Analysis of the Brun Gcd Algorithm

V. Berthé, L. Lhote, B. Vallée

SpecFun- Novembre 2016
Brun gcd algorithm

- A multiple gcd algorithm that is a natural extension of the usual Euclid algorithm for $(d + 1)$ integers.
- It coincides with it for two entries.
- It performs Euclidean divisions, between the largest entry and the second largest entry.
- This is the discrete version of a multidimensional continued fraction algorithm due to Brun ('57).

Also called Podsypanin modified Jacobi–Perron algorithm, $d$-dimensional Gauss transformation, ordered Jacobi–Perron algorithm, etc.

and also an algorithm for efficient exponentiation with precomputation [de Rooij]
We perform the worst-case and the average-case analysis of this algorithm for the number of steps. We prove that the worst-case and the mean number of steps are linear with respect to the size of the entry. The method relies on dynamical analysis, and is based on the study of the underlying Brun dynamical system. The dominant constant of the average-case analysis is related to the entropy of the system. We provide asymptotic estimates for the Brun entropy. We also compare this algorithm to Knuth’s extension of the Euclid algorithm.
Euclid algorithm and continued fractions

- We start with two (coprime) integers
- One divides the largest by the smallest
- Euclid’s algorithm yields the digits of the continued fraction expansion of their quotient
- Euclid’s algorithm becomes in its continuous version the Gauss transformation

\[ T : [0, 1] \to [0, 1], \ x \mapsto \{1/x\} \]

- Rational trajectories behave like generic trajectories for the Gauss transformation (methods from Dynamical Analysis [Baladi-Vallée])
- Our strategy: consider the generalizations of Euclid’s algorithm issued from multidimensional continued fraction algorithms endowed with a “good” dynamical system (Brun, Jacobi-Perron, Selmer etc.)
Brun algorithm

We divide the largest entry by the second largest entry and reorder.

\[(74, 37, 13, 5, 3) \mapsto (37, 13, 5, 3) \mapsto (13, 11, 5, 3) \mapsto (11, 5, 3, 2) \mapsto (5, 3, 2, 1) \mapsto (3, 2, 1) \mapsto (2, 1) \mapsto (1)\]
Brun algorithm

We divide the largest entry by the second largest entry and reorder.

\[(74, 37, 13, 5, 3) \rightarrow (37, 13, 5, 3) \rightarrow (13, 11, 5, 3) \rightarrow (11, 5, 3, 2) \rightarrow (5, 3, 2, 1) \rightarrow (3, 2, 1) \rightarrow (2, 1) \rightarrow (1)\]

Start with \((u_0, u_1, \ldots, u_d)\) with \(u_0 > u_1 > u_2 > \ldots > u_d > 0\)

- In each step, the first component \(u_0\) is divided by the second component \(u_1\), and creates a remainder \(v_0\)

\[v_0 := u_0 - mu_1\] Remainder \[m := \left\lfloor \frac{u_0}{u_1} \right\rfloor\] Partial quotient

- The second component \(u_1\) becomes the largest one.
- There are different cases for the insertion (or not) of \(v_0\).
The algorithm BrunGcd\((d)\)

\[ u_0 > u_1 > u_2 > \ldots > u_d > 0 \]

We divide the largest entry \(u_0\) by the second largest entry \(u_1\) and we reorder.

\[ v_0 := u_0 - \left\lfloor \frac{u_0}{u_1} \right\rfloor u_1 \]

\((G)\) (Generic case) if \(v_0\) is not present in \((u_1, \ldots, u_d)\), we perform a usual insertion;

\((Z)\) (Zero case) if \(v_0 = 0\), we do not insert \(v_0\);

\((E)\) (Equality case) if \(v_0 \neq 0\) is already present (at position \(i\), say), we do not insert \(v_0\).
The algorithm \texttt{BrunGcd}(d) decomposes into \(d\) phases, labelled from \(\ell = 0\) to \(\ell = d - 1\). During each phase, a component is “lost”, and the \(\ell\)-th phase transforms an element of \(\Omega_{(d-\ell)}\) into an element of \(\Omega_{(d-\ell-1)}\).

The phase ends as soon as it looses a component:
- if \(\nu_0 = 0\);
- or else, if \(\nu_0 \neq 0\) is already present in \((u_1, \ldots, u_k)\).

The algorithm stops at the end of the \((d-1)\)-th phase with an element of \(\Omega_{(0)}\) which equals the \texttt{gcd}.
The algorithm $\text{BrunGcd}(d)$

We divide the largest entry by the second largest entry and reorder.

The algorithm $\text{BrunGcd}(d)$ computes the gcd of $(d + 1)$ positive integers. It deals with the input set

$$\Omega(d) := \{u = (u_0, u_1, \ldots, u_d) \mid u_0 > u_1 > u_2 > \ldots > u_d > 0\}.$$

During the execution of the algorithm, some components “disappear” and the algorithm deals with the disjoint union

$$\biguplus_{\ell=0}^{d-1} \Omega(d-\ell).$$
Results
Maximum number of steps

The worst-case of the BrunGcd algorithm arises when

- the quotients are the smallest possible (all equal to 1, except the last one, equal to 2),
- and the insertion positions the largest possible.
Maximum number of steps

The worst-case of the BrunGcd algorithm arises when

- the quotients are the smallest possible (all equal to 1, except the last one, equal to 2),
- and the insertion positions the largest possible.

