

Think parallel

Möglichkeiten der Parallelisierung

Beate Künneke Unternehmensberatung
bk@berlin.de

DOAG Konferenz 2012, Nürnberg

Agenda

1 Motivation und Zielsetzung

2 Beispiel

3 Begriffswelt

4 Theorie PRAM

5 Von der Theorie zur Praxis

6 Praxis

Moore'sches Gesetz:

Die Komplexität integrierter Schaltkreise (Anzahl der Schaltkreiskomponenten auf einem Computerchip) verdoppelt sich regelmäßig.

Anforderungen an die Entwicklung:

- Steigende Leistung
- Geringerer Platzbedarf
- Geringere Kosten

Zurzeit geht man von 18-20 Monaten aus.

Aber: dies es gibt eine physikalische Grenze, die voraussichtlich in 10-15 Jahren erreicht wird.

Anstieg Datenvolumen (Big Data):

Die Menge an digitalen Daten wird in den kommenden Jahren exponentiell anwachsen, prognostiziert IDC.

- 2010 durchbrach das globale digitale Universum erstmals die Zettabyte-Barriere (1 mit 21 Nullen),
- 2011 kalkulierte man die weltweit produzierte Datenmenge auf ein Volumen von 1,8 Zettabyte, das sind 1,8 Billionen Gigabyte.

=> 90 Prozent des derzeitigen globalen Datenbestands sind in den beiden zurückliegenden Jahren entstanden.

4 Entwicklungen bestimmen die Veränderungen:

- **Volumen:** Die Menge der anfallenden Informationen wird weiter drastisch wachsen. Grenzen sind keine erkennbar.
- **Quellen:** Neben dem Volumen wächst auch die Zahl der Datenquellen. Waren es früher hauptsächlich Transaktionssysteme, die strukturierte Daten produzierten, kommen heute vielfältige weitere Quellen hinzu wie beispielsweise Sensoren, Social Networks und mobile Endgeräte.
- **Geschwindigkeit:** In Zukunft werden Analysen und Auswertungen ad hoc erwartet und gefordert. Wo früher ein monatlicher Report ausreichte, geht heute der Trend in Richtung Echtzeitanalysen.
- **Nutzer:** Mit dem steigenden Datenvolumen wächst auch die Zahl der Nutzer von Datenanalysen in den Unternehmen. Die Herausforderung liegt darin, die unterschiedlichen Rollen mit den passenden Auswertungen zu versorgen.

Konsequenzen:

- Zur weiteren Steigerung des Datendurchsatzes sind auf herkömmlichen sequentiellen Weg Grenzen gesetzt und schon in greifbarer Nähe.

- Die Datenverarbeitung muss geteilt und damit verteilt werden. Teil-Ergebnisse müssen zusammengeführt und verarbeitet werden.

- Die Teilung und Verteilung der Arbeitspakete kann
 - Allgemein betriebssystemseitig oder
 - Durch geeignete dem Problem **maßgeschneiderte Algorithmen** gelöst werden (→ Zukunft).

Beispiele:

- **Seti@Home:**

SETI@home (Search for extraterrestrial intelligence at home, englisch für „Suche nach außerirdischer Intelligenz zu Hause“) ist ein Verteiltes-Rechnen-Projekt der Universität Berkeley, das sich mit der Suche nach außerirdischem intelligenten Leben befasst.

- **IBM Watson:**

Eine intelligente Software und 2880 Prozessorkerne sind beim Beantworten von Quiz-Fragen schneller als ein Mensch mit der besten Allgemeinbildung. Diesen Beweis hat jetzt "Watson" in der amerikanischen Quiz-Show "Jeopardy" erbracht.

Häufige wiederkehrende Problemstellungen:

- Sortieren
- Suchen (Min, Max, ..)
- Strategie (Schach, Traveling Sales Man)
- Graphenalgorithmen
- Logikprogrammierung
- Lösung von Differentialgleichungen
- Matrixmultiplikationen
- ...

