

# Grundlagen Rechnernetze und Verteilte Systeme

IN0010, SoSe 2019

## Übungsblatt 9

1. Juli – 5. Juli 2019

**Hinweis:** Mit \* gekennzeichnete Teilaufgaben sind ohne Lösung vorhergehender Teilaufgaben lösbar.

In dieser Woche wird in den Übungen auch Aufgabe 3 der Midterm besprochen.

### Aufgabe 1 Distanz-Vektor-Routing

Gegeben sei die in Abbildung 1 dargestellte Topologie mit den vier Routern A bis D. Die Linkkosten sind jeweils an den Kanten angegeben. Wir notieren die Routingtabellen in Kurzform als Vektor  $[(x_A, y_A), \dots, (x_D, y_D)]$ . Die Tupel  $(x, y)$  geben dabei die Kosten sowie den Next-Hop zum Ziel an.

Zu Beginn seien die Routingtabellen noch leer, d. h. die Router kennen noch nicht einmal ihre direkten Nachbarn. Dies wird durch die Schreibweise  $(/, /)$  angedeutet. Sich selbst erreichen die Router natürlich mit Kosten 0.

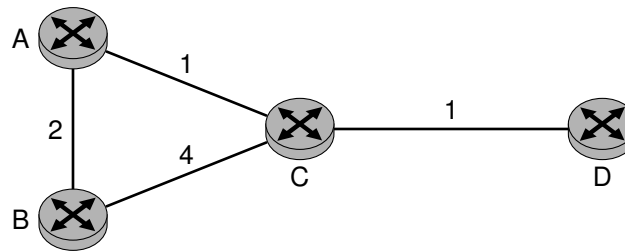


Abbildung 1: Netztopologie (Aufgabe 3)

Die Router beginnen nun damit, in periodischen Zeitabständen ihre Distanz-Vektoren mit ihren direkten Nachbarn auszutauschen. Dabei schickt beispielsweise Router B ein Update an Router C, welches lediglich die Distanz zum jeweiligen Ziel enthält (nicht aber den Next-Hop). Wenn nun Router A ein solches Update von B erhält und darin eine Route zu D finden würde, so wüsste A, dass er D über B erreicht. Die Kosten zu D entsprechen dann den Kosten zu B zuzüglich der Kosten, mit denen B das Ziel erreichen kann.

Im Folgenden wollen wir dieses Verhalten untersuchen. Da das Ergebnis allerdings davon abhängt, in welcher Reihenfolge Updates ausgetauscht werden, treffen wir die idealisierte Annahme, dass alle Router exakt zeitgleich ihre Updates verschicken.

a)\* Geben Sie gemäß obiger Tabelle die Routingtabellen aller vier Router in den folgenden Schritten an. Brechen Sie ab, sobald ein konvergenter Zustand erreicht ist.

| Schritt | Router A                           | Router B                           | Router C   | Router D                           |
|---------|------------------------------------|------------------------------------|--|------------------------------------|
| 0       | $[(0, A), (/, /), (/, /), (/, /)]$ | $[(/, /), (0, B), (/, /), (/, /)]$ | $[(/, /), (/, /), (0, C), (/, /)]$                       | $[(/, /), (/, /), (/, /), (0, D)]$ |
| 1       | $(0, A), (2, B), (1, C), (/, /)$   | $(2, A), (0, B), (4, C), (/, /)$   | $(1, A), (4, B), (0, C), (1, D)$                         | $(1, /), (1, /), (1, /), (0, D)$   |
| 2       | $(0, A), (2, B), (1, C), (2, D)$   | $(2, A), (0, B), (3, A), (5, C)$   | $(1, A), (3, B), (0, C), (1, D), (2, A), (5, C), (0, D)$ | $(1, /), (1, /), (1, /), (0, D)$   |
| 3       | $(0, A), (2, B), (1, C), (2, D)$   | $(2, A), (0, B), (3, A), (4, C)$   | $(1, A), (3, B), (0, C), (1, D), (2, A), (4, C), (0, D)$ | $(1, /), (1, /), (1, /), (0, D)$   |
| 4       |                                    |                                    |  |                                    |
| 5       |                                    |                                    |  |                                    |

b) Welcher (Graph-)Algorithmus findet hier Verwendung?

Belman ford  
Dezentralisiert.

