MULTIPLE GCDs:
Probabilistic analysis of the plain algorithm
The number case.

Valérie Berthé, Loïck Lhote, Brigitte Vallée

Work in progress...

Rencontre du projet DynA3S,
Octobre 2013, Paris.

Computing GCDs of d inputs

For $d = 2$: the "classical" Euclid algorithm,

For $d \geq 3$, there are various strategies.

The plain algorithm performs a sequence of computations on two entries;

On the input $(x_1, x_2, \ldots, x_d)$, it computes

- first: $y_2 := \gcd(x_1, x_2)$
- then, for $k \in [3..d]$: $y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k)$.

The "total" gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$ is obtained after $d - 1$ phases.

Each phase performs a call to the classical Euclid algorithm.

Computing GCDs of d inputs

For $d = 2$: the "classical" Euclid algorithm,

For $d \geq 3$, there are various strategies.

The plain algorithm performs a sequence of computations on two entries;

On the input $(x_1, x_2, \ldots, x_d)$, it computes

- first: $y_2 := \gcd(x_1, x_2)$
- then, for $k \in [3..d]$: $y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k)$.

The "total" gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$ is obtained after $d - 1$ phases.

Each phase performs a call to the classical Euclid algorithm.

The same formal scheme

- for polynomials over a finite field: $\mathbb{F}_q[X]$
- for numbers : positive integers.

Computing GCDs of d inputs

For $d = 2$: the "classical" Euclid algorithm,

For $d \geq 3$, there are various strategies.

The plain algorithm performs a sequence of computations on two entries;

On the input $(x_1, x_2, \ldots, x_d)$, it computes

– first: $y_2 := \gcd(x_1, x_2)$

– then, for $k \in [3..d]$: $y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k)$.

The "total" gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$ is obtained after $d - 1$ phases.

Each phase performs a call to the classical Euclid algorithm.

The same formal scheme

– for polynomials over a finite field: $\mathbb{F}_q[X]$

– for numbers : positive integers.

A very natural scheme, proposed in Knuth's book, but not yet analyzed.

In this talk, – we focus on the analysis of the number case,

– we explain the similarities/differences between the two cases.

Which behavior can be expected?

Knuth wrote: "In most cases, the length of the partial gcd decreases rapidly during the first few phases of the calculation.
This will make the remainder of the computation quite fast".

Which behavior can be expected?

Knuth wrote: "In most cases, the length of the partial gcd decreases rapidly during the first few phases of the calculation.
This will make the remainder of the computation quite fast".

Our analysis exhibits a more precise phenomenon:
A strong difference between the first phase and the subsequent phases.
In most cases, "almost all the calculation" is done during the first phase.

Which behavior can be expected?

Knuth wrote: "In most cases, the length of the partial gcd decreases rapidly during the first few phases of the calculation.
This will make the remainder of the computation quite fast".

Our analysis exhibits a more precise phenomenon:
A strong difference between the first phase and the subsequent phases.
In most cases, "almost all the calculation" is done during the first phase.

In the integer case, we prove the following facts about the number of divisions performed, measured with respect to the size of the input:

- during the first phase:
  - it is linear on average,
  - it asymptotically follows a QUASI–beta law;

Which behavior can be expected?

Knuth wrote: "In most cases, the length of the partial gcd decreases rapidly during the first few phases of the calculation.
This will make the remainder of the computation quite fast".

Our analysis exhibits a more precise phenomenon:
A strong difference between the first phase and the subsequent phases.
In most cases, "almost all the calculation" is done during the first phase.

In the integer case, we prove the following facts about the number of divisions performed, measured with respect to the size of the input:

- during the first phase:
  - it is linear on average,
  - it asymptotically follows a QUASI–beta law;
- during subsequent phases:
  - it is constant in average
  - it asymptotically follows a QUASI–geometric law

Which behavior can be expected?

Knuth wrote: "In most cases, the length of the partial gcd decreases rapidly during the first few phases of the calculation.
This will make the remainder of the computation quite fast".

Our analysis exhibits a more precise phenomenon:
A strong difference between the first phase and the subsequent phases.
In most cases, "almost all the calculation" is done during the first phase.

In the integer case, we prove the following facts about the number of divisions performed, measured with respect to the size of the input:

- during the first phase:
  - it is linear on average,
  - it asymptotically follows a QUASI–beta law;
- during subsequent phases:
  - it is constant in average
  - it asymptotically follows a QUASI–geometric law

The same phenomena occur for the size of the partial gcd.

Probabilistic analysis of the plain d–GCD algorithm on $\mathbb{Z}$.

On the input $(x_1, x_2, \ldots, x_d)$,

- the algorithm computes the total gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$
- with $d-1$ phases.
- The k-th phase computes the k–th gcd,

$$y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k) \ .$$

Probabilistic analysis of the plain d–GCD algorithm on $\mathbb{Z}$.

On the input $(x_1, x_2, \ldots, x_d)$,

- the algorithm computes the total gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$
- with $d-1$ phases.
- The k-th phase computes the k–th gcd,

$$y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k) \ .$$

- each phase performs the classical Euclid algorithm

via a sequence of Euclidean divisions

Probabilistic analysis of the plain d–GCD algorithm on $\mathbb{Z}$.

On the input $(x_1, x_2, \ldots, x_d)$,

- the algorithm computes the total gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$
- with $d-1$ phases.
- The k-th phase computes the k–th gcd,

$$y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k) \ .$$

- each phase performs the classical Euclid algorithm

via a sequence of Euclidean divisions

The set of inputs is $\Omega = \{\underline{x} := (x_1, \ldots, x_d); \quad x_i \in \mathbb{N}\}$

The size of an input: $\mathrm{d}(\underline{x}) := \mathrm{d}(x_1 x_2 \ldots x_d)$ with $\mathrm{d}(x) := \lceil \log x \rceil$

"almost additive" $\mathrm{d}(\underline{x}) \approx \mathrm{d}(x_1) + \ldots + \mathrm{d}(x_d)$

Probabilistic analysis of the plain d–GCD algorithm on $\mathbb{Z}$.