Beispiel für Parallelisierung eines Problems

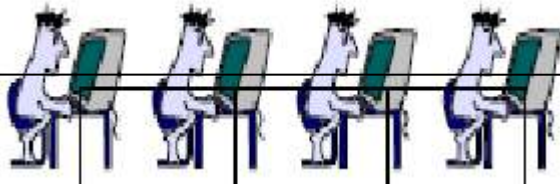
$$1+8+7+9+6+2+6+13= ???$$



$$1+8=9, 9+7=16, 16+9=25, \dots$$

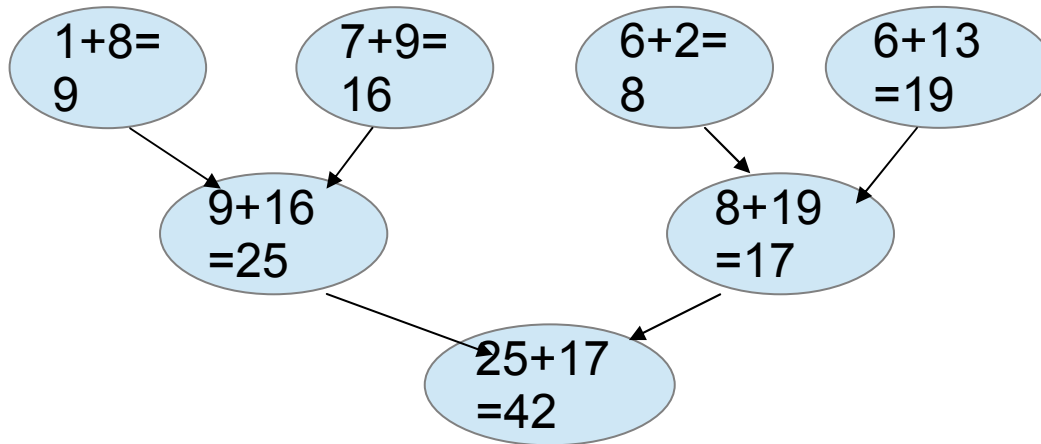
Zeit : 7 x Addition = $O(n)$

$n=8$ Zahlen, 7 Additionen
Gesamtzeit 7 Zeiteinheiten



7 x Addition = $O(\log n)$

Gesamtzeit 3 Zeiteinheiten



Aber: $3 \cdot 4 = 12$

Was fällt am Beispiel auf?

- Die Gesamtzeit bei der parallelen Verarbeitung kann geringer als bei der sequentiellen Abarbeitung sein.
- Betrachtet man das die Anzahl der Arbeitseinheiten*Zeit, so ist man schlechter als sequentiell.
- Die Anzahl der Arbeitseinheiten spielt eine Rolle
 - Zu viel bringt nichts
 - Zu wenig aber auch nicht
- Der Algorithmus zur parallelen Verarbeitung ist zur Erreichung eines besseren Gesamtergebnisses oft anders als sequentiell.
- Für die parallele Verarbeitung muss Kommunikation stattfinden.
- Es ist schwierig alle Arbeitseinheiten gleich auszulasten.

Fragen:

- Lassen sich alle Verarbeitungsschritte parallelisieren?
- Wann ist ein parallele Verarbeitung fertig?

- **parallel [griechisch]**
 - allgemein: nebeneinander verlaufend, in gleichem Abstand [Brockhaus]
 - Informatik: gleichzeitig ablaufend
- **Grad der Parallelisierung**
 - Anzahl der Prozessoreinheiten (Arbeitseinheiten) die beteiligt sind
- **Durchsatz**
 - Menge an Problemen die in einer Zeiteinheit gelöst werden können
- **Latenz**
 - Ausführungszeit der Berechnung eines speziellen Problems
- **Speedup**
 - Verhältnis zwischen Abarbeitung auf einer Einheit und mehreren Einheiten
 - Gegeben sei ein Problem, das von einer Einheit P nicht schneller als in der Zeit T gelöst werden kann.
 - n Einheiten vom Typ P kann dasselbe Problem nicht schneller als in der Zeit T/n
 - lösen, mit anderen Worten:
 - Bei n Prozessoren ist der Speedup höchstens n .

