Verteilte Matrixen

Nina Herrmann

November 18, 2020

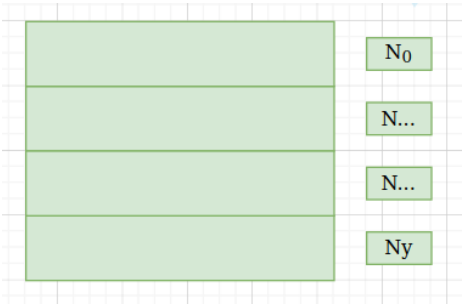
1 Konzept

1.1 Aufteilung auf Rechenknoten

Zunächst wird die Matrix reihenweise auf die Rechnerknoten aufgeteilt. Das heißt der erste Index für einen Rechenknoten ergibt sich aus:

$$\text{globalIndexNode} = \text{id} * (\text{n/nNodes}) \tag{1}$$

wobei n die Gesamtanzahl der Elemente ist, id die id des Rechenknoten und $nNodes$ die Anzahl der Rechenknoten.



Damit die Aufteilung effizient sein kann gilt, dass die Anzahl der Reihen ein Vielfaches der Anzahl der Knoten ist.

$$\text{locRows} * \text{nNodes} = \text{nRows} \quad \text{locRows} \in N \tag{2}$$

Beispiel: Bei einer 10 * 10 Matrix und zwei Rechenknoten werden Reihe 0-4 dem ersten Rechenknoten zugewiesen, und Reihe 5-9 dem zweiten Rechenknoten. Eine Aufteilung auf drei Rechenknoten ist nicht möglich.

1.2 Aufteilung auf GPUs und CPUs

Danach werden die lokalen Elemente reihenweise auf die GPUs und CPUs aufgeteilt. Dabei wird angenommen, dass die Anzahl der lokalen Elemente sich auf CPU und zu gleichen Teilen auf die GPUs abhängig von der angegebenen `cpu_fraction` aufteilen lassen.
Wenn die vorherige 10×10 Matrix auf zwei Rechenknoten aufgeteilt wurde haben wir eine 5×10 Matrix lokal auf dem Rechenknoten 0. Wenn dieser Rechenknoten eine CPU und vier GPUs hat und ein Anteil von 28% auf der CPU berechnet werden soll werden die ersten 14 Elemente auf der CPU berechnet. Jede GPU berechnet dann neun Elemente.
Eine nicht zulässige `cpu_fraction` wäre in diesem Fall 30%. 15 Elemente würden auf der CPU bearbeitet werden, aber die restlichen 35 Elemente lassen sich nicht gleichmäßig auf vier GPUs aufteilen.