On the input $(x_1, x_2, \ldots, x_d)$,

- the algorithm computes the total gcd $y_d := \gcd(x_1, x_2, \ldots, x_d)$
- with $d-1$ phases.
- The k-th phase computes the k–th gcd,

$$y_k := \gcd(x_k, y_{k-1}) = \gcd(x_1, x_2, \ldots, x_k) \ .$$

- each phase performs the classical Euclid algorithm

via a sequence of Euclidean divisions

The set of inputs is $\Omega = \{\underline{x} := (x_1, \ldots, x_d); \quad x_i \in \mathbb{N}\}$

The size of an input: $\mathrm{d}(\underline{x}) := \mathrm{d}(x_1 x_2 \ldots x_d)$ with $\mathrm{d}(x) := \lceil \log x \rceil$

"almost additive" $\mathrm{d}(\underline{x}) \approx \mathrm{d}(x_1) + \ldots + \mathrm{d}(x_d)$

Main parameters of interest

- the number L_k of divisions during the k–th phase

i.e. on the input (x_k, y_{k-1})

- the size D_k of the k–th gcd

(at the beginning of the k-th phase).

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

`Euclid`(a_1, a_2), [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < \ a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

$\texttt{Euclid}(a_1, a_2)$, [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < \ a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The Euclid Algorithm is then extended to the case when $a_1 < a_2$, by letting $\texttt{Euclid}(a_1, a_2) := \texttt{Euclid}(a_2, a_1)$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

$\texttt{Euclid}(a_1, a_2)$, [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < \ a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The Euclid Algorithm is then extended to the case when $a_1 < a_2$, by letting $\texttt{Euclid}(a_1, a_2) := \texttt{Euclid}(a_2, a_1)$

The pair (a_1, a_2) of positive integers is entirely determined by

– the sequence of quotients $(m_1, m_2, \ldots, m_r)$,

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

$\texttt{Euclid}(a_1, a_2)$, [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < \ a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The Euclid Algorithm is then extended to the case when $a_1 < a_2$, by letting $\texttt{Euclid}(a_1, a_2) := \texttt{Euclid}(a_2, a_1)$

The pair (a_1, a_2) of positive integers is entirely determined by

- the sequence of quotients $(m_1, m_2, \ldots, m_r)$, where
  - the first quotient m_1 is positive, with

$$m_1 \geq 0 \ [a_1 \geq a_2] \text{ or } m_1 \geq 1 \ [a_1 < a_2]$$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

$\texttt{Euclid}(a_1, a_2)$, [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < \ a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The Euclid Algorithm is then extended to the case when $a_1 < a_2$, by letting $\texttt{Euclid}(a_1, a_2) := \texttt{Euclid}(a_2, a_1)$

The pair (a_1, a_2) of positive integers is entirely determined by

- the sequence of quotients $(m_1, m_2, \ldots, m_r)$, where
  - the first quotient m_1 is positive, with
    $$m_1 \geq 0 \ [a_1 \geq a_2] \text{ or } m_1 \geq 1 \ [a_1 < a_2]$$
  - any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

$\texttt{Euclid}(a_1, a_2)$, [case $a_1 \geq a_2$].

$$
\begin{array}{lcllll}
a_1 & = & m_1\, a_2 & + & a_3 & 0 < a_3 < a_2 \\
a_2 & = & m_2\, a_3 & + & a_4 & 0 < a_4 < a_3 \\
\ldots & = & \ldots & + & & \\
a_{r-1} & = & m_{r-1}\, a_r & + & a_{r+1} & 0 < a_{r+1} < a_r \\
a_r & = & m_r\, a_{r+1} & + & 0 &
\end{array}
$$

The last non zero remainder is the gcd y. Here $y = a_{r+1}$.

The Euclid Algorithm is then extended to the case when $a_1 < a_2$, by letting $\texttt{Euclid}(a_1, a_2) := \texttt{Euclid}(a_2, a_1)$

The pair (a_1, a_2) of positive integers is entirely determined by

- the sequence of quotients $(m_1, m_2, \ldots, m_r)$, where
  - the first quotient m_1 is positive, with
    $$m_1 \geq 0 \ [a_1 \geq a_2] \text{ or } m_1 \geq 1 \ [a_1 < a_2]$$
  - any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$
- the gcd $y = a_{r+1}$ satisfies $y \geq 1$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The pair (a_1, a_2) of positive integers is entirely determined by

– the first quotient m_1 with $m_1 \geq 0$ $[a_1 \geq a_2]$ or $m_1 \geq 1$ $[a_1 < a_2]$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The pair (a_1, a_2) of positive integers is entirely determined by

- the first quotient m_1 with $m_1 \geq 0$ $[a_1 \geq a_2]$ or $m_1 \geq 1$ $[a_1 < a_2]$
- any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The pair (a_1, a_2) of positive integers is entirely determined by

- the first quotient m_1 with $m_1 \geq 0$ $[a_1 \geq a_2]$ or $m_1 \geq 1$ $[a_1 < a_2]$
- any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$
- the gcd y satisfies $y \geq 1$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The pair (a_1, a_2) of positive integers is entirely determined by

- the first quotient m_1 with $m_1 \geq 0$ $[a_1 \geq a_2]$ or $m_1 \geq 1$ $[a_1 < a_2]$
- any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$
- the gcd y satisfies $y \geq 1$

`Euclid`(a_1, a_2), [case $a_1 \geq a_2$].

$$y := \gcd(a_1, a_2); \qquad a_1 = u_1\, y, \quad a_2 = u_2\, y, \quad a_3 = u_3\, y$$

$$\gcd(u_1, u_2) = 1, \gcd(u_2, u_3) = 1$$

$$\frac{u_2}{u_1} = \cfrac{1}{m_1 + \cfrac{1}{m_2 + \cfrac{1}{m_3 + \cfrac{1}{\ddots + \cfrac{1}{m_r}}}}}, \qquad \frac{u_3}{u_2} = \cfrac{1}{m_2 + \cfrac{1}{m_3 + \cfrac{1}{\ddots + \cfrac{1}{m_r}}}},$$

$$\frac{u_2}{u_1} = h_{m_1} \circ h_{m_2} \circ \ldots \circ h_{m_r}(0), \qquad \frac{u_3}{u_2} = h_{m_2} \circ \ldots \circ h_{m_r}(0)$$

$$\frac{1}{u_1^2} = (h_{m_1} \circ h_{m_2} \circ \ldots \circ h_{m_r})'(0) \qquad \frac{1}{u_1^2} = (h_{m_2} \circ \ldots \circ h_{m_r})'(0)$$

