

# Theorie und Einsatz von Verbindungseinrichtungen in parallelen Rechnersystemen

*Klassifizierung von Verbindungsnetzwerken*

14. April 2011

Andy Georgi

INF 1046  
Nöthnitzer Straße 46  
01187 Dresden

0351 - 463 38783

## Foliensatz

### Verfügbarkeit der Folien

Vorlesungswebseite:

[http://tu-dresden.de/die\\_tu\\_dresden/zentrale\\_einrichtungen/zih/lehre/ss2010/tevpr](http://tu-dresden.de/die_tu_dresden/zentrale_einrichtungen/zih/lehre/ss2010/tevpr)

## Agenda

- 1 Verbindungsart
  - Statische Verbindungsnetzwerke
  - Dynamische Verbindungsnetzwerke

## Statische Verbindungsnetzwerke

### Definition

Innerhalb eines statischen Verbindungsnetzwerks besitzt jedes Prozessorelement eine feste Anzahl von Verbindungsleitungen zu benachbarten Prozessorelementen, entlang derer Signale unmittelbar von Knoten zu Knoten gesendet werden können. Die Vermittlungsfunktion wird durch die Prozessorelemente selbst ausgeübt, indem die Signale auf den entsprechenden Ausgang gegeben werden. Somit beschränkt sich das Netz auf Verbindungsleitungen.

## Definition

Dynamische Verbindungsnetzwerke führen alle Vermittlungsaufgaben mittels aktiver Koppellemente eigenständig aus. Es ist daher häufig ausreichend wenn ein Prozessorelement lediglich über einen Ein- und einen Ausgang mit dem Netz verbunden ist. Der Aufbau eines dynamischen Verbindungsnetzwerks erfolgt mittels einer oder mehrerer Stufen von Koppellementen, mit deren Hilfe die Nachrichten durch das Netz geleitet werden.

- 2 Topologische Klassifikationen
  - Topologie
  - Durchmesser und mittlere Weglänge
  - Grad und Regularität
  - Bisektionsweite
  - Symmetrie
  - Skalierbarkeit
  - Permutationsmöglichkeiten
  - Partitionierbarkeit
  - Plättbarkeit
  - Einbettung

# Topologie

- Repräsentation von Netzen mit Hilfe von endlichen Graphen
  - Kanten = Verbindungsleitungen
  - Knoten = Prozessorelemente
  - Quelle = Sender
  - Senke = Empfänger
- Topologische Charakteristiken können zur Beurteilung der Leistungsfähigkeit herangezogen werden

# Durchmesser

## Definition

Besteht ein Verbindungsnetzwerk aus  $N$  Knoten, indiziert von 0 bis  $N - 1$  und sei  $d_{A,B}$  die minimale Anzahl von Schritten, die notwendig ist, um von Knoten  $A$  zu Knoten  $B$  zu gelangen, so ist der Durchmesser  $\Phi$  gegeben durch

$$\Phi = \max d_{A,B} \quad 0 \leq A < N, \quad 0 \leq B < N$$

- Durchmesser entspricht Abschätzung im ungünstigsten Fall
- Aussage über die *mittlere Weglänge*  $\bar{P}$  häufig sinnvoller:

$$\bar{P} = \frac{1}{N(N-1)} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} d_{j,k} ; \text{ für } j \neq k$$

### Grad

Der *Grad*  $\Gamma$  eines Knotens entspricht der Anzahl seiner ein- und ausgehenden Verbindungen. In einem Netz  $G$  ist  $\delta(G)$  die Mindest- und  $\Delta(G)$  die Maximalzahl von Verbindungsleitungen, welche mit einem Knoten des Netzes verbunden sind.

### Regularität

Ein Netz  $G$  wird als *n-regulär* bezeichnet, sobald gilt:

$$\Delta(G) = \delta(G) = n$$

Demzufolge besitzt in einem *n-regulären* Netz jeder Knoten den gleichen *Grad*  $\Gamma$  von  $n$ .

