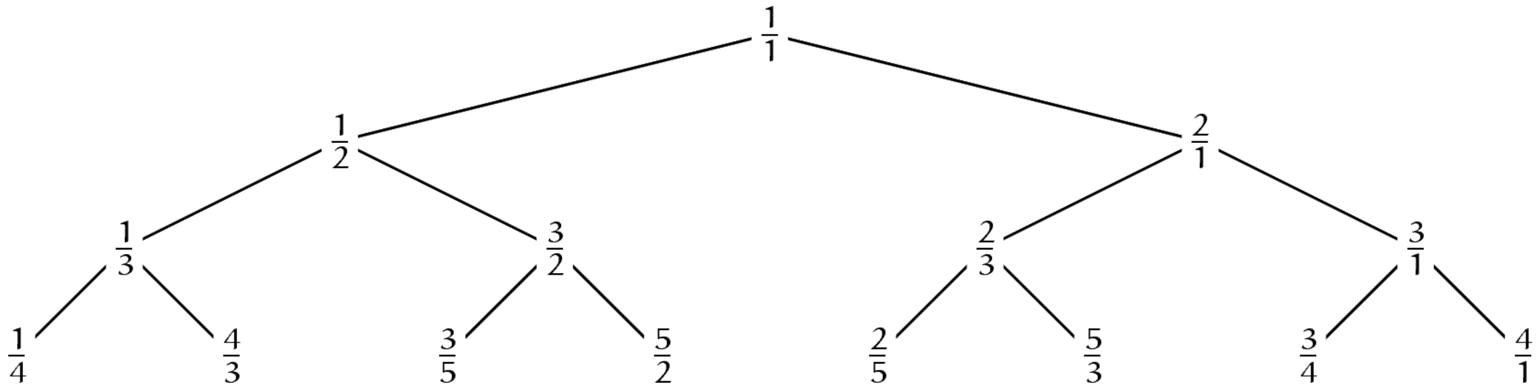
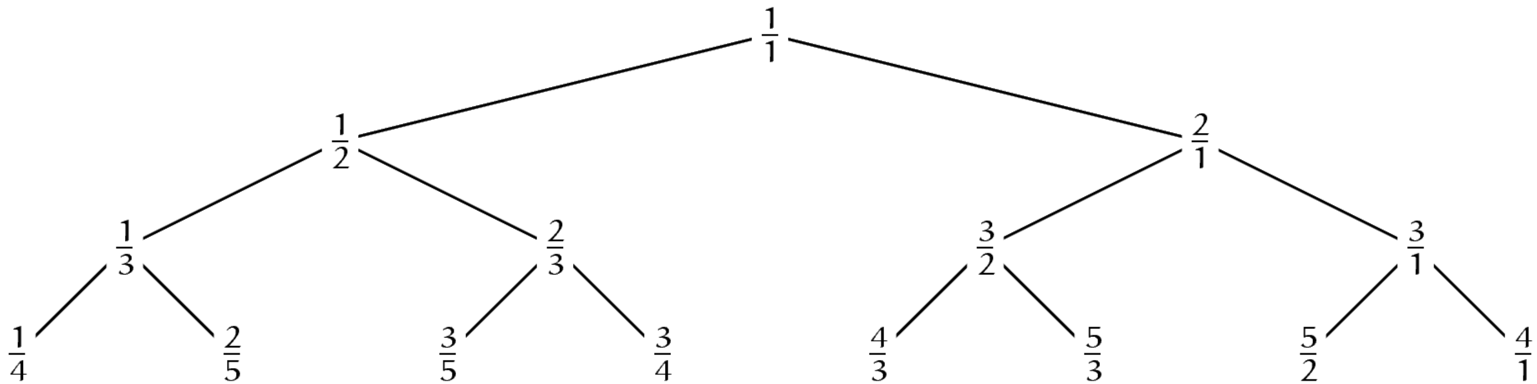
Enumerating the Rationals: Twice!