The combinatorial bijection induced by the Euclid Algorithm $[d = 2]$

The pair (a_1, a_2) of positive integers is entirely determined by

- the first quotient m_1 with $m_1 \geq 0$ $[a_1 \geq a_2]$ or $m_1 \geq 1$ $[a_1 < a_2]$
- any quotient m_i for $i \in [2..r]$ satisfies $m_i \geq 1$
- the gcd y satisfies $y \geq 1$

`Euclid`(a_1, a_2), [case $a_1 \geq a_2$].

$$y := \gcd(a_1, a_2); \qquad a_1 = u_1\, y, \quad a_2 = u_2\, y, \quad a_3 = u_3\, y$$

$$\gcd(u_1, u_2) = 1, \gcd(u_2, u_3) = 1$$

$$\frac{u_2}{u_1} = \cfrac{1}{m_1 + \cfrac{1}{m_2 + \cfrac{1}{m_3 + \cfrac{1}{\ddots + \cfrac{1}{m_r}}}}}, \qquad \frac{u_3}{u_2} = \cfrac{1}{m_2 + \cfrac{1}{m_3 + \cfrac{1}{\ddots + \cfrac{1}{m_r}}}},$$

$$\frac{u_2}{u_1} = h_{m_1} \circ h_{m_2} \circ \ldots \circ h_{m_r}(0), \qquad \frac{u_3}{u_2} = h_{m_2} \circ \ldots \circ h_{m_r}(0)$$

$$\frac{1}{u_1^2} = (h_{m_1} \circ h_{m_2} \circ \ldots \circ h_{m_r})'(0) \qquad \frac{1}{u_1^2} = (h_{m_2} \circ \ldots \circ h_{m_r})'(0)$$

Dirichlet generating functions relative to the Euclid algorithm $(d = 2)$.

Dirichlet generating functions relative to the Euclid algorithm $(d = 2)$.

The Dirichlet generating function of the inputs is

$$\sum_{(a_1,a_2)\in\mathbb{N}^2} \frac{1}{a_1^{s_1}} \frac{1}{a_2^{s_2}} = \zeta(s_1)\cdot\zeta(s_2), \qquad \zeta(s) = \sum_{p\in\mathbb{N}} \frac{1}{p^s},$$

where the Riemann zeta function $\zeta(s)$ is the gen. function of the set $\mathbb{N}$.

With the $\gcd y$

$$\zeta(s_1)\zeta(s_2) = \left(\sum_{y\geq 1} \frac{1}{y^{s_1+s_2}}\right) \sum_{\substack{(u_1,u_2)\in\mathbb{N}^2 \\ \gcd(u_1,u_2)=1}} \frac{1}{u_1^{s_1}} \frac{1}{u_2^{s_2}}$$

Dirichlet generating functions relative to the Euclid algorithm $(d = 2)$.

The Dirichlet generating function of the inputs is

$$\sum_{(a_1,a_2)\in\mathbb{N}^2} \frac{1}{a_1^{s_1}} \frac{1}{a_2^{s_2}} = \zeta(s_1)\cdot\zeta(s_2), \qquad \zeta(s) = \sum_{p\in\mathbb{N}} \frac{1}{p^s},$$

where the Riemann zeta function $\zeta(s)$ is the gen. function of the set $\mathbb{N}$. With the $\gcd y$

$$\zeta(s_1)\zeta(s_2) = \left(\sum_{y\geq 1} \frac{1}{y^{s_1+s_2}}\right) \sum_{\substack{(u_1,u_2)\in\mathbb{N}^2 \\ \gcd(u_1,u_2)=1}} \frac{1}{u_1^{s_1}} \frac{1}{u_2^{s_2}}$$

With the previous decomposition with the LFT's

$$2 \sum_{(u_1,u_2)} \frac{1}{u_1^{s_1}} \frac{1}{u_2^{s_2}} = (I - \mathbf{G}_{s_1+s_2})^{-1} \circ (\mathbf{G}_{s_1} + \mathbf{G}_{s_2})[1](0)$$

where $\mathbf{G}_s$ is the generating operator for the quotients,

$$\mathbf{G}_s[f](x) = \sum_{m\geq 1} |h_m'(x)|^{s/2} f\circ h_m(x) = \sum_{m\geq 1} \left(\frac{1}{m+t}\right)^s f\left(\frac{1}{m+x}\right)$$

The Euclid algorithm ($d = 2$) on integers

translates as a product of Dirichlet generating functions

$$2\zeta(s)\,\zeta(t) = \zeta(s+t) \cdot \left[(I - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)[1](0)\right]$$

The Euclid algorithm ($d = 2$) on integers

translates as a product of Dirichlet generating functions

$$2\zeta(s)\,\zeta(t) = \zeta(s+t) \cdot \left[(I - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)[1](0)\right]$$

which involve – the Riemann Dirichlet series $\zeta(s) = \sum_{p \geq 1} p^{-s}$

– the generating operator $\mathbf{G}_s$ for the quotients

The Euclid algorithm ($d = 2$) on integers

translates as a product of Dirichlet generating functions

$$2\zeta(s)\,\zeta(t) = \zeta(s+t) \cdot \left[(I - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)[1](0) \right]$$

which involve – the Riemann Dirichlet series $\zeta(s) = \sum_{p \geq 1} p^{-s}$

– the generating operator $\mathbf{G}_s$ for the quotients

This is a functional operator which depends on a complex parameter s,

$$\mathbf{G}_s[f](t) = \sum_{m \geq 1} \left(\frac{1}{m+t} \right)^s f\left(\frac{1}{m+t} \right)$$

This is the transfer operator of the Euclidean underlying dynamical system,

The analysis is more involved than the previous one,

but provides the same type of results.