### Definition

Werden in einem Netz der Größe  $N$  zwei Mengen  $M_1$  und  $M_2$  die Knoten so zugeordnet, dass  $\lfloor N/2 \rfloor$  Knoten in  $M_1$  und  $\lceil N/2 \rceil$  Knoten in  $M_2$  enthalten sind, so wird die *Bisektionsweite*  $W_k(M_1, M_2)$  dieser Aufteilung über die minimale Anzahl der Verbindungsleitungen zwischen den Knoten in  $M_1$  und denen in  $M_2$  definiert. Die *Bisektionsweite*  $W$  des gesamten Netzes ergibt sich anschließend aus dem minimalen Wert von  $W_k$  für alle möglichen Verteilungen der Knoten auf die beiden Mengen  $M_1$  und  $M_2$ .

### Isomorphismus, Automorphismus

Seien  $G_1 = (V_1, E_1)$  und  $G_2 = (V_2, E_2)$  endliche ungerichtete Graphen, so heißen  $G_1$  und  $G_2$  *isomorph*, wenn eine bijektive Abbildung  $\varphi : V_1 \rightarrow V_2$  existiert mit  $\{u, v\} \in E_1 \Leftrightarrow \{\varphi(u), \varphi(v)\} \in E_2$ . In diesem Fall heißt  $\varphi$  *Isomorphismus* zwischen  $G_1$  und  $G_2$ . Gilt zusätzlich  $G_1 = G_2$ , so heißt  $\varphi$  *Automorphismus*.

## Knotensymmetrie

Sei  $G = (V, E)$  ein ungerichteter Graph, so heißt  $G$  knotensymmetrisch (engl.: vertex-transitive), wenn für alle  $u, v \in V$  ein Automorphismus  $\varphi_{u,v}$  existiert mit  $\varphi_{u,v}(u) = v$ .

## Kantensymmetrie

Sei  $G = (V, E)$  ein ungerichteter Graph, so heißt  $G$  kantensymmetrisch (engl.: edge-transitive), falls für je zwei Kanten  $\{u, v\}, \{u', v'\} \in E$  ein Automorphismus  $\varphi$  existiert mit  $\varphi(u) = u'$  und  $\varphi(v) = v'$ .

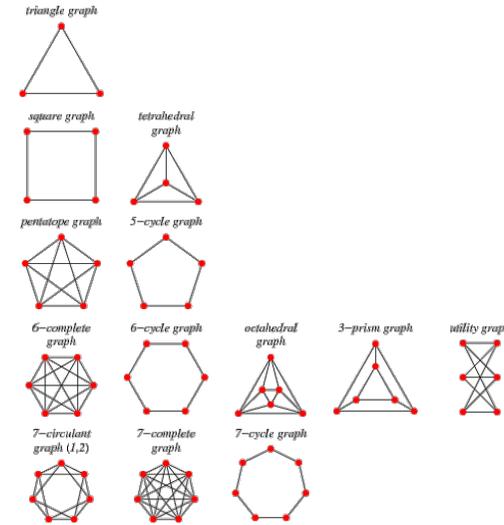


Abbildung: Beispiele für Knotensymmetrie [Vert]

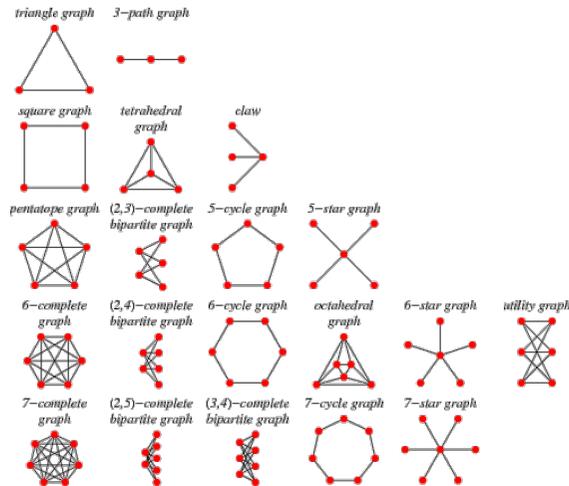


Abbildung: Beispiele für Kantensymmetrie [Edge]