• Kosten einer parallelen Berechnung

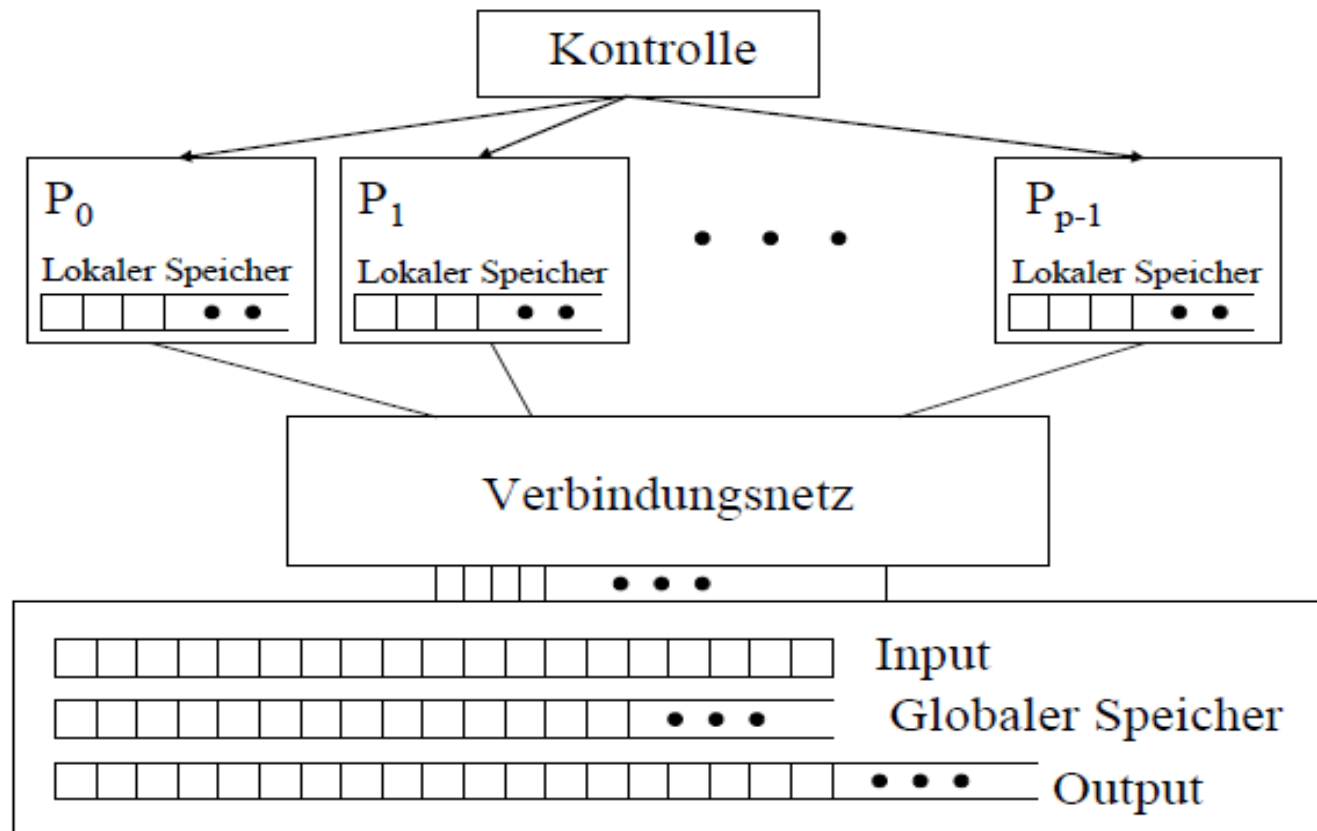
- Die Kosten einer parallelen Berechnung ist das Produkt aus Parallelitätsgrad und Latenz der Berechnung.
- Die Kosten eines parallelen Algorithmus für ein gegebenes Problem können nie kleiner sein als die des besten sequentiellen Algorithmus.
 $C_{par} \geq C_{seq}$

• Optimaler paralleler Algorithmus

- Ein Algorithmus mit dem Parallelitätsgrad p heißt optimal, wenn für seine Kosten gilt: $C_{par} \leq k \cdot C_{seq}$
- wobei k eine Konstante ist, die nicht von p oder von der Problemgröße n abhängt.

Wie kommt man auf einen möglichst optimalen Algorithmus?

PRAM: Parallel Random Access Machine



PRAM:

- Theoretisches Modell für einen Parallelrechner.
- Lokaler Speicher in den Prozessoren (distributed memory).
- Globaler Speicher (shared memory).
- Eingabe- und Ausgabeband im globalen Speicher.
- Erlaubt beliebige Parallelität.

Gutes Modell für die Ermittlung unterer Schranken für parallele Algorithmen.

Arten von PRAMs

- EREW: exclusive read exclusive write
- CREW: concurrent read exclusive write
- ERCW: exclusive read concurrent write
- CRCW: concurrent read concurrent write

Techniken bei der Vorgehensweise

Devide & Conquer: Die Lösung eines Problems wird die die Kombination der Lösungen der Teilprobleme gefunden.

Beispiel: Mergesort – Sortieren durch Mischen

Branch & Bound: Die Lösung eines Problems wird durch die Lösung irgendeines Teilproblems gelöst.

