Analysis für Informatiker **WiSe 2020-21**

Über die Komplexität komplexer Division **8. Februar 2021 (neueste Version)**

valentinpi
Freie Universität Berlin

Zusammenfassung

Diese Ausarbeitung zeigt die Division komplexer Zahlen in algebraischer Darstellung mithilfe linearer Gleichungssysteme und zeigt eine Reduktion der Komplexität des Algorithmus. Diese ist analog zu der Reduktion der Komplexität komplexer Multiplikation, welche bereits Gauß entdeckt hatte.[1] Die Reduktion selbst wurde von Aleksandr Cariow entdeckt. Hier soll dieses sehr kurze Paper erklärt werden.[2]

Anmerkung. Wie in der Vorlesung eingeführt ist in Polardarstellung die Multiplikation und Division komplexer Zahlen sehr leicht, jedoch benötigen wir den Arcus Cosinus für das Argument der Zahl. Jener liegt aber nicht in jeder Hardwarearchitektur vor. Die hier besprochene Reduktion ist also hardwareorientiert. Wir können so etwa eine effiziente Divisionsinstruktion logisch implementieren.
Wir optimieren hier ausserdem nach Multiplikationen. Während jene auf modernen CPU-Architekturen zwar sehr schnell sind, so haben sie aber den Nachteil, dass deren Schaltkreise sehr viel Strom verbrauchen. Im IoT-Bereich ist dies schwer zu verkraften.[3]

Aufgabe: Gegeben sind zwei komplexe Zahlen $x = x_r + ix_i$, $y = y_r + iy_i \neq 0$; $x, y \in \mathbb{C}$. Berechne:

$$z = \frac{x}{y}$$

mit möglichst wenigen arithmetischen Operationen.

Komplexe Division Zunächst bestimmen wir zu einem beliebigen $x = x_r + ix_i \in \mathbb{C}$ die reziproke komplexe Zahl $\frac{1}{x} = x'_r + ix'_i$. Es muss gelten:

$$x \cdot \frac{1}{x} = 1 \qquad (x_r + ix_i) \cdot (x'_r + ix'_i) = \underbrace{(x_r x'_r - x_i x'_i)}_{1} + i \underbrace{(x_r x'_i + x_i x'_r)}_{0} = 1$$

Es ergibt sich das folgende lineare Gleichungssystem, einmal in Gleichungen, einmal in Matrixschreibweise:

$$\begin{array}{ll} \text{I} & x_r x'_r - x_i x'_i = 1 \\ \text{II} & x_r x'_i + x_i x'_r = 0 \end{array} \qquad \begin{pmatrix} x_r & -x_i \\ x_i & x_r \end{pmatrix} \cdot \begin{pmatrix} x'_r \\ x'_i \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Wir lösen das LGS:

$$x_r \text{II} - x_i \text{I}\colon x_r^2 x'_i + x_i^2 x'_i = -x_i \Rightarrow x'_i = \frac{-x_i}{x_r^2 + x_i^2}$$

$$\left(x'_i = \frac{-x_i}{x_r^2 + x_i^2} \right) \text{ in I: } x_r x'_r + \frac{x_i^2}{x_r^2 + x_i^2} = 1 \Rightarrow x'_r = \frac{1}{x_r} - \frac{x_i^2}{x_r(x_r^2 + x_i^2)} = \frac{x_r}{x_r^2 + x_i^2}$$

Damit folgt:

$$\frac{1}{x} = \frac{x_r}{x_r^2 + x_i^2} + i \frac{-x_i}{x_r^2 + x_i^2}$$

Für die ursprüngliche Aufgabe folgt demnach:

$$z = \frac{x}{y} = (x_r + ix_i) \cdot \left(\frac{y_r}{y_r^2 + y_i^2} + i \frac{-y_i}{y_r^2 + y_i^2} \right) = \frac{x_r y_r + x_i y_i}{y_r^2 + y_i^2} + i \frac{-x_r y_i + x_i y_r}{y_r^2 + y_i^2}$$

Komplexität: 4 Multiplikationen, 3 Additionen, 2 Quadrate, 2 Divisionen reeller Zahlen.