Roland Backhouse and João Ferreira*





* Funded by Fundação para a Ciência e a Tecnologia, Portugal

Positive Rationals

- A positive rational in "lowest-form" is an ordered pair of positive, coprime integers
- Every rational m/n has unique "lowest-form" representation:

$$\frac{m/(m \triangledown n)}{n/(m \triangledown n)}$$

where $\triangledown$ ("nabla") denotes the GCD.

Euclid's Algorithm

$\{\ 0<m \ \wedge\ 0<n\ \}$

$x, y := m, n\ ;$

$\{$ Invariant: $0<x \ \wedge\ 0<y \ \wedge\ x \triangledown y = m \triangledown n \}$

do

$\quad x<y \longrightarrow y := y-x$

$\square\ y<x \longrightarrow x := x-y$

od

$\{\ x=y=m \triangledown n\ \}$

Termination: $x=y$

So, $x \triangledown y = x \triangledown x = y \triangledown y$

$= x = y$.

Extended Euclid's Algorithm

$\{ \ 0<m \ \wedge \ 0<n \ \}$

$x, y := m, n \ ; \quad C := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$\{$ Invariant: $(x \ \ y) = (m \ \ n) \times C$

where $A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$ and $B = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \ \}$

do

$x<y \longrightarrow (x \ \ y) := (x \ \ y) \times A \ ; \quad C := C \times A$

$\square \ y<x \longrightarrow (x \ \ y) := (x \ \ y) \times B \ ; \quad C := C \times B$

od

$\{ \ x = y = m \nabla n \ \wedge \ (x \ \ y) = (m \ \ n) \times C \ \}$

Extended Euclid's Algorithm

On termination, for arbitrary m and n :

$$(m \triangledown n \quad m \triangledown n) = (m \quad n) \times C$$

It follows that

$$(1 \quad 1) \times C^{-1} = \left({}^{m}/_{(m \triangledown n)} \quad {}^{n}/_{(m \triangledown n)} \right)$$

Note

- m and n uniquely define C
- C^{-1} uniquely defines a rational m/n
  (all finite products of A^{-1} and B^{-1} are different)

$$A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = A^{-1} \qquad L = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = B^{-1}$$

$$R^T = L$$

$\{\ 0 < m \ \wedge\ 0 < n\ \}$

$x, y := m, n \ ; \quad D := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$\{$ Invariant: $(x \ \ y) \times D = (m \ \ n)\ \}$

do

$\quad x < y \longrightarrow (x \ \ y) := (x, y) \times A \ ; \quad D := R \times D$

$\square \ y < x \longrightarrow (x \ \ y) := (x, y) \times B \ ; \quad D := L \times D$

od

$\{\ (1 \ \ 1) \times D = \left({}^{m}/_{(m \triangledown n)} \quad {}^{n}/_{(m \triangledown n)} \right)\ \}$

$$A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = A^{-1} \qquad L = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = B^{-1}$$

$$R^T = L$$

$$\{\ 0 < m \ \wedge\ 0 < n\ \}$$

$$x, y := m, n \ ; \quad E := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\{\ \text{Invariant:}\ \ E \times \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} m \\ n \end{pmatrix}\ \}$$

$$\begin{array}{l} \mathbf{do} \\ \quad x < y \ \rightarrow\ \begin{pmatrix} x \\ y \end{pmatrix} := B \times \begin{pmatrix} x \\ y \end{pmatrix} \ ; \quad E := E \times L \\ \ \square\ y < x \ \rightarrow\ \begin{pmatrix} x \\ y \end{pmatrix} := A \times \begin{pmatrix} x \\ y \end{pmatrix} \ ; \quad E := E \times R \\ \mathbf{od} \end{array}$$