- Erweiterbarkeit massiv paralleler Systeme von essentieller Bedeutung
- Dabei werden folgende Randbedingungen angestrebt:
  - Weitesgehende Beibehaltung der Topologie
  - Nutzung bestehender Verbindungsleitungen
  - Keine negative Beeinflussung der Leistungsfähigkeit
  - Eingeschränkte Zunahme der Netzkomplexität

## Permutationsmöglichkeiten I

### Permutation

Sei  $S \neq \emptyset$  eine beliebige endliche Menge und sei  $n$  die Anzahl der Elemente in  $S$ , so entspricht eine  $n$ -stellige *Permutation* einer bijektiven Abbildung  $\sigma : S_n \rightarrow S_n$ .

## Permutationsmöglichkeiten II

### Blockierendes Netz

Ein Netz wird als *blockierend* bezeichnet, wenn es eine Untermenge  $P_S$  aller möglichen Permutationen  $P_M$  unterstützt – Permutationen aus der Menge  $P_M - P_S$  hingegen zu Konflikten innerhalb des Netzes führen.

## Permutationsmöglichkeiten III

### Rearrangierbares Netz

Ein *rearrangierbares* Netz erlaubt alle Permutationen  $P_M$ , allerdings müssen bei Änderungen der aktuellen Permutationen unter Umständen bestehende Verbindungen rekonfiguriert werden.

## Permutationsmöglichkeiten IV

### Streng blockierungsfreies Netz

Streng blockierungsfreie Netze erlauben alle Permutationen  $P_M$ , ohne das Auftreten von Blockierungen oder die Rekonfiguration bereits bestehender Verbindungen.

### Definition

Ein Netz der Größe  $N$  heißt *partitionierbar*, wenn es in  $K$  sich einander nicht beeinflussende Subnetze der Größe  $N_i < N$ ,  $0 \leq i < K$  aufgeteilt werden kann, wobei die so entstandenen Subnetze alle Eigenschaften des Netzes besitzen, welches in der Größe  $N$  konstruiert wurde.

### Definition

Ein Graph  $G$  wird als *plättbar* oder *planar* bezeichnet, insofern es möglich ist, den Graphen  $G$  innerhalb einer Ebene so zu zeichnen, dass sich die Kanten von  $G$  nicht überschneiden. Somit dürfen sich die Kanten von  $G$  ausschließlich in den Knoten berühren.

### Definition

Für zwei endliche ungerichtete Graphen  $G = (V_G, E_G)$  und  $H = (V_H, E_H)$  heißt eine Abbildung der Knotenmengen,  $\varphi : V_G \rightarrow V_H$  *Einbettung* von  $G$  in  $H$ . Dabei wird  $G$  als *Gast-* und  $H$  als *Wirtsgraph* bezeichnet.

### Dilatation

Unter der *Dilatation* einer Grapheneinbettung wird die maximale Knotendistanz der Bilder von benachbarten Knoten aus  $G$  in  $H$  verstanden:

$$dilation(\varphi'(E_G)) = \max|\varphi'(u, u')| \text{ mit } (u, u') \in E_G$$

### Kongestion

Die *Kongestion* einer Grapheneinbettung ist die maximale Anzahl von Kanten, deren Wege hinsichtlich der Wegewahl  $\varphi'(E_G)$  eine einzelne Kante aus  $H$  benutzen:

$$\text{cong}(\varphi'(E_G)) = \max\{|(u, u') \in E_G | (v, v') \in \varphi'(u, u')\} \text{ mit } (v, v') \in E_H$$

### Reduktion

Die *Reduktion* einer Grapheneinbettung entspricht der maximalen Anzahl von Knoten des Gastgraphen, welche einem einzelnen Knoten des Wirtsgraphen zugeordnet werden:

$$\text{load}(\varphi) = \max\{|u \in V_G | \varphi(u) = v\} \text{ mit } v \in V_H$$

