# Übungsblatt 11

1. Vermittlungsschicht und Routing: Theorie

Welche Funktionalitäten stellt die Netzwerkschicht der Transportschicht zur Verfügung?

Transport segment from sending to receiving host

On sending side encapsulates segments into datagrams

On receiving side, delivers segments to transport layer

Network layer protocols in every host, router, interm. node

Router examines header fields in all IP datagrams passing through it

Network Layer Functions

Forwarding: move packets from router’s input to appropriate router output

Routing: determine route taken by packets from source to destination

Forwarding (Richtiges Abbiegen an einer Weggabelung): Weiterleitung von Paketen von Eingangs-Link zu korrektem Ausgangs-Link

Routing (Planen einer Wanderung): Bestimmen des Weges auf dem Pakete von Quelle zum Ziel weitergeleitet werden (Routing-Algorithmen)

Ist es möglich, dass Pakete (aus der Sicht der Netzwerkschicht) unterschiedliche Routen zwischen gleichen Start- und Endknoten nehmen können? Begründen Sie Ihre Antwort.

Ja, Routing protocol goal: determine “good” paths (equivalently, routes), from sending hosts to receiving host, through network of routers kein Verbindungsaufbau zu Zielhost, kein fixer Pfad

Haben Router eigentlich IP-Adressen? Wenn ja, wie viele? Begründen Sie Ihre Antwort. Erklären Sie den grundsätzlichen Aufbau einer IP-Adresse.

IP address: 32-bit identifier for host, router interface

• Interface: connection between host/router and physical link

– Router’s typically have multiple interfaces

– Host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)

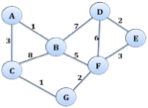
• IP addresses associated with each interface

Router haben IP-Adressen. Sie können auch mehrere haben. Die Anzahl hängt von der Anzahl der Netzwerkkarten im Router ab, d.h. zu wie vielen unterschiedlichen IP-Netzen ein Router gehört.

IPv4 benutzt 32-[Bit](https://de.wikipedia.org/wiki/Bit)-Adressen, daher können in einem Netz maximal 4.294.967.296 Adressen vergeben werden. IPv4-Adressen werden üblicherweise dezimal in vier Blöcken geschrieben, zum Beispiel 207.142.131.235. Zwei und dreistellige Zahlen dürfen hierbei nicht mit der Ziffer 0 beginnen (Ist nach RFC nicht erlaubt und wird häufig als Oktalzahl interpretiert). Jeder Block repräsentiert 8 Bit; somit ergibt sich für jeden Block ein Wertebereich von 0 bis 255. Bei der Weiterentwicklung [IPv6](https://de.wikipedia.org/wiki/IPv6) werden 128-Bit-Adressen verwendet.

Eine IP-Adresse besteht aus einem Netzanteil und einem Hostanteil. Der Netzanteil identifiziert ein Teilnetz, der Hostanteil identifiziert ein Gerät (Host) innerhalb eines Teilnetzes.

1. **Vermittlungsschicht und Routing: Link-State-Algorithmus**

Gegeben Sei ein Netzwerk wie in Abbildung 1 dargestellt. Berechnen Sie, unter Zuhilfenahme des Algorithmus von Dijkstra, den kürzesten Pfad ausgehend von Knoten A zu allen Netzwerkknoten. Benutzen Sie dazu eine ähnliche Tabelle wie Sie sie in der Vorlesung kennengelernt haben. Die Zahlen geben die Kosten für den jeweiligen Link an. 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Step | N’ | D(B) | D(C) | D(D) | D(E) | D(F) | D(G) |
| 0 | A | 1,A | 3,A | ∞ | ∞ | ∞ | ∞ |
| 1 | AB |  | 3,A | 8,B | ∞ | 6,B | ∞ |
| 2 | ABC |  |  | 8,B | ∞ | 6,B | 4,C |
| 3 | ABCG |  |  | 8,B | ∞ | 6,B |  |
| 4 | ABCGF |  |  | 8,B | 9,G |  |  |
| 5 | ABCGFD |  |  |  | 9,G |  |  |
| 6 | ABCGFDG |  |  |  |  |  |  |

A->B: 1

A->C: 3

A->G: 4

A->F: 6

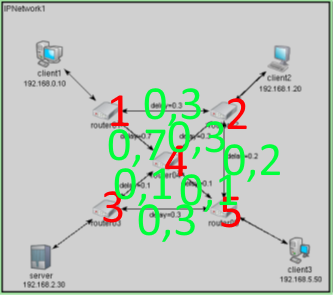
A->D: 8

A->E: 9

5. **OMNeT++-Netzwerkschicht und Routingalgorithmus**

Nutzen Sie die in U 11.4 fertig gestellte Implementierung und optimieren Sie die Forwarding-Tabelle in der Datei **IP.h**. Ihr Ziel ist es dabei alle Daten in möglichst kurzer Zeit an den Zielhost zu schicken. Die unterschiedlichen Latenzen (in Sekunden) sollen dabei wie in Abbildung 5 dargestellt angenommen werden.

Hinweis: Ermitteln Sie dazu eine ähnliche Tabelle wie in U 11.2.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Step | N’ | D(R2) | D(R3) | D(R4) | D(R5) |
| 0 | R1 | 0.3,R1 | ∞ | 0.7,R1 | ∞ |
| 1 | R1R2 |  | ∞ | 0.6,R2 | 0.5,R2 |
| 2 | R1R2R5 |  | 0.9,R5 | 0.6,R2 |  |
| 3 | R1R2R5R4 |  | 0.7,R4 |  |  |