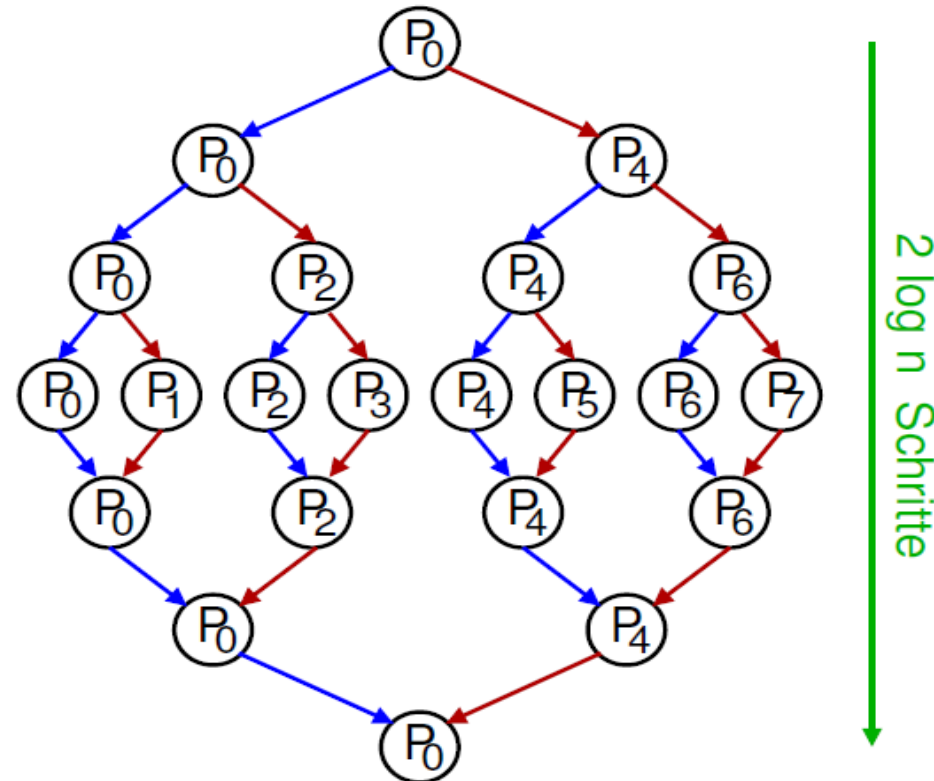
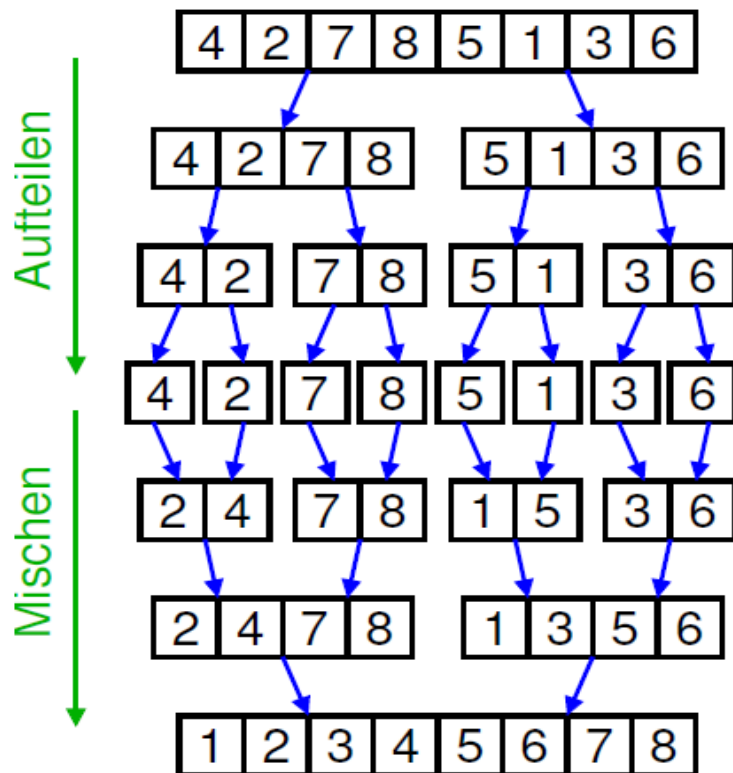
Beispiel: Traveling Salesman

Kürzester Rundreiseweg durch die 15 größten Städte Deutschlands. Insgesamt sind $14!/2 = 43.589.145.600$ verschiedene Wege möglich.