### Expansion

Die *Expansion* einer Grapheneinbettung ist das Verhältnis zwischen den Kardinalitäten der gesamten Knotenmenge des Wirtsgraphen und der durch die Einbettung bestimmten Bildmenge von Knoten des Gastgraphen:

$$\text{expansion}(\varphi) = \frac{|V_H|}{|\varphi(V_G)|}$$

1. Einbettungen
2. Schnittstellen
- 3. Schnittstellenstruktur**
  - Schnittstellen in statischen Netzen
  - Schnittstellen in dynamischen Netzen
  - Kommunikationsprozessoren

## Schnittstellen in statischen Netzen I

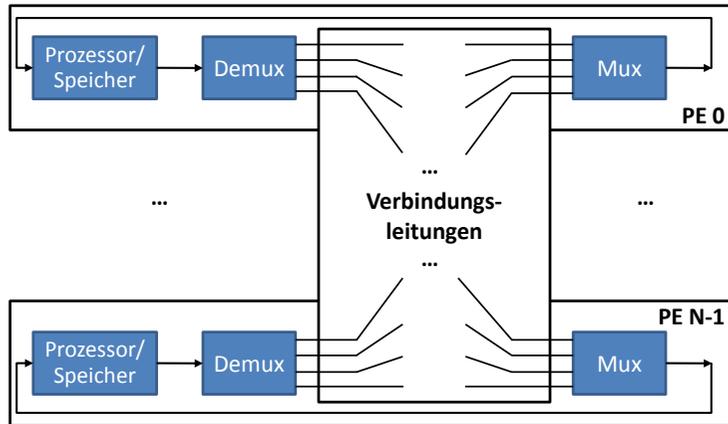


Abbildung: Statisches Netz mit Multiplexer-Demultiplexer Schnittstellen

## Schnittstellen in statischen Netzen II

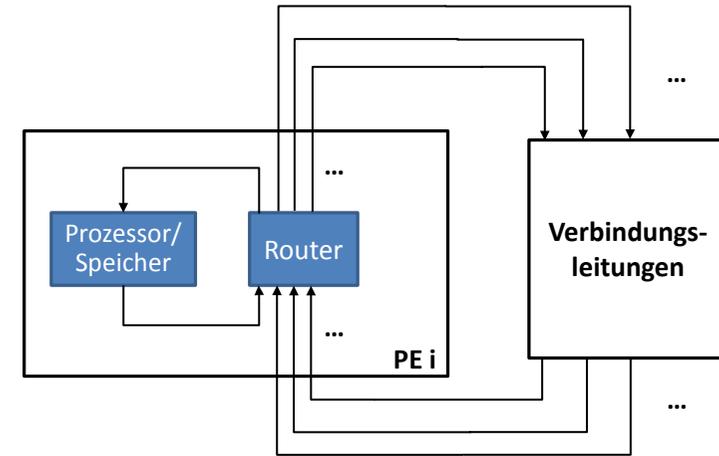


Abbildung: Statisches Netz mit integriertem Router

## Schnittstellen in dynamischen Netzen

- I.d.R. besitzt jedes PE einen bidirektionalen Zugang zum Netz
- Keine besonderen Hardwarekomponenten notwendig, da Vermittlungsfunktionen vom Netz ausgeführt werden
- Lediglich die für die Vermittlung notwendigen Informationen müssen vom PE berechnet und den Nutzdaten hinzugefügt werden

## Kommunikationsprozessoren

- Belastung des Prozessors durch Protokollverarbeitung
- Auslagerung dieser Aufgaben auf zusätzlich bereitgestellte anwendungsspezifische integrierte Schaltungen
- Komplexität von Kommunikationsprozessoren abhängig vom Umfang der zu übernehmenden Aufgaben

# Agenda

- ④ Operationsmodi
  - Synchron
  - Asynchron

# Synchron

- Verbindungsaufbau: Alle Prozesse eines Kommunikators fordern gleichzeitig eine Verbindung an
- Datenübertragung: Signale werden im festen Verhältnis zu einem zentralen Takt gesendet bzw. empfangen

# Asynchron

- Verbindungsaufbau: Anforderung von Verbindungen kann zu beliebigen Zeitpunkten erfolgen
- Datenübertragung: Nutzung von Handshake-Protokollen oder mitgeführten lokalen Takten

# Agenda

- ⑤ Kommunikationsflexibilität

## Unicast (1-to-1)

In einem *Unicast*-Netz sind ausschließlich Einzelverbindungen möglich, bei denen eine Quelle zeitgleich nur mit einer Senke kommuniziert.

