Demnach ist also U(w,v) meist von der Größenordnung log w.

Wir betrachten die unendliche Folge

$$A = (a_1, e_1, a_2, e_2, a_3, e_3, \ldots)$$

mit $2\nmid a_j$, $e_j \in \{-1,1\}$, $a_j + e_j > 0$ $(j>0)$ und die davon hergeleitete Folge $\{q_0, q_1, q_2, \ldots\}$ mit $q_0 := 1$, $q_1 := a_1$, $q_n := a_n q_{n-1} + e_{n-1}\, q_{n-2}$ $(n>1)$. Der Vergleich mit (2) liefert

$$r_0 = q_n. \tag{3}$$

Nach dem Vorbild von Fibonacci definieren wir $f : \mathbb{Z} \to \mathbb{Z}$ vermöge $f_0 := 1$, $f_1 := 1$, $f_n - f_{n-1} - f_{n-2} := 0$ $(n \in \mathbb{Z})$. Es ist

$$f_n = \frac{G^{n+1} + (-1)^n\, G^{-n-1}}{G + G^{-1}} > \frac{G^n}{\sqrt{5}} \qquad (n \geqq 0). \tag{4}$$

Hilfssatz 1. *Für die Folge* A *gilt* $q_n \geqq f_n$ $(n \geqq 0)$.

Beweis. Für $n=0$ und $n=1$ ist das klar. Es sei $n>1$. Durch Induktion nach k beweisen wir zunächst

$$q_n \geqq f_k q_{n-k} + f_{k-1} e_{n-k} q_{n-k-1} \qquad (0<k<n). \tag{5}$$

Für $k = 1$ ist das richtig wegen $q_n = a_n q_{n-1} + e_{n-1} q_{n-2}$. Es sei also (5) bereits bewiesen für $0<k<n-1$. Dann genügt es zu zeigen

$$f_k q_{n-k} + f_{k-1} e_{n-k} q_{n-k-1} \geqq f_{k+1} q_{n-k-1} + f_k e_{n-k-1} q_{n-k-2}. \tag{6}$$

Durch Einsetzen von $q_{n-k} = a_{n-k} q_{n-k-1} + e_{n-k-1} q_{n-k-2}$ lautet (6) nach Division mit q_{n-k-1} jetzt

$$f_k a_{n-k} + f_{k-1} e_{n-k} \geqq f_{k+1}, \tag{7}$$

was wegen $f_{k+1} := f_k + f_{k-1}$ und $a_{n-k} + e_{n-k} > 1$ richtig ist. Aus (5) für $k = n-1$ folgt wegen $a_1 + e_1 > 1$ die Behauptung.

Ein anderer Beweis von Hilfssatz 1 findet sich in [5].

Aus (3), (1), Hilfssatz 1, (4) folgt

$$U(r_0, r_1) \leqq \frac{\log (r_0 \sqrt{5})}{\log G} \qquad (0 < r_1 \leqq r_0). \tag{8}$$

Wir kehren jetzt den ungeraden Algorithmus um. Nach Vorgabe von $n>0$, $r_{n+1} = 0$, $r_n > 0$ und von a_j, $e_j \in \{-1,1\}$ mit $2\nmid a_j$, $a_j + e_j > 0$ $(n \geqq j > 0)$ sei $r_{j-1} := a_j r_j + e_j r_{j+1}$ $(n \geqq j > 1)$. Dann sind r_1, r_0 eindeutig bestimmt und erfüllen $r_0 \geqq r_1 > 0$. Für $e_n, e_{n-1}, \ldots, e_1$ hat man höchstens 2^n Möglichkeiten. Mit

$$b_j := \begin{cases} a_j & \text{falls } e_j = 1 \\ a_j - 1 & \text{falls } e_j = -1 \end{cases} \qquad (n \geqq j > 0)$$

gilt $b_j > 0$, $r_{j-1} \geqq b_j r_j$ $(n \geqq j > 0)$ und daher

$$r_0 \geqq b_1 b_2 \ldots b_n r_n.$$

Für $n>0$, $x>1$ bezeichne $L_n(x)$ die Anzahl der Paare v,w mit $0<v\leqq w\leqq x$ und $U(w,v)=n$. Mit r_0, r_1 statt w,v folgt sofort

$$(9)\qquad L_n(x) \leqq 2^n \sum_{\substack{b_1>0,\ \dots,\ b_n>0,\ r_n>0\\ b_1 \dots b_n r_n \leqq x}} \cdots \sum \sum 1.$$

Für $1<\sigma \in \mathbb{R}$ sei $\zeta(\sigma) := \sum_{b=1}^{\infty} b^{-\sigma}$. Durch Multiplikation der rechten Seite von (9) mit $x^{\sigma}(b_1 \dots b_n r_n)^{-\sigma}$ folgt

$$L_n(x) < x^{\sigma}(2\zeta(\sigma))^{n+1} \qquad (1<\sigma\in\mathbb{R}).$$

Für $0<\tau \in \mathbb{R}$ bezeichne $S_\tau(x)$ die Anzahl der Paare v,w mit $0<v\leqq w\leqq x$ und $U(w,v) \leqq \tau \log w$. Dann ist

$$S_\tau(x) \leqq \sum_{0<n\leqq\tau\log x} L_n(x) < x^{\sigma}\,\frac{(2\zeta(\sigma))^{2+\tau\log x}}{2\zeta(\sigma)-1}.$$

Wählt man σ hinreichend nahe an 1 und τ hinreichend nahe an 0, erhält man daraus mit gewissen $\alpha>0$, $\beta>0$ sofort

$$(10)\qquad S_\tau(x) < \alpha\, x^{2-\beta}.$$

Der **Beweis von Satz 1,** dessen Methode eine Weiterführung von [1] und von [3] ist, ergibt sich unmittelbar aus (10) und (8).

Wählt man etwa $\sigma=\frac{3}{2}$ und $\tau=\frac{1}{8}$, so kann man wegen $1<\zeta(\frac{3}{2})<3$ in (10) und damit Satz 1 dazu $\alpha=36$ und $\beta=\frac{1}{4}$ wählen.

Literatur

[1] J.D. Dixon, A simple estimate of the number of steps in the euclidean algorithm. Amer. Math. Soc. Monthly **78**, 374–376, 1971.

[2] O. Perron, Die Lehre von den Kettenbrüchen. 3. Aufl., Stuttgart 1954.

[3] G. J. Rieger, Über die Schrittanzahl beim Algorithmus von Harris und dem nach nächsten Ganzen. Arch. d. Math. **34**, 421–427, 1980.

[4] G.J. Rieger, Ein Heilbronn-Satz für Kettenbrüche mit ungeraden Teilnennern. Math. Nachr. (im Druck).

[5] G. J. Rieger, Continued fractions and related algorithms. Proc. Conference Exeter 1980.