

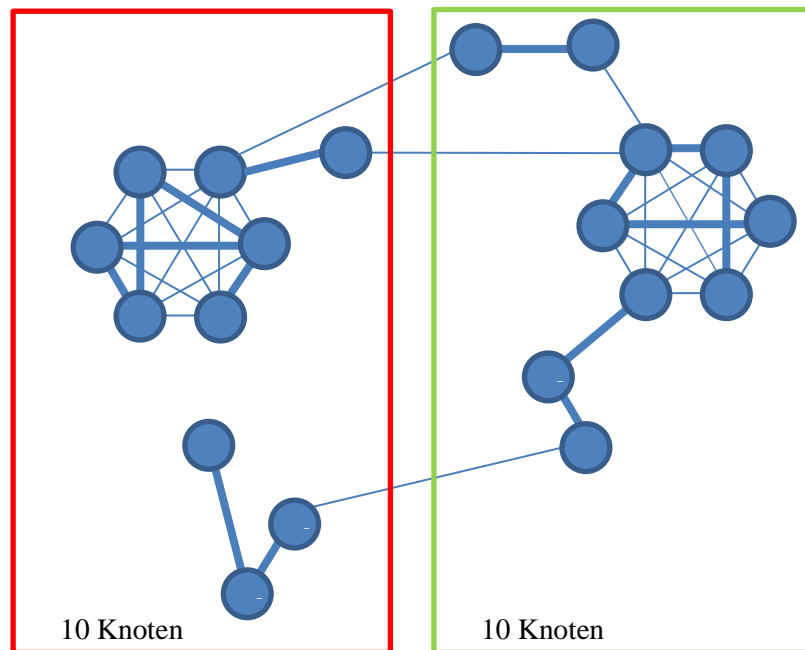
SPP Übung 2

Aufgabe 1

a)

Diameter: Die Größte minimale Distanz liegt zwischen Knoten 21 und den Knoten 0, 1, 2, 3, und 4 vor und beträgt 9

Bisektionsbandbreite: Durch Teilen des Netzwerkes in die beiden abgebildeten Teilnetzwerke Erhält man eine Bisektionsbandbreite von 300 Mb/s.



Netzwerkgrad: Der Netzwerkgrad und somit der Maximale Knotengrad beträgt 7 und wird von den Knoten 5 und 14 Verursacht.

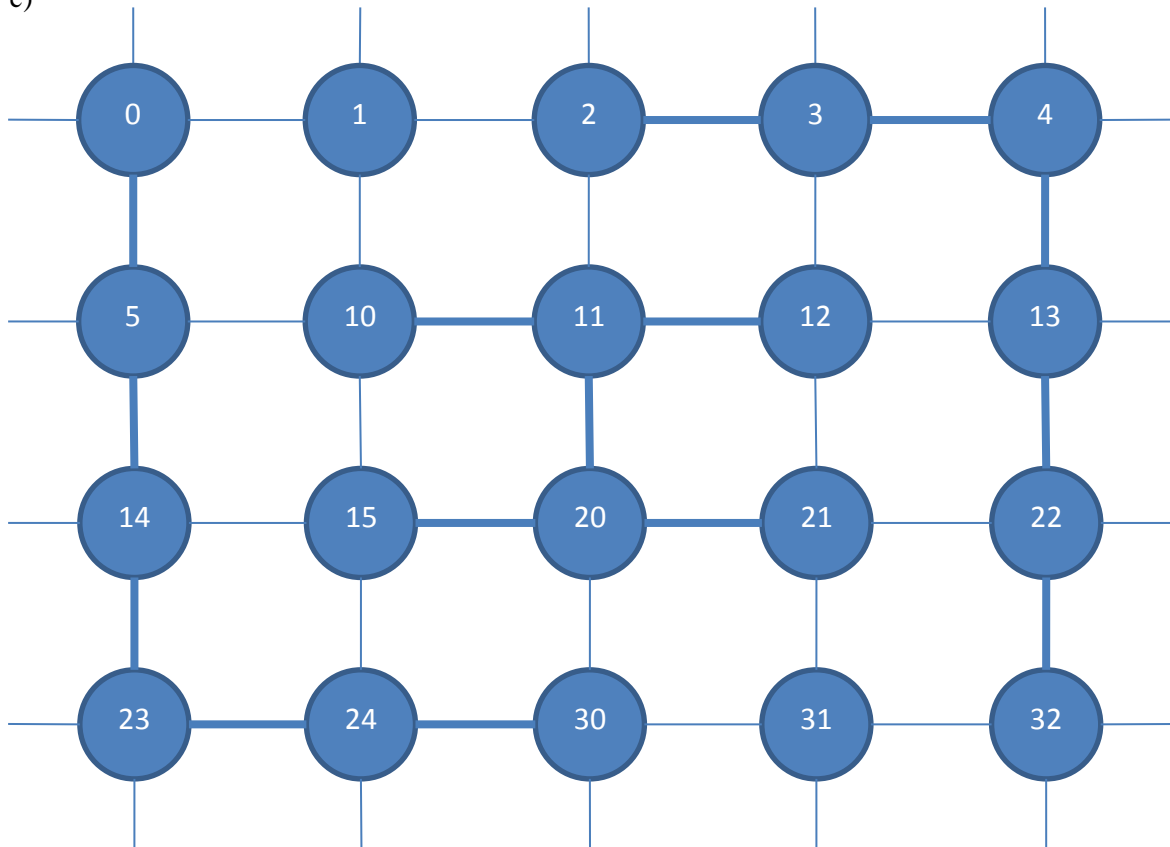
Node connectivity: Die Node connectivity beträgt 1 da das Netzwerk durch entfernen einer der Knoten 5, 14, 20, 22, 23, 24 oder 11 nicht mehr verbunden wäre.

