
Leistungsnachweis in
Rechnerkommunikation

Wintersemester 2011/2012

29. März 2012

Name: _____

Matrikelnummer: _____

Geburtsdatum: _____

Studienfach: _____

Fachsemester: _____

Extra Nachweis: benoteter Schein unbenoteter Schein

- Bitte verwenden Sie einen blauen oder schwarzen Kugelschreiber (kein rot, keinen Bleistift).
- Schriftliche Aufzeichnungen (sowohl eigene Aufzeichnungen wie auch Bücher) sind als Hilfsmittel zugelassen. Auch ein Taschenrechner ist erlaubt und hilfreich. Nicht zugelassen sind dagegen Computer, PDAs, Mobiltelefone und sonstige Kommunikationsmittel.
- Legen Sie den Ausweis (mit Lichtbild) griffbereit auf den Platz.
- Bitte überprüfen Sie, ob Sie alle 13 Blätter erhalten haben.
- Schreiben Sie die Antworten jeweils in den freien Raum hinter den Fragen. Sollte dieser nicht ausreichen, steht noch freier Raum am Ende der Klausur zur Verfügung. Bitte kennzeichnen Sie dort deutlich, welche Aufgabe Sie bearbeiten. Gesondert beigelegte Blätter werden nicht bewertet!
- Schreiben Sie deutlich! Unleserliche Antworten gehen nicht in die Bewertung ein!

Ich habe die Hinweise auf dieser Seite zur Kenntnis genommen und alle 13 Blätter der Klausur empfangen:

Unterschrift

Bewertung:	1	2	3	4	Σ

1 Allgemeine Fragen (24 Punkte gesamt)

1.1 Leitungskodierung (2 Punkte)

Zu welchem Zweck werden digitale Signale vor der Übertragung einer Leitungskodierung unterzogen? Nennen Sie zwei Eigenschaften.

1.2 Manchesterkodierung (3 Punkte)

Skizzieren Sie die in Abbildung 1 gezeigte Bitfolge nach der einfachen Manchesterkodierung in die selbe Abbildung. Welcher Zusammenhang besteht bei der einfachen Manchesterkodierung zwischen Signal- und Übertragungsrate?

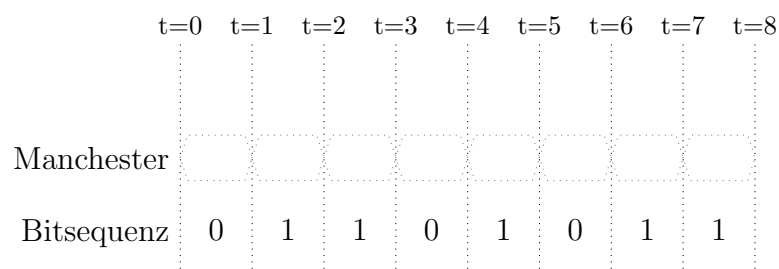


Abbildung 1: Leitungskodierung

1.3 Möglichkeiten der Leitungskodierung (2 Punkte)

Nennen Sie vier weitere Möglichkeiten der Leitungskodierung.

1.4 Signalabtastung (2 Punkte)

Wir wollen ein analoges Signal durch periodisches Abtasten wieder vollständig durch die Abtastwerte zurückgewinnen. Welche Frequenz darf somit ein analoges Signal maximal haben, wenn es alle 0.2ms abgetastet wird?

1.5 Bandbreite (5 Punkte)

Bei einer Datenübertragung soll eine maximale Datenrate von 96 kbit s^{-1} erzielt werden. Wie groß muss die Bandbreite B gewählt werden, wenn man Rauschen vernachlässigt und wenn pro Schritt ein Signal aus vier Bits übertragen werden kann? Wie verändert sich die Bandbreite B , wenn wir ein Signal-Rausch-Verhältnis von $\frac{S}{N} = 15$ annehmen?

1.6 Marathonläufer (5 Punkte)

Ein Marathonläufer trägt eine Schachtel von 5 CD-ROMs mit einer Speicherkapazität von jeweils 700 MB mit sich. Bei einem Marathon erreicht der Läufer eine Durchschnittsgeschwindigkeit von 21.6 km h^{-1} . Bis zu welcher Entfernung überträgt der Marathonläufer die Daten schneller als eine DSL-Verbindung, wenn diese eine Datenrate von 4 Mbit s^{-1} beziehungsweise 16 Mbit s^{-1} hat? Gehen Sie von einer unendlich schnellen Ausbreitungsverzögerung der DSL Leitung aus. Das Brennen der CD-ROMs nimmt keine Zeit in Anspruch. Nehmen Sie vereinfachend für Ihre Rechnung an, dass gilt: $1 \text{ MB} = 10^6 \text{ B}$.

1.7 Harmonische Schwingungen (5 Punkte)

In einem neuartigen Telefonnetz besteht ein periodisches Signal aus 2 B. Bestimmen Sie für die Bitraten $B_1 = 2400 \text{ bit s}^{-1}$, $B_2 = 4800 \text{ bit s}^{-1}$ und $B_3 = 9600 \text{ bit s}^{-1}$ die Frequenz der ersten harmonischen Schwingung. Berechnen Sie die Bandbreitenbeschränkung f_c , wenn bei der Bitrate B_1 insgesamt 20 harmonische Schwingungen übertragen wurden.

2 Transportschicht (22 Punkte gesamt)

2.1 TCP Kontrollmechanismen (4 Punkte)

Welche zwei Kontrollmechanismen gibt es bei TCP und wofür sind sie da (kurze Erklärung)?

2.2 TCP Signalisierung (10 Punkte)

In Abbildung 2 baut ein Host A zu einem Host B eine TCP Verbindung auf. Nach einer Anfrage mit einer Größe von 5 B beginnt Host B mit maximaler Geschwindigkeit Daten an Host A zu übertragen. Vervollständigen Sie das in Abbildung 2 gezeigte Ablaufdiagramm durch Eintragen der Sequenznummern (Seq), der Acknowledgment Nummern (Ack) und des Congestion Windows (CW). Gehen Sie davon aus, dass die Puffer immer unendlich groß sind. Die Fenstergrößen sind nach Erhalten und vor dem Aussenden eines Segments in der Maximum Segment Size (MSS) Einheit anzugeben, wobei gilt: 1 MSS = 10 B. Host A startet mit Seq=50 und Host B mit Seq=0. Es gibt kein Selective Acknowledgment (SACK) und keinen weiteren Timeout. Zu welcher Zeit (t_1 oder t_2) könnte Host B neue Daten übermitteln? Begründen Sie Ihre Antwort.

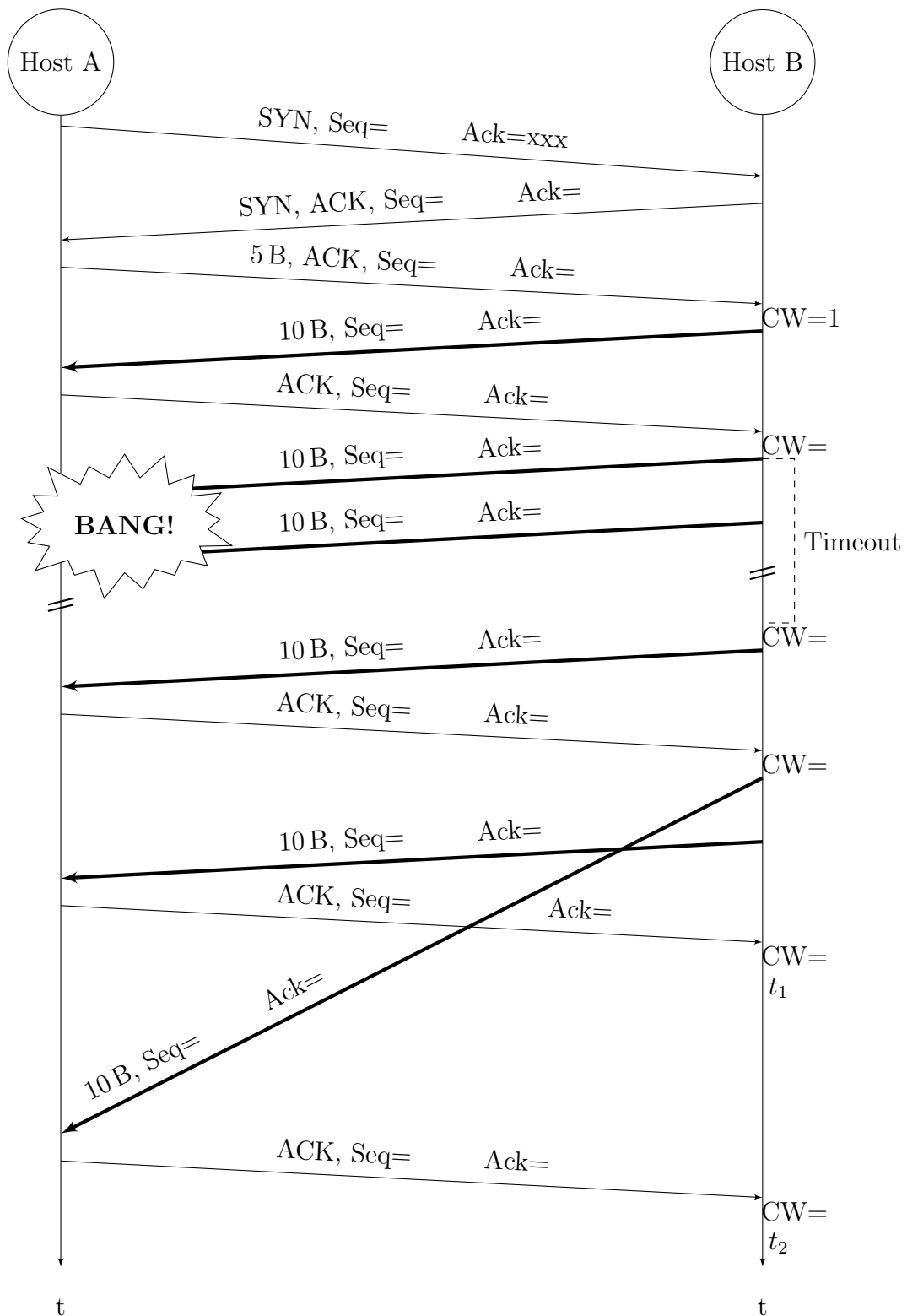


Abbildung 2: Ablaufdiagramm für eine TCP-Verbindung

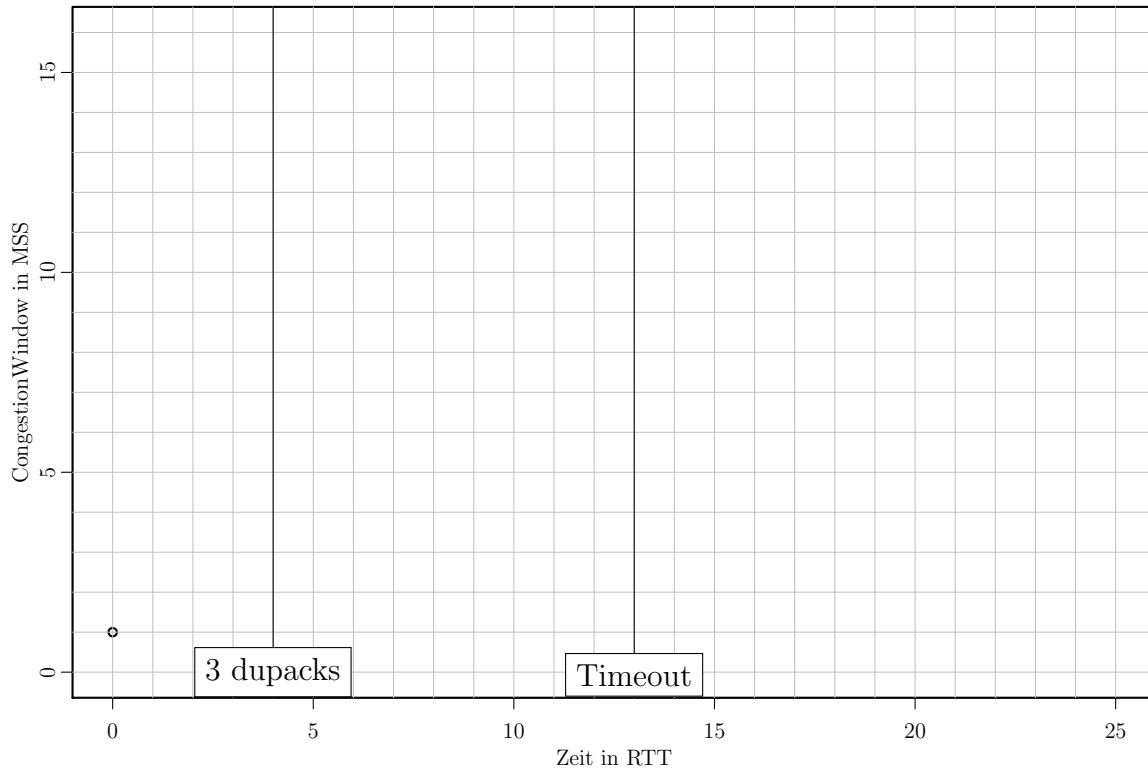


Abbildung 3: Zeitlicher Ablauf des CongestionWindow

2.3 TCP Überlastkontrolle (8 Punkte)

Zeichnen Sie in Abbildung 3 den Verlauf des Congestion Windows über die Zeit in Round Trip Times (RTT) möglichst exakt ein. Sie können vereinfachend annehmen, dass sich im Zustand des *Additive Increase* das Congestion Window um exakt eine MSS pro RTT erhöht. Beschreiben Sie für die drei Zeiträume 0 – 4, 5 – 13 und 14 – 25 das Kurvenverhalten und nennen Sie die zugrunde liegende Vorschrift. Geben Sie für jeden Zeitraum den Threshold an.

3 Zustandsdiagramm (20 Punkte)

In drahtlosen Multihop-Netzwerken müssen in der Regel Nachrichten über mehrere Stationen geroutet (öfter versendet) werden bis sie ihren Bestimmungsort erreichen. Broadcast Übertragungen (= Senden an alle Nachbarstationen in Reichweite) sind hier eine gängige Kommunikationsmethode. Wenn nun eine Nachbarstation dieselbe Nachricht weiterleitet (erneut aussendet) wird sie von der ursprünglichen Station wieder empfangen. Dies kann zu Übertragungsschleifen führen. Zur Überwindung dieses Problems kann die *Overhearing*-Methode verwendet werden. Dabei wird das mehrfache Aussenden von redundant empfangenen Nachrichten verhindert.

Das folgende Codefragment in Listing 1 beschreibt diesen Algorithmus. Jede Station sendet oder empfängt Broadcast-Nachrichten und speichert lokal pro Nachricht die folgenden Informationen: Status, Timeout und Encounters (= Begegnungen). Der Status einer Nachricht wird auf der Grundlage eines Zustandsdiagramms geändert. Verwenden Sie den Code aus Listing 1, um das lokale Zustandsdiagramm für eine Nachricht auf einer Station zu generieren. Sie können für die Zeichnung die Rückseite dieses Blattes verwenden um nicht ständig umblättern zu müssen.

Tipp: Das Zustandsdiagramm hat neben einem Startzustand ● auch einen Endzustand ⊙ (= die Nachricht wird gelöscht).

Listing 1: Broadcast Routing

```
/* Erzeugt eine lokale message in einem Puffer */
BroadcastPacket createMessage(String data)
{
    message = new BroadcastPacket(data,T); //Objekt erzeugen, Timeout T
    addMessage(message); //im lokalen Puffer speichern
    setStatus(message, "Ready_To_Send"); //setzt den Status
    setEncounters(message, 0); //setzt die Begegnungen zu 0
    return message;
}

/* Neue Informationen sollen versendet werden */
void initTransmission(String data) {
    message = createMessage(data);
}

/* Eine Broadcast-Nachricht wurde empfangen */
void receive(BroadcastPacket message) {
    if (!isAvailable(message)) //falls lokal noch nicht vorhanden
        message = createMessage(message.getData());
    switch (getStatus(message)) { //lokale Statusabfrage
    case "Ready_To_Send":
    case "Overhearing":
        incEncounters(message); //Begegnungen +1
        break;
    case "Idle":
        setStatus(message, "Overhearing");
        break;
    }
    setTimeout(message, T); //setzt erneut den Timeout auf T
}

/* Ein Timer einer Message ist abgelaufen */
void timeout(BroadcastPacket message) {
    switch (getStatus(message)) {
    case "Ready_To_Send":
        if (getEncounters(message) < 2) //Anzahl der Begegnungen
            send(message); //senden der Nachricht
        setStatus(message, "Overhearing");
        break;
    case "Overhearing":
        setStatus(message, "Idle");
        break;
    case "Idle":
        delMessage(message); //entfernt message aus Puffer
        break;
    }
    setTimeout(message, T);
}
```

4 Peer-to-Peer Routing (24 Punkte gesamt)

Peers bilden ein ringförmiges Overlay-Netzwerk. Jeder Peer erhält einen zufälligen aber eindeutigen Bezeichner $p \in [0; 2^m - 1]$ (hier: $m = 5$). Jedem Datenelement wird mittels Hash-Funktion ein Schlüssel k ebenfalls aus diesem Raum zugewiesen. Um den Speicherort eines Elementes mit dem Schlüssel k zu finden muss entlang des Ringes geroutet werden. Das aus der Vorlesung bekannte Chord System verwaltet das Overlay-Netzwerk effizienter indem es sogenannte Sprungtabellen (*Fingertabellen*) bereitstellt.

4.1 Fingertabellen (12 Punkte)

Ein Netzwerk besteht aus 6 Teilnehmern mit den Bezeichnern $p \in [2, 9, 11, 18, 24, 27]$. Füllen Sie unter Verwendung der aus der Vorlesung bekannten Notation und Rechenregeln die Fingertabelle (Tabelle 1) für das Chord System. (Im Gegensatz zu der Vorlesung sind die Tabellen der einzelnen Peers zu einer zusammen gefasst.)

i	FT_2	FT_9	FT_{11}	FT_{18}	FT_{24}	FT_{27}
1						
2						
3						
4						
5						

Tabelle 1: Fingertabelle aller Peers

4.2 Lookup (4 Punkte)

Folgende zwei Lookups werden auf dem Chord System aus Sektion 4.1 durchgeführt. Nennen Sie jeweils den Suchpfad und markieren Sie, durch unterstreichen, den Peer auf dem das gesuchte Datenelement gespeichert ist.

1. Der Startknoten ist Peer 2, der Schlüssel ist $k = 25$:

2 →

2. Der Startknoten ist Peer 11, der Schlüssel ist $k = 1$:

11 →

4.3 Rekonfiguration (8 Punkte)

Zwischen den Peers 2 und 9 aus Sektion 4.1 wird ein neuer Knoten $p = 5$ eingefügt. Vervollständigen Sie die Fingertabelle des neuen Peers und dessen direkten Nachbarn (in Tabelle 2).

i	FT_2	FT_5	FT_9
1			
2			
3			
4			
5			

Tabelle 2: Fingertabelle der Peers 2,5 und 9

Könnten sich dadurch auch Fingertabellen von nicht auf der Ringstruktur direkt benachbarten Peers ändern? Begründen Sie Ihre Antwort kurz.

Zusatzblatt

Zusatzblatt