Continued Logarithm Algorithm: A probabilistic study

Pablo Rotondo
LIGM, Paris-Est Marne-la-Vallée

Work with
Brigitte Vallée and Alfredo Viola

The origins

Introduced by Gosper as a mutation of continued fractions:

- gives rise to a $\gcd$ algorithm akin to Euclid’s.
- quotients are powers of two:
  - small information parcel.
  - employs only shifts and substractions.
- appears to be simple and efficient.
The origins

Introduced by Gosper as a mutation of continued fractions:
- gives rise to a \texttt{gcd} algorithm akin to Euclid’s.
- \textbf{quotients are powers of two}:
  - small information parcel.
  - employs only shifts and substractions.
- appears to be \textit{simple and efficient}.

More recently:
- Shallit studied its \texttt{worst-case} performance in 2016.
- We consider its \texttt{average} performance!
Continued Logarithm Algorithm

A sequence of binary “divisions” beginning from \((p, q)\):

\[q = 2^a p + r, \quad 0 \leq r < 2^a p.\]
Continued Logarithm Algorithm

A sequence of binary “divisions” beginning from \((p, q)\):

\[ q = 2^a p + r, \quad 0 \leq r < 2^a p. \]

**Note.** \( a = \max\{k \geq 0 : 2^k p \leq q\} \)
Continued Logarithm Algorithm

A sequence of binary “divisions” beginning from \((p, q)\):

\[ q = 2^a p + r, \quad 0 \leq r < 2^a p. \]

**Note.** \( a = \max\{k \geq 0 : 2^k p \leq q\} \)

Continue with the new pair

\[(p, q) \mapsto (p', q') = (r, 2^a p), \]

until the remainder \( r \) equals 0.

Example. Let us find \( \gcd(13, 31) \).

\[
\begin{array}{c|c|c}
 p & q & r \\
\hline
 1 & 31 & 0 \\
 2 & 13 & 2 \\
 4 & 6 & 0 \\
 8 & 0 & 8 \\
\end{array}
\]

\(\Rightarrow\) Ended with \((0, 8)\), what is the \(\gcd\)?

Parasitic powers of 2.
Continued Logarithm Algorithm

A sequence of binary “divisions” beginning from \((p, q)\):

\[ q = 2^a p + r, \quad 0 \leq r < 2^a p. \]

**Note.** \(a = \max\{k \geq 0 : 2^k p \leq q\}\)

Continue with the new pair

\[ (p, q) \mapsto (p', q') = (r, 2^a p), \]

until the remainder \(r\) equals 0.

**Example.** Let us find \(\gcd(13, 31)\).

<table>
<thead>
<tr>
<th></th>
<th>(a)</th>
<th>(p)</th>
<th>(q)</th>
<th>(r)</th>
<th>(2^a p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>31</td>
<td>5</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>26</td>
<td>6</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>20</td>
<td>8</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

Ended with \((0, 8)\), what is the \(\gcd\)?

⇒ parasitic powers of 2.
Continued Logarithm Algorithm

A sequence of binary “divisions” beginning from \((p, q)\):

\[ q = 2^a p + r, \quad 0 \leq r < 2^a p. \]

**Note.** \(a = \max\{k \geq 0 : 2^k p \leq q\}\)

Continue with the new pair

\[(p, q) \mapsto (p', q') = (r, 2^a p),\]

until the remainder \(r\) equals 0.

**Example.** Let us find \(\gcd(13, 31)\).

<table>
<thead>
<tr>
<th>(a)</th>
<th>(p)</th>
<th>(q)</th>
<th>(r)</th>
<th>(2^a p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>31</td>
<td>5</td>
<td>26</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>26</td>
<td>6</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>20</td>
<td>8</td>
<td>12</td>
</tr>
<tr>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>8</td>
</tr>
</tbody>
</table>

Ended with \((0, 8)\), what is the \(\gcd\)？ ⇒ parasitic powers of 2.
Consider
\[ \Omega_N = \{(p, q) \in \mathbb{N} \times \mathbb{N} : p \leq q \leq N\} . \]

Worst-case studied by Shallit (2016): \( 2 \log_2 N + O(1) \) steps.
Consider
\[ \Omega_N = \{(p, q) \in \mathbb{N} \times \mathbb{N} : p \leq q \leq N\} . \]

Worst-case studied by Shallit (2016): \(2 \log_2 N + O(1)\) steps.