Theorem [Lam-Shallit-Vanstone] The maximum number $Q(d,N)$ of steps of the BrunGcd Algorithm on the set

$$\Omega_{(d,N)} := \{ u = (u_0, u_1, \ldots, u_d) \mid N \geq u_0 > u_1 > u_2 > \ldots > u_d > 0 \}$$

satisfies

$$Q(d,N) \sim \frac{1}{|\log \tau_d|} \log N \quad (N \to \infty)$$

Let $\tau_d \in ]0, 1[$ be the smallest real root of $X^{d+1} + X - 1$

$$1/|\log \tau_d| \sim \frac{(d + 1)}{\log d} \quad (d \to \infty)$$
Mean number of steps

The algorithm BrunGcd acts on the set

$$\Omega_{(d,N)} = \{(u_0, u_1, \ldots, u_d) \mid N \geq u_0 > u_1 > u_2 > \ldots > u_d > 0\}$$

endowed with the uniform distribution

- The total number of steps $L_d$ is on average linear in the size $\log N$ of the entries
Mean number of steps

The algorithm BrunGcd acts on the set

\[ \Omega_{(d,N)} = \{ (u_0, u_1, \ldots, u_d) \mid N \geq u_0 > u_1 > u_2 > \ldots > u_d > 0 \} \]

dowed with the uniform distribution

- The total number of steps \( L_d \) is on average linear in the size \( \log N \) of the entries

**Theorem** Here \( d \) is fixed, \( N \) tends to \( \infty \). One has

\[
E_N[L_d] \sim \frac{d + 1}{\mathcal{E}_d} \cdot \log N \quad (N \to \infty)
\]

\( \mathcal{E}_d \): entropy of the Brun dynamical system
Mean number of steps

The algorithm BrunGcd acts on the set

\[ \Omega_{(d,N)} = \{(u_0, u_1, \ldots, u_d) \mid N \geq u_0 > u_1 > u_2 > \ldots > u_d > 0\} \]

endowed with the uniform distribution

- The total number of steps \( L_d \) is on average linear in the size \( \log N \) of the entries
- Number of steps performed during the first phase: \( M_d \)
  
  Theorem \( \mathbb{E}_N[L_d] \sim \mathbb{E}_N[M_d] \sim \frac{d+1}{\mathcal{E}_d} \cdot \log N \quad (N \to \infty) \)

- Number of steps performed after the first phase: \( R_d \)
  
  Theorem \( \mathbb{E}_N[R_d] \sim r_d \quad (N \to \infty) \)

- One has a strong difference between the first phase, where most of the work is done, and the remainder of the execution, where \( R_d \) is on average asymptotically constant
Comparison between the worst and the average case

- Both dominant constants behave as $d/\log d$ for $d \to \infty$

$$
\mathbb{E}_N[L_d] \sim \frac{d + 1}{\mathcal{E}_d} \cdot \log N \quad Q(d, N) \sim \frac{1}{|\log \tau_d|} \cdot \log N \quad (N \to \infty)
$$

$$
1/|\log \tau_d| \sim \frac{(d + 1)}{\log d} \quad \mathcal{E}_d \sim \log d \quad (d \to \infty)
$$

- This indicates the same behavior for the algorithm in the average-case and in the worst-case.

- As the worst-case is reached when the quotients are all equal to 1, this seems to indicate that the BrunGcd Algorithm deals with quotients which are very often equal to 1.
On the quotients equal to 1

- Number of steps performed during the first phase: \( M_d \)
- Number of quotients equal to 1 during the first phase: \( O_d \)

**Theorem**

\[
\frac{E_N[O_d]}{E_N[M_d]} \sim \rho_d \quad (N \to \infty)
\]

\[
\rho_d = 1 + O(2^{-d/\log d}) \quad (d \to \infty)
\]

- Number of steps of the subtractive version of BrunGcd during the first phase: \( \Sigma_d \)

**Theorem**

\[
\frac{E_N[\Sigma_d]}{E_N[M_d]} \sim \sigma_d \quad (d \to \infty)
\]

\[
1 \leq \sigma_d \leq 2 + (\log d)^{-1/2}
\]
On the proportion of quotients equal to 1

The following figure exhibits the proportion of quotients equal to 1 during the first phase as a function of the dimension \( d \). This proportion tends quickly to 1:

- when \( d = 16 \), more than 99% of the Euclidean divisions are in fact subtractions
- for \( d = 50 \), the proportion is 99.99%.
On the constants

