

mi-bildbalken

Parallele Sortierung

Björn Rathjen Patrick Winterstein
Freie Universität Berlin

Proseminar Algorithmen, SS14

Inhalt

Motivation

Grundlage des Sortierens

- Komparator

Sortiernetzwerk

- Aufbau

- Korrektheit

Laufzeit

- Herleitung

- Vergleich mit Software sortieren

Gegenüberstellung

Zusammenfassung

Ausblick

- Hypercube

- Anhang

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

Motivator : Allgemein

ist Basis für :

- ▶ Suche
- ▶ (Sortierung)
 - ▶ Listen
 - ▶ Wörterbücher
 - ▶ ...
- ▶ Ist dies auch in Hardware möglich ?

Motivation

Grundlage des Sortierens

Komparator

Sortiernetzwerk

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

Aufbau

- ▶ 2 Eingänge
- ▶ vergleichender Baustein
- ▶ 2 Ausgänge

Vergleichender Baustein (ii)

```
void comp(chan in1, in2, out1 out2){  
    a = <- in1;  
    b = <- in2;  
  
    if (a < b){  
        out1 <- a;  
        out2 <- b;  
        return void;  
    }  
    out1 <- b;  
    out2 <- a;  
    return void;  
}
```

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Aufbau

Korrektheit

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

- ▶ mehrere Eingabeleitungen
- ▶ Vergleichende Schritte müssen dazwischen laufen
- ▶ mehrere Ausgabeleitungen sortierte Ausgabe

nativ : Aufgabe

Aufgabe

- ▶ Resultat soll sortierte Eingabe sein

nativ : Aufgabe

Aufgabe

- ▶ Resultat soll sortierte Eingabe sein
- grundlegendes Prinzip
- ▶ intuitiver Einsatz von Vergleichen
 - ▶ Schrittweises sortieren

nativ : grundlegendes Prinzip

Bild
siehe 2

Demonstration

Bild kleiner Zahlenfolge 4-8-16
Beispiel

0,1-Prinzip

Theorem

Wenn es eine Folge A gibt, die ein Sortiennetzwerk nicht sortiert, so existiert auch eine 0,1-Folge, die von diesem Netzwerk nicht sortiert wird.

Proof.

fill



Ansatz

man kann jede Zahlenfolge durch eine 0,1 Folge repräsentieren

$$f(c) = \begin{cases} 0, & \text{if } c < k \\ 1, & \text{if } c \geq k \end{cases}$$

0,1- Beispiel

Beispiel an der Tafel ? Bild

effektiverer Ansatz

- ▶ Aufgabe
- ▶ grundlegendes Prinzip
- ▶ Demonstration (kleines Beispiel)
- ▶ Laufzeit

effektiver

Aufgabe :

- ▶ Resultat soll sortierte Eingabe sein
- ▶ soll effizient sein

effektiver

Aufgabe :

- ▶ Resultat soll sortierte Eingabe sein
- ▶ soll effizient sein

grundlegendes Prinzip :

- ▶ intuitiver Einsatz von Vergleichen
+ Einbezug von Teile und Herrscher

Aufteilung

Bild Buch Biton-Sortierer

Biton -Sortierer : Aufbau

Bild

Demonstration

Bild kleiner Zahlenfolge 4-8-16
Beispiel

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Laufzeit

Herleitung

Vergleich mit Software sortieren

Gegenüberstellung

Zusammenfassung

Ausblick

$$\frac{1}{2} \cdot \log_2 n (\log_2 n + 1)$$

1

$$\frac{1}{2} \cdot \log_2 n (\log_2 n + 1)$$

Herleitung

Herleitung

Unterschiedliche Betrachtungen Schritte gegen Vergleiche, versuch der Darstellung

Bubblesort im Hardwarenetz

Bild Bubblesort
siehe 1

Mergesort im Hardwarenetz

Bild Mergesort
siehe 4

Bild Quicksort

Bild 5

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

- ▶ Geschwindigkeit vs Variabilität
 - ▶ hohe Geschwindigkeit durch direkte Hardware Implementierung
 - ▶ starre Struktur , bildet Rahmen der Möglichkeiten
 - ▶ stark typisierte Eingabe
- ▶ Hardwareaufwand vs Softwareaufwand
 - ▶ Software zur Auswertung keine zum sortieren
 - ▶ geringe Skalierbarkeit
 - ▶ hoher Aufwand wenn Eingabelimit überschritten wird
 - ▶ nur lokal
 - ▶ Hardware Konzeption eventuell aufwendiger

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

Zusammenfassung

- ▶ paralleles sortieren ist schnell und effizient
- ▶ stark Problemabhängig
- ▶ ... muss noch gefüllt werden

Motivation

Grundlage des Sortierens

Sortiernetzwerk

Laufzeit

Gegenüberstellung

Zusammenfassung

Ausblick

Hypercube

Anhang

Ausblick

weiter

Hyprecube

?i

Aufbau

structur

Funktion

Ende

Fragen, Anregungen?
(keine Liederwünsche)

For Further Reading I



A. Author.

Taschenbuch der Algorithmen.

Springer Verlag , 2008.



Tom Leighton.

Einführung in Parallele Algorithmen und Architekturen

Gitter, Bäume und Hypercubes.

Thomson Publishing , 1997.



S. Someone.

<http://www.iti.fh->

[flensburg.de/lang/algorithmen/sortieren/networks/nulleins.htm](http://www.iti.fh-flensburg.de/lang/algorithmen/sortieren/networks/nulleins.htm)