- Family \((p, q) = (1, 2^n - 1)\) gives the bound asymptotically.
Consider

$$\Omega_N = \{(p, q) \in \mathbb{N} \times \mathbb{N} : p \leq q \leq N\}.$$ 

**Worst-case** studied by Shallit (2016): $2 \log_2 N + O(1)$ steps.

º Family $(p, q) = (1, 2^n - 1)$ gives the bound asymptotically.

We studied the **average** number of steps over $\Omega_N$, posed by Shallit.
Consider
\[ \Omega_N = \{(p, q) \in \mathbb{N} \times \mathbb{N} : p \leq q \leq N\} . \]

**Worst-case** studied by Shallit (2016): \(2 \log_2 N + O(1)\) steps.

- Family \((p, q) = (1, 2^n - 1)\) gives the bound asymptotically.

We studied the average number of steps over \(\Omega_N\), posed by Shallit.

**Main result** [RVV18].

Mean number of steps \(E_N[K]\) and shifts \(E_N[S]\) are \(\Theta(\log N)\).

More precisely

\[
E_N[K] \sim k \log N , \quad E_N[S] \sim \frac{\log 3 - \log 2}{2 \log 2 - \log 3} E_N[K]
\]

for an explicit constant \(k \doteq 1.49283 \ldots\) given by

\[
k = \frac{2}{H} , \quad H = \text{entropy of appropriate DS}
\]
Consider
\[ \Omega_N = \{ (p, q) \in \mathbb{N} \times \mathbb{N} : p \leq q \leq N \} . \]

**Worst-case** studied by Shallit (2016): \(2 \log_2 N + O(1)\) steps.

- Family \((p, q) = (1, 2^n - 1)\) gives the bound asymptotically.

We studied the **average** number of steps over \(\Omega_N\), posed by Shallit.

**Main result [RVV18].**

Mean number of steps \(E_N[K]\) and shifts \(E_N[S]\) are \(\Theta(\log N)\).

More precisely

\[ E_N[K] \sim k \log N , \quad E_N[S] \sim \frac{\log 3 - \log 2}{2 \log 2 - \log 3} E_N[K] \]

for an **explicit constant** \(k \approx 1.49283 \ldots\) given by

\[ k = \frac{2}{H} , \quad H = \frac{1}{\log(4/3)} \left( \frac{\pi^2}{6} + 2 \sum_j \frac{(-1)^j}{2^j j^2} - (\log 2) \frac{\log 27}{\log 16} \right) \]
Process depends **only** on $p/q$ rather than $(p, q)$.

- Map $p/q \mapsto p'/q'$ can be extended to $\mathcal{I} = (0, 1)$

  $$T: \mathcal{I} \to \mathcal{I}, \quad T(x) = \frac{1}{2^a x} - 1,$$

  where $a = \lfloor \log_2(1/x) \rfloor$.

- Iteration gives a special continued fraction

  $$\frac{p}{q} = \frac{1}{2^a \left(1 + \frac{p'}{q'}\right)}.$$
Process depends **only** on $p/q$ rather than $(p, q)$.

- Map $p/q \mapsto p'/q'$ can be extended to $\mathcal{I} = (0, 1)$
  
  $$T: \mathcal{I} \rightarrow \mathcal{I}, \quad T(x) = \frac{1}{2^a x} - 1,$$
  
  where $a = \lfloor \log_2(1/x) \rfloor$.

- Iteration gives a special continued fraction
  
  $$\frac{p}{q} = \frac{1}{2^a \left(1 + \frac{p'}{q'}\right)}.$$

- For Euclid's algorithm, we get the Gauss map
  
  $$S: \mathcal{I} \rightarrow \mathcal{I}, \quad S(x) = \frac{1}{x} - m,$$
  
  where $m = \lfloor 1/x \rfloor$.