The constants $E_d, \rho_d, \sigma_d, r_d$ are dynamical constants. They are defined via the dynamical system underlying the BrunGcd algorithm. It is defined on the simplex

$$J_d = \{ x = (x_1, \ldots, x_d \mid 1 \geq x_1 \geq \ldots \geq x_d \geq 0 \}$$

and admits an invariant density defined on $J_d$

$$\Psi_d(x) = \sum_{\sigma \in \mathcal{S}_d} \prod_{i=1}^d \frac{1}{1 + x_{\sigma(1)} + x_{\sigma(2)} + \ldots + x_{\sigma(i)}}$$

Consider the measure $\nu_d$ associated with $\Psi_d$, and the function

$$\mu_d : [0, 1] \to [0, 1], \ y \mapsto \nu_d(yJ_d)$$

$$E_d = (d+1) \int_0^1 \mu_d(y) \frac{dy}{y}, \quad \rho_d = 1 - \mu_d \left( \frac{1}{2} \right), \quad \sigma_d = \sum_{m \geq 1} \mu_d \left( \frac{1}{m} \right)$$
On the number of steps
Gauss map and continued fractions

\( T_G : [0, 1] \to [0, 1], \ x \mapsto \{1/x\}, \) if \( x \neq 0, \) and \( T_G(0) = 0 \)

\[ x = \frac{1}{a_1 + \frac{1}{a_2 + \cdots}} \]

\[ a_n = \left[ \frac{1}{T^{n-1}(x)} \right], \ n \geq 1 \]

\[
\begin{bmatrix}
  x \\
  1
\end{bmatrix} = x \begin{bmatrix}
  0 & 1 \\
  1 & \left\lfloor \frac{1}{x} \right\rfloor
\end{bmatrix} \begin{bmatrix}
  T(x) \\
  1
\end{bmatrix} = \theta(x) \begin{bmatrix}
  0 & 1 \\
  1 & a_1(x)
\end{bmatrix} \begin{bmatrix}
  T(x) \\
  1
\end{bmatrix}
\]

\[ A_n(x) = A(x)A(T(x)) \ldots A(T^{n-1}(x)) \]

\[ \theta_n(x) = \theta(x) \ldots \theta(T^{n-1}(x)) \]

\[ A_n(x) = \begin{bmatrix}
  p_{n-1} & p_n \\
  q_{n-1} & q_n
\end{bmatrix} \]

\[ \theta_n(x) = |q_n x - p_n| \begin{bmatrix}
  x \\
  1
\end{bmatrix} = \theta_n(x)A_n(x) \begin{bmatrix}
  T^n(x) \\
  1
\end{bmatrix} \]
Gauss map and continued fractions

\( T_G : [0, 1] \to [0, 1], \ x \mapsto \{1/x\}, \) if \( x \neq 0, \) and \( T_G(0) = 0 \)

\[
x = \frac{1}{a_1 + \frac{1}{a_2 + \cdots}}
\]

\[
a_n = \left\lfloor \frac{1}{T^{n-1}(x)} \right\rfloor, \quad n \geq 1
\]

\[
\begin{bmatrix}
  x \\
  1
\end{bmatrix} = x
\begin{bmatrix}
  0 & 1 \\
  1 & \frac{1}{x}
\end{bmatrix}
\begin{bmatrix}
  T(x) \\
  1
\end{bmatrix} = \theta(x)
\begin{bmatrix}
  0 & 1 \\
  1 & a_1(x)
\end{bmatrix}
\begin{bmatrix}
  T(x) \\
  1
\end{bmatrix}
\]

\[
A_n(x) = A(x)A(T(x)) \ldots A(T^{n-1}(x)) \quad \theta_n(x) = \theta(x) \ldots \theta(T^{n-1}(x))
\]

\[
A_n(x) = \begin{bmatrix}
  p_{n-1} & p_n \\
  q_{n-1} & q_n
\end{bmatrix}
\theta_n(x) = |q_n x - p_n|
\begin{bmatrix}
  x \\
  1
\end{bmatrix} = \theta_n(x)A_n(x)
\begin{bmatrix}
  T^n(x) \\
  1
\end{bmatrix}
\]

**Thm** For a.e. \( x, \) \( \lim \frac{1}{n} \log q_n = \frac{\pi^2}{12 \log 2} = 1.18 \ldots = \lambda_1 \) first Lyapunov exponent

first Lyapunov exponent = “log largest eigenvalue” \( \sim \) size of the matrices/convergents

\[
A_n(x) \sim q_n(x) \sim e^{\lambda_1 n} \sim \text{Number of steps} = \text{size/ log eigenvalue} = \log N/\lambda_1
\]
Lyapunov exponents and continued fractions

Let $X \subset [0, 1]^{d-1}$

A $d$-dimensional continued fraction map over $X$ is given by measurable maps

$$T: X \rightarrow X, \ A: X \rightarrow GL(d, \mathbb{Z}), \ \theta: X \rightarrow \mathbb{R}_+$$

that satisfy the following: for a.e. $x \in X$, one has

$$\begin{bmatrix}
    x \\
    1
\end{bmatrix} = \theta(x)A(x) \begin{bmatrix}
    T(x) \\
    1
\end{bmatrix}$$