Mergesort

Beispiel: 8 Elemente, 8 Prozesse



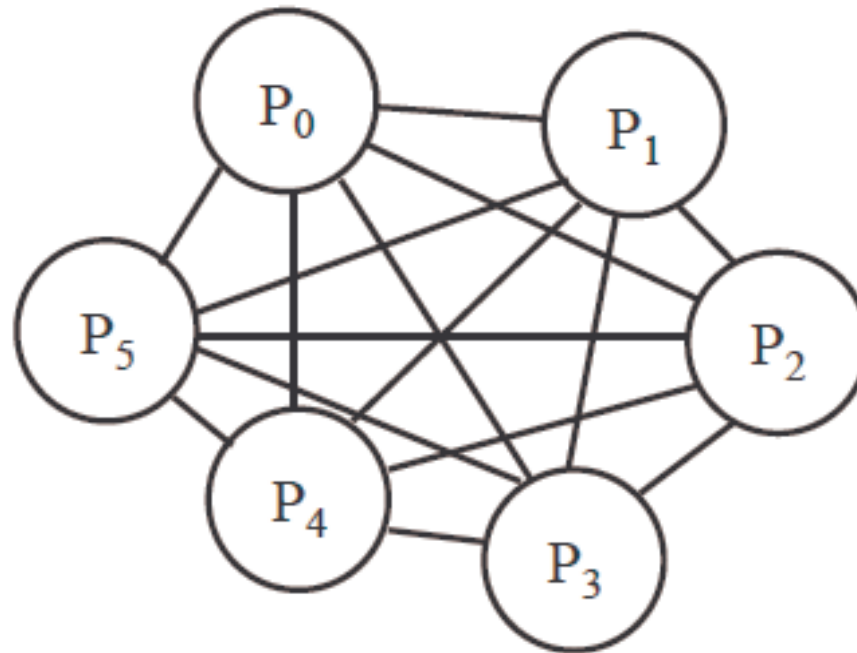
Was ist zu beachten?

- Verständnis für Parallelität spielt eine wichtige Rolle
 $[(a+b)+c]+d=(a+b)+(c+d)$
- Der Algorithmus muss zur Architektur passen.
- Kommunikationskosten müssen berücksichtigt werden.

Was ist zu beachten?

Es gibt zahlreiche Ansätze, der PRAM in einer realen Architektur möglichst nahezukommen.

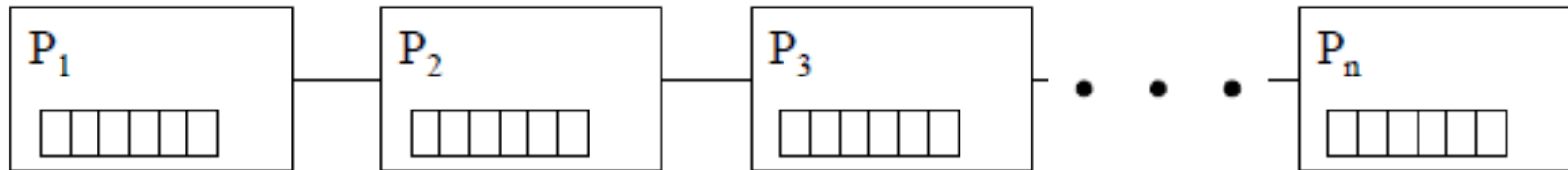
Das Problem ist der Zugriff auf den globalen Speicher, der ja nur konstante Zeit kosten darf. Ein naheliegender Ansatz ist das Verteilen des Speichers auf die physikalischen Speicher der Prozessoren und das Vernetzen der Prozessoren.



Prozessornetzwerke

- Array
- Ring
- Mesh (Gitter)
- Torus
- Hypercube
- CCC
- Shuffle Exchange
- Butterfly