- Iteration gives classical continued fractions
  
  $$\frac{p}{q} = \frac{1}{m + \frac{p'}{q'}}.$$
Process depends **only** on $p/q$ rather than $(p, q)$.

- Map $p/q \mapsto p'/q'$ can be extended to $\mathcal{I} = (0, 1)$

  \[ T : \mathcal{I} \rightarrow \mathcal{I}, \quad T(x) = \frac{1}{2^a x} - 1, \]

  where \( a = \lfloor \log_2(1/x) \rfloor \).

- Iteration gives a special continued fraction

  \[ \frac{p}{q} = \frac{1}{2^a \left( 1 + \frac{p'}{q'} \right)}. \]

- For Euclid’s algorithm, we get the Gauss map

  \[ S : \mathcal{I} \rightarrow \mathcal{I}, \quad S(x) = \frac{1}{x} - m, \]

  where \( m = \lfloor 1/x \rfloor \).

- Iteration gives classical continued fractions

  \[ \frac{p}{q} = \frac{1}{m + \frac{p'}{q'}}. \]

The continued fraction expansion ends (is finite) when we get 0.
The CL dynamical system [Chan05]

The map $T : I \to I$

**Branches**

For $x \in \mathcal{I}_a := \left[2^{-a-1}, 2^{-a}\right]$

$$x \mapsto T_a(x) := \frac{2^{-a}}{x} - 1.$$ 

where $a(x) := \left\lfloor \log_2(1/x) \right\rfloor$.

**Inverse branches**

$$h_a(x) := \frac{2^{-a}}{1 + x}, \quad \mathcal{H} := \{h_a : a \in \mathbb{N}\},$$

and at depth $k$

$$\mathcal{H}^k := \{h_{a_1} \circ \cdots \circ h_{a_k} : a_1, \ldots, a_k \in \mathbb{N}\}.$$
Dynamical system \((\mathcal{I}, T)\)

The map for the CL algorithm

The map for Euclid’s algorithm.
Density transformer

Question: If $g \in C^0(I)$ were the density of $x \mapsto$ density of $T(x)$?
Density transformer

**Question:** If $g \in C^0(\mathcal{I})$ were the density of $x \mapsto$ density of $T(x)$?

\[ T(x) \]

\[ d_y \]

\[ |d_h_3(y)| \ |d_h_2(y)| \ |d_h_1(y)| \ |d_h_0(y)| \]

\[ x \]

\[ 1 \]

\[ 1 \]
Density transformer

Question: If \( g \in C^0(\mathcal{I}) \) were the density of \( x \mapsto \) density of \( T(x) \)?

Answer: The density is

\[
H[g](x) = \sum_{h \in \mathcal{H}} |h'(x)| \ g(h(x)) \]

\[
= \frac{1}{(1 + x)^2} \sum_{a \geq 0} 2^{-a} \ g \left( \frac{2^{-a}}{1 + x} \right). 
\]
Density transformer

Question: If $g \in \mathcal{C}^0(I)$ were the density of $x \mapsto$ density of $T(x)$?

Answer: The density is

$$H[g](x) = \sum_{h \in \mathcal{H}} |h'(x)| \cdot g(h(x))$$

$$= \frac{1}{(1 + x)^2} \sum_{a \geq 0} 2^{-a} g \left( \frac{2^{-a}}{1 + x} \right).$$

In general $T^k(x)$ has density

$$H^k[g](x) = \sum_{h \in \mathcal{H}^k} |h'(x)| \cdot g(h(x)).$$
Density transformer

**Question:** If \( g \in C^0(I) \) were the density of \( x \mapsto \text{density of } T(x) \)?

**Answer:** The density is

\[
H[g](x) = \sum_{h \in H} |h'(x)| \ g(h(x))
= \frac{1}{(1 + x)^2} \sum_{a \geq 0} 2^{-a} g \left( \frac{2^{-a}}{1 + x} \right).
\]

In general \( T^k(x) \) has density

\[
H^k[g](x) = \sum_{h \in H^k} |h'(x)| \ g(h(x)).
\]

\[\Rightarrow\] Transfer operator \( H_s \) extends \( H \), introducing a variable \( s \)

\[
H_s[g](x) = \sum_{h \in H} |h'(x)|^s g(h(x)).
\]
Principles of dynamical analysis [Vallée, Flajolet, Baladi, ...]:

Generating functions.

