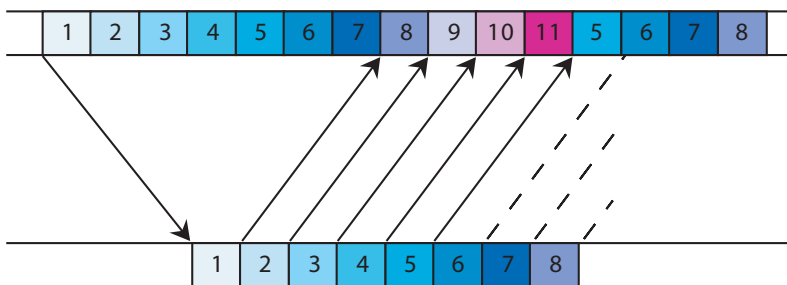


Fachpraktikum

Fehlerkontrolle mittels Automatischer Übertragungswiederholung in Kommunikationsnetzen mit Paketvermittlung

Versuch KT 42

Skizze 31. August 2016



Die theoretischen Fragen im Kapitel 3 (Aufgabe 1-5) müssen vor dem Praktikum gelöst werden. Das Kapitel 4 wird während des Praktikums bearbeitet.

Inhaltsverzeichnis

1	Ziel des Versuchs	1
2	Fehlerkontrolle	2
2.1	Fehlererkennung und Fehlerkorrektur	2
2.2	Einbettung der Fehlerkontrolle im OSI-System	2
2.2.1	Physikalische Schicht	3
2.2.2	Sicherungsschicht	3
2.2.3	Transportschicht	4
2.3	Definitionen	4
2.3.1	Bestätigungsmeldung	4
2.3.2	Umlaufzeit	5
2.3.3	Leistungsfähigkeit eines ARQ Protokolls	5
2.4	Paketfehlerrate	6
2.4.1	Rückkanal fehlerfrei	6
2.4.2	Rückkanal fehlerbehaftet	6
2.5	Durchsatz	7
2.5.1	Stop-and-Wait-ARQ	7
2.5.2	Go-Back-N-ARQ	9
2.5.3	Selective-Repeat-ARQ	10
2.5.4	Vergleich	11
2.6	Anwendung in UMTS (Universal Mobile Telecommunications System)	12
3	Theoretische Aufgaben	13
4	Praktische Aufgaben	14
4.1	Aufgabe 1	14
4.2	Aufgabe 2	16
4.3	Aufgabe 3	16

Kapitel 1

Ziel des Versuchs

Das Ziel des Versuchs liegt im Kennenlernen der Prinzipien der Fehlerkontrolle mittels automatischer Übertragungswiederholung in Kommunikationsnetzen mit Paketvermittlung. Die automatische Übertragungswiederholung (englisch: automatic repeat request (ARQ)) stellt einen Fehlerkontrollmechanismus mit sehr hoher Zuverlässigkeit dar und lässt sich relativ einfach implementieren. Sie kommt daher in zahlreichen Anwendungen wie z.B. dem transfer control protocol (TCP) des Internets zum Einsatz.

In dieser Versuchsanleitung werden zuerst die grundlegenden Techniken von ARQ-Protokollen diskutiert. Es wird gezeigt, wie die Leistungsfähigkeit der Protokolle unter vereinfachenden Annahmen analytisch berechnet werden kann. Es folgt eine kurze Übersicht über den Einsatz von ARQ-Protokollen in modernen Telekommunikationsnetzen. Anschliessend wird der Aufbau einer Simulationsumgebung erläutert, mit deren Hilfe die analytisch berechneten Leistungskenngrößen verifiziert werden können. Zudem können in der Simulation verschiedene Parameter der ARQ-Protokolle variiert werden, was einen vertieften Einblick in die Funktionsweise der Mechanismen erlaubt.

Kapitel 2

Fehlerkontrolle in Kommunikationsnetzen mit Paketvermittlung

2.1 Fehlererkennung und Fehlerkorrektur

Die Methoden zur Fehlerkontrolle in Kommunikationsnetzen mit Paketvermittlung sind vielfältig, haben jedoch alle die Gemeinsamkeit, dass der Sender die Pakete Zusatzinformation (Redundanz) in Form von Codesymbolen versieht, welche es dem Empfänger ermöglichen, Übertragungsfehler festzustellen und allenfalls zu korrigieren. Falls Übertragungsfehler mit Hilfe dieser Codesymbole korrigiert werden können, spricht man von Vorwärtsfehlerkontrolle oder Vorwärtsfehlerkorrektur (forward error correction (FEC)).

Manche Fehlerkontrollmechanismen verfügen zudem über einen Rückkopplungskanal (oder Rückkanal), über welchen der Empfänger die Übertragungswiederholung fehlerhaft empfangener Pakete anfordern kann (Abbildung 2.1). Diese Art der Fehlerkontrolle wird automatische Übertragungswiederholung (ARQ) genannt [1,2].

ARQ tritt in den verschiedensten Aufgabengebieten eines Kommunikationssystems auf. Eine systematische Beschreibung der Verwendung des ARQ-Verfahrens folgt in den nächsten Abschnitten.

2.2 Einbettung der Fehlerkontrolle im OSI-System

Um die Designkomplexität eines Kommunikationsprozesses zu reduzieren, hat sich eine Strukturierung des Kommunikationsprozesses in Form von Schichten durchgesetzt. Dabei stellt jede Schicht ihrer übergeordneten Schicht Dienste zur Verfügung. Eines der meist verwendeter Schichtenmodelle ist das Open Systems Interconnection (OSI) Modell. Das Modell wurde von der International Standards Organization (ISO) im Jahre 1983 als Standard eingeführt [3,4]. Durch die Standardisierung des OSI-Modells wurde der Kommunikationsprozess in sieben verschiedene Schichten aufgeteilt, wie in der Abbildung 2.2 dargestellt.

In diesem Fachpraktikum wird zunächst detailliert auf die zwei untersten Schichten, die Sicherungsschicht und die physikalische Schicht, eingegangen, da sie gute Beispiele für die Anwendung

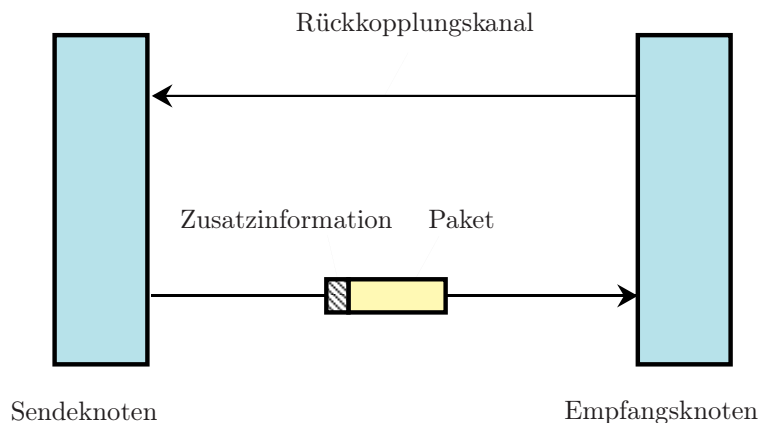


Abbildung 2.1: Übertragung von Zusatzinformation für die Fehlererkennung und -korrektur sowie Rückkopplungskanal für die Anforderung von Übertragungswiederholungen.

der ARQ Protokolle liefern. Weiter gehen wir auf die Transportschicht ein, da ARQ Protokolle ebenfalls auf dieser Ebene implementiert werden können, wie zum Beispiel beim TCP.

2.2.1 Physikalische Schicht

Die Hauptaufgabe der physikalischen Schicht ist die Übertragung der Information über das Medium, wobei nicht nur die Herstellung einer Verbindung zwischen zwei Kommunikationsknoten, sondern auch z.B. Synchronisationsprobleme über den Kanal gelöst werden. Das Medium kann ein typischer Kanal aus Kupferdraht, eine optische Verbindung oder ein kabelloser Kanal sein. Der Aufbau der physikalischen Schicht ist dementsprechend abhängig von der Charakteristik des physikalischen Kanals und sie muss möglichst allen Störungen des Kanals entgegenwirken.

Ein typisches Designziel für die physikalische Schicht ist das Zurverfügungstellen einer möglichst hohen Datenrate über das Medium, wobei gleichzeitig eine möglichst tiefe Fehlerrate angestrebt wird. Dabei hängen die Tiefe der Fehlerrate und die Höhe der Datenrate von der Charakteristik des Kanals und der Anwendung ab.

2.2.2 Sicherungsschicht

Die Sicherungsschicht ist verantwortlich für die Behandlung von Übertragungsfehlern und reguliert zudem den Datenfluss, damit langsame Empfänger nicht mit Daten von schnellen Sendern überflutet werden.

Pakete werden von der Vermittlungsschicht der Sicherungsschicht übergeben, wobei Pakete in Frames zerlegt und für die Übertragung vorbereitet werden. Aufgabe der physikalischen Schicht ist nun die Reduktion der Fehlerrate bei der Übertragung über den Kanal. Da der Fall auftreten kann, dass fehlerbehaftete Pakete den Empfangsknoten trotz Kanalcodierung erreichen, muss ARQ eingesetzt werden. Durch ein ausgeklügeltes Feedbacksystem kann der Empfangsknoten dem Sendeknoten mitteilen, welche Pakete fehlerhaft sind und nochmals gesendet werden müssen. Die automatische Übertragungswiederholung hat somit ihren Ursprung in der Sicherungsschicht.

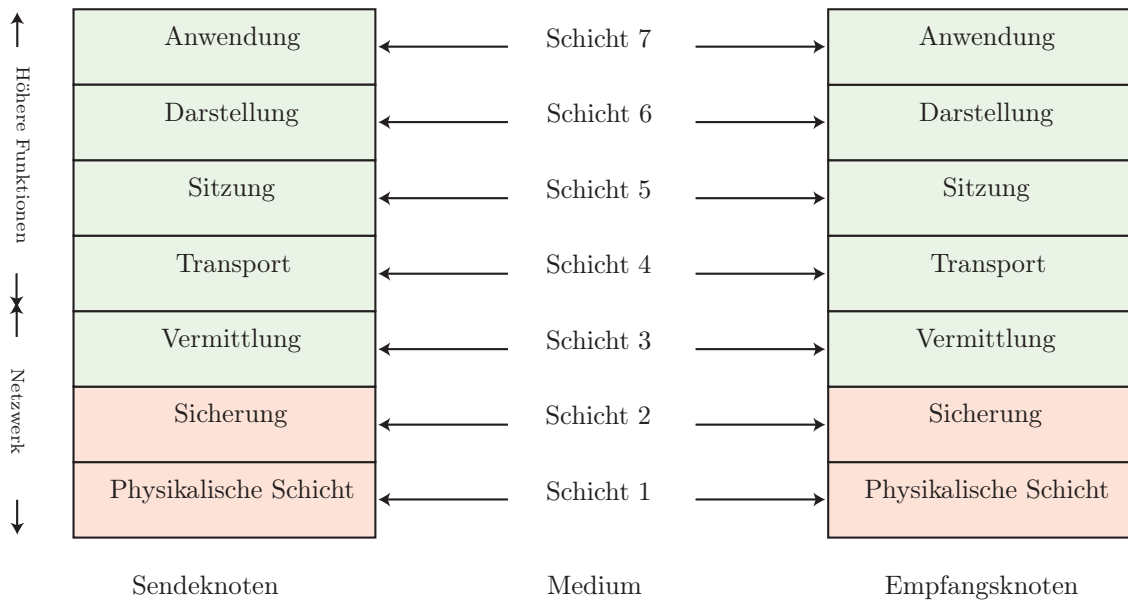


Abbildung 2.2: Die sieben Schichten des OSI-Modells.

2.2.3 Transportschicht

Die Transportschicht kontrolliert und steuert den Auf- und Abbau von virtuellen Kanälen. Virtueller Kanal ist eine Bezeichnung für verschiedene Wege im Netz, die vom Sender zum Empfänger führen. Diese Schicht hat die Möglichkeit, mehrere Übertragungen zum gleichen Endgerät auf einen einzigen virtuellen Kanal zu reduzieren. Andererseits kann sie den Datenverkehr auf mehrere virtuelle Kanäle verteilen, um eine höhere Übertragungsgeschwindigkeit zu erreichen.

Durch ARQ Protokolle können fehlende oder falsche Pakete erneut vom Sender angefordert werden. ARQ-Verfahren sind auch nützlich, um die Datenübertragung über einen neuen virtuellen Kanal wiederaufzunehmen, wenn vorher der bestehende Kanal unterbrochen worden ist (beispielsweise durch einen Netzfehler).

2.3 Definitionen

2.3.1 Bestätigungsmeldung

Wie bereits erwähnt, müssen bei einem System mit automatischer Übertragungswiederholung die Pakete mit einem Code zur Fehlererkennung versehen werden. Zusätzlich erhält jedes Paket eine Sequenznummer (SN). Der Empfänger kann fehlende oder fehlerhaft empfangene Pakete nochmals anfordern, indem er Bestätigungsmeldungen über den Rückkopplungskanal an den Sender zurückschickt. Die Bestätigungsmeldung kann entweder die Sequenznummern der erfolgreich empfangenen Pakete enthalten (positive Bestätigung, Englisch: Acknowledgement (ACK)) oder derjenigen Pakete, welche nochmals übertragen werden müssen (negative Bestätigung, Englisch: Non Acknowledgement (NACK)). Für den logischen Ablauf des ARQ-Protokolls sind positive und negative Bestätigungen äquivalent. In manchen Realisierungen werden Kombinationen von positiven und negativen Bestätigungen eingesetzt. In einem praktischen System ist aber die Zahl der Über-

tragungsfehler typischerweise relativ klein, so dass negative Bestätigungen weniger Verkehr auf dem Rückkopplungskanal erzeugen als positive Bestätigungen.

2.3.2 Umlaufzeit

In ARQ Systemen existieren verschiedene Arten von Verzögerungen. Hier wird nur die durchschnittliche Paketverzögerung betrachtet.

Die Zeit, welche zwischen dem Absenden eines Pakets und dem Eintreffen der entsprechenden Bestätigung beim Sender verstreicht, wird **Umlaufzeit** genannt (englisch: round-trip-delay) und hier mit der Variablen T_{rt} bezeichnet (Abbildung 2.3).

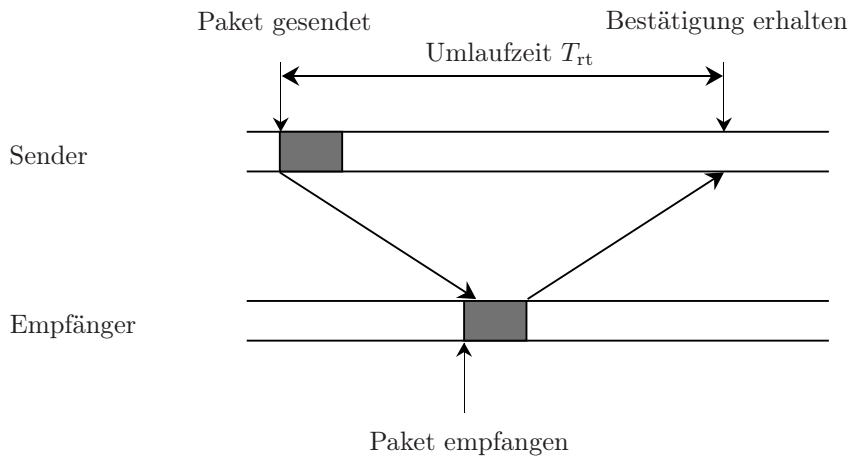


Abbildung 2.3: Umlaufzeit T_{rt} , welche zwischen dem Senden eines Pakets und dem Eintreffen der entsprechenden Bestätigung verstreicht.

Die Umlaufzeit enthält die beiden Ausbreitungsverzögerung (Englisch: propagation delay) $T_{prop}^{TX \rightarrow RX}$ (vom Sender zum Empfänger) und $T_{prop}^{RX \rightarrow TX}$ (vom Empfänger zum Sender), die Dauer der Bestätigung T_{ACK} und ebenfalls die beiden Verarbeitungsverzögerungen T_{prc}^{RX} und T_{prc}^{TX} :

$$T_{rt} = T_{prop}^{TX \rightarrow RX} + T_{prop}^{RX \rightarrow TX} + T_{ACK} + T_{prc}^{TX} + T_{prc}^{RX} \quad (2.1)$$

Zur Vereinfachung nehmen wir diese Verzögerungen für den weiteren Verlauf dieses Praktikums als konstant an. Die Umlaufzeit hat einen grossen Einfluss auf die Wahl des ARQ-Protokolls und bestimmt dessen Leistungskenngrössen massgebend.

2.3.3 Leistungsfähigkeit eines ARQ Protokolls

Die Leistungsfähigkeit eines ARQ-Protokolls wird durch die Zuverlässigkeit (Paketfehlerrate) und Effizienz (Durchsatz) bestimmt.

Paketfehlerrate

Die Zuverlässigkeit kann mit Hilfe der Rate $P(E)$ der irrtümlich akzeptierten Pakete ausgedrückt werden.

Die Paketfehlerrate $P(E)$ ist der Anteil der irrtümlich vom Empfänger als korrekt eingestuften Pakete bezogen auf die Gesamtzahl von Paketen [3].

Durchsatz

Die Effizienz von ARQ-Protokollen lässt sich anhand des Durchsatzes η vergleichen, welcher definiert ist als

$$\eta = \frac{1}{\mathbb{E}[N_{\text{slot}}]}, \quad (2.2)$$

wobei N_{slot} die Anzahl Zeitschlitze bezeichnet, welche für die erfolgreiche Übertragung eines Pakets gebraucht werden. Der Durchsatz bezeichnet somit das Verhältnis der effektiv genutzten zur aufgewendeten Übertragungskapazität. Alternativ findet sich in [3] folgende Definition des Durchsatzes η :

Der **Durchsatz** η eines ARQ-Systems ist die Anzahl von Informationsbits, die in einer gewissen Zeit übertragen und vom Empfänger akzeptiert wurden, bezogen auf die Anzahl der Bits, die in dieser Zeit insgesamt über den Kanal hätten übertragen werden können.

2.4 Paketfehlerrate

2.4.1 Rückkanal fehlerfrei

Die Paketfehlerrate $P(E)$ ergibt sich als die Summe der Wahrscheinlichkeiten für die einzelnen möglichen Fehlerereignisse, d.h. ein