Array als Verbindungsnetz

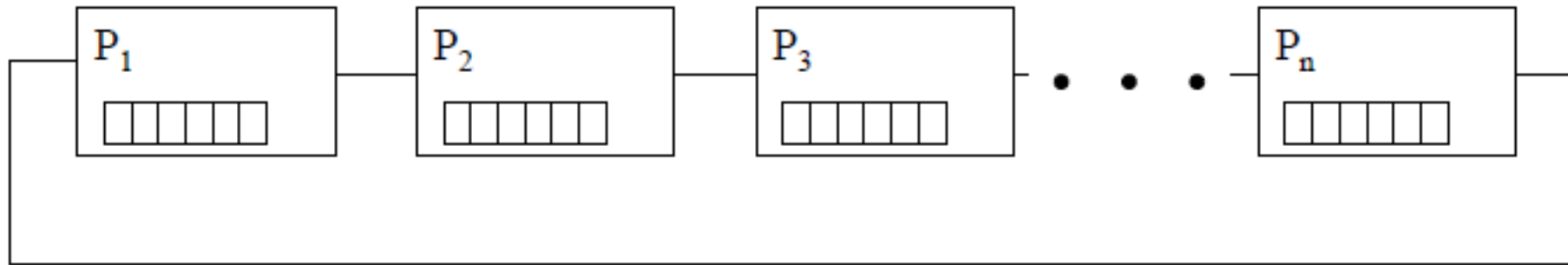


Durchmesser: $n - 1$

Schnittbreite: 1

Leitungslänge: $O(1)$

Ring als Verbindungsnetz

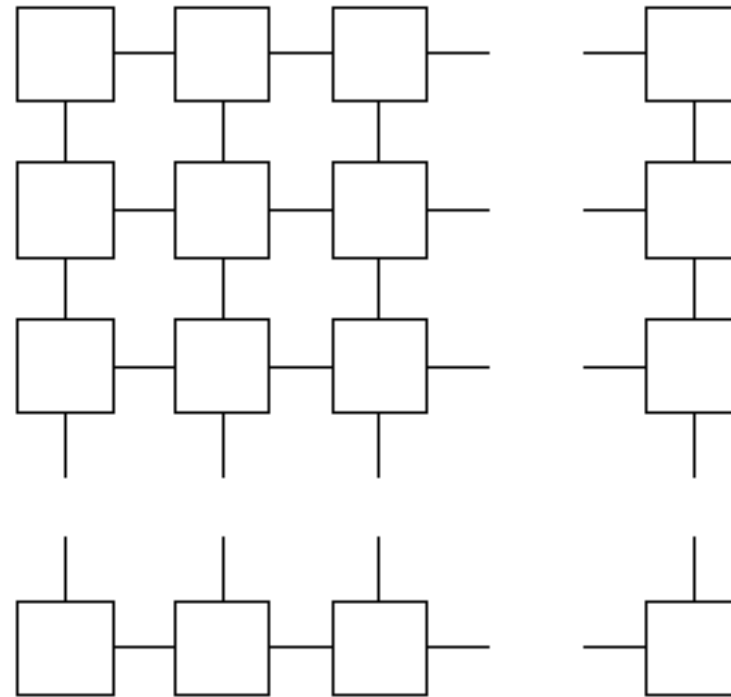


Durchmesser: $n/2$

Schnittbreite: 2

Leitungslänge: $O(1)$

Mesh als Verbindungsnetz

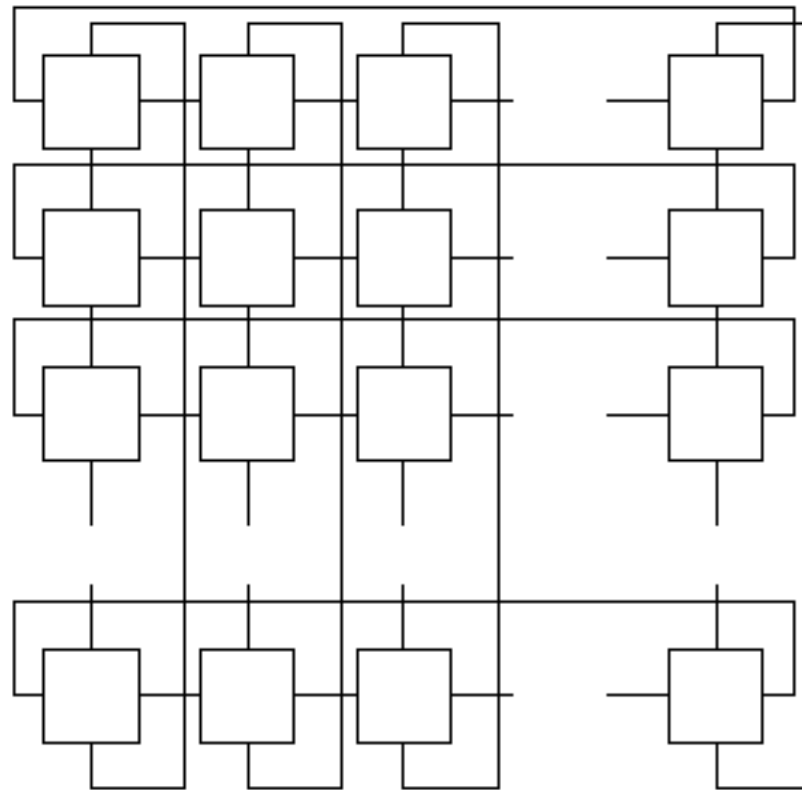


Durchmesser: $2n^{1/2} - 2$

Schnittbreite: $n^{1/2}$

Leitungslänge: $O(1)$

Torus als Verbindungsnetz

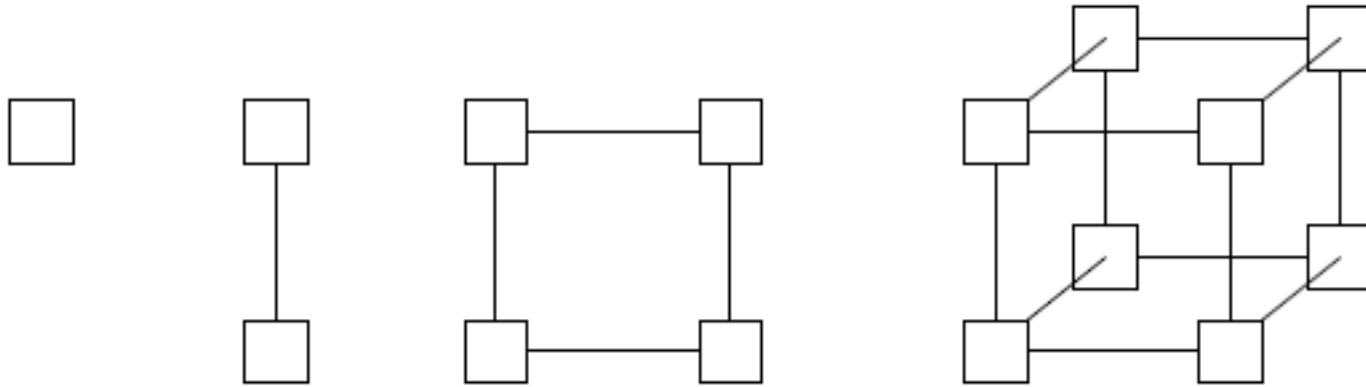


Durchmesser: $n^{1/2} - 1$

Schnittbreite: $2 n^{1/2}$

Leitungslänge: $O(1)$

Hypercube als Verbindungsnetz

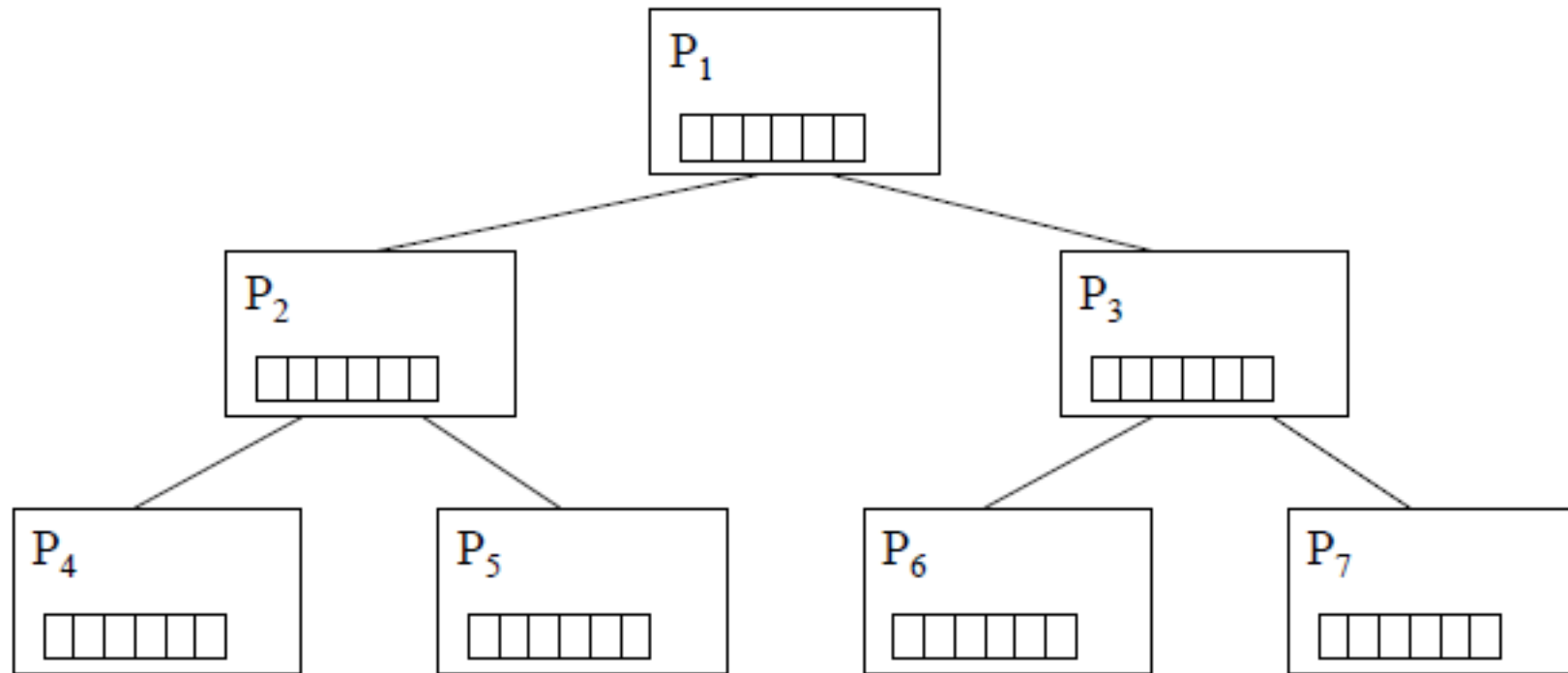


Durchmesser: $\log n$

Schnittbreite: $n/2$

Leitungslänge: nicht konstant

