Some comments on last week’s assignment

Overview current assignment

Solving a 2020 Dijkstra exam question

Time for you to solve the tasks

Solutions will be published next week
Task 1.2: Internet Organization

A network of multiple autonomous systems (AS).

The network on the left consists of multiple autonomous systems (AS). Single-headed arrows point from providers to their customers. Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3.
- Tier-2 ISPs: AS5, AS7, AS8.
- IXPs: AS6, AS9.

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.
  - Advantages: Still connected if one of the provider fails. For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...
  - Disadvantages: Network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
The Internet has a hierarchical structure

| Tier-1      |  |
|-------------|  |
| international | have no provider  |

| Tier-2      |  |
|-------------|  |
| national    | provide transit to Tier-3s have at least one provider  |

| Tier-3      |  |
|-------------|  |
| local       | do not provide any transit have at least one provider  |
Task 1.2-a) Internet Organization

A network of multiple autonomous systems (AS).

The network on the left consists of multiple autonomous systems (AS). Single-headed arrows point from providers to their customers. Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3.
- Tier-2 ISPs: AS5, AS7, AS8.
- IXPs: AS6, AS9.

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.
- Advantages: Still connected if one of the provider fails. For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...
- Disadvantages: Network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
Task 1.2-a) Internet Organization

A network of multiple autonomous systems (AS).

- Single-headed arrows point from providers to their customers.
- Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3.
- Tier-2 ISPs: AS5, AS7, AS8.
- IXPs: AS6, AS9.

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.
- Advantages: Still connected if one of the providers fails.
- For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...
- Disadvantages: network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
Task 1.2-a) Internet Organization

A network of multiple autonomous systems (AS).

The network on the left consists of multiple autonomous systems (AS). Single-headed arrows point from providers to their customers. Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3
- Tier-2 ISPs: AS5, AS7, AS8
- Tier-3 ISPs: AS4, AS10, AS11, AS12
- IXPs: AS6, AS9

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.

Advantages: Still connected if one of the providers fails. For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...

Disadvantages: Network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
Task 1.2-a) Internet Organization

A network of multiple autonomous systems (AS).

The network on the left consists of multiple autonomous systems (AS). Single-headed arrows point from providers to their customers. Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3.
- Tier-2 ISPs: AS5, AS7, AS8.
- IXPs: AS6, AS9.

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.

Advantages: Still connected if one of the providers fails. For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...

Disadvantages: Network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
Task 1.2-a) Internet Organization

A network of multiple autonomous systems (AS).

The network on the left consists of multiple autonomous systems (AS). Single-headed arrows point from providers to their customers. Double-headed arrows represent peer connections.

a) For each AS, identify if it is a Tier-1, Tier-2 or Tier-3 ISP network or an IXP.

Solution:
- Tier-1 ISPs: AS1, AS2, AS3.
- Tier-2 ISPs: AS5, AS7, AS8.
- IXPs: AS6, AS9.

b) AS7 has two different providers (AS1 and AS2). How is this type of interconnection called? What are the advantages of multiple different providers for AS7? Can you see any disadvantages?

Solution:
- Multihoming.
- Advantages: Still connected if one of the provider fails. For destinations (IP prefixes) which can be reached over both providers, AS7 can choose the better one based on cost, trust, ...
- Disadvantages: Network configuration is slightly more difficult. The total cost could be higher compared to an ISP with only one provider.
Task 1.6-g) Packet vs. Circuit Switching

![Graph Showing Packet vs. Circuit Switching]

- **Demand [Mbps]**
  - 5 Mbps between 1-5 seconds
  - 10 Mbps between 5-10 seconds
  - 5 Mbps between 15-17 seconds

- **Time [s]**
  - From 1 to 17 seconds

- **Nodes**
  - Node A
  - Node B
  - Node C
  - Node S
Task 1.6-g) Packet vs. Circuit Switching

If A and B use packet switching

Node A
Node B

Demand [Mbps]

<table>
<thead>
<tr>
<th>Time [s]</th>
<th>Node A</th>
<th>Node B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>18.5</td>
<td></td>
</tr>
</tbody>
</table>
Communication Networks

Exercise 2

Some comments on last week’s assignment

Overview current assignment

Solving a 2020 Dijkstra exam question

Time for you to solve the tasks

Solutions will be published next week
Task 2.1: Dijkstra’s algorithm
Let’s compute the shortest-paths from $u$ using Dijkstra’s algorithm

$S = \{u\}$ set of nodes for which we know the shortest-path

$D(v)$ the smallest distance currently known by $u$ to reach $v$

$c(u, v)$ the weight of the link connecting $u$ and $v$
Initialization

\[ S = \{u\} \]

for all nodes \( v \):

if \( (v \text{ is adjacent to } u) \):

\[ D(v) = c(u, v) \]

else:

\[ D(v) = \infty \]
D is initialized based on u’s weight, and S only contains u itself.

\[ D(.) = \]

\[ S = \{u\} \]

\[
\begin{array}{c|c}
A & 3 \\
B & \infty \\
C & \infty \\
D & \infty \\
E & 2 \\
F & \infty \\
G & \infty \\
\end{array}
\]
while not all nodes in $S$:
    add $w$ with the smallest $D(w)$ to $S$
    update $D(v)$ for all adjacent $v$ not in $S$: 
    \[ D(v) = \min\{D(v), D(w) + c(w, v)\} \]
$D(.) = \{u\}$

$S = \{u\}$

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>\infty</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td>\infty</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td>\infty</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>\infty</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>\infty</td>
</tr>
</tbody>
</table>
$D(.) = \{u, E\}$

$S = \{u, E\}$

$D(.) = \begin{align*}
A & \quad 3 \\
B & \quad \infty \\
C & \quad \infty \\
D & \quad \infty \\
E & \quad 2 \\
F & \quad \infty \\
G & \quad \infty 
\end{align*}$
\[ D(.) = \]

\[ S = \{u, E\} \]

\begin{align*}
A & \quad 3 \\
B & \quad \infty \\
C & \quad 3 \quad \text{D}(v) = \min\{\infty, 2 + 1\} \\
D & \quad \infty \\
E & \quad 2 \\
F & \quad \infty \\
G & \quad 6 \quad \text{D}(v) = \min\{\infty, 2 + 4\}
\end{align*}
Skipping a few steps...