Let

$$A_n(x) = A(x)A(T(x)) \ldots A(T^{n-1}(x)),$$

$$\theta_n(x) = \theta(x) \ldots \theta(T^{n-1}(x))$$

$$\begin{bmatrix}
    x \\
    1
\end{bmatrix} = \theta_n(x)A_n(x) \begin{bmatrix}
    T^n(x) \\
    1
\end{bmatrix}$$

First Lyapunov exponent $\lambda_1 = \log$ eigenvalue $\sim$ size of the matrices $A_n(x) = e^{\lambda_1 n} \sim$ Number of steps $\sim \log N/\lambda_1$
Number of steps $\ell(u, v)$

$\ell(u, v)$: number of steps in Euclid algorithm $0 < v < u$

- **Worst case**

  $\ell(u, v) = O(\log v)$ \quad ($\leq 5 \log_{10} v$, Lamé 1844)

  Reynaud 1821 $[\ell(u, v) < v/2]$, see Shallit’s survey
Number of steps $\ell(u, \nu)$

$\ell(u, \nu)$: number of steps in Euclid algorithm $0 < \nu < u$

- **Worst case**
  
  $$\ell(u, \nu) = O(\log \nu) \quad (\leq 5 \log_{10} \nu, \text{ Lamé 1844})$$

- **Mean case**
  
  $0 < \nu < u \leq N \quad \gcd(u, \nu) = 1$

  $$\mathbb{E}_N(\ell) \sim \frac{12 \log 2}{\pi^2} \cdot \log N$$

[see Knuth, Vol. 2]
Number of steps $\ell(u, v)$

$\ell(u, v)$: number of steps in Euclid algorithm $0 < v < u$

- **Worst case**

  $$\ell(u, v) = O(\log v) \quad (\leq 5 \log_{10} v, \text{ Lamé 1844})$$

- **Mean case** $0 < v < u \leq N$ $\gcd(u, v) = 1$

  $$\frac{12 \log 2}{\pi^2} \cdot \log N + \eta + O(N^{-\gamma})$$

  $\eta$ Porter’s constant

  asymptotically normal distribution

[Heilbronn’69, Dixon’70, Porter’75, Hensley’94, Baladi-Vallée’05…]
Distributional dynamical analysis

\[ \gcd(u_0, u_1) = 1 \quad N \geq u_0 > u_1 > \cdots \quad u_{k-1} = a_k u_k + u_{k+1} \]

Cost of moderate growth \( c(a) = O(\log a) \)

- Number of steps in Euclid algorithm \( c \equiv 1 \)
- Number of occurrences of a quotient \( c = 1_a \)
- Binary length of a quotient \( c(a) = \log_2(a) \)

Theorem [Baladi-Vallée'05]

\[ E\{\text{Cost}\} = 12 \log 2 \pi^2 \cdot \hat{\mu}(\text{Cost}) \cdot \log N + O(1) \]

The distribution is asymptotically Gaussian (CLT)
Distributional dynamical analysis

\[ \gcd(u_0, u_1) = 1 \quad N \geq u_0 > u_1 > \cdots \quad u_{k-1} = a_k u_k + u_{k+1} \]

**Cost** of moderate growth \( c(a) = O(\log a) \)
- Number of steps in Euclid algorithm \( c \equiv 1 \)
- Number of occurrences of a quotient \( c = 1_a \)
- Binary length of a quotient \( c(a) = \log_2(a) \)

**Theorem** [Baladi-Vallée’05]

\[ E_N[\text{Cost}] = \frac{12 \log 2}{\pi^2} \cdot \hat{\mu}(\text{Cost}) \cdot \log N + O(1) \]

The distribution is asymptotically Gaussian (CLT)

Discrete framework-Euclid algorithm
Ergodic theorem

We are given a dynamical system \((X, T, B, \mu)\)

\[ T : X \to X \]

- **Average time values**: one particle over the long term
- **Ergodic theory**
- **Average space values**: all particles at a particular instant, average over all possible sets

\[ \mu(B) = \mu(T^{-1}B) \quad T\text{-invariance} \]

\[ T^{-1}B = B \implies \mu(B) = 0 \text{ or } 1 \quad \text{ergodicity} \]

Ergodic theorem  space mean = average mean

\[ \frac{1}{N} \sum_{0 \leq n \leq N} f(T^n)x = \int f \, d\mu \quad \text{a.e. } x \]
Ergodic theorem

**Theorem** [Baladi-Vallée’05]

\[
\mathbb{E}_N[\text{Cost}] = \frac{12 \log 2}{\pi^2} \cdot \hat{\mu}(\text{Cost}) \cdot \log N + O(1)
\]
Ergodic theorem

**Theorem** [Baladi-Vallée’05]

\[ \mathbb{E}_N[\text{Cost}] = \frac{12 \log 2}{\pi^2} \cdot \hat{\mu}(\text{Cost}) \cdot \log N + O(1) \]

\[ \mathbb{E}_N[c] = \frac{\text{dimension}}{\text{entropy}} \cdot \hat{\mu}(c) \cdot \log N + O(1) \]

\[ \hat{\mu}(c) = \int_0^1 c([1/x]) \cdot \frac{1}{\log 2} \cdot \frac{1}{1 + x} \, dx \]