Baum als Verbindungsnetz



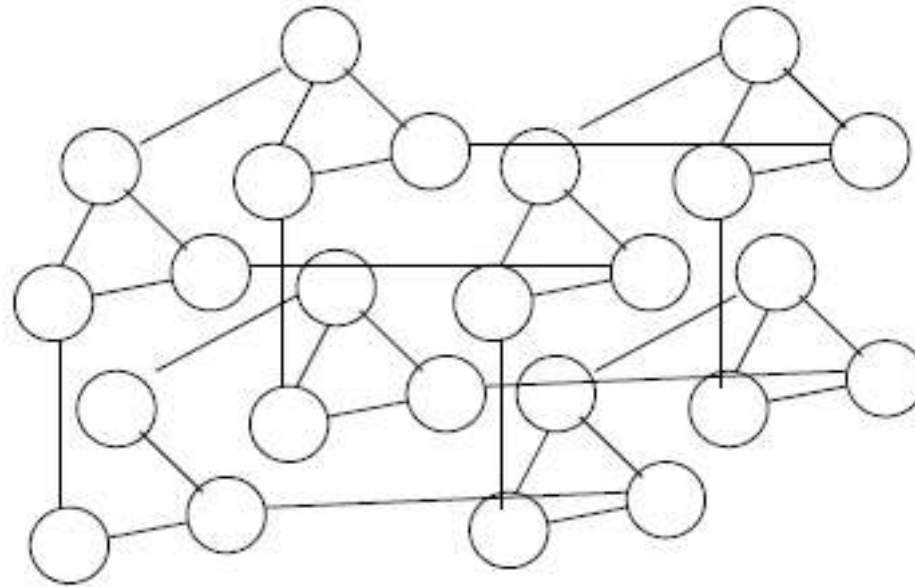
• • •

Durchmesser: $2 \log (n+1) - 2$

Schnittbreite: 1

Leitungslänge: $O(n^{1/2})$

Cube Connected Cycles als Verbindungsnetz



Durchmesser: $O(\log p)$

Schnittbreite: $O(p)$

Leitungslänge: nicht konstant

Übersicht über die gängigen Topologien

	Knoten	Durchmesser	Schnittbreite	Grad	konstante Kantenlänge
Array	p	$p-1$	1	2	Ja
Ring	p	$p/2$	2	2	Ja
Mesh	$p=k^2$	$2k-2$	k	4	Ja
Torus	$p=k^2$	$k-1$	$2k$	4	Ja
3-D-Mesh	$p=k^3$	$3k-3$	k^2	6	Ja
Baum	$p-1$	$2(\log p - 1)$	1	3	Nein
Hypercube	p	$\log p$	$p/2$	$\log p$	Nein
CCC	$p=2^{k*k}$	$2k$	2^{k-1}	3	Nein

Aktuelle Architekturen

- Multikern-Prozessoren
 - integrieren mehrere unabhängige CPUs auf einem Chip können z.B. gemeinsame Busse, Cache-Speicher haben
 - Standard
 - Multiprozessor-Rechner
 - Ein Rechner mit mehreren Prozessoren, oft auf gleicher Hauptplatine eingebaut. Prozessoren können andere Ressourcen
 - (z.B. Hauptspeicher, Ein- /Ausgabe) gemeinsam nutzen
 - Verteilte Systeme → Rechnerbündel (Cluster)
 - besteht aus einer Menge von vollständigen Rechnern
 - durch ein Kommunikationsnetzwerk verbunden
 - **Oracle Grid**
- => alles MIMD (MIMD: Multiple Instruction Stream, Multiple Data Stream)

Noch was ...



Danke für Ihre
Aufmerksamkeit



Guten Heimweg

Alles Gute bis zur
nächsten Konferenz 2013



Fragen?
Beate Künneke
Unternehmensberatung
bk@berlin.de