D(.) =

S = \{u, E, A, C\}

\begin{align*}
A & : 3 \\
B & : 5 \\
C & : 3 \\
D & : 7 \\
E & : 2 \\
F & : 8 \\
G & : 6
\end{align*}
\[ D(.) = \{ \{u, E, A, C\} \}

\begin{align*}
A & \quad 3 \\
B & \quad 5 \text{ \small{smallest D(w)}} \\
C & \quad 3 \\
D & \quad 7 \\
E & \quad 2 \\
F & \quad 8 \\
G & \quad 6
\end{align*}
$D(v) = \min\{7, 5 + 1\}$

$S = \{u, E, A, C, B\}$

<table>
<thead>
<tr>
<th>Vertex</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
</tr>
<tr>
<td>B</td>
<td>5</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
</tr>
<tr>
<td>D</td>
<td>6</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
</tr>
<tr>
<td>F</td>
<td>8</td>
</tr>
<tr>
<td>G</td>
<td>6</td>
</tr>
</tbody>
</table>
Here is the final state

\[ D(.) = \begin{align*}
A & \quad 3 \\
B & \quad 5 \\
C & \quad 3 \\
D & \quad 6 \\
E & \quad 2 \\
F & \quad 8 \\
G & \quad 6
\end{align*} \]

\[ S = \{u, A, B, C, D, E, F, G\} \]
From the shortest-paths, \( u \) can directly compute its forwarding table.
Task 2.2: Changing weights

Consider dynamic weights

**Important:** different weights for both link directions

Next-hop as tie-break value

e — incoming traffic
Task 2.3: Link Weight Configuration

The Abilene network in the US
Task 2.4: Source-and-Destination-Based Routing

Is it possible to route packets based on the source address?

What are advantages/disadvantages?
Communication Networks

Exercise 2

Some comments on last week’s assignment

Overview current assignment

Solving a 2020 Dijkstra exam question

Time for you to solve the tasks

Solutions will be published next week
Task 2.5-a) **Reverse Dijkstra (Exam question 2020)**

Network with 10 nodes and unknown links and link weights

Full Dijkstra output for node U

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>100</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>10</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>10</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>10</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>10</td>
<td>13</td>
<td>11</td>
</tr>
</tbody>
</table>
Goal

Identify all the links and their weights visible in the Dijkstra output

There is at most one link between two nodes

All links have a positive weight

In case of nodes with equal shortest paths, the algorithm prefers the node coming first in the alphabet
Start with the first iteration of the Dijkstra output

![Graph with nodes U, A, B, C, D, E, F, H, G, and I. Each node is connected by undirected edges.]

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
</tbody>
</table>
Start with the first iteration of the Dijkstra output

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra’s algorithm performed on node U towards all other nodes.

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
</tbody>
</table>
Then continue with the node with the smallest distance
Add new links and update weights

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node $U$ towards all other nodes.
Add new links and update weights

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node $U$ towards all other nodes.
Add new links and update weights

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.
Your turn

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>100</td>
<td>-</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
</tbody>
</table>
Next we look at node D

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra’s algorithm performed on node U towards all other nodes.
Your turn

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
</tbody>
</table>
Node E is next, nothing changes

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.
Your turn

![Network Diagram](image)

**Figure 6:** A network consisting of 10 nodes with unknown links and link weights.

<table>
<thead>
<tr>
<th>#</th>
<th>U</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td>10</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>10</td>
<td>14</td>
<td>11</td>
</tr>
</tbody>
</table>

**Table 1:** For each iteration (1 to 10) the table shows the shortest path found by Dijkstra’s algorithm performed on node U towards all other nodes.
Next we look at node G

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.
Final network (a few steps skipped)

Figure 6: A network consisting of 10 nodes with unknown links and link weights.

Table 1: For each iteration (1 to 10) the table shows the shortest path found by Dijkstra's algorithm performed on node U towards all other nodes.
Task 2.5-b) Reverse Dijkstra (Exam question 2020)

Given Dijkstra’s output, could we have missed a link starting at U?
If yes, which one and with which weight? If no, why not?
Task 2.5-b) Reverse Dijkstra \textbf{(Exam question 2020)}

Given Dijkstra’s output, could we have missed a link \textit{starting at U}? If yes, which one and with which weight? If no, why not?

Not possible! During it initialization phase, Dijkstra considers all directly connected nodes. We will therefore see all links (with their cost) starting from node U.
Task 2.5-c) Reverse Dijkstra (Exam question 2020)

Given Dijkstra’s output, could we have missed a link starting at C?

If yes, which one and with which weight? If no, why not?
Task 2.5-c) Reverse Dijkstra (Exam question 2020)

Given Dijkstra’s output, could we have missed a link starting at C?
If yes, which one and with which weight? If no, why not?
Possible! For example a link between C and G with weight > 8.
Communication Networks
Exercise 2

Some comments on last week’s assignment

Overview current assignment

Solving a 2020 Dijkstra exam question

Time for you to solve the tasks

Solutions will be published next week