Continuous framework-truncated trajectories
Cost of truncated trajectories

Cost of moderate growth

\[ c(a_i) = O(\log a_i) \text{ for } a_i \text{ partial quotient} \]

\[ x = \frac{1}{a_1 + \frac{1}{\frac{1}{a_2 + \frac{1}{a_3 + \cdots}}}} \]
Cost of truncated trajectories

Cost of moderate growth

\[ c(a_i) = O(\log a_i) \text{ for } a_i \text{ partial quotient} \]

Cost of a truncated trajectory

\[ C_n(x) = \sum_{i=1}^{n} c(a_i(x)) \quad a_i = \left[ \frac{1}{T^{i-1}(x)} \right] \]

According to the ergodic theorem, for a.e. \( x \in [0, 1] \)

\[ C_n(x)/n \to \widehat{\mu}(x) \]

\[ \widehat{\mu}(C) = \int_{0}^{1} c \left( \left\lfloor 1/x \right\rfloor \right) \cdot \frac{1}{\log 2} \cdot \frac{1}{1 + x} \cdot dx \]

\[ \mathbb{E}_N[C] = \frac{2}{\pi^2/(6 \log 2)} \cdot \widehat{\mu}(C) \cdot \log N \]
Multidimensional Euclid’s algorithms and continued fractions

- **Jacobi-Perron** We subtract the first one to the two other ones with $u_0 \geq u_1, u_2 \geq 0$

$$ (u_0, u_1, u_2) \mapsto (u_2, u_0 - \left\lfloor \frac{u_0}{u_2} \right\rfloor u_2, u_1 - \left\lfloor \frac{u_1}{u_2} \right\rfloor u_2) $$

- **Brun** We subtract the second largest entry and we reorder. If $u_0 \geq u_1 \geq u_2 \geq 0$

$$ (u_0, u_1, u_2) \mapsto (u_0 - u_1, u_1, u_2) $$

- **Poincaré** We subtract the previous entry and we reorder

$$ (u_0, u_1, u_2) \mapsto (u_0 - u_1, u_1 - u_2, u_2) $$

- **Selmer** We subtract the smallest to the largest and we reorder

$$ (u_0, u_1, u_2) \mapsto (u_0 - u_2, u_1, u_2) $$

- **Fully subtractive** We subtract the smallest one to the other ones and we reorder

$$ (u_0, u_1, u_2) \mapsto (u_0 - u_2, u_1 - u_2, u_2) $$
Number of steps for the Euclid algorithm

Consider

$$\Omega_m := \{(u_1, u_2) \in \mathbb{N}^2, \ 0 \leq u_1, u_2 \leq m\}$$

endowed with the uniform distribution

- Theorem  The mean value $\mathbb{E}_m[L]$ of the number of steps satisfies

$$\mathbb{E}_m[L] \sim \frac{2}{\pi^2/(6 \log 2)} \log m = \frac{1}{\lambda_1} \log m$$

$\lambda_1$ is the first Lyapunov exponent of the Gauss map

$\pi^2/(6 \log 2)$ is the entropy

[Heilbronn’69, Dixon’70, Hensley’94, Baladi-Vallée’03...]

Consider parameters $(u_1, \cdots, u_d)$ with $0 \leq u_1, \cdots, u_d \leq m$

To be expected

$$E_m[L] \sim \frac{\text{dimension}}{\text{Entropy}} \times \log m = \frac{1}{\text{first Lyapounov exponent}} \times \log m$$

The first Lyapounov exponent governs the growth of the denominators of the convergents $q_n$
Comparison of gcd algorithms

We consider three Euclid algorithms for polynomials in $\mathbb{F}_q[X]$

$$\Omega := \{ R = (R_1, R_2, R_3) \mid \deg R_3 > \max(\deg R_1, \deg R_2), \ R_3 \text{ monic} \}$$

- One chooses one **specific component**. This is
  - the first component for the Jacobi-Perron algorithm
  - the second largest component for the Brun algorithm
  - and the smallest component for the Fully Subtractive algorithm

- Each algorithm divides the other two components by this specific component, and replaces these components by their remainders in the division by the specific component.

- After having performed these divisions, this specific component becomes the largest one, and it is thus placed at the third position.

The algorithm stops when there remains only one non-zero component. This is the **gcd**.
Costs

**Theorem [B.-Nakada-Natsui-Vallée]**

\[ \Omega_m := \{ R = (R_1, R_2, R_3) \mid m = \deg R_3 > \max(\deg R_1, \deg R_2) \} \]

- Number of steps
  \[ \frac{3}{\text{Entropy}} \cdot m \]
- Bit-complexity
  Quadratic \( m^2 \)  Brun < Jacobi-Perron < Fully Subtractive
- Fine bit-complexity (non-zero terms)
  We find the same value for the three algorithms!
  \[ \frac{3(q - 1)}{2q} \cdot m^2 \]
On Knuth gcd algorithm
Knuth gcd algorithm

Consider the input \((u_0, u_1, \ldots, u_d)\)

- \(v_0 := u_0\)
- For \(k \in [1..d]\), one successively computes

\[
v_k := \gcd(u_k, v_{k-1}) = \gcd(u_0, u_1, \ldots, u_k)
\]

The total gcd \(v_d := \gcd(u_0, u_1, \ldots, u_d)\) is obtained after \(d\) phases.