Similarities and differences between the two analyses

	Polynomials	Integers
GF	Power GF	Dirichlet GF and operators
Basic tool	$G(z) = \sum_m z^{\mathrm{d}(m)}$	$\mathbf{G}_s[f](t) = \sum_{m\geq 1} \left(\frac{1}{m+t}\right)^s f\left(\frac{1}{m+t}\right)$
Phase GF	$\frac{U(z^k)+U(z)-1}{1-G(z^{k+1})}$	$(I-\mathbf{G}_{(k+1)s})^{-1} \circ [\mathbf{G}_{ks}+\mathbf{G}_s]$
Singularities	z s.t. $G(z^{k+1})=1$	s s.t. $\lambda((k+1)s)=1$
Extraction	Cauchy Formula	Perron Formula
Contours	Disks	Vertical lines

Similarities and differences between the two analyses

	Polynomials	Integers
GF	Power GF	Dirichlet GF and operators
Basic tool	$G(z) = \sum_m z^{\mathrm{d}(m)}$	$\mathbf{G}_s[f](t) = \sum_{m\geq 1} \left(\frac{1}{m+t}\right)^s f\left(\frac{1}{m+t}\right)$
Phase GF	$\frac{U(z^k)+U(z)-1}{1-G(z^{k+1})}$	$(I-\mathbf{G}_{(k+1)s})^{-1} \circ [\mathbf{G}_{ks}+\mathbf{G}_s]$
Singularities	z s.t. $G(z^{k+1})=1$	s s.t. $\lambda((k+1)s)=1$
Extraction Contours	Cauchy Formula Disks	Perron Formula Vertical lines

$\lambda(s)$ is the dominant eigenvalue of $\mathbf{G}_s$

$\lambda(2)=1$; $\lambda'(2)$ closely related to the entropy

Generating function relative to the d-Euclid Algorithm

We have shown that the Euclid algorithm ($d = 2$) translates as a product

$$\zeta(s_1)\,\zeta(s_2) = T(s_1, s_2)\;\zeta(s_1 + s_2),$$

with $\quad 2T(s,t) = (1 - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)\,[1](0)$

Generating function relative to the d-Euclid Algorithm

We have shown that the Euclid algorithm ($d = 2$) translates as a product

$$\zeta(s_1)\,\zeta(s_2) = T(s_1, s_2)\ \zeta(s_1 + s_2),$$

$$\text{with} \quad 2T(s,t) = (1 - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)\,[1](0)$$

Then, for any $d \geq 2$, the d–Euclid algorithm translates as the product

$$\zeta(s_1) \cdot \ldots \cdot \zeta(s_d) = \zeta(t_d) \prod_{k=1}^{d-1} T(s_{k+1}, t_k) \qquad [t_k := s_1 + s_2 + \ldots + s_k,]$$

Generating function relative to the d-Euclid Algorithm

We have shown that the Euclid algorithm ($d = 2$) translates as a product

$$\zeta(s_1)\,\zeta(s_2) = T(s_1, s_2)\ \zeta(s_1 + s_2),$$

with $\quad 2T(s,t) = (1 - \mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)\,[1](0)$

Then, for any $d \geq 2$, the d–Euclid algorithm translates as the product

$$\zeta(s_1) \cdot \ldots \cdot \zeta(s_d) = \zeta(t_d) \prod_{k=1}^{d-1} T(s_{k+1}, t_k) \qquad [t_k := s_1 + s_2 + \ldots + s_k,]$$

Now, with $s = s_1 = \ldots = s_d$,
the (plain) generating function $S(s)$ of $\mathbb{N}^d$ has an alternative expression

$$S(s) = \zeta(s)^d = \zeta(ds) \prod_{k=1}^{d-1} T(s, ks)$$

which is an exact translation of the d-Euclid algorithm.

T is the "phase generating function".

Generating functions relative to the d-Euclid Algorithm

We start with: $$S(s) = \zeta(s)^d = \zeta(ds) \prod_{k=1}^{d-1} T(s, ks)$$

Generating functions relative to the d-Euclid Algorithm

We start with: $$S(s) = \zeta(s)^d = \zeta(ds) \prod_{k=1}^{d-1} T(s, ks)$$

For studying the distribution of the two parameters :

- L_k (number of steps in the k-th phase)
- D_k (size of the gcd at the beginning of the k-th phase)

we use bivariate generating functions, with an extra variable u

Generating functions relative to the d-Euclid Algorithm

We start with: $$S(s) = \zeta(s)^d = \zeta(ds) \prod_{k=1}^{d-1} T(s, ks)$$

For studying the distribution of the two parameters :

- L_k (number of steps in the k-th phase)
- D_k (size of the gcd at the beginning of the k-th phase)

we use bivariate generating functions, with an extra variable u

$$L_k(s,u) = \zeta(s)^d \cdot \frac{T(s, ks, u)}{T(s, ks)}, \qquad D_k(s,u) = \zeta(s)^d \cdot \frac{Z(ks, u)}{\zeta(ks)},$$

$$2T(s,t,u) = u(1-u\mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)[1](0), \qquad Z(s,u) = \sum_{n \geq 1} \frac{u^{\mathrm{d}(n)}}{n^s}$$

Generating functions relative to the d-Euclid Algorithm

We start with: $$S(s) = \zeta(s)^d = \zeta(ds) \prod_{k=1}^{d-1} T(s, ks)$$

For studying the distribution of the two parameters :

- L_k (number of steps in the k-th phase)
- D_k (size of the gcd at the beginning of the k-th phase)

we use bivariate generating functions, with an extra variable u

$$L_k(s,u) = \zeta(s)^d \cdot \frac{T(s, ks, u)}{T(s, ks)}, \qquad D_k(s,u) = \zeta(s)^d \cdot \frac{Z(ks, u)}{\zeta(ks)},$$

$$2T(s,t,u) = u(1-u\mathbf{G}_{s+t})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_t)[1](0), \qquad Z(s,u) = \sum_{n \geq 1} \frac{u^{\mathrm{d}(n)}}{n^s}$$

For the expectations, the cumulative generating functions are useful:

$$\widehat{L}_k(s) := \frac{\partial}{\partial u} L_k(s,u)|_{u=1} = \frac{\zeta(s)^d}{T(s,ks)} \frac{\partial}{\partial u} T(s, ks, u)|_{u=1}$$

$$\widehat{D}_k(s) := \frac{\partial}{\partial u} D_k(s,u)|_{u=1} = \frac{\zeta(s)^d}{\zeta(ks)} \frac{\partial}{\partial u} Z(s,u)|_{u=1}$$

Towards the distributional analysis of L_k and D_k.

The generating functions of the events $[L_k > m]$ and $[D_k > m]$ are

$$\widehat{L}_k^{[m]}(s) := \sum_{j>m} [u^j] L_k(s,u), \qquad \widehat{D}_k^{[m]}(s) = \sum_{j>m} [u^j] D_k(s,u)$$

Towards the distributional analysis of L_k and D_k.