- $H_s$ describes all executions of depth 1.
- $H_s^2 = H_s \circ H_s$ describes all executions of depth 2.
- ...
- and $(I - H_s)^{-1} = I + H_s + H_s^2 + \ldots$ describes all executions.
Principles of dynamical analysis [Vallée, Flajolet, Baladi, . . .]:

Generating functions.

- $H_s$ describes all executions of depth 1.
- $H_s^2 = H_s \circ H_s$ describes all executions of depth 2.
- 
- and $(I - H_s)^{-1} = I + H_s + H_s^2 + \ldots$ describes all executions.
Reduced denominators and inverse branches

Euclidean algorithm:

- **Homographies**
  
  \[ h_m(x) = \frac{1}{m + x}, \]

  with \( \det h_m = -1 \).

- **For** \( h = h_{m_1} \circ \cdots \circ h_{m_k} \)

  \[ h(0) = \frac{p}{q} \Rightarrow |h'(0)| = \frac{1}{q^2}, \]

  \( p/q \) reduced.
## Reduced denominators and inverse branches

<table>
<thead>
<tr>
<th>Euclidean algorithm:</th>
<th>CL algorithm:</th>
</tr>
</thead>
<tbody>
<tr>
<td>▶ Homographies</td>
<td>▶ Homographies</td>
</tr>
<tr>
<td>[ h_m(x) = \frac{1}{m + x} , ]</td>
<td>[ h_a(x) = \frac{1}{2^a(1 + x)} , ]</td>
</tr>
<tr>
<td>with ( \det h_m = -1 ).</td>
<td>with ( \det h_a = -2^a ).</td>
</tr>
<tr>
<td>▶ For ( h = h_{m_1} \circ \cdots \circ h_{m_k} )</td>
<td>▶ For ( h = h_{m_1} \circ \cdots \circ h_{m_k} )</td>
</tr>
<tr>
<td>[ h(0) = \frac{p}{q} \Rightarrow \left</td>
<td>h'(0) \right</td>
</tr>
<tr>
<td>( p/q ) reduced.</td>
<td>( p/q ) reduced.</td>
</tr>
</tbody>
</table>
Reduced denominators and inverse branches

**Euclidean algorithm:**

- **Homographies**
  \[ h_m(x) = \frac{1}{m + x}, \]
  with \( \det h_m = -1. \)
- For \( h = h_{m_1} \circ \cdots \circ h_{m_k} \)
  \[ h(0) = \frac{p}{q} \implies |h'(0)| = \frac{1}{q^2}, \]
  \( p/q \) reduced.

**CL algorithm:**

- **Homographies**
  \[ h_a(x) = \frac{1}{2^a(1 + x)}, \]
  with \( \det h_a = -2^a. \)
- For \( h = h_{m_1} \circ \cdots \circ h_{m_k} \)
  \[ h(0) = \frac{p}{q} \implies |h'(0)| \text{ vs. } \frac{1}{q^2}? \]
  \( p/q \) reduced.

**Problem:** Denominator retrieved is engorged by powers of two.
Recording the dyadic behaviour

**Solution:** Dyadic numbers $\mathbb{Q}_2$!

Dyadic topology = Divisibility by 2 constraints, using the dyadic norm $|\cdot|_2$. 
Recording the dyadic behaviour

**Solution:** Dyadic numbers $\mathbb{Q}_2$!

Dyadic topology = Divisibility by 2 constraints, using the dyadic norm $|\cdot|_2$.

- Introduce dyadic component
  $\Rightarrow$ mixed dynamical system $(x, y) \in \mathcal{I} \times \mathbb{Q}_2$
Recording the dyadic behaviour

**Solution:** Dyadic numbers $\mathbb{Q}_2$!

Dyadic topology $= \text{Divisibility by 2 constraints}$, using the dyadic norm $| \cdot |_2$.