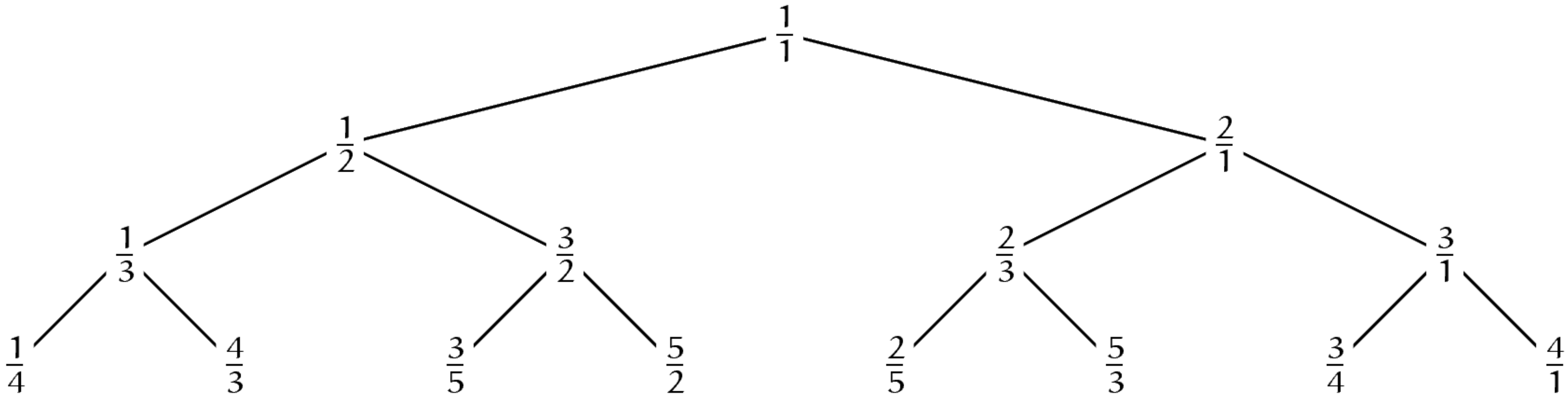
$$\{\ E \times \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} m/(m\nabla n) \\ n/(m\nabla n) \end{pmatrix}\ \}$$

Tree of Matrices

$$
\begin{tikzcd}
 & & & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \arrow[dll, dash, "L"'] \arrow[drr, dash, "R"] & & & \\
 & \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \arrow[dl, dash, "L"'] \arrow[dr, dash, "R"] & & & & \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \arrow[dl, dash, "L"'] \arrow[dr, dash, "R"] & \\
\begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} & & \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} & & \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} & & \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}
\end{tikzcd}
$$

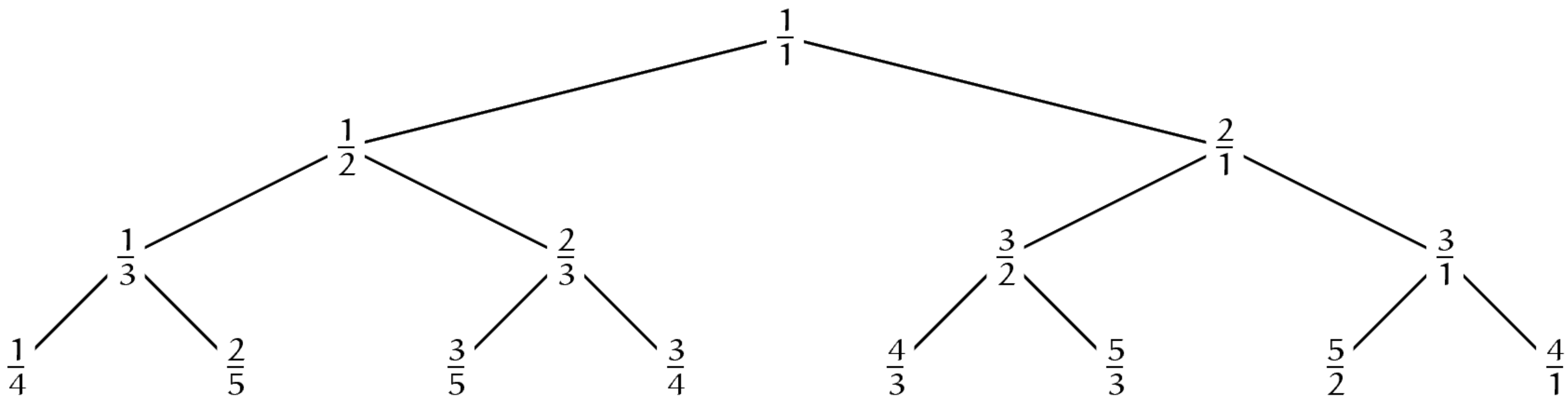
$$L = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