The generating functions of the events $[L_k > m]$ and $[D_k > m]$ are

$$\widehat{L}_k^{[m]}(s) := \sum_{j>m} [u^j] L_k(s,u), \qquad \widehat{D}_k^{[m]}(s) = \sum_{j>m} [u^j] D_k(s,u)$$

They admit the alternative expressions

$$\widehat{L}_k^{[m]}(s) = \frac{\zeta(s)^d}{T(ks,s)} \cdot \mathbf{G}^m_{(k+1)s} \circ (1 - \mathbf{G}_{(k+1)s})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_{ks})[1](0)$$

$$D_k^{[m]}(s) = \zeta(s)^d \cdot \frac{\zeta_{e^m}(ks)}{\zeta(ks)}, \qquad \zeta_M(s) := \sum_{n \geq M} \frac{1}{n^s}$$

Towards the distributional analysis of L_k and D_k.

The generating functions of the events $[L_k > m]$ and $[D_k > m]$ are

$$\widehat{L}_k^{[m]}(s) := \sum_{j>m} [u^j] L_k(s,u), \qquad \widehat{D}_k^{[m]}(s) = \sum_{j>m} [u^j] D_k(s,u)$$

They admit the alternative expressions

$$\widehat{L}_k^{[m]}(s) = \frac{\zeta(s)^d}{T(ks,s)} \cdot \mathbf{G}^m_{(k+1)s} \circ (1 - \mathbf{G}_{(k+1)s})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_{ks})[1](0)$$

$$D_k^{[m]}(s) = \zeta(s)^d \cdot \frac{\zeta_{e^m}(ks)}{\zeta(ks)}, \qquad \zeta_M(s) := \sum_{n \geq M} \frac{1}{n^s}$$

both of type $B(s) \cdot A_{k,m}(s)$, with $A_{k,m}(s) \approx A_k^m(s)$

$A_k(s) = \lambda((k+1)s)$ [L –case] $\qquad A_k(s) = \exp[1 - ks]$ [D –case]

Towards the distributional analysis of L_k and D_k.

The generating functions of the events $[L_k > m]$ and $[D_k > m]$ are

$$\widehat{L}_k^{[m]}(s) := \sum_{j>m} [u^j] L_k(s,u), \qquad \widehat{D}_k^{[m]}(s) = \sum_{j>m} [u^j] D_k(s,u)$$

They admit the alternative expressions

$$\widehat{L}_k^{[m]}(s) = \frac{\zeta(s)^d}{T(ks,s)} \cdot \mathbf{G}^m_{(k+1)s} \circ (1 - \mathbf{G}_{(k+1)s})^{-1} \circ (\mathbf{G}_s + \mathbf{G}_{ks})[1](0)$$

$$D_k^{[m]}(s) = \zeta(s)^d \cdot \frac{\zeta_{e^m}(ks)}{\zeta(ks)}, \qquad \zeta_M(s) := \sum_{n \geq M} \frac{1}{n^s}$$

both of type $B(s) \cdot A_{k,m}(s)$, with $A_{k,m}(s) \approx A_k^m(s)$

$A_k(s) = \lambda((k+1)s)$ [L –case] $A_k(s) = \exp[1 - ks]$ [D –case]

The asymptotics depends on the value $a := A_k(1)$ at the pole $s = 1$

For $k = 1$, one has $a = 1$ – For $k \geq 2$, one has $a < 1$.

Towards the distributional analysis of L_k and D_k.

We need extractors. For a Dirichlet generating function,

$$S(s) = \sum_{p \geq 1} \frac{a_p}{p^s}$$

we need to isolate the terms with indices p with $\mathrm{d}(p) = n$, and we define

$$\Psi_n[S] := \sum_{p;\mathrm{d}(p)=n} a_p = \sum_{p=e^{n-1}+1}^{e^n} a_p \qquad \mathrm{d}(p) := \lceil \log p \rceil$$

Towards the distributional analysis of L_k and D_k.

We need extractors. For a Dirichlet generating function,

$$S(s) = \sum_{p \geq 1} \frac{a_p}{p^s}$$

we need to isolate the terms with indices p with $\mathrm{d}(p) = n$, and we define

$$\Psi_n[S] := \sum_{p;\mathrm{d}(p)=n} a_p = \sum_{p=e^{n-1}+1}^{e^n} a_p \qquad \mathrm{d}(p) := \lceil \log p \rceil$$

Then : $$\mathbb{P}_n[L_k > m] = \frac{\Psi_n\left[\widehat{L}_k^{[m]}(s)\right]}{\Psi_n\left[S(z)\right]}, \qquad \mathbb{P}_n[D_k > m] = \frac{\Psi_n\left[\widehat{D}_k^{[m]}(s)\right]}{\Psi_n\left[S(z)\right]},$$

Towards the distributional analysis of L_k and D_k.

We need extractors. For a Dirichlet generating function,

$$S(s) = \sum_{p \geq 1} \frac{a_p}{p^s}$$

we need to isolate the terms with indices p with $\mathrm{d}(p) = n$, and we define

$$\Psi_n[S] := \sum_{p;\mathrm{d}(p)=n} a_p = \sum_{p=e^{n-1}+1}^{e^n} a_p \qquad \mathrm{d}(p) := \lceil \log p \rceil$$

Then : $$\mathbb{P}_n[L_k > m] = \frac{\Psi_n\left[\widehat{L}_k^{[m]}(s)\right]}{\Psi_n\left[S(z)\right]}, \qquad \mathbb{P}_n[D_k > m] = \frac{\Psi_n\left[\widehat{D}_k^{[m]}(s)\right]}{\Psi_n\left[S(z)\right]},$$

We need theorems which relate

– the asymptotic properties of $\Psi_n[S]$ (for $n \to \infty$)

– and analytic properties of $S(s)$ near its dominant singularity,

Dominant singularity = singularity with the largest real part = here $s = 1$.

We are interested in the case when S contains a large m-th power.

A general result.

Consider the sequence of Dirichlet series

$$F^{[m]}(s) = B(s)\ A(s)^m = \sum_{p \geq 1} \frac{a_p^{[m]}}{p^s}, \qquad a_p^{[m]} \geq 0$$

A general result.

Consider the sequence of Dirichlet series

$$F^{[m]}(s) = B(s)\ A(s)^m = \sum_{p \geq 1} \frac{a_p^{[m]}}{p^s}, \qquad a_p^{[m]} \geq 0$$

If there exists a vertical strip $\mathcal{V} = \{s; |\Re s - 1| < \delta$ for which

(i) $B(s)$ has an only pole at $s = 1$ of order $d \geq 2$;

assume $\lim_{s \to 1} B(s)/(s-1)^d = 1$

(ii) $A(s)$ is analytic on $\mathcal{V}$

A general result.

