

Projektarbeit PC1

Dokumentation

Maurice Wenig

Inhaltsverzeichnis

1	Vorgehen	2
1.1	Kommunikation	2
1.2	Berechnung	2
2	Implementierung	3
3	Benutzung	3

1 Vorgehen

Das initiale Feld wird gleichmäßig auf alle Prozesse aufgeteilt. In jedem Prozess werden die neuen Temperaturen im lokalen *chunk* berechnet. Dafür gibt es einen Austausch der aktuellen Werte mit den Nachbarprozessen. Diese Werte werden in den Ghost Cells in einem Halo der Breite g um den lokalen *chunk* gespeichert. Ab hier werden die Ghost Cells als Teil des lokalen Chunks angesehen. Der Teil des Chunks, der keine Ghost Cells beinhaltet, wird als innerer Chunk bezeichnet. Am Ende der Berechnung werden die inneren Chunks der einzelnen Prozesse wieder zusammengesetzt.

```

1: split_up_domain()
2: for  $i = 0 \rightarrow n\_iterations$  do
3:   if  $i \% g == 0$  then
4:     exchange_ghost_cells()
5:   calculate()
6: collect()

```

1.1 Kommunikation

Die angewandten Techniken wurden größtenteils dem Paper “Ghost Cell Pattern” von [Kjolstad and Snir](#) entnommen. Es wird ein Deep Halo benutzt und die Corner Cells sollen auch effizient übertragen werden. Da die Ost-West-Kommunikation immer vor der Nord-Süd-Kommunikation passieren muss, damit die Ecken richtig übertragen werden, wird zwischen der Kommunikation von den beiden Richtungen gewartet.

```

1: irecv_east()
2: irecv_west()
3: isend_east()
4: isend_west()
5: wait()
6: irecv_north()
7: irecv_south()
8: isend_north()
9: isend_south()
10: wait()

```

Falls ein Nachbar in eine Richtung nicht existiert, werden die Ghost Cells mit Padding gefüllt. Dabei wird immer der nächste Wert des inneren Chunks kopiert.

1.2 Berechnung

Es werden immer nur so viele Zellen berechnet, wie für den nächsten Schritt benötigt werden. Dafür wird eine *border* eingeführt. Am Anfang umfasst die *border* den ganzen Chunk, inklusive Halo, bis auf den äußersten Ring. Am Ende umfasst die *border* nur noch den inneren Chunk.

Um die neuen Werte zu berechnen werden zwei Arrays benutzt. Eines, das die alten Werte enthält und eines, das die Ergebnisse der Berechnung enthält. Als Vorbereitung für den nächsten

Schritt werden am Ende der Berechnung die Arrays vertauscht.

```
1: border := adapt_border()
2: for  $x, y$  in the border do
3:   results[ $x, y$ ] := calculation_step( $x, y$ )           ▷ calculation_step uses old_values
4: swap(results, old_values)
```

2 Implementierung

3 Benutzung

Literatur

Fredrik Berg Kjolstad and Marc Snir. Ghost cell pattern. In *Proceedings of the 2010 Workshop on Parallel Programming Patterns - ParaPloP '10*. ACM Press, 2010. doi: 10.1145/1953611.1953615.

Abbildungsverzeichnis

Tabellenverzeichnis