

Übungsblatt Routing

Aufgabe 1:

In der Vorlesung haben Sie zwei Klassen von Algorithmen zur Berechnung der kürzesten Pfade kennengelernt.

- Was sind die Voraussetzungen für den Distanz-Vektor-Algorithmus? Nennen Sie ein Beispiel für ein Distanz-Vektor-Routingprotokoll.
- Was sind die Voraussetzungen für den Link-State-Algorithmus? Nennen Sie ein Beispiel für ein Link-State-Routingprotokoll.
- Implementieren Sie den in der Vorlesung vorgestellten Dijkstra Algorithmus mit Java. Auf der Webseite der Übung finden Sie ein beispielhaftes Netz, für welches Sie mit Hilfe Ihrer Implementierung die kürzesten Pfade berechnen sollen. Am Tag der Übung wird außerdem eine weitere Topologie veröffentlicht, welche Sie nutzen sollen, um die Korrektheit Ihrer Implementierung zu verifizieren.

Aufgabe 2:

Beim Distanz-Vektor-Routing tauschen benachbarte Knoten periodisch Echo- und Hello-Dateneinheiten aus. Diese dienen zum einen dazu, die Nachbarschaftsinformationen zu erlangen (d.h. welche Knoten sind direkt erreichbar und wie sind die Kosten zu diesen), zum anderen können durch die in den Dateneinheiten enthaltenen Routingtabellen optimale Wege zu entfernten Knoten bestimmt werden (Knoten x ändert seine Route zu Knoten y , wenn er aufgrund der Routingtabelle eines Nachbarn einen besseren Weg bestimmen kann). Für die folgende Aufgabe soll das Netz aus Abbildung 1 betrachtet werden.

- Die Routingtabelle von Knoten 5 enthalte die Einträge wie in Tabelle 1 dargestellt. Wie ändern sich die Einträge dieser Routingtabelle, wenn Knoten 5 folgende Vektoren von seinen Nachbarn erhält:
 - Knoten 6: (11,7,10,3,2,0),
 - Knoten 4: (9,5,2,0,1,3)?

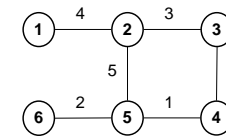


Abbildung 1: Beispielnetz

Zielknoten	1	2	3	4	5	6
Erreichbar über Knoten	4	2	2	4	5	6
Kosten	10	5	8	1	0	2

Tabelle 1: Routingtabelle von Knoten 5

- Betrachten Sie jetzt die Einträge der Routingtabellen aller Knoten bzgl. Zielknoten 6. Anfänglich seien die Einträge aller Routingtabellen mit 0 (Knoten x zu x) bzw. ∞ (Knoten x zu Knoten y , $x \neq y$) initialisiert. Skizzieren Sie die einzelnen Schritte des Verfahrens inklusive der ausgetauschten Kostenwerte, bis ein stabiler Zustand erreicht ist.
- Nehmen Sie an, Knoten 6 fällt aus. Wie ändern sich jetzt die Routingtabellen in den folgenden 4 Schritten? Wie lautet der Fachbegriff für dieses Verhalten?

Aufgabe 3:

In Abbildung 2 sei eine beispielhafte Netzkonfiguration dargestellt: Ein Netz der Klasse B (160.25.0.0) enthalte die zwei dargestellten unterschiedlichen Subnetze (160.25.41 und 160.25.52), wobei jedes Subnetz einem Ethernet-Segment entspricht. Host 1 bis Host 5 sind dabei über einen Router an die Außenwelt angebunden. Für jedes System sind IP-Adresse(n) angegeben, sowie die Namen der vorhandenen Netzwerkkarten.

- Wie lautet die Konfiguration von Host 1 und Host 5 bzgl. Netzmaske und Default-Router? Welche Systeme sind somit direkt und welche indirekt erreichbar? Welche maximale Anzahl von Hosts können in den Subnetzen mit einer entsprechenden Subnetzmaske unterschieden werden?
- Wie sieht die IP-Routingtabelle des lokalen Routers aus?
- Wäre eine IP-Adresse von 160.25.41.255 (Netzwerkkarte eth0) für den Router zulässig?
- Wie viele Einträge hat ein beliebiger Router im Internet für die in Abbildung 2 dargestellten Subnetze? Erhöht sich diese Zahl beim Hinzufügen weiterer Subnetze?

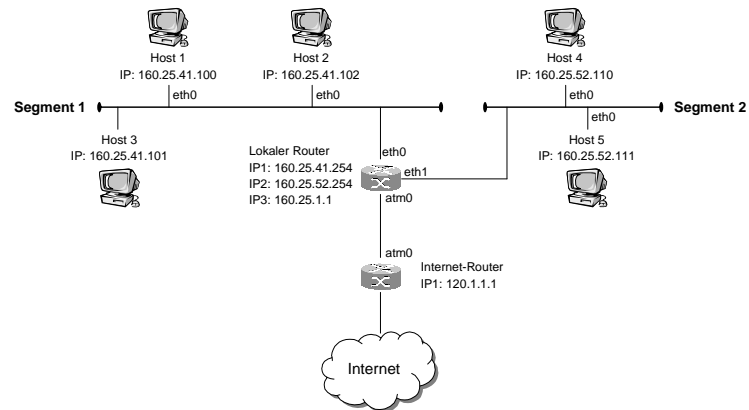


Abbildung 2: Beispielhaftes IP-Netz

Aufgabe 4:

In der folgenden Aufgabe soll ein näheres Verständnis für die in der Vorlesung vorgestellte Datenstruktur *Patricia Trie* entwickelt werden.

Betrachten Sie die in den folgenden Teilaufgaben einzusortierenden Buchstaben mit ihrem binären ASCII-Code:

Bitindex	4	3	2	1	0
A	0	0	0	0	1
E	0	0	1	0	1
I	0	1	0	0	1
K	0	1	0	1	1
L	0	1	1	0	0
M	0	1	1	0	1
T	1	0	1	0	0

- Fügen Sie die folgenden Buchstaben in einen *Binären Trie* ein.
T - E - L - E - M - A - T - I - K
- Wie würde ein *Patricia Trie* nach Einfügen der Buchstaben aussehen?
- Welche Vor- und Nachteile hat der Patricia Trie im Vergleich zum binären Trie?