Nun fällt die Verbindung zwischen den Knoten C und D aus. Die Knoten C und D bemerken dies und setzen die entsprechenden Pfadkosten auf unendlich.

c) Was passiert in den folgenden Schritten, in denen die aktiven Knoten weiter ihre Distanzvektoren austauschen? Geben Sie nach jedem Schritt die Distanztabellen an, bis das weitere Ergebnis klar ist.

| Schritt | Router A                | Router B                | Router C                | Router D                |
|---------|-------------------------|-------------------------|-------------------------|-------------------------|
| 0       | (0,A) (2,B) (2,C) (2,D) | (2,A) (0,B) (3,A) (4,A) | (2,A) (3,A) (0,C) (4,C) | (4,A) (4,B) (4,C) (0,D) |
| 1       | (0,A) (2,B) (2,C) (4,B) | (2,A) (0,B) (3,A) (4,A) | (3,A) (3,A) (0,C) (4,C) | (4,A) (4,B) (4,C) (0,D) |
| 2       |                         | (4,B) (7,D)             | (3,A) (7,A)             |                         |
| 3       |                         |                         |                         |                         |
| 5       |                         |                         |                         |                         |
| 6       |                         |                         |                         |                         |
| 7       |                         |                         |                         |                         |
| ⋮       |                         |                         |                         |                         |

*keine gleichen Routen werden versendet, stattd. sich was ändert ändern alle sich*

d)\* In der Vorlesung wurden **Split Horizon**, **Triggered Updates** und **Path Vector** als mögliche Gegenmaßnahmen für das Count-to-Infinity-Problem genannt. Erläutern Sie in der Gruppe die Funktionsweise dieser Verfahren.

*ganzer Pfad wird versendet*

## Aufgabe 2 Subnetting (Hausaufgabe)

Der TUMexam AG werden die Adressbereiche 131.159.32.0/22 und 131.159.36.0/24 zugewiesen. Für die Aufteilung dieses Adressbereichs ist die TUMexam AG selbst verantwortlich. Nach einer sorgfältigen Bedarfsanalyse ergeben sich die folgenden Anforderungen an die Subnetze und die Mindestanzahl **nutzbarer** IP-Adressen:

| Subnetz | NET 1 | NET 2 | NET 3 | NET 4 | NET 5 |
|---------|-------|-------|-------|-------|-------|
| IPs     | 300   | 300   | 15    | 40    | 4     |

*32 64 8*

Bei der Erhebung dieser Zahlen wurde die an das jeweilige Router-Interface zu vergebende IP-Adresse bereits berücksichtigt.

a)\* Geben Sie jeweils die erste und letzte IP-Adresse der beiden vergebenen Adressbereiche an.

b) Wie viele IP-Adressen stehen der TUMexam AG insgesamt zur Verfügung? Können alle davon zur Adressierung von Hosts verwendet werden?

c)\* Ist es möglich, den von den beiden Adressblöcken gebildeten Adressbereich in einem einzigen Subnetz zusammenzufassen?

d)\* Teilen Sie nun die beiden Adressbereiche gemäß der Bedarfsanalyse auf, so dass Subnetze der passenden Größe entstehen. Gehen Sie mit den Adressen so sparsam wie möglich um. Es soll am Ende ein möglichst großer zusammenhängender Adressbereich für zukünftige Nutzung frei bleiben. Für jedes Subnetz ist anzugeben:

- die Größe des Subnetzes
- die Anzahl nutzbarer Adressen
- das Subnetz in Präfixschreibweise
- die Subnetzmaske in Dotted-Decimal-Notation
- die Netz- und Broadcastadresse

2d) 137.159.32.0/22 <sup>32 Bits</sup>  
 erste: 137.159.32.0  
 letzte: 137.159.35.255

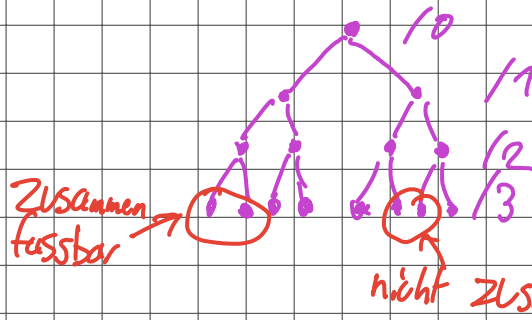
131.159.36.0/24  
 erste: 131.159.36.0  
 letzte: 131.159.36.255

137.159.32.0/22

000100100: 00000000  
 11: 11111111  
 35.255

b)  $2^{10} + 2^8 - 4 = 1024 + 256 - 4 = 1276$

c) Nein, weil sie nicht gleich groß sind



Man kann Netze nur zusammenfassen wenn sie gleich groß sind und benachbart sind.

d)  
 Größe

Net 1  
 512

Anzahl Adressen  
 Präfix

510

Schreibweise

137.159.32.0/23

Netz-adresse

137.159.32.0

Broadcast-adresse

137.159.33.255