One performs a sequence of \(d\) gcd computations on two entries.
Knuth gcd algorithm

Consider the input \((u_0, u_1, \ldots, u_d)\)

- \(v_0 := u_0\)
- For \(k \in [1..d]\), one successively computes

\[
v_k := \gcd(u_k, v_{k-1}) = \gcd(u_0, u_1, \ldots, u_k)
\]

The total gcd \(v_d := \gcd(u_0, u_1, \ldots, u_d)\) is obtained after \(d\) phases

One performs a sequence of \(d\) gcd computations on two entries

The same formal scheme can be applied to
- positive integers
- polynomials with coefficients in \(\mathbb{F}_q\)
The following figure compares the number of steps of the BrunGcd and the PlainGcd algorithms, as a function of dimension $d$, when the binary size is fixed to $\log_2 N = 5000$.

- The complexity of BrunGcd algorithm appears to be sublinear with respect to $d$.
- The complexity of the PlainGcd algorithm appears to be independent of $d$.  

![Comparison PlainGcd/BrunGcd in function of the dimension (N is fixed)](image-url)
Number of steps for Knuth gcd algorithm

A different notion of size

\[ \Omega'_{(d,N)} := \{(u_0, \ldots, u_d) \mid u_0 u_1 \ldots u_d \leq N\} \]

The expectation of the number of steps \( L_d \) during the first phase is linear with respect to the size \( N \) and satisfies

\[ \mathbb{E}_N[L_d] \sim \frac{6 \log 2}{\pi^2} \cdot \frac{\log N}{(d + 1)} \]

First phase linear on average
For the other phases \( k \geq 2 \) constant in average
Almost all the calculation is done during the first phase
Analogous results for formal power series with coefficients in a finite field
Average-case and distributional analysis

[B.-Creusefond-Lhote-Vallée], ISSAC 13
Comparison of gcd algorithms

- Brun algorithm for $d + 1$ integers

  Number of steps \( \mathbb{E}_N[L] \sim \frac{d + 1}{\mathcal{E}^B_d} \cdot \log N \)

  Entropy \( \mathcal{E}^B_d \sim \log d \)

- Knuth algorithm

  Number of steps \( \mathbb{E}_N[L] \sim \frac{1}{\mathcal{E}^K_2} \cdot \frac{\log N}{(d + 1)} \)

  Entropy \( \mathcal{E}^K_2 = \pi^2/(6 \log 2) \)

😊 For Brun algorithm, \( \log N \) is the size of the maximal input, whereas for Knuth algorithm, \( \log N \) is the cumulative size
Method
Method

- A bijection between the set of entries and the sets of quotients together with possible insertion places and gcd’s.

  Inputs \( \sim \) quotients \( \times \) possible insertion places \( \times \) gcd

- Expression of associated Dirichlet series in terms of transfer operators of the dynamical system which highlight the singularities

- This proves in particular that the first phase dominates (dominant singularity)

- We use a Delange type theorem
Brun dynamical system

A continuous extension of the algorithm that provides an exact characterization of the trajectories that are related to the execution of the algorithm. It acts on the simplex $\mathcal{J}(d) \subset \mathbb{R}^d$

$$\mathcal{J}(d) := \{ \mathbf{x} = (x_1, \ldots, x_d) \mid 1 \geq x_1 \geq \ldots \geq x_d \geq 0 \}$$

$$T_d(0^d) = 0^d, \quad T_d(\mathbf{x}) = \text{Ins} \left( \left\{ \frac{1}{x_1} \right\}, \frac{1}{x_1} \text{End} \mathbf{x} \right) \quad \text{for } \mathbf{x} \neq 0^d$$

The algorithm BrunSD($d$) The map $\text{Ins}(y_0, \mathbf{y})$ is the insertion “in front of”, with two cases:

- $(G)$ if $y_0$ is not present in the list $\mathbf{y}$, this is an usual insertion;
- $(P)$ if $y_0$ is already present in the list $\mathbf{y}$, we insert $y_0$ in front of the block of components equal to $y_0$.

