

Übungsblatt 4: Verzögerungen und Flusskontrolle

Übung

Aufgabe 1 Ausbreitungs- vs. Serialisierungsverzögerung

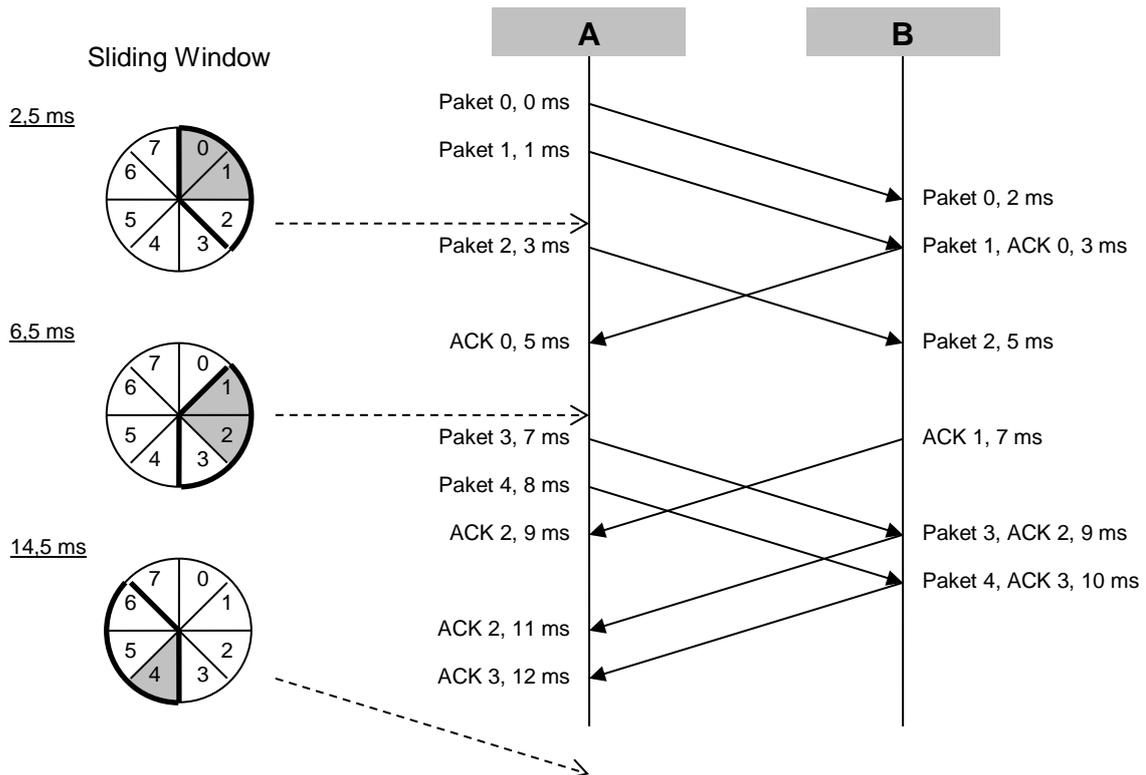
- Ausbreitungsverzögerung: $t_A = s/v = 2 \cdot 10^3 \text{ m} / 2 \cdot 10^8 \text{ m/s} = 10^{-5} \text{ s} = 10 \mu\text{s}$
- Serialisierungsverzögerung: $t_S = P/R$ (Bitübertragungsrate: $R = P/t_S$)
- Ansatz: $t_A = t_S$
- $R = P/t_A = 12.000 \text{ Bit} / 10 \mu\text{s} = 1,2 \cdot 10^9 \text{ Bit/s} = 1,2 \text{ Gbit/s}$