- Introduce dyadic component
  $\Rightarrow$ mixed dynamical system $(x, y) \in I \times \mathbb{Q}_2$

- Incorporate $\mathbb{Q}_2$ into the Transfer Operator?
Recording the dyadic behaviour

**Solution:** Dyadic numbers $\mathbb{Q}_2$!

Dyadic topology $= \text{Divisibility by 2 constraints}$, using the dyadic norm $| \cdot |_2$.

- Introduce **dyadic component**
  $\Rightarrow$ mixed dynamical system $(x, y) \in I \times \mathbb{Q}_2$
- Incorporate $\mathbb{Q}_2$ into the **Transfer Operator**?

**Idea works!**
The extended dynamical system

Introduce $\mathcal{I} := \mathcal{I} \times \mathbb{Q}_2$ and $T : \mathcal{I} \to \mathcal{I}$ as follows

$$T(x, y) = (T_a(x), T_a(y)), \quad \text{for } x \in \mathcal{I}_a = [2^{-a-1}, 2^{-a}].$$

This gives inverse branches

$$h_a(x, y) = (h_a(x), h_a(y)), \quad (x, y) \in \mathcal{I}.$$
The extended dynamical system

- Introduce $\mathcal{I} := \mathcal{I} \times \mathbb{Q}_2$ and $T : \mathcal{I} \to \mathcal{I}$ as follows

$$T(x, y) = (T_a(x), T_a(y)),$$

for $x \in \mathcal{I}_a = [2^{-a-1}, 2^{-a}]$. This gives inverse branches

$$h_a(x, y) = (h_a(x), h_a(y)),$$  \hspace{1cm} (x, y) \in \mathcal{I}.

Evolution is lead by the real component, which determines $a$. 
The extended dynamical system

- Introduce $\mathcal{I} := \mathcal{I} \times \mathbb{Q}_2$ and $\mathcal{T} : \mathcal{I} \rightarrow \mathcal{I}$ as follows

$$\mathcal{T}(x, y) = (T_a(x), T_a(y)),$$

for $x \in I = [2^{-a-1}, 2^{-a}]$. This gives inverse branches

$$h_a(x, y) = (h_a(x), h_a(y)),$$

$(x, y) \in \mathcal{I}$. Evolution is lead by the real component, which determines $a$.

- For Transfer operator ⇒ need change of variables formula!
Introduce \( \mathcal{I} := \mathcal{I} \times \mathbb{Q}_2 \) and \( T: \mathcal{I} \to \mathcal{I} \) as follows

\[
T(x, y) = (T_a(x), T_a(y)) ,
\]

for \( x \in \mathcal{I}_a = [2^{-a-1}, 2^{-a}] \). This gives inverse branches

\[
h_a(x, y) = (h_a(x), h_a(y)) , \quad (x, y) \in \mathcal{I}.
\]

Evolution is lead by the real component, which determines \( a \).

For Transfer operator \( \Rightarrow \) need change of variables formula!

Haar (translation invariant) measure \( \nu \) on \( \mathbb{Q}_2 \) has one!
Functional space $\mathcal{F}$ for the extended operator $H_s$

Real component directs the dynamical system:

- *sections* $F_y$ fixing $y \in \mathbb{Q}_2$ asked to be $C^1(\mathcal{I})$.
- The dyadic component follows, demanding only integrability of

$$y \mapsto \sup_x F_y, \quad \text{and} \quad y \mapsto \sup_x \partial_x F_y.$$
Functional space $\mathcal{F}$ for the extended operator $\underline{H}_s$

Real component directs the dynamical system:

- *sections* $F_y$ fixing $y \in \mathbb{Q}_2$ asked to be $C^1(\mathcal{I})$.
- the dyadic component follows, demanding only integrability of

\[
y \mapsto \sup_x F_y, \quad \text{and} \quad y \mapsto \sup_x \partial_x F_y.
\]