[1] https://en.wikipedia.org/wiki/Multiplication_algorithm#Complex_multiplication_algorithm
[2] https://arxiv.org/ftp/arxiv/papers/1608/1608.07596.pdf
[3] https://en.wikipedia.org/wiki/Operation_Reduction_for_Low_Power
Allesamt zuletzt am 07.02.2021, 13:55 Uhr, aufgerufen.

Optimierung nach Cariow Das Ziel ist, die multiplikative Komplexität zu reduzieren. Wir nutzen hier Matrizen. Zunächst stellen wir die Zahl $z = z_r + iz_i$ als Matrix aus dem Real- und Imaginärteil dar:

$$\begin{pmatrix} z_r \\ z_i \end{pmatrix} = \frac{1}{y_r^2 + y_i^2} \begin{pmatrix} x_r y_r + x_i y_i \\ -x_r y_i + x_i y_r \end{pmatrix} = \begin{pmatrix} \delta & 0 \\ 0 & \delta \end{pmatrix} \begin{pmatrix} x_r & x_i \\ x_i & -x_r \end{pmatrix} \begin{pmatrix} y_r \\ y_i \end{pmatrix} \text{ mit } \delta := \frac{1}{y_r^2 + y_i^2}$$

Nun führen wir eine Faktorisierung durch. Wir faktorisieren die mittlere Matrix der Teile von x. Dieser Teil stellt die durch das Paper erbrachte, zentrale Beobachtung dar:

$$\begin{pmatrix} x_r & x_i \\ x_i & -x_r \end{pmatrix} = \begin{pmatrix} x_r - x_i + x_i & x_i \\ x_i & -x_r - x_i + x_i \end{pmatrix} = \begin{pmatrix} x_r - x_i & 0 & x_i \\ 0 & x_r + x_i & x_i \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \\ 1 & 1 \end{pmatrix}$$

Wir stellen uns dies folgendermaßen vor: In den diagonalen Einträgen werden geschickte Nullen addiert, dann wird eine 3×2-Matrix eingeführt, welche je die x_i addiert und den Eintrag $x_r - x_i$, sowie $x_r + x_i$ negiert auffasst. Nun erstellen wir aus der Matrix der x_r und x_i eine Diagonalmatrix:

$$\begin{pmatrix} x_r - x_i & 0 & x_i \\ 0 & x_r + x_i & x_i \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_r - x_i & 0 & 0 \\ 0 & x_r + x_i & 0 \\ 0 & 0 & x_i \end{pmatrix}$$

Das Gesamtergebnis:

$$\begin{pmatrix} z_r \\ z_i \end{pmatrix} = \begin{pmatrix} \delta & 0 \\ 0 & \delta \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_r - x_i & 0 & 0 \\ 0 & x_r + x_i & 0 \\ 0 & 0 & x_i \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} y_r \\ y_i \end{pmatrix}$$

Komplexität: 3 Multiplikationen, 6 Additionen, 2 Quadrate, 2 Divisionen reeller Zahlen.
Die Multiplikationen mit den 1en und (-1)en können wir je weglassen. In der Diagonalmatrix in der Mitte finden zwei Additionen statt. Das letzte Produkt rechts ergibt:

$$\begin{pmatrix} y_r \\ -y_i \\ y_r + y_i \end{pmatrix}$$

Also eine Addition. Die darauffolgende Multiplikation mit der Diagonalmatrix erfordert exakt drei Multiplikationen. Es werden nämlich schlicht die nichtleeren Einträge jeder Zeile multipliziert. Im letzten Produkt kommen dann nur noch zwei Additionen vor. Dazu kommt die Addition, die zwei Divisionen und die zwei Quadrate für δ.

Tatsächlich befindet sich im Paper von Cariow dazu ein Fehler: Die dritte Matrix von links hat einen Vorzeichenfehler bei der eins in $(2, 2)$.