Aufgabe 2 Latenz bei Zwischensystemen

- a) $t_S = 36.000 \text{ Bit} / 100 \cdot 10^6 \text{ Bit/s} = 0,36 \text{ ms}$ // t_S : Serialisierung kompl. Nachricht
 $L = 3 \cdot t_S = 1,08 \text{ ms}$ // Latenz
- b) $t_{Sp} = 12.200 \text{ Bit} / 100 \cdot 10^6 \text{ Bit/s} = 0,122 \text{ ms}$ // t_{Sp} : Serialisierung eines Pakets
nach $3 \cdot t_{Sp} = 0,366 \text{ ms}$ hat B Paket 1 empfangen
nach $4 \cdot t_{Sp} = 0,488 \text{ ms}$ hat B Paket 2 empfangen
nach $5 \cdot t_{Sp} = 0,610 \text{ ms}$ hat B Paket 3 empfangen (Paket 3 trifft $2 \cdot t_{Sp}$ nach Paket 1 ein)
- c) $t_{Sa} = 400 \text{ Bit} / 100 \cdot 10^6 \text{ Bit/s} = 4 \mu\text{s}$ // t_{Sa} : Serialisierung einer Bestätigung
 $\text{RTT} = 3 \cdot (t_{Sp} + t_{Sa}) = 0,378 \text{ ms}$
nach $3 \cdot \text{RTT} = 1,134 \text{ ms}$ wurde die Bestätigung des 3. Pakets von A empfangen
nach $3 \cdot \text{RTT} - 3 \cdot t_{Sa}$ (oder $2 \cdot \text{RTT} + 3 \cdot t_{Sp}$) = $1,122 \text{ ms}$ hat B das 3. Paket empfangen
- d) $t_{Sh} = 200 \text{ Bit} / 100 \cdot 10^6 \text{ Bit/s} = 2 \mu\text{s}$ // t_{Sh} : Serialisierung eines Paketheader
 $\text{RTT}_C = 2 \cdot t_{Sh} + t_{Sp} + 2 \cdot t_{Sh} + t_{Sa} = 4 \cdot t_{Sh} + t_{Sp} + t_{Sa} = 0,134 \mu\text{s}$
nach $3 \cdot \text{RTT}_C = 0,402 \text{ ms}$ wurde die Bestätigung des 3. Pakets von A empfangen
nach $3 \cdot \text{RTT}_C - (2 \cdot t_{Sh} + t_{Sa}) = 0,394 \text{ ms}$ hat B das 3. Paket empfangen

Aufgabe 3 Flusskontrolle: Skizzierung des Sliding Window-Verfahrens

a)



- b) A hat sich nicht zu jeder Zeit an das Sliding Window Protokoll gehalten. So hätte A nämlich nicht zum Zeitpunkt 8 ms das Paket 4 abschicken dürfen, da das Sliding Window voll war (Pakete 1 – 3). B hat von dieser Verletzung gar nichts mitbekommen, da B vor dem Empfang des Paketes 4 das Paket 2 bestätigt hat und somit definitiv für die Pakete 3 – 5 empfangsbereit war. (Dies könnte man ggf. durch eine Darstellung des Sliding-Windows bei B veranschaulichen.)

Selbststudium

Aufgabe 4 Komplexaufgabe Fluss-Steuerung

Lösung

- a) Es geht dabei um das Problem, dass der Sender die Daten schneller schickt als der Empfänger sie verarbeiten kann. Im dargestellten Fall könnte dies bspw. dadurch passieren, dass
- der Rechner B einfach langsamer ist und deshalb die Komm.-SW gar nicht so schnell in der Lage ist, die Prüfsummen der einzelnen Nachrichten zu überprüfen wie A die Nachrichten schicken kann.
 - das Programm P_B die Daten nur zögerlich oder vielleicht in zu großen Abständen der Komm.-SW abnimmt.

Der Endeffekt wäre dabei letztlich immer irgendwie ein Pufferüberlauf beim Empfänger, d. h. es kommen Daten beim Empfänger an, für die kein freier Platz zum Ablegen vorhanden ist und die deshalb verworfen werden müssen.

- b) Zum einen besteht eine Möglichkeit darin, bei fehlendem Speicherplatz Nachrichten zu verwerfen und halt einfach auf die Methoden der Fehlerbehandlung zu setzen.

In der Regel ist es jedoch günstiger, den Verlust von Nachrichten durch fehlenden Speicherplatz beim Empfänger zu vermeiden. Dafür muss der Sender irgendwelche Informationen über die Aufnahme-

fähigkeit des Empfängers bekommen, damit der Sender seine Sendeleistung ggf. einschränken kann. Genau dies versucht eine Fluss-Steuerung. Sollte es trotzdem mal zu einem überlaufenden Puffer beim Empfänger kommen, können dann immer noch Verfahren der Fehlerbehandlung zum Einsatz kommen.

- c) Es ergibt nun eine Reihe von Möglichkeiten, die angewendet werden können, die im Folgenden kurz aufgelistet sind: (*in der Übung muss diese Reihenfolge sicherlich nicht eingehalten werden, vielleicht ist sie aber hilfreich!!!*)

Start- & Stopp-Mechanismus

- hier darf der Sender zunächst ohne Einschränkungen senden
- wenn der Empfänger merkt, dass der Puffer sich zunehmend füllt, kommt Stopp-Signal
- dann muss der Sender wieder auf Start-Signal warten, bevor er weitersenden kann
- Nachteile:
 - **sehr ungleichmäßiges Senden**
 - bei großen Wegstrecken (→ große Übertragungsverzögerungen) sind große Puffer nötig, da Stopp-Signal ja erst beim Sender ankommen muss, bevor dieser aufhört, **Änderungen RTT?**
 - ca. doppeltes Bandbreite-Verzögerungs-Produkt muss noch frei sein beim Absenden des Stopp-Signals!