Consider the sequence of Dirichlet series

$$F^{[m]}(s) = B(s)\ A(s)^m = \sum_{p\geq 1} \frac{a_p^{[m]}}{p^s}, \qquad a_p^{[m]} \geq 0$$

If there exists a vertical strip $\mathcal{V} = \{s; |\Re s - 1| < \delta$ for which

(i) $B(s)$ has an only pole at $s = 1$ of order $d \geq 2$;

assume $\lim_{s\to 1} B(s)/(s-1)^d = 1$

(ii) $A(s)$ is analytic on $\mathcal{V}$

(iii) [on vertical lines], $|A(s)| \leq |A(\Re s)|$.

(iv) [on the real axis] $s \mapsto A(s)$ strictly decreasing and log convex

$$a := A(1) \neq 0, \quad b := A'(1) < 0,$$

A general result.

Consider the sequence of Dirichlet series

$$F^{[m]}(s) = B(s)\ A(s)^m = \sum_{p\geq 1} \frac{a_p^{[m]}}{p^s}, \qquad a_p^{[m]} \geq 0$$

If there exists a vertical strip $\mathcal{V} = \{s; |\Re s - 1| < \delta$ for which

(i) $B(s)$ has an only pole at $s = 1$ of order $d \geq 2$;

assume $\lim_{s\to 1} B(s)/(s-1)^d = 1$

(ii) $A(s)$ is analytic on $\mathcal{V}$

(iii) [on vertical lines], $|A(s)| \leq |A(\Re s)|$.

(iv) [on the real axis] $s \mapsto A(s)$ strictly decreasing and log convex

$$a := A(1) \neq 0, \quad b := A'(1) < 0,$$

(v) $s \mapsto B(s)$ of polynomial growth in $\mathcal{V}$ for $|\Im s| \to \infty$

A general result.

Consider the sequence of Dirichlet series

$$F^{[m]}(s) = B(s)\ A(s)^m = \sum_{p\geq 1} \frac{a_p^{[m]}}{p^s}, \qquad a_p^{[m]} \geq 0$$

If there exists a vertical strip $\mathcal{V} = \{s; |\Re s - 1| < \delta$ for which

(i) $B(s)$ has an only pole at $s = 1$ of order $d \geq 2$;

assume $\lim_{s\to 1} B(s)/(s-1)^d = 1$

(ii) $A(s)$ is analytic on $\mathcal{V}$

(iii) [on vertical lines], $|A(s)| \leq |A(\Re s)|$.

(iv) [on the real axis] $s \mapsto A(s)$ strictly decreasing and log convex

$$a := A(1) \neq 0, \quad b := A'(1) < 0,$$

(v) $s \mapsto B(s)$ of polynomial growth in $\mathcal{V}$ for $|\Im s| \to \infty$

Then, when $m/n \in [0, c_0]$ with $c_0 < a/|b|$, one has:

$$\Psi_n\left(F^{[m]}\right) := \sum_{p;\mathrm{d}(p)=n} a_p^{[m]} = e^n(e-1)\frac{n^{d-1}}{(d-1)!}\ a^m \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right].$$

Result based on the Landau Theorem: it uses the Perron formula

which relates $\Psi_n[S]$ to $\displaystyle\int_{\mathcal{L}} e^{ns}\frac{S(s)}{s}ds$

Result based on the Landau Theorem: it uses the Perron formula

$$\text{which relates } \Psi_n[S] \text{ to } \int_{\mathcal{L}} e^{ns} \frac{S(s)}{s} ds$$

– Main asymptotic term related to the residue at $s = 1$ of $e^{ns} B(s) A(s)^m$ itself related to the $(d-1)$-th derivative at $s = 1$ of $s \mapsto e^{ns} A^m(s)$

$$\left[e^{ns} e^{m \log A(s)} \right]_{s=1}^{(d-1)} \approx \left[n + m \frac{A'(1)}{A(1)} \right]^{d-1} [e^n A(1)^m]$$

Result based on the Landau Theorem: it uses the Perron formula

$$\text{which relates } \Psi_n[S] \text{ to } \int_{\mathcal{L}} e^{ns} \frac{S(s)}{s} ds$$

– Main asymptotic term related to the residue at $s = 1$ of $e^{ns} B(s) A(s)^m$ itself related to the $(d-1)$-th derivative at $s = 1$ of $s \mapsto e^{ns} A^m(s)$

$$\left[e^{ns} e^{m \log A(s)} \right]_{s=1}^{(d-1)} \approx \left[n + m \frac{A'(1)}{A(1)} \right]^{d-1} [e^n A(1)^m]$$

– Remainder term: related to the asymptotic growth for $|\tau| := |\Im s| \to \infty$.

Here, the dependence with respect to m be made more precise,

Result based on the Landau Theorem: it uses the Perron formula

$$\text{which relates } \Psi_n[S] \text{ to } \int_{\mathcal{L}} e^{ns}\frac{S(s)}{s}ds$$

– Main asymptotic term related to the residue at $s=1$ of $e^{ns}B(s)A(s)^m$ itself related to the $(d-1)$-th derivative at $s=1$ of $s\mapsto e^{ns}A^m(s)$

$$\left[e^{ns}e^{m\log A(s)}\right]_{s=1}^{(d-1)} \approx \left[n+m\frac{A'(1)}{A(1)}\right]^{d-1}[e^nA(1)^m]$$

– Remainder term: related to the asymptotic growth for $|\tau|:=|\Im s|\to\infty$.

Here, the dependence with respect to m be made more precise,

We have to adapt the previous result, as $A_m(s)$ is not a true m–th power.

When $|\tau|$ is small: $A_m(s)=a^m(s)f(s)[1+\theta_m(s)]$, $|\theta_m(s)|\le\theta^m$, $(\theta<1)$

$$\left[e^{ns}e^{\log A_m(s)}\right]_{s=1}^{(d-1)} \approx \left[n+m\frac{a'(1)}{a(1)}\right]^{d-1}[e^nA_m(1)]$$

When $|\tau|\to\infty$: $|A_m(s)|\le|\tau|^\xi|A_m(\sigma)|\ll|\tau|^\xi|a^m(\sigma)|$

Application to our context.

