Titel des Papers

Autor

May 17, 2024

Abstract

Die Art und Weise, wie auf Daten zugegriffen wird, kann einen erheblichen Einfluss auf die Performance eines Algorithmus haben. Insbesondere wenn große Datenmengen häufig zwischen langsamen Speichermedien und dem Cache ausgetauscht werden, spricht man von einem Engpass des Speicherinterfaces. Durch effiziente Datenzugriffe und optimale Nutzung des Caches kann nicht nur die Performance signifikant gesteigert werden, sondern es wird auch sichergestellt, dass die Leistungsfähigkeit selbst bei der Verarbeitung großer Datenmengen stabil bleibt. In diesem Paper werden Methoden wie Loop Unrolling, Loop Fusion, Blocking analysiert. Um die Effektivität mancher Methoden zu demonstrieren, werden diese gebenchmarkt und mit den Standardmethoden verglichen.

1 Einleitung

Im letzten Jahrzehnt hat sich die Rechenleistung von Prozessoren erheblich gesteigert. In Abbildung 1 und 1.1 ist zu erkennen, dass seit der Einführung der 4th Generation im Jahr 2013 sich die Anzahl der Kerne bei Intel, die für den allgemeinen Verbrauchermarkt verfügbar sind, von 6 auf 24 Kernen erhöht hat. Es ist wichtig zu beachten, dass es zwar Prozessoren mit einer noch höheren Anzahl von Kernen gibt, diese jedoch in der Regel nicht für den Standard-Endverbraucher bestimmt sind. Ebenso zeigt sich in den Abbildungen 2 und 2.1 ein Anstieg der maximalen Taktfrequenz über die Jahre. Wenn man nun die theoretische maximale Rechenleistung, definiert als:
P_{max} = Anzahl der Kerne $*$ Turbo Taktfrequenz $*$ Flops pro Taktzyklus,
und der Entwicklung der Speicherbandbreite gegenüberstellt, wird in Abbildung 3 deutlich, dass die Zunahme der Speicherbandbreite nicht im gleichen Maße wie die Rechenleistung ansteigt. Im Detail hat sich die Bandbreite von 51.2 GB/s im Jahr 2013 auf 89.6 GB/s im Jahr 2024 erhöht. Parallel dazu ist die Performance im gleichen Zeitraum von 187.2 FLOPS/s auf 1945.6 FLOPS/s gestiegen. Diese Diskrepanz zwischen der gesteigerten Rechenleistung und der vergleichsweise langsamer wachsenden Speicherbandbreite verdeutlicht die Notwendigkeit einer Optimierung von Datenzugriffen. Ein effizienter Einsatz des Caches ist dabei unerlässlich, um die Leistung zu maximieren.

2 Berechnung der Performancegrenzen

In den nachfolgenden Unterabschnitten werden die Formeln vorgestellt, die zur Bewertung von loop-basiertem Code und zur Berechnung der Performancegrenzen herangezogen werden. Es ist jedoch wichtig zu berücksichtigen, dass diese Formeln lediglich eine Annäherung darstellen und nur unter bestimmten Bedingungen gültig sind. Beispielsweise wird vorausgesetzt, dass alle Ressourcen vollständig ausgeschöpft werden. Zudem basieren die Formeln auf Aspekten des idealen Cache-Modells, wie einem unendlich schnellen Cache und der Nichtexistenz von Latenzzeiten.

2.1 Maschinenbalance

Die Maschinenbalance B_m ist das Verhältnis aus der maximalen Bandbreite und der maximalen theoretischen Rechenleistung $B_m = \frac{B_{max}}{P_{max}}$
Es beschreibt also wie viele Daten pro Flop übertragen werden könnnen. Um dies zu verdeutlichen, betrachten wir die B_m für den i7-9700k Prozessor, der eine maximale Bandbreite von 41.6 GB/s und eine maximale Rechenleistung von 8 Kerne * 4.9 GHz * 16 FLOPS/Taktzyklus = 627,2 FLOPS/s hat. 41.6 GB/s / 8 Byte = 5.2 GWords/s Somit ergibt sich $B_m = \frac{41.6\text{GB/s}}{627.2\text{FLOPS/s}} = 0.066\text{GB/FLOP}$.

2.2 Codebalance

In diesem Unterabschnitt erklären wir das Konzept der Codebalance und führen die entsprechenden Berechnungen durch.

2.3 Lichtgeschwindigkeit

Hier erläutern wir das Konzept der Lichtgeschwindigkeit in Bezug auf die Performance und führen die entsprechenden Berechnungen durch.

2.4 Maximale Performance

In diesem Unterabschnitt berechnen wir die maximale erreichbare Performance für loop-basierten Code.

2.5 Zusammenfassung

Hier fassen wir die Ergebnisse der Berechnungen zusammen und diskutieren ihre Bedeutung für die Optimierung von Datenzugriffen und die effiziente Nutzung des Caches.

3 Ergebnisse

Hier werden die Ergebnisse und deren Interpretation präsentiert.

4 Diskussion

Hier wird über die Bedeutung und Implikationen der Ergebnisse diskutiert.

5 Fazit

Hier wird das Fazit des Papers gezogen und ein Ausblick gegeben.

References

[1] Autor1, Titel des Referenzartikels, Journal, Jahr.

[2] Autor2, Titel des Referenzartikels, Journal, Jahr.