

Aufgabenblatt 1

(zum 31. Oktober 2012)

Björn Stabel 222128

Tim Strehlow 316594

Friedrich Maiwald 350570

Aufgabe 1.1: Topologien

a) *Wie viele verschiedene kürzeste Wege der Länge k gibt es auf einem Hypercube der Dimension d ?*

Es muss mindestens $d \cdot 2^{d-1}$ verschiedene kürzeste Wege geben, da das die Anzahl der Kanten (bei Weglänge $k = 1$) in einem Hypercube in Abhängigkeit von der Dimension ist.

Wie viele Wegalternativen für kürzeste Wege der Länge k gibt es auf einem Hypercube der Dimension d ?

Es gibt $k!$ verschiedene Möglichkeiten für einen kürzesten Weg der Länge k in einem Hypercube. Da es sich um einen kürzesten Weg handelt, wird jede Dimension nur höchstens einmal während dieses Weges durchschritten. Wir gehen dabei davon aus, dass $k \leq d$ ist.

b) *Der in der Vorlesung gezeigte Broadcast für Hypercubes erzeugt einen Spannbaum. Wie viele verschiedene Spannbäume sind möglich, wenn man die Reihenfolge der d Dimensionen variiert?*

Bei einem Hypercube der Dimension d kann man durch Variation der Reihenfolge der Dimensionen beim Broadcast insgesamt $d!$ verschiedene Spannbäume erzeugen. Im ersten Schritt kann man in d Richtungen gehen. In jedem folgenden Schritt ist jeweils eine Dimension weniger übrig. Also gibt es zusammengerechnet $d!$ verschiedene mögliche Spannbäume.

Wie können verschiedene Broadcasts parallel ablaufen, ohne dass es dabei zu einer Überlagerung kommt?

Es können insgesamt d Broadcasts parallel ablaufen. Wir benutzen dabei das in der Vorlesung gezeigte Verfahren mit dem Broadcast auf einem Spannbaum. Für jeden der parallel laufenden Broadcasts wechselt die Reihenfolge der Dimensionen. Ein Beispiel für $d = 3$:

Broadcast 1: Dim 1 \rightarrow Dim 2 \rightarrow Dim 3

Broadcast 2: Dim 2 \rightarrow Dim 3 \rightarrow Dim 1

Broadcast 3: Dim 3 \rightarrow Dim 1 \rightarrow Dim 2

Wenn alle drei Broadcasts gleichzeitig starten und in einem gleichen „Takt“ gesendet werden, sind pro „Takt“ immer nur höchstens eine Nachricht auf einer Kante unterwegs und es kommt nicht zu einer Überlagerung. Dabei muss es egal sein, in welcher Reihenfolge die verschiedenen Broadcast-Nachrichten auf den Knoten ankommen.

Aufgabe 1.2: Fluten mit Bestätigung

Implementieren Sie mit Hilfe des Simulationsframeworks teachnet (zu finden in ISIS) den Flutungsalgorithmus mit Bestätigung. Testen Sie den Algorithmus auf einem Ring und vergleichen Sie die Nachrichtenanzahl mit dem "Broadcast auf unidirektionalen Ringen" auf derselben Topologie.

Auf einem Ring mit der Kantenanzahl e benötigt der Flutungsalgorithmus genau $2e$ Nachrichten, nämlich einen Explorer und eine Quittung pro Kante. Beim Broadcast auf unidirektionalen Ringen werden nur e Nachrichten benötigt, weil die Information einmal um den Ring herum geht (inklusive „Bestätigung“ durch Vollendung des Zyklus).

Aufgabe 1.3: Echo Algorithmus

Implementieren Sie mit Hilfe von teachnet den Echo-Algorithmus und überprüfen Sie die Richtigkeit der Nachrichtenanzahl ($2e$).

Die Implementierung ähnelt dem Flutungsalgorithmus mit Bestätigung, allerdings wird bei zwei Explorern auf einem Kanal auf die Quittung verzichtet, da beide Knoten augenscheinlich schon informiert sind. Dadurch laufen immer nur zwei Nachrichten auf einem Kanal (entweder zwei Explorer oder ein Explorer und dessen Quittung) und unser Algorithmus hat insgesamt die Nachrichtenanzahl $2e$.