frac{n}{2} \ln n\right)
\]

\[
\leq \cdots \leq 2^k C(n - 2^{-k} n, n) + kM_Z\left(\frac{n}{2} \ln n\right).
\]

Making \( k = \ell \):

\[
C(0, n) \leq O(n) + M_Z\left(\frac{n}{2} \ln n\right) \ln n = O\left(M_Z(n \ln n) \ln n\right).
\]

Remark:

- With FFT, \( C(0, n) = O(M_Z(n \ln n) \ln n) = O(n \ln^3 n) \).
- When \( M_Z(n) \) is super-linear in \( O(n^\alpha) \), the analysis replaces a \( \ln n \) by a geometric series, resulting in \( C(0, n) = O(M_Z(n \ln n)) = O(n^\alpha \ln^\alpha n) \).
### General hypergeometric sequences: order 1, degree $d$

- Consider numerators and denominators of rational $a_n$ separately.
- Size of integers is multiplied by $d$.
- Complexity becomes $O\left(M_{\mathbb{Z}}(dn \ln n) \ln n\right)$.

### General P-recursive sequences: order $r$, degree $d$

- Perform binary splitting on matrix factorial $M(n-1) \cdots M(0)$ before multiplying by the vector $A_0$.
- Complexity becomes $O\left(MM(r)M_{\mathbb{Z}}(dn \ln n) \ln n\right)$.

Remark: the same discussion as for $n!$ applies, by distinguishing multiplication by FFT and super-linear multiplication in $O(n^\alpha)$. 

Frédéric Chyzak

AECF, MPRI C-2-22
Applications: high-precision computation of constants

\[ e = \exp(1) \text{ to } 10^{-N} \]

Define \( e_n = 1 + \cdots + 1/n! \), so that \( 0 < e - e_n < 1/(n n!) \).

From \( e_{n+1} - e_n = 1/(n + 1)! \) follows \( (n + 2)e_{n+2} - (n + 3)e_{n+1} + e_n = 0 \), then

\[
\begin{pmatrix}
  e_{n+1} \\
  e_{n+2}
\end{pmatrix} = \frac{1}{n + 2} \begin{pmatrix}
  0 & n + 2 \\
  -1 & n + 3
\end{pmatrix}
\begin{pmatrix}
  e_n \\
  e_{n+1}
\end{pmatrix}.
\]

As \( n = O(N/\ln N) \) is enough, the complexity follows.

\[ \pi = 4 \arctan(1), \text{ etc.} \]

Similarly.

Chudnovsky and Chudnovsky gave a faster formula for \( 1/\pi \).

Much more involved: evaluation of (D-finite) functions at any point of the complex plane (requires controlled analytic continuation).
Application: Shape Prediction of Biomembranes

(Melczer, Mezzarobba, 2020)

- result: fixed genus + minimization of energy $\rightarrow$ unique shape
- uses rigorous numeric analytic continuation to prove positivity

unique shape $\uparrow$

the solution $f$ of

$$c_3 f''' + c_2 f'' + c_1 f' + c_0 f = 0$$

is positive $\uparrow$

its Taylor coefficients are positive

$$
\begin{align*}
c_3(z) &= 8388593z^2(3z^4 - 164z^3 + 370z^2 - 164z + 3)(z + 1)^2(z^2 - 6z + 1)^2(z - 1)^3 \\
c_2(z) &= 8388593(z + 1)(z^2 - 6z + 1)(66z^8 - 3943z^7 + 18981z^6 - 16759z^5 - 30383z^4 + 47123z^3 - 17577z^2 + 971z - 15)(z - 1)^2 \\
c_1(z) &= 16777186(z - 1)(210z^{12} - 13761z^{11} + 101088z^{10} - 178437z^9 + 248334z^8 + 930590z^7 - 446064z^6 - 694834z^5 + 794998z^4 - 267421z^3 + 241 \times 10^{-1} \\
c_0(z) &= 6341776308z^{12} - 427012938072z^{11} + 2435594423178z^{10} - 2400915979716z^9 - 10724094731502z^8 + 26272536406048z^7 - 8496738740956z^6 - \ldots
\end{align*}
$$
Baby-step giant-step approach to computing $n!$.

Idea: $n = m^2 \Rightarrow n! = P(0, n) = P(0, m)P(m, 2m) \cdots P((m - 1)m, m^2)$.

Input  An integer $n = m^2 \in \mathbb{N}$.
Output  The integer $n!$.

1. *(Baby-step)* Compute $Q := (X + 1)(X + 2) \cdots (X + m)$ from the binary tree of subproducts with $X + i$ at the leaves.
2. *(Giant-step)* Obtain $(Q(0), Q(m), \ldots, Q((m - 1)m))$ by multipoint evaluation.
3. Compute and return the product $Q(0) \times Q(m) \times \cdots \times Q((m - 1)m)$.

Theorem

The algorithm computes $n!$ in $O(M(\sqrt{n}) \ln n)$ ops.

Proof: Baby-step fits in $O(M(m) \ln m)$. Giant-step fits in $O(M(m) \ln m)$. Final step fits in $O(m)$.  

Frédéric Chyzak  AECF, MPRI C-2-22
Baby-step giant-step for a P-recursive sequence

Input  A regular recurrence \( p_r(n)a_{n+r} + \cdots + p_0(n)a_n = 0 \) with coefficients \( p_i \in \mathbb{Z}[X] \); initial conditions \( a_0, \ldots, a_{r-1} \) in \( \mathbb{Z} \); some integer \( n \geq r \).

Output  The terms \( a_{n-r+1}, \ldots, a_n \) of the solution.

1. Compute \( Q(X) := M(X + m - 1) \cdots M(X + 1)M(X) \) where \( m = \left\lceil \sqrt{n/d} \right\rceil \).
2. Evaluate the matrices \( Q(0), Q(m), \ldots, Q(m'm') \) for \( m' = \left\lfloor \frac{n-r-m}{m} \right\rfloor \).
3. Evaluate \( M(n - r), M(n - r - 1), \ldots, M(m''m') \) for \( m'' = \left\lfloor \frac{n-r}{m} \right\rfloor \).
4. Compute the product \( R = M(n - r) \cdots M(m''m')Q(m'm') \cdots Q(0) \).
5. If \( p_r \neq 1 \), compute \( b := p_r(n - r) \cdots p_r(0) \) by the same algorithm; otherwise set \( b := 1 \).
6. Return \( b^{-1} R (a_0, \ldots, a_{r-1})^T \).

Theorem

The algorithm computes in \( O(MM(r)M(\sqrt{dn}) \ln(dn)) \) ops.

Proof: Main steps: \( Q \) by a product tree in \( O(MM(r, md) \ln m) \); all \( Q(jm) \) by \( r^2 \) multipoint evaluations in \( O(r^2 \frac{n}{m} M(md)/(md)) \); \( R \) by matrix products in \( O(MM(r)(m + n/m)) \). Use \( MM(r, \delta) = O(MM(r)M(\delta)) \) to get the result.