

Sei GKZ die Menge der Zahlen eines gewählten GKZ-Formates, die Basis des Formats sei dabei als 2 vorgeschrieben, worauf aber auch verzichtet werden könnte.

Zunächst eine kurze Zusammenfassung zu Basisdarstellungen reeller Zahlen. Sei $b \in \mathbb{N}, b > 2$. Z.B. unter Anwendung des euklidischen Algorithmus zeigt sich, dass sich jede natürliche Zahl n eindeutig als Summe $a_0b^0 + a_1b^1 + a_2b^2 + \dots + a_kb^k$ mit $a_i < b$ und $n < b^{k+1}$ darstellen lässt, diese Eigenschaft überträgt sich analog auf ganze Zahlen, als Quotientenkörper dieser auch auf die rationalen Zahlen und über Vervollständigung auf die reellen Zahlen, kurzum: Jede reelle Zahl ist darstellbar als

$$\sum_{i=-\infty}^{\infty} a_i b^i, a_i < b,$$

auch bekannt als Dezimalzahldarstellung und geschrieben als $\dots a_3 a_2 a_1 a_0 . a_{-1} a_{-2} \dots$. Zu erwähnen ist dabei noch, dass $0.(b-1)(b-1)\dots = 1$ und die Darstellung bis auf diesen Fall eindeutig ist. Durch die bedingungslose Basiswahl heißt dies insbesondere, dass absolut kein Unterschied zwischen den Rechnungen bzgl. verschiedener Basen besteht, oder anders ausgedrückt binär und dezimal ist vollkommen gleich bedeutend und darum egal. Bezeichne von diesem Punkt an B die Menge der reellen Zahlen mit den Dezimaldarstellungen bzgl. b .

Es gilt

$$\text{id} : GKZ \xrightarrow{\text{Ordnung, Norm}} B$$

und

$$\#GKZ \xrightarrow{\text{Arithmetik, Darstellung}} B.$$

¹Darstellung: $GKZ \subseteq B$, genauer gibt es in GKZ nur endliche Summen, also

$$GKZ \subseteq \bigcup_{k-l=m} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\}.$$

Als Konvention gelte stets $l \leq k$. Es gibt jetzt die natürliche Kette

$$\bigcup_{k-l=1} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\} \subseteq \bigcup_{k-l=2} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\} \subseteq \bigcup_{k-l=3} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\} \subseteq \dots$$

und wir definieren die Mantissenlänge m_{GKZ} oder fortan einfach m als

$$m = \min \left\{ n : GKZ \subseteq \bigcup_{k-l=n} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\} \right\}$$

(wohldefiniert da Minimum unter Einfluss einer Kette). Dasselbe Konzept lässt sich für k und l verwenden.

$$\dots \subseteq \bigcup_{k \leq 1} \left\{ \sum_{i=-\infty}^k a_i b^i : a_i < b \right\} \subseteq \bigcup_{k \leq 2} \left\{ \sum_{i=-\infty}^k a_i b^i : a_i < b \right\} \subseteq \dots$$

¹ \hookrightarrow heißt injektiver Morphismus, Morphismus bzgl. der angegebenen Eigenschaften

und wir definieren den Maximalexponenten e_{max} als

$$e_{max} = \min \left\{ e + 1 : GKZ \subseteq \bigcup_{k \leq e} \left\{ \sum_{i=-\infty}^k a_i b^i : a_i < b \right\} \right\},$$

sowie

$$\dots \supseteq \bigcup_{l \geq 2} \left\{ \sum_{i=l}^{\infty} a_i b^i : a_i < b \right\} \supseteq \bigcup_{l \geq 1} \left\{ \sum_{i=l}^{\infty} a_i b^i : a_i < b \right\} \supseteq \dots$$

und Minimalexponenten e_{min} als

$$e_{min} = \max \left\{ e + m : GKZ \subseteq \bigcup_{l \geq e} \left\{ \sum_{i=l}^{\infty} a_i b^i : a_i < b \right\} \right\}$$

(für die Wohldefiniertheit wählt man einfach einen geeigneten Startpunkt für eine Kette, der wiederum existiert da GKZ beschränkt und nicht leer). Fügt man dies zusammen, dann gilt folgendes

Korollar. *Sei $GKZ \subseteq B$ und existieren die Werte m, e_{max} und e_{min} (GKZ ist beschränkt und die Maximalzahl der Summanden in den Summen ist beschränkt). Dann gilt*

$$GKZ \text{ ist GKZ-Format} \Leftrightarrow GKZ = \bigcup_{\substack{k-l=m \\ e_{min} \leq l \leq k \leq e_{max}}} \left\{ \sum_{i=l}^k a_i b^i : a_i < b \right\}.$$