Application to our context.

For the L case, with the constants $a = \lambda(k+1), b = (k+1)\lambda'(k+1)$.

$$\mathbb{P}_n[L_k > m] = \mathbf{G}_{k+1}^m[1](0) \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

$$= a^m[1 + O(\theta^m)] \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

Application to our context.

For the L case, with the constants $a = \lambda(k+1), b = (k+1)\lambda'(k+1)$.

$$\mathbb{P}_n[L_k > m] = \mathbf{G}_{k+1}^m[1](0) \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

$$= a^m[1 + O(\theta^m)] \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

For the D-case, with the constants $a = e^{1-k}, b = -ke^{1-k}$.

$$\mathbb{P}_n[D_k > m] = \frac{\zeta_{e^m}(k)}{\zeta(k)} \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

$$= a^m[1 + O(\theta^m)] \left(1 - \frac{|b|}{a}\frac{m}{n}\right)^{d-1} \left[1 + O\left(\frac{1}{n}\right)\right]$$

For the first phase: $a = 1 \implies$ A quasi –beta behavior $(1, d-1)$

For the subsequent phases: $a < 1 \implies$ A quasi–geometric behavior

Main result for the number of divisions L_k – First phase ($k = 1$)

Main result for the number of divisions L_k – First phase ($k = 1$)

The number of divisions L_1 performed by the d-Euclid algorithm during the first phase has a mean value of linear order

$$\mathbb{E}_n[L_1] = \frac{6\log 2}{\pi^2}\,\frac{n}{d}\left[1 + O\left(\frac{1}{n}\right)\right] \qquad \frac{\pi^2}{6\log 2} = \text{entropy}$$

Main result for the number of divisions L_k – First phase ($k = 1$)

The number of divisions L_1 performed by the d-Euclid algorithm during the first phase has a mean value of linear order

$$\mathbb{E}_n[L_1] = \frac{6\log 2}{\pi^2}\,\frac{n}{d}\left[1 + O\left(\frac{1}{n}\right)\right] \qquad \frac{\pi^2}{6\log 2} = \text{entropy}$$

It follows an asymptotic quasi-beta law of parameter $(1, d-1)$ and its distribution satisfies when $n \to \infty$, and $m/n \in [0, c_0]$ with $c_0 \in [0, (6\log 2)/\pi^2[$

$$\mathbb{P}[L_1 > m] = \left(1 - \frac{m}{n}\frac{\pi^2}{6\log 2}\right)^{d-1}\left[1 + O\left(\frac{1}{n}\right) + O(\theta^m)\right],$$

θ related to the subdominant spectral radius .

Main result for the number of divisions L_k – First phase $(k = 1)$

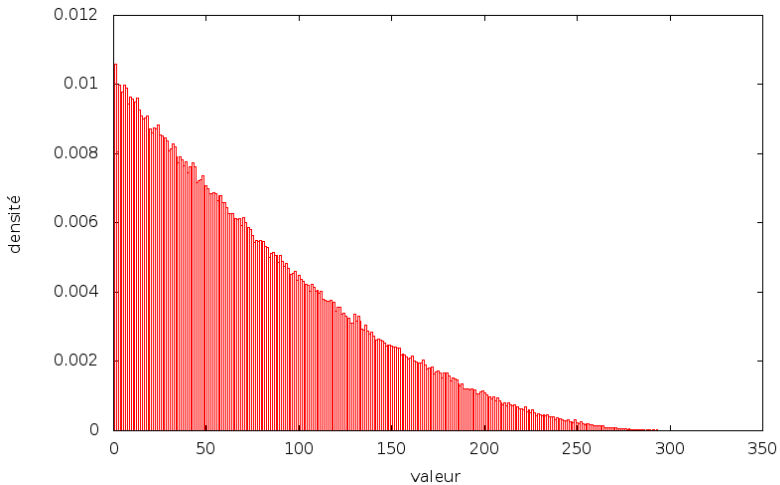
The number of divisions L_1 performed by the d-Euclid algorithm during the first phase has a mean value of linear order

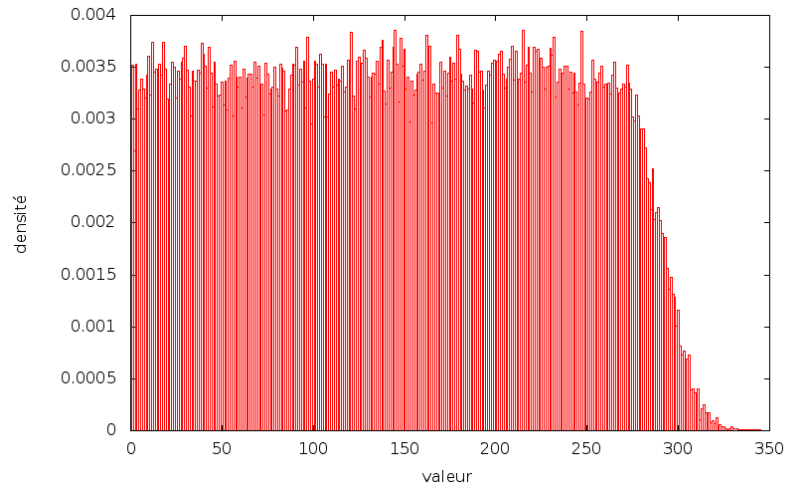
$$\mathbb{E}_n[L_1] = \frac{6\log 2}{\pi^2}\,\frac{n}{d}\left[1 + O\left(\frac{1}{n}\right)\right] \qquad \frac{\pi^2}{6\log 2} = \text{entropy}$$

It follows an asymptotic quasi-beta law of parameter $(1, d-1)$ and its distribution satisfies when $n \to \infty$, and $m/n \in [0, c_0]$ with $c_0 \in [0, (6\log 2)/\pi^2[$

$$\mathbb{P}[L_1 > m] = \left(1 - \frac{m}{n}\frac{\pi^2}{6\log 2}\right)^{d-1}\left[1 + O\left(\frac{1}{n}\right) + O(\theta^m)\right],$$

θ related to the subdominant spectral radius .