Calkin-Wilf Tree



Premultiply by $\begin{pmatrix} 1 & 1 \end{pmatrix}$

Stern-Brocot Tree



Postmultiply by $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$

Enumerating the Matrices

- There is a bijection between strings of "R"s and "L"s and products of Rs and Ls
- Finding the "next" matrix is equivalent to finding the next string in the lexicographic ordering
- Next string of the same length:

$$tLR^j \longrightarrow tRL^j$$

- Next matrix in the same level:

$$T \times L \times R^j \longrightarrow T \times R \times L^j$$

Computing the "next" matrix

$$(T \times L \times R^{j}) \times (R^{-j} \times L^{-1} \times R \times L^{j}) = T \times R \times L^{j}$$

where

$$R^{-j} \times L^{-1} \times R \times L^{j} = \begin{pmatrix} 2j+1 & 1 \\ -1 & 0 \end{pmatrix}$$

How to determine j ?

do
$x < y \longrightarrow (x \ \ y) := (x, y) \times A$; $D := R \times D$
$\Box$ $y < x \longrightarrow (x \ \ y) := (x, y) \times B$; $D := L \times D$
od

Therefore, $j = \left\lfloor \frac{n-1}{m} \right\rfloor$

Computing the "next" matrix

$$j = \left\lfloor \frac{n-1}{m} \right\rfloor \qquad \begin{pmatrix} 2j+1 & 1 \\ -1 & 0 \end{pmatrix}$$

On termination of the algorithm:

$$(1 \quad 1) \times D = (m \quad n)$$

That is, if

$$D = \begin{pmatrix} D_{00} & D_{01} \\ D_{10} & D_{11} \end{pmatrix}$$

then $m = D_{00} + D_{10}$ and $n = D_{01} + D_{11}$.

Therefore,

$$j = \left\lfloor \frac{n-1}{m} \right\rfloor = \left\lfloor \frac{D_{01} + D_{11} - 1}{D_{00} + D_{10}} \right\rfloor .$$

Enumerating the matrices

- D is a power of R exactly when the rationals are integers — easy to test!

$$D := I$$

do

$$D_{00} + D_{10} = 1 \longrightarrow D := \begin{pmatrix} 1 & 0 \\ D_{01} + D_{11} & 1 \end{pmatrix}$$

$\Box$

$$D_{00} + D_{10} \neq 1 \longrightarrow j := \left\lfloor \frac{D_{01} + D_{11} - 1}{D_{00} + D_{10}} \right\rfloor \; ;$$

$$D := D \times \begin{pmatrix} 2j+1 & 1 \\ -1 & 0 \end{pmatrix}$$

od

Optimising Calkin-Wilf Enumeration

- The matrix $\begin{pmatrix} 2j+1 & 1 \\ -1 & 0 \end{pmatrix}$ is a function of $(1\ 1) \times D$

- Next matrix:

$$D := D \times J.((1\ 1) \times D)$$

- Next Calkin-Wilf rational:

$$(1\ 1) \times (D \times J.((1\ 1) \times D))$$

- Matrix multiplication is associative:

$$((1\ 1) \times D) \times J.((1\ 1) \times D)$$

Optimising Calkin-Wilf Enumeration

We get the following optimised algorithm:

$$
\begin{array}{l}
m, n := 1, 1 \\
\textbf{do} \\
\quad m = 1 \longrightarrow m, n := n+1, m \\
\quad \square \\
\quad m \neq 1 \longrightarrow m, n := \left(2 \times \left\lfloor \frac{n-1}{m} \right\rfloor + 1\right) \times m - n, m \\
\textbf{od}
\end{array}
$$