Bis hier gibt es noch nichts gegen eine Einbettung, das Problem liegt jetzt darin, wie wir diese Zahlen aufschreiben, denn wir wählen für ein GKZ-Format jetzt das Standardintervall

$$I_m := \left\{ \sum_{i=1}^m a_i b^{-i} \right\}$$

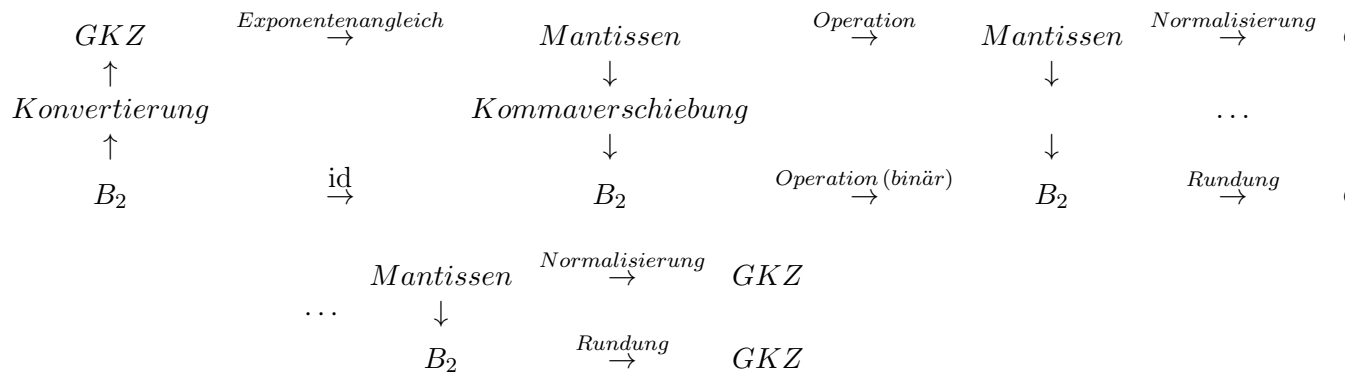
und schreiben die Elemente von GKZ als

$$GKZ = \bigcup_{e_{min} \leq e \leq e_{max}} I_m \cdot b^e.$$

Wir schreiben GKZen demnach als eine Skalierung des Standardintervalls, teilen die Darstellung also streng in zwei Teile, was unseren Dezimaldarstellungen aber widerspricht (Skalierung entspricht einer Repositionierung des Kommas, in GKZ-Formaten lassen wir verschieden Positionen zu, in Dezimaldarstellungen nicht). Die fehlende Einbettung bzgl. der Arithmetik ist klar.

Nun zur Arithmetik von GKZen. Beginnen wir mit unserem letzten Resultat. Sei a eine reelle Zahl mit $a \in GKZ$ und bezeichne a_\circ a in der Dartsellung \circ , sowie $[a_\circ] = a$ ($[\cdot]$ entfernt die Darstellung von der Zahl). Dann gilt $a_2 = a_{10}$ und $a_2 \neq a_{GKZ}$, aber $[a_2] = [a_{GKZ}]$.

Die Arithmetik im GKZ-Format läuft jetzt in der strengen Abfolge Angleichung der Exponenten (die zuvor unabhängigen Mantissen sind jetzt gleichwertig und können verrechnet werden; die Mantissen sind in Binärdarstellung, werden also mit binär-Arithmetik bearbeitet), Berechnung der neuen Mantisse, Normalisierung. Was passiert aber, wenn man einfach rechnet, und erst danach konvertiert? Knackpunkt ist, dass da man im Binären keinen Exponentenangleich durchführen muss direkt im Schritt der Mantissenberechnung ansetzt. Bleibt nachzuweisen, dass die Ergebnisse gleich sind. Für die Binärarithmetik ist nicht wichtig wo sich das Komma befindet, nur dass es bei allen Beteiligten gleich verstanden wird; der Exponentenangleich in der GKZ-Arithmetik hebt die Rechnung auf ein Niveau, dass also kongruent zur Binärarithmetik ist, das anschließende Normalisieren erfolgt nach denselben Regeln wie denen sich auch die ursprüngliche Konvertierung unterwerfen lässt. Zusammengefasst bewegen wir uns in folgendem Zusammenhang (zur Vereinfachung sei o.B.d.A. die Operation als unär angesehen).



Die Wege lassen sich wiederum nur unter der Abbildung $[\cdot]$ vergleichen! Anders ausgedrückt: Man rechnet unter Einfluss von $[\cdot]$ und 'kehrt' danach diese wider (heißt man fügt willkürlich eine Darstellung hinzu, also eine zusätzliche Struktur auf etwas, was bis dahin ohne diese betrachtet wurde, was die schwere des Vergehens aufzeigen sollte).