b)

Der Diameter von 9 führt zu einer hohen Latenz Zwischen dem Knoten 21 und den Knoten 0, 1, 2, 3, und 4, da die Zeit bis das erste Bit den Zielknoten erreicht hoch ist.

Eine andere Schwachstelle des Netzwerkes ist die geringe node connectivity von 1. Diese führt dazu das z.B. durch den Ausfall des Knotens 20 auch die Knoten 22, 23 und 24 ausfallen. Das Netzwerk ist also nicht robust gegen Ausfälle.

c)



Diameter: 4

Bisektionsbandbreite: 2100 Mb/s

Netzwerkgrad: 4

Node connectivity: 4

Aufgabe 2

a)

P1 : W(P2;b;3)W(P1;a;2)W(P2;c;1)W(P1;b;2)W(P2;a;1)W(P1;c;3)W(P1;a;3)W(P2;b;1)

P2 : W(P2;b;3)W(P1;a;2)W(P1;c;3)W(P1;b;2)W(P2;a;1)W(P2;c;1)W(P1;a;3)W(P2;b;1)

Bei sequentiell konsistentem Speicher nicht möglich

Bei kohärentem Speicher nicht möglich

b)

P1 : W(P1;a;2)W(P1;b;2)W(P2;b;3)W(P1;a;3)W(P2;c;1)W(P2;a;1)W(P1;c;3)W(P2;b;1)

P2 : W(P1;b;2)W(P2;b;3)W(P2;c;1)W(P1;a;2)W(P1;a;3)W(P2;a;1)W(P2;b;1)W(P1;c;3)

Bei sequentiell konsistentem Speicher nicht möglich

Bei kohärentem Speicher möglich

c)

P1 : W(P1;a;2)W(P2;a;1)W(P1;a;3)W(P1;b;2)W(P2;b;3)W(P2;b;1)W(P1;c;3)W(P2;c;1)

P2 : W(P1;c;3)W(P2;c;1)W(P1;b;2)W(P2;b;3)W(P2;b;1)W(P1;a;2)W(P2;a;1)W(P1;a;3)

Bei sequentiell konsistentem Speicher nicht möglich

Bei kohärentem Speicher möglich

d)

P1 : W(P2;b;3)W(P1;a;2)W(P1;b;2)W(P1;a;3)W(P1;c;3)W(P2;a;1)W(P2;c;1)W(P2;b;1)

P2 : W(P2;b;3)W(P1;a;2)W(P1;b;2)W(P1;a;3)W(P1;c;3)W(P2;a;1)W(P2;c;1)W(P2;b;1)

Bei sequentiell konsistentem Speicher möglich

Bei kohärentem Speicher möglich

e)

P1 : W(P1;a;2)W(P1;b;2)W(P2;b;3)W(P1;a;3)W(P2;c;1)W(P2;a;1)W(P1;c;3)W(P2;b;1)

P2 : W(P1;a;2)W(P1;b;2)W(P2;b;3)W(P1;a;3)W(P2;c;1)W(P2;a;1)W(P1;c;3)W(P2;b;1)

Bei sequentiell konsistentem Speicher möglich

Bei kohärentem Speicher möglich

Begründungen:

Bei sequentiell konsistentem Speicher müssen alle Prozesse die Operationen in der gleichen Reihenfolge sehen. Das ist nur bei d) und e) der Fall.

Bei kohärentem Speicher darf die Reihenfolge von Operationen auf unterschiedliche Speicherstellen von Prozess zu Prozess variieren. Schreiboperationen auf dieselbe Speicherstelle müssen jedoch von allen Prozessen in derselben Reihenfolge gesehen werden.

Das ist bei allen angegebenen Reihenfolgen der Fall außer bei a), dort sieht P1 W(P2;c;1) und W(P2;c;3) P2 sieht jedoch W(P2;c;3) gefolgt von W(P2;c;1).

Aufgabe 3

a) Ein Thread besteht aus den folgenden Komponenten:

- Thread-ID
- program Counter
- register Set
- stack

b) Thread kommunizieren via geteilter Variablen/geteiltem Speicher, während hingegen Prozesse eine explizite Kommunikationsstruktur benötigen.

c) Parallelisierung mittels Threads bietet sich an, wenn dieselben Variablen von allen Threads genutzt werden müssen, da diese auf dieselben Speicheradressen zugreifen und somit immer aktuelle Werte abrufen können. Bei der Parallelisierung mittels Prozessen bekommt hingegen jeder Prozess eine eigene Kopie der ursprünglichen Daten, was sich besonders anbietet, wenn für jeden Prozess unabhängige Daten vorliegen sollen.

