

TECHNISCHE UNIVERSITÄT DRESDEN

ZENTRUM FÜR INFORMATIONSDIENSTE
UND HOCHLEISTUNGSRECHNEN
PROF. DR. WOLFGANG E. NAGEL

Komplexpraktikum "Paralleles Rechnen"
B - Thread-parallele Ausführung von Conways
Game-of-Life

Bengt Lennicke

Dresden, 22. Januar 2024

Inhaltsverzeichnis

1	Aufgabenstellung	3
1.1	Conways 'Game of Life'	3
2	Umsetzung	3
3	Ausführung	3
3.1	Hardware	3
3.2	Programm-Versionen	4
3.3	Messung	4
4	Auswertung	4
	Literatur	5

1 Aufgabenstellung

Implementieren Sie eine thread-parallele Variante von Conways 'Game-of-Life' mit periodic boundary conditions. Nutzen Sie dazu OpenMP Compiler-Direktiven. Benutzen Sie double buffering um Abhängigkeiten aufzulösen.

- Beschreiben Sie Ihren Ansatz und gehen Sie sicher, dass die Arbeit thread-parallel ausgeführt wird.
- Messen und Vergleichen Sie die Ausführungszeiten für 1,2,4,8,16 und 32 Threads, für den GCC, als auch den Intel Compiler bei Feldgrößen von 128x128, 512x512, 2048x2048, 8192x8192 und 32768x32768.
- Nutzen Sie für die Berechnung eine geeignete Anzahl an Schleifendurchläufen (Zyklen des Spiels), sodass der genutzte Timer genau genug ist.
- Nutzen Sie dafür die "romeo" Partition von taurus.
- Achten Sie darauf, dass benachbarte Threads möglichst nah einander gescheduled sind.
- Testen Sie für die Feldgröße 128x128, welchen Einfluss die OpenMP Schleifenschedulingverfahren haben (OMP_SCHEDULE), indem Sie für die Ausführung mit 32-Threads des mit Intel kompilierten Benchmarks die Verfahren static, dynamic, guided, und auto bei default chunk size vergleichen
- Führen Sie jeweils 20 Messungen durch und analysieren Sie die Ergebnisse mit geeigneten statistischen Mitteln.

1.1 Conways 'Game of Life'

Conways 'Game of Life' ist ein Gedankenspiel bei dem auf einem zweidimensionalen Spielbrett(board) Felder(cell) 'lebendig' oder 'tot' sind. Im Spielverlauf können die Felder 'lebendig werden', 'sterben', 'am Leben' bzw. 'Tot bleiben'. Das Aktualisieren basiert auf 4 Regeln:[2]

- Jede lebende Zelle mit weniger als 2 lebendigen Nachbarn stirbt (Unterpopulation)
- Jede lebende Zelle mit 2 oder 3 lebendigen Nachbarn bleibt am Leben
- Jede lebende Zelle mit mehr als 3 lebendigen Nachbarn stirbt (Überpopulation)
- Jede tote Zelle mit genau 3 lebendigen Nachbarn wird lebendig (Reproduktion)

Mit diesen einfachen Regeln und wiederholtem Aktualisieren des Boards bildet es ein komplexes System, welches mehrere Interpretationsweisen erlaubt.[2]

In diesem Bericht ist das Spiel selbst nicht von besonderem Interesse, sondern die Programmiertechnische Umsetzung der Aktualisierung des Boards unter der Verwendung von openMP.

2 Umsetzung

3 Ausführung

3.1 Hardware

Die Messung für die Bearbeitung der Aufgaben sind auf dem CPU Cluster Romeo der TU Dresden ausgeführt worden. Dieser Cluster bietet 192 nodes mit jeweils [1]:

- 2 x AMD EPYC CPU 7702 (64 cores) @ 2.0 GHz, Multithreading möglich

- 512 GB RAM
- 200 GB SSD Speicher
- Betriebssystem: Rocky Linux 8.7

3.2 Programm-Versionen

Relevant für die Reproduzierbarkeit sind die Versionen der verwendeten Bibliotheken und Programme.

- GNU Make 4.2.1
- gcc (GCC) 10.3.0

3.3 Messung

4 Auswertung

Literatur

[1] HPC Compendium, 'HPC Resources', 12.01.2024

https://doc.zih.tu-dresden.de/jobs_and_resources/hardware_overview/#romeo

[2] Wikipedia Seite, 'Conways Spiel des Lebens', 22.01.2024

https://de.wikipedia.org/wiki/Conways_Spiel_des_Lebens#Die_Spielregeln