(Note that the test for change in level is also a function of $(1\ 1) \times D$)

Optimising Calkin-Wilf Enumeration

- $\left\lfloor \frac{n-1}{m} \right\rfloor = \left\lfloor \frac{n}{m} \right\rfloor$ for $m \perp n$ and $m \neq 1$
- $\left(2 \times \left\lfloor \frac{n}{m} \right\rfloor + 1\right) \times m - n = n + 1$ when $m = 1$

Hence, we can eliminate the case analysis:

$$
\begin{array}{l}
m, n := 1, 1 \\
\textbf{do} \\
\quad m, n := \left(2 \times \left\lfloor \frac{n}{m} \right\rfloor\right) \times m - n,\ m \\
\textbf{od}
\end{array}
$$

(This is the algorithm discovered by Newman)

Overview

- Goal: to enumerate the rationals
- Rationals involve GCD: Euclid's Algorithm
- Bijection between rationals and products of matrices
- New problem: enumerating rationals

  $\downarrow$

  enumerating matrices
- Optimising Calkin-Wilf Enumeration

Conclusions

- Our goal is to improve the effectiveness of algorithmic problem solving
- Matrices in Extended Euclid's Algorithm combine concision and precision
- Our construction makes "obvious" how to enumerate the rationals in Stern-Brocot order
- More importantly: we show the duality between both trees

← Made by Brocot!
(17 teeth)

Questions?

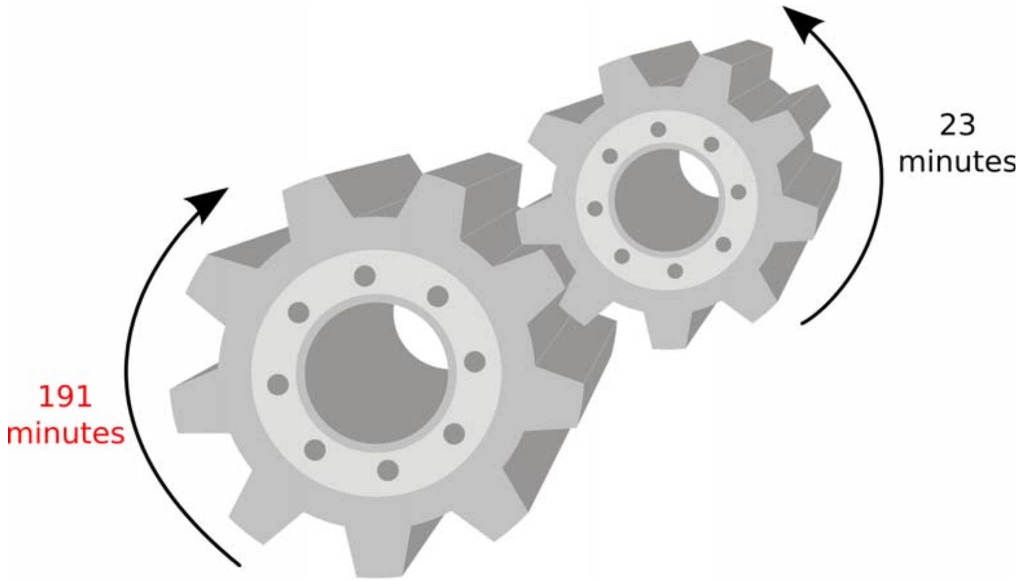
Historical note on Stern-Brocot tree

Brocot's example:

A shaft turns once in 23 minutes. We want to make suitable cogs, so that another shaft completes a revolution in 191 minutes.

- Speed ratio: $191/23$

- There is a technological limitation!



Historical note on Stern-Brocot tree

Brocot's method:

- $\frac{8}{1} < \frac{191}{23} < \frac{9}{1}$

$8 \quad 1 \quad -7$

- $191 = 8 \times 23 + 7$

  (7 min. earlier)

- $191 = 9 \times 23 - 16$

  (16 min. later)

$9 \quad 1 \quad +16$