Kredit-Mechanismen

- hier teilt der Empfänger dem Sender mit, dass er **bestimmte Menge an Daten schicken darf**
- (der Sender bekommt also einen Kreditrahmen, der sich mit jeder versendeten Nachricht verringert)
- um weitere Pakete schicken zu dürfen, bedarf es Erhöhung des Kreditrahmens durch den Empfänger
- Veränderungen des Kreditrahmens zu jedem Zeitpunkt möglich, auch Verringerungen!

Sliding-Window

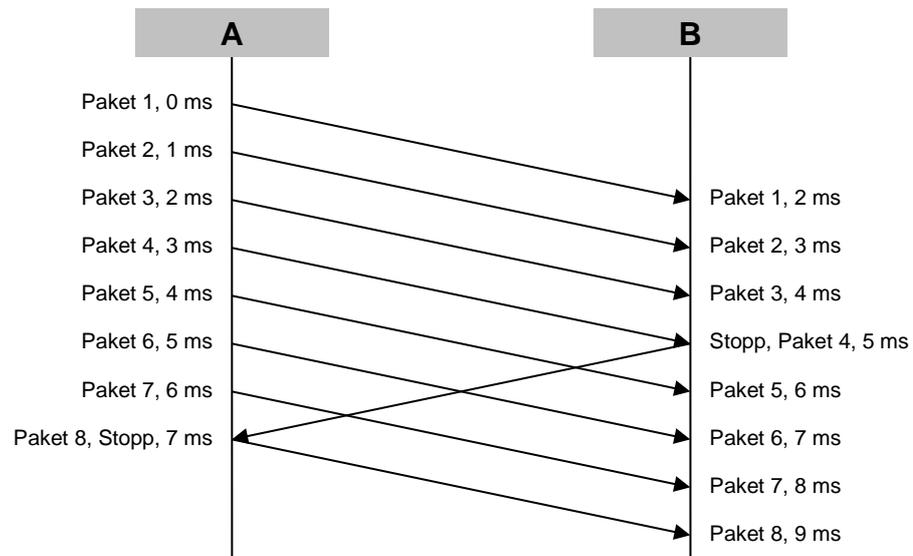
- bei Kredit-Mechanismen sind jeweils Veränderungen des Kreditrahmens vom Empfänger an den Sender zu schicken (mit Angabe, wie groß diese Änderung ist)
- andererseits werden aber meist sowieso Bestätigungen vom Empfänger an den Sender geschickt (für die Fehlerbehandlung, anderes Thema!)
- wenn der Empfänger die Daten nun aber halbwegs gleichmäßig abarbeiten kann, dann folgende Festlegung möglich:
 - **Fluss-Steuerung und Fehlerbehandlung werden miteinander verschmolzen**
 - wie bei anderen Kreditmechanismen gibt der Empfänger dem Sender ein bestimmtes Kontingent an zu verschickenden Nachrichten
 - **mit einer Bestätigung wird jedoch zugleich mitgeteilt, dass diese Nachricht aus dem Puffer genommen werden konnte und der Sender eine weitere schicken darf**
- Vorteil:
 - u. U. weniger umfangreiche Nachrichten für Rückrichtung notwendig

Raten-basierte Verfahren

- die bisher genannten Verfahren basieren alle darauf, dem Sender ein bestimmtes Kontingent mitzuteilen, in dessen Rahmen der Sender völlig frei senden darf
- **häufig ist es aber so, dass der Empfänger eine ziemlich konstanten Geschwindigkeit hat, mit der er die Daten verarbeiten kann**
- bei einem Raten-basierten Verfahren kann der Empfänger nun dem Sender diese Geschwindigkeit mitteilen – und der Sender muss dann seine Sendeleistung entsprechend anpassen
 - Veränderungen bzgl. dieser Rate kann der Empfänger jederzeit an den Sender schicken
- Vorteil:
 - in vielen Fällen können **die Steuerungsnachrichten der Fluss-Steuerung weitgehend vermieden** werden

Aufgabe 5 Start- & Stopp-Mechanismus

Lösung



Anmerkung

Die Fluss-Steuerung habe ich deshalb in die Anwendungsebene verlegt, damit wirklich klar ist, dass Pakete erst verarbeitet werden, wenn sie vollständig empfangen sind.