Ensuing space $\mathcal{F}$ makes $\underline{H}_s$

- have a dominant eigenvalue and spectral gap
  relying strongly on the real component.
Functional space $\mathcal{F}$ for the extended operator $H_s$

Real component directs the dynamical system:

- sections $F_y$ fixing $y \in \mathbb{Q}_2$ asked to be $C^1(\mathcal{I})$.
- the dyadic component follows, demanding only integrability of

$$y \mapsto \sup_x F_y, \quad \text{and} \quad y \mapsto \sup_x \partial_x F_y.$$

Ensuing space $\mathcal{F}$ makes $H_s$

- have a dominant eigenvalue and spectral gap relying strongly on the real component.

We can finish the dynamical analysis!
Conclusion and further questions

Conclusions:

- We have studied the average number of shifts and subtractions for the CL algorithm.
- Study makes an interesting use of the dyadics in the framework of dynamical analysis.

Questions:

1. Conjecture: The successive pairs \((p_i, q_i)\) given by the algorithm satisfy
   \[
   \lim_{i \to \infty} \frac{1}{i} \log_2 \gcd(p_i, q_i) = \frac{1}{2}.
   \]

2. Comparison to other binary algorithms: binary GCD, LSB.
Conclusion and further questions

Conclusions:
- We have studied the average number of shifts and subtractions for the CL algorithm.
- Study makes an interesting use of the dyadics in the framework of dynamical analysis.

Questions:
1. **Conjecture:** The successive pairs \((p_i, q_i)\) given by the algorithm satisfy

\[
\lim_{i \to \infty} \frac{1}{i} \log_2 \gcd(p_i, q_i) = 1/2.
\]

Back to \((13, 31)\)

<table>
<thead>
<tr>
<th>(i)</th>
<th>(a_i)</th>
<th>(p_i)</th>
<th>(q_i)</th>
<th>(\gcd(p_i, q_i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>13</td>
<td>31</td>
<td>(2^0)</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>26</td>
<td>(2^0)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>6</td>
<td>20</td>
<td>(2^1)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>(2^2)</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>(2^2)</td>
</tr>
</tbody>
</table>
Conclusion and further questions

Conclusions:

- We have studied the average number of shifts and substractions for the CL algorithm.
- Study makes an interesting use of the dyadics in the framework of dynamical analysis.

Questions:

1. **Conjecture**: The successive pairs \((p_i, q_i)\) given by the algorithm satisfy

\[
\lim_{i \to \infty} \frac{1}{i} \log_2 \gcd(p_i, q_i) = 1/2.
\]

Back to \((13, 31)\)

<table>
<thead>
<tr>
<th>(i)</th>
<th>(a_i)</th>
<th>(p_i)</th>
<th>(q_i)</th>
<th>(\gcd(p_i, q_i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>13</td>
<td>31</td>
<td>2^0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>26</td>
<td>2^0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>6</td>
<td>20</td>
<td>2^1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>2^2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>2^2</td>
</tr>
</tbody>
</table>

2. **Comparison to other binary algorithms**: binary GCD, LSB.
Conclusion and further questions

Conclusions:
- We have studied the average number of shifts and substractions for the CL algorithm.
- Study makes an interesting use of the dyadics in the framework of dynamical analysis.

Questions:
1. **Conjecture:** The successive pairs \((p_i, q_i)\) given by the algorithm satisfy

\[
\lim_{i \to \infty} \frac{1}{i} \log_2 \gcd(p_i, q_i) = \frac{1}{2}.
\]

Back to \((13, 31)\)

<table>
<thead>
<tr>
<th>(i)</th>
<th>(a_i)</th>
<th>(p_i)</th>
<th>(q_i)</th>
<th>(\gcd(p_i, q_i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>13</td>
<td>31</td>
<td>(2^0)</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>26</td>
<td>(2^0)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>6</td>
<td>20</td>
<td>(2^1)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>(2^2)</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>(2^2)</td>
</tr>
</tbody>
</table>

2. **Comparison to other binary algorithms:** binary GCD, LSB.