We use here the existence of an ergodic absolutely continuous invariant measure, and contraction properties of Brun Dynamical system [Broise]
Transfer operators and Gauss map $T: x \mapsto \{1/x\}$

Perron-Frobenius operator  
Think of $f$ as a density function

$$P[f](x) = \sum_{y: T(y) = x} \frac{1}{|T'(y)|} f(y) = \sum_{k \geq 1} \left( \frac{1}{k + x} \right)^2 f \left( \frac{1}{k + x} \right)$$

Let $\mathcal{H}$ stand for the set of inverse branches of the Gauss map

$$P[f](x) = \sum_{h \in \mathcal{H}} h'(x) \cdot f \circ h(x)$$
Transfer operators and Gauss map $T : x \mapsto \{1/x\}$

Perron-Frobenius operator Think of $f$ as a density function

$$P[f](x) = \sum_{y : T(y) = x} \frac{1}{|T'(y)|} f(y) = \sum_{k \geq 1} \left( \frac{1}{k + x} \right)^2 f \left( \frac{1}{k + x} \right)$$

Let $\mathcal{H}$ stand for the set of inverse branches of the Gauss map

$$P[f](x) = \sum_{h \in \mathcal{H}} h'(x) \cdot f \circ h(x)$$

Ruelle operator

$$P_s[f](x) = \sum_{h \in \mathcal{H}} h'(x)^s \cdot f \circ h(x) \quad s \in \mathbb{C}$$

Dirichlet series

Take $x = 0$, $f = 1$ $\sim_{\mathcal{H}^*} (\text{Id} - P_s)^{-1} \sim \sum_{\ell \geq 1} 1/\ell^{2s}$ 😊
Transfer operators and Gauss map $T: x \mapsto \{1/x\}$

**Perron-Frobenius operator**  
Think of $f$ as a density function

$$P[f](x) = \sum_{y : T(y) = x} \frac{1}{|T'(y)|} f(y) = \sum_{k \geq 1} \left( \frac{1}{k + x} \right)^2 f \left( \frac{1}{k + x} \right)$$

Let $\mathcal{H}$ stand for the set of inverse branches of the Gauss map

$$P[f](x) = \sum_{h \in \mathcal{H}} h'(x) \cdot f \circ h(x)$$

**Ruelle operator**

$$P_s[f](x) = \sum_{h \in \mathcal{H}} h'(x)^s \cdot f \circ h(x) \quad s \in \mathbb{C}$$

**Dirichlet series**

Take $x = 0$, $f = 1 \leadsto_{\mathcal{H}^*} (\text{Id} - P_s)^{-1} \leadsto \sum_{\ell \geq 1} 1/\ell^{2s}$

Involving additive costs

$$P_{s,w}[f](x) = \sum_{h \in \mathcal{H}} h'(x)^s \cdot w^{c(h)} \cdot f \circ h(x)$$
Transfer operators and Brun algorithm

Each step of the algorithm is a linear fractional transformation

Let \( h \) be an inverse branch and \( J[h] \) its Jacobian

\[
P_s[f](x) = \sum_{h \in \mathcal{H}} J[h](x)^s \cdot f \circ h(x)
\]

\[
T(x) = \text{Ins} \left( \left\{ \frac{1}{x_1} \right\}, \left( \frac{x_1}{x_2}, \ldots, \frac{x_d}{x_1} \right) \right)
\]

\[
m(x) = \begin{bmatrix} 1 \\ \frac{1}{x_1} \end{bmatrix}, \quad j(x) = \text{Pos} \left( \left\{ \frac{1}{x_1} \right\}, \left( \frac{x_2}{x_1}, \ldots, \frac{x_d}{x_1} \right) \right)
\]

Inverse branch

\[
h_{(m,j)}(y_1, y_2, \ldots, y_d) = \left( \frac{1}{m+y_j}, \frac{y_1}{m+y_j}, \ldots, \frac{y_{j-1}}{m+y_j}, \frac{y_j+1}{m+y_j}, \ldots, \frac{y_d}{m+y_j} \right)
\]

Jacobian

\[
J[h_{(m,j)}](y) = \frac{1}{(m+y_j)^{d+1}} \quad \sim \mathcal{H}^* \quad J[h](0) = \frac{1}{u_0^{d+1}}
\]
Generating functions and transfer operators

\[ \mathbf{u} = (u_0, u_1, \ldots, u_d), \quad u_0 > u_1 > \cdots > u_d > 0, \quad \|\mathbf{u}\| := u_0 \]

Dirichlet series

\[ \sum_{\mathbf{u}} \frac{C(\mathbf{u})}{\|\mathbf{u}\|^s} = \sum_{n \geq 1} n^{-s} \sum_{\|\mathbf{u}\|=n} C(\mathbf{u}) \]

We then introduce a further indeterminate \( w \)

\[ \sum_{\mathbf{u}} \frac{w^{C(\mathbf{u})}}{\|\mathbf{u}\|^s} \]

The derivative w.r.t. \( w \) at \( w = 1 \) yields cumulative generating functions
Generating functions and transfer operators

Generating function \[ \sum_{u} \frac{w^{C(u)}}{||u||^s} \]

Operator \( P_{s,w}[f](x) = \sum_{h \in H} J[h](x)^s \cdot w^{c(h)} \cdot f \circ h(x) \)

Jacobian \( J[h](0) = \frac{1}{||u||^{d+1}} \)