Main result for the number of divisions L_k – Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the number of divisions performed by the d-Euclid algorithm during the k-th phase

Main result for the number of divisions L_k – Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the number of divisions performed by the d-Euclid algorithm during the k-th phase

– has a mean value of constant order

$$\mathbb{E}_n[L_k] = (I - \mathbf{G}_{k+1})^{-1}[1](0) \left[1 + O\left(\frac{1}{n}\right)\right] = \left(2\frac{\zeta(k)}{\zeta(k+1)} - 1\right) \left[1 + O\left(\frac{1}{n}\right)\right]$$

Main result for the number of divisions L_k – Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the number of divisions performed by the d-Euclid algorithm during the k-th phase

– has a mean value of constant order

$$\mathbb{E}_n[L_k] = (I - \mathbf{G}_{k+1})^{-1}[1](0) \left[1 + O\left(\frac{1}{n}\right)\right] = \left(2\frac{\zeta(k)}{\zeta(k+1)} - 1\right) \left[1 + O\left(\frac{1}{n}\right)\right]$$

– follows an asymptotic quasi-geometric law, with quasi–ratio $\lambda(k+1)$

$$\mathbb{P}_n[L_k > m] = \mathbf{G}_{k+1}^m[1](0) \left[1 + O\left(\frac{m}{n}\right)\right] \quad \text{for } m = o(n),$$

Main result for the number of divisions L_k – Subsequent phases (case $k \geq 2$)

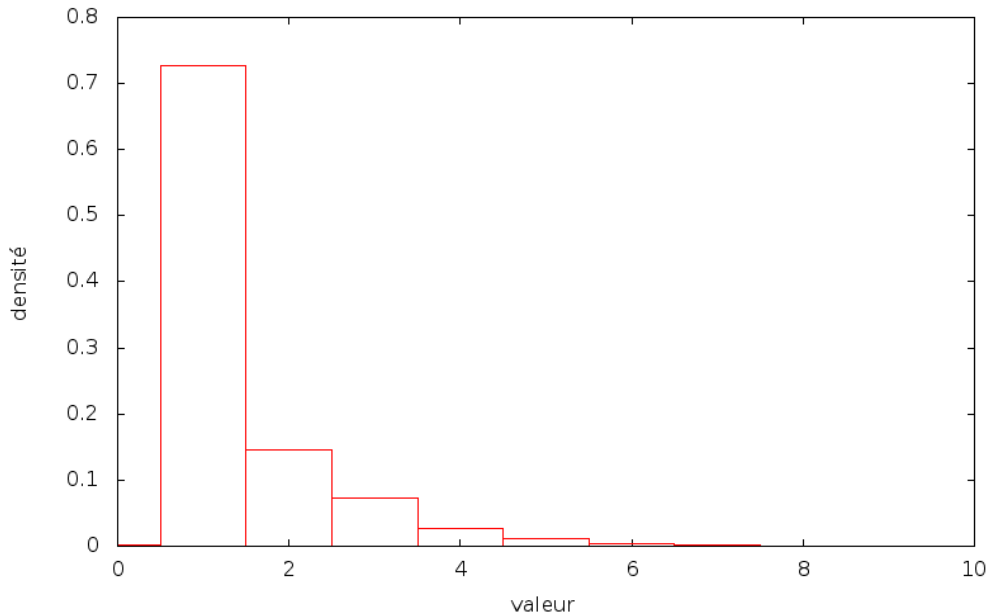
For $k \geq 2$, the number of divisions performed by the d-Euclid algorithm during the k-th phase

– has a mean value of constant order

$$\mathbb{E}_n[L_k] = (I - \mathbf{G}_{k+1})^{-1}[1](0) \left[1 + O\left(\frac{1}{n}\right)\right] = \left(2\frac{\zeta(k)}{\zeta(k+1)} - 1\right) \left[1 + O\left(\frac{1}{n}\right)\right]$$

– follows an asymptotic quasi-geometric law, with quasi–ratio $\lambda(k+1)$

$$\mathbb{P}_n[L_k > m] = \mathbf{G}_{k+1}^m[1](0) \left[1 + O\left(\frac{m}{n}\right)\right] \quad \text{for } m = o(n),$$



Main result for the size D_k of the $\gcd$ – First phase ($k = 1$)

Main result for the size D_k of the $\gcd$ – First phase ($k = 1$)

The size D_1 of the $\gcd$ at the beginning of the first phase has a mean value of linear order

$$\mathbb{E}_n[D_1] = \frac{n}{d}\left[1 + O\left(\frac{1}{n}\right)\right]$$

Main result for the size D_k of the $\gcd$ – First phase ($k = 1$)

The size D_1 of the $\gcd$ at the beginning of the first phase has a mean value of linear order

$$\mathbb{E}_n[D_1] = \frac{n}{d}\left[1 + O\left(\frac{1}{n}\right)\right]$$

It follows an asymptotic quasi-beta law of parameter $(1, d-1)$ and its distribution satisfies when $n \to \infty$, and $m/n \in [0, c_0]$ with $c_0 \in [0, 1[$

$$\mathbb{P}[D_1 > m] = \left(1 - \frac{m}{n}\right)^{d-1}\left[1 + O\left(\frac{1}{n}\right) + O(e^{-m})\right],$$

Main result for the size D_k of the $\gcd$– Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the size D_k at the beginning of the k–th phase

Main result for the size D_k of the gcd– Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the size D_k at the beginning of the k–th phase

– has a mean value of constant order

$$\mathbb{E}_n[L_k] = \frac{\widehat{\zeta}'(k)}{\zeta(k)} \left[1 + O\left(\frac{1}{n}\right)\right] \qquad \widehat{\zeta}'(s) = \sum_{p \geq 1} \frac{\lceil \log p \rceil}{p^s}$$

Main result for the size D_k of the gcd– Subsequent phases (case $k \geq 2$)

For $k \geq 2$, the size D_k at the beginning of the k–th phase

– has a mean value of constant order

$$\mathbb{E}_n[L_k] = \frac{\widehat{\zeta}'(k)}{\zeta(k)} \left[1 + O\left(\frac{1}{n}\right)\right] \qquad \widehat{\zeta}'(s) = \sum_{p \geq 1} \frac{\lceil \log p \rceil}{p^s}$$

– follows an asymptotic quasi-geometric law, with quasi–ratio e^{1-k}

$$\mathbb{P}_n[L_k > m] = \frac{\zeta_{e^m}(k)}{\zeta(k)} \left[1 + O\left(\frac{m}{n}\right)\right] \quad \text{for } m = o(n),$$