

Bei uns war ursprünglich noch ein Dritter in unserer Abgabegruppe eingeteilt. Wir haben ihn vor über einer Woche versucht per E-Mail zu erreichen, leider erfolglos.

Nach Ablauf der Anmeldefrist zu den Abgabegruppen haben wir gesehen, dass diese Person leider unsere Abgabegruppe verlassen hat.

Bisher konnten wir noch keinen Dritten für unsere Abgabegruppe finden.

Uns wurde auch seit dem letzten Blatt keine weitere Person zugeteilt.

Aufgabe 4.1

- a) Maximale Rahmengröße:

$$12B + 1600B \hat{=} 96b + 12800b = 12896b$$

Übertragung eines Rahmens:

$$\frac{4}{1000}s + \frac{12896b}{100 \cdot 10^6 \frac{b}{s}} = 0.00412896s$$

Übertragung eines ACK:

$$\frac{4}{1000}s + \frac{96b}{100 \cdot 10^6 \frac{b}{s}} = 0.00400096s$$

Maximale Nutzdatenrate:

$$\frac{12800b}{0.00412896s + 0.00400096s} = 1574431.23672 \frac{b}{s} \hat{=} 1.57443123672 \frac{Mb}{s}$$

- b) Wahrscheinlichkeit, dass eine komplette Übertragung (Rahmen + ACK) erfolgreich ist:

$$(1 - 10^{-4})^{12896b+96b} = 0.272732187262$$

Mittlere Nutzdatenrate:

$$0.272732187262 \cdot 1574431.23672 \frac{b}{s} = 429398.074885 \frac{b}{s} \hat{=} 429.398074885 \frac{kb}{s}$$

- c) i) Um die Nutzdatenrate aus a) zu erreichen, können wir die Rahmengröße auf 1 setzen. Dadurch arbeitet unser Sliding-Window-Verfahren genauso wie das Stop-And-Wait-Protokoll, weshalb die Nutzdatenrate gleich ist.
ii) Es dauert mindestens

$$0.00412896s + 0.00400096s = 0.00812992s$$

, bis der erste Rahmen des Fensters bestätigt wird.

Das Absenden eines Rahmens beträgt

$$\frac{12896b}{100 \cdot 10^6 \frac{b}{s}} = 0.00012896s$$

Um die Leistung des Kanals komplett ausnutzen zu können, müssen wir mindestens so viele Rahmen in unserer Fensterzeit 0.00812992s senden, dass keine Pausen entstehen. Damit benötigen wir eine Fenstergröße von mindestens

$$\lceil \frac{0.00812992\text{s}}{0.00012896\text{s}} \rceil = \lceil 63.0421836228 \rceil = 64$$

Rahmen.

Ab unter 64 Rahmen pro Fenster, sprich z.B. 63 Rahmen, hätten wir schon eine Wartezeit von mindestens 0.00000544s. In dieser Zeit ist der Kanal ungenutzt.

Aufgabe 4.2

a) i) Latenz:

$$\frac{1000\text{km}}{200000\frac{\text{km}}{\text{s}}} = \frac{1}{200}\text{s}$$

Senden eines Rahmens:

$$\frac{2500 \cdot 8\text{b}}{1 \cdot 10^9\frac{\text{b}}{\text{s}}} = 0.00002\text{s}$$

Es kann also ein Rahmen alle 0.00002s auf die Leitung gelegt werden. Es braucht nun weitere $(\frac{1}{200}\text{s} + 0.00002\text{s}) + \frac{1}{200}\text{s} = 0.01002\text{s}$ bis die Bestätigung des Rahmens ankommt.

Anzahl Rahmen:

$$\frac{0.01002\text{s}}{0.00002\text{s}} = 501$$

ii) Da wir 501 Rahmen "gleichzeitig" senden können, benötigen wir mind. $501 + 1$ Sequenznummern. Diese können wir mit 9 Bit darstellen:

$$\lceil \lg(502) \rceil = 9$$

iii) Es gibt keinen Unterschied in der Rahmengröße hier. Aus diesem Grund kann man wie bei der ii) 9 Bit für die Darstellung der Sequenznummern verwenden.

b)

$$\frac{501 \cdot 2450 \cdot 8\text{b}}{0.01002\text{s}} = 980000000\frac{\text{b}}{\text{s}} \hat{=} 980\frac{\text{Mb}}{\text{s}}$$

Aufgabe 4.3

1. Da alle Weiterleitungstabellen leer sind, broadcastet jeder Switch (und jeder Hub sowieso) den ersten Rahmen. Damit bekommen auch alle Rechner (außer A, Switch 1 broadcastet nicht an Port 1.1, da der Rahmen von dort kam) diesen Rahmen.

Switch 1:

1.1	A
1.2	
1.3	
1.4	

Switch 2:

3.1	A
3.2	
3.3	

Switch 3:

4.1	A
4.2	
4.3	

Switch 4:

6.1	A
6.2	

2. C sendet nun über 2.2 einen Rahmen nach A an Hub 1. Hub 1 broadcastet den Rahmen an Switch 1 und Switch 2. Weder Switch 1 kennt den Port von A (1.1) und sendet ihm diesen. Switch 2 kennt auch den Port von A, jedoch liegt er an 3.1, also an dem Port, an dem er den Rahmen erhalten hat. Switch 2 dropped also den Rahmen.

Switch 1:

1.1	A
1.2	
1.3	C
1.4	

Switch 2:

3.1	A	C
3.2		
3.3		

Switch 3:

4.1	A
4.2	
4.3	

Switch 4:

6.1	A
6.2	

3. G sendet nun über 6.2 einen Rahmen nach A an Switch 4. Switch 4 weiß, dass A an Port 6.1 anliegt, also gibt er Switch 2 den Rahmen weiter. Switch 2 weiß, dass A an 3.1 anliegt, also gibt er den Rahmen an Hub 1 weiter. Hub 1 broadcastet nun das Signal an Rechner C und Switch 1. Switch 1 weiß, dass A an 1.1 anliegt.

Switch 1:

1.1	A
1.2	
1.3	C G
1.4	

Switch 2:

3.1	A C
3.2	
3.3	G

Switch 3:

4.1	A
4.2	
4.3	

Switch 4:

6.1	A
6.2	G

Aufgabe 4.2

- a) Man könnte einen solchen Switch immer als root bridge wählen, indem man ihm die niedrigste ID gibt (zB 0), und dann als root bridge immer den Switch mit der geringsten ID wählt.

Es ist sinnvoll einen solchen Switch als root bridge zu wählen, wenn ein hoher Anteil an Kommunikation mit dem Internet stattfindet. Denn dann ist der ST so aufgebaut, dass alle Switches einen möglichst schnellen Weg (minimale Kosten) zur root bridge haben. Damit kann also die Datenrate im Netzwerk erhöht werden.