For the number of steps \( C \), take \( x = 0 \), \( f = 1 \), \( c = 1 \), and \( \frac{\partial}{\partial w} \big|_{w=1} \)

\[ \sum_{u} \frac{C(u)}{||u||^s} \xrightarrow{h \in H^*} (\text{Id} - P_{s,w})^{-1}[1](0) \xrightarrow{\text{Perron-Frobenius}} \frac{1}{1 - \lambda_s} \]

Singularity for \( s \) such that \( \lambda_s = 1 \) with \( \lambda_s \) dominant eigenvalue of the operator \( P_s \) (cf. invariant measure)
Branches and inverse branches

For any \( x \in J(d) \), the map \( T(d) \) uses a digit

\[(m, j) \in A(d) := \mathbb{N}^* \times [1..d]\]

with a quotient \( m(x) \geq 1 \) and an insertion index \( j(x) \in [1..d] \).

Let \( K_{(d,m,j)} := \{ x \in J(d) \mid m(x) = m, \ j(x) = j \} \)

When \( (m, j) \) varies in \( A(d) \)

– the subsets \( K_{(d,m,j)} \) form a topological partition of \( J(d) \)

– the restriction \( T_{(d,m,j)} \) of \( T(d) \) to \( K_{(d,m,j)} \) is a bijection from \( K_{(d,m,j)} \) onto \( J(d) \)

\[
T_{(d,m,j)}(x_1, x_2, \ldots, x_d) = \left( \frac{x_2}{x_1}, \ldots, \frac{x_{j-1}}{x_1}, \frac{1}{x_1} - m, \frac{x_{j+1}}{x_1}, \ldots, \frac{x_d}{x_1} \right)
\]

Its inverse is a bijection from \( J(d) \) onto \( K_{(d,m,j)} \)

\[
h_{(d,m,j)}(y_1, \ldots, y_d) = \left( \frac{1}{m+y_j}, \frac{y_1}{m+y_j}, \ldots, \frac{y_{j-1}}{m+y_j}, \frac{y_{j+1}}{m+y_j}, \ldots, \frac{y_d}{m+y_j} \right)
\]
The Brun Perron–Frobenius operator

\[ H_{(d)}[f](x) = \sum_{h \in \mathcal{H}_{(d)}} |J[h](x)| f \circ h(x) \]

A convenient functional space is \( C^1(\mathcal{J}_{(d)}), \| \cdot \|_1 \)

\[ \|f\|_1 = \sup_{x \in \mathcal{J}_{(d)}} |f(x)| + \sup_{x \in \mathcal{J}_{(d)}} \|Df(x)\| \]

\( Df(x) \) is the differential of \( f \) at \( x \) and \( \| \cdot \| = \) a norm on \( \mathbb{R}^d \)

\( H_{(d)} \) acts on \( (C^1(\mathcal{J}_{(d)}), \| \cdot \|_1) \) and is quasi-compact: the “upper” part of its spectrum is formed with isolated eigenvalues of finite multiplicity. The quasi-compacity is due to:

- A contraction ratio

\[ \tau_d := \lim_{n \to \infty} \sup_{h \in \mathcal{H}_{(d)}^n} \sup_{x \in \mathcal{J}_{(d)}} \left\| Dh(x) \right\|^{1/n} < 1 \]

\( \tau_d \) is the smallest real root of \( z^{d+1} + z - 1 = 0 \)

- A distortion constant

\[ \exists L > 0, \quad \| DJ[h](x) \| \leq L |J[h](x)|, \quad \forall h \in \mathcal{H}_{(d)}^*, \forall x \in \mathcal{J}_{(d)} \]
Spectral properties of $H_d$ acting on $\mathcal{C}^1(J_d)$

- $\lambda = 1$ is the unique simple dominant eigenvalue of maximum modulus, isolated from the remainder of the spectrum by a spectral gap
- The dominant eigenfunction is explicit

$$\psi_d(x) = \sum_{\sigma \in S_d} \prod_{i=1}^{k} \frac{1}{1 + x_{\sigma(1)} + x_{\sigma(2)} + \ldots + x_{\sigma(i)}}$$

- Except for small $d$, there is no explicit expression known for the integral

$$\kappa_d := \int_{J_d} \psi_d(x) \, dx$$

The invariant density $\Psi_d$ and the invariant measure $\nu_d$ are not explicit.
Conclusion and future work

- We have used the Brun underlying dynamical system to describe the probabilistic behaviour of the BrunGcd algorithm.
- We have studied the asymptotics (for $d \to \infty$) of the main constants that intervene in the analysis.
- We conclude that the BrunGcd algorithm is less efficient than the Knuth gcd algorithm.
- This is probably the case for all the gcd algorithms which are based on multidimensional continued fraction algorithms.
- We plan to study other costs such as the bit-complexity or to perform a distributional analysis.
- More needs for the properties of dynamical systems.

We plan to study finite and periodic trajectories.

We want to conduct a systematic comparison of continued fraction algorithms with respect to Lyapunov exponents.

We plan to analyze the extended gcd algorithm based on the LLL algorithm, even if its underlying system is quite complex to deal with.