## Multicast (1-to-N)

Ein Netz ist *Multicast*-fähig sobald eine Quelle zeitgleich mit einer Untermenge aller verfügbaren Senken kommunizieren kann.

## Broadcast (1-to-all)

Ein Netz mit *Broadcast*-Möglichkeiten unterstützt Kommunikation zwischen einer Quelle und allen verfügbaren Senken. Somit ist der *Broadcast* ein Spezialfall des *Multicast*.

# Zentral

- Steuerung der Netzwerkkomponenten über eine gemeinsam genutzte zentrale Instanz
- Pro: Effektive Nutzung aller verfügbaren Ressourcen möglich
- Contra: Hoher Aufwand bei Rekonfiguration

- 6 Kontrollstrategie
  - Zentral
  - Dezentral

# Dezentral

- Individuelle und selbständige Steuerung der Koppellemente durch die Netzwerkkomponenten selbst
- Pro: Vereinfachung des Rekonfigurationsprozesses
- Contra: Lokale Sicht auf das Netz

- 7 Vermittlungsverfahren
  - Schaltmethodik
  - Vermittlungsfunktion
  - Datenflusssteuerung

- Aufbau einer vollständigen, durchgängigen Verbindung zwischen Quelle und Senke
- Reihenfolgengetreue Übertragung des Bitdatenstroms
- Kommunikationsweg bleibt während der gesamten Informationsübertragung bestehen

- Verteilung einer Nachricht auf mehrere Dateneinheiten (Pakete)
- Vermittlung erfolgt aufgrund von Steuerinformationen innerhalb der Dateneinheiten
- Teilstrecken sind nur während der Übertragung einer Dateneinheit blockiert

- Vermittlungseinheiten entsprechen anwendungsorientierten Gesichtspunkten
- I.d.R. erfolgt dennoch eine Unterteilung in mehrere Dateneinheiten
- Freigabe einer Teilstrecke erfolgt nach Übertragung einer vollständigen Nachricht

## Verbindungsorientierte Kommunikation

- Etablierung einer Verbindung zwischen Quelle und Senke, welche für die gesamte Dauer der Datenübertragung genutzt wird
- Verbindung kann physikalisch auf eine feste Leitung abgebildet werden (Leitungsvermittlung) oder lediglich virtuell existieren (Paketvermittlung)

## Verbindungslose Kommunikation

- Ausschließlich in paketvermittelnden Systemen möglich
- Auswahl des Wegs erfolgt an allen Zwischenstationen für jedes Paket einzeln anhand definierter Metriken

## Store-and-Forward

- Weiterleitung einer Dateneinheit an einer Vermittlungsstation erfolgt erst nach vollständiger Zwischenspeicherung
- Speicherstrategien:
  - Input Buffering
  - Output Buffering
  - Common Buffering

## Wormhole-Routing

- Aufteilung des Headers in kleinere Einheiten (sog. *Flits*)
- Ein Flit enthält Schaltungsinformation für jeweils eine Vermittlungsstelle
- Mit Hilfe der Flits wird der Weg zwischen Quelle und Senke etabliert
- Die Freigabe erfolgt über eine Kennung am Ende der Dateneinheit
- Im Falle einer Blockierung wird das Senden unterbrochen bis die Blockierung aufgehoben ist

- Ablauf einer Kommunikation identisch zum Wormhole-Routing
- Im Falle einer Blockierung kann ein Paket allerdings zwischengespeichert werden

- 8 Konflikte
  - Konflikttypen
  - Konfliktbehandlung

## Konflikttypen

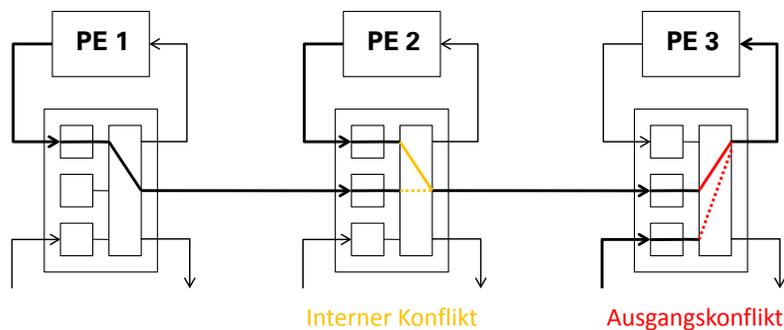


Abbildung: Mögliche Netzkonflikte

## Konfliktbehandlung bei Leitungsvermittlung

- Block:
  - Verbindungsaufbau schreitet bis zur Blockierung fort
  - Aufgebaute Verbindungen bleiben bestehen, bis zur Auflösung des Konflikts
- Drop:
  - Abbruch des Verbindungsaufbaus im Falle einer Blockierung
- Modified Drop:
  - Verbindungsaufbau wird nach einer definierten Wartezeit abgebrochen, wenn eine Blockierung auftritt

- Block:
  - Zwischenspeicherung der eintreffenden Pakete im Falle einer Blockierung
  - Signalisierung an vorhergehende Stufen, bei Erschöpfung der verfügbaren Ressourcen
- Drop:
  - Ablauf identisch zum Block-Verfahren, allerdings erfolgt keine Signalisierung bei einem Pufferüberlauf
  - Paketverluste sind möglich
- Modified Drop:
  - Signalisierung der vorhergehenden Stufe erfolgt bereits bei einer definierten Auslastung des verfügbaren Speichers
  - Kommt es dennoch zu einem Pufferüberlauf, werden die Pakete verworfen

- 9 Fehlerbeherrschung
  - Fehlersichere Systeme
  - Fehlertolerante Systeme

## Fehlersichere Systeme

### Definition

In einem *fehlersicheren System* wird ein auftretender Fehler erkannt und das System in einen sicheren Zustand überführt. Dabei ist keine Rekonfiguration der Hard- oder Software notwendig.

## Fehlertolerante Systeme

### Fehlerkompensierendes System

In einem *fehlerkompensierendem System* ist das Auftreten eines Fehlers transparent für den Anwender. Das System bleibt im Fehlerfall unverändert, allerdings werden die Auswirkungen des Fehlers unterdrückt, so dass das Ergebnis dennoch korrekt ist.

### Fehlerbehebendes System

Bei einer *Fehlerbehebung* wird das System im Fehlerfall durch Rekonfiguration der Hard- oder Software in einen fehlerfreien Zustand überführt. Diese Umstrukturierung ist i.d.R. mit Einschränkungen der Leistungsfähigkeit verbunden und somit für den Nutzer nicht mehr transparent.

## 10 Literaturverzeichnis

-  [Gil91] Giloi, Wolfgang K.  
Rechnerarchitektur, Springer-Verlag 1991.  
ISBN: 3-540-56355-5
-  [Cul99] D. E. Culler, J. P. Singh  
Parallel Computer Architecture, Morgan Kaufmann Publishers 1999.  
ISBN: 1-55860-343-3
-  [SchJ96] Th. Schwederski, M. Jurczyk  
Verbindungsnetze, B. G. Teubner Stuttgart 1996.  
ISBN: 3-519-02134-X
-  [TanGoo99] A. S. Tanenbaum, J. Goodman  
Computerarchitektur, Prentice Hall 1999.  
ISBN: 3-8272-9573-4

-  [Vert] Weisstein, Eric W.  
Vertex-Transitive Graph, 2010.  
<http://mathworld.wolfram.com/Vertex-TransitiveGraph.html>
-  [Edge] Weisstein, Eric W.  
Edge-Transitive Graph, 2010.  
<http://mathworld.wolfram.com/Edge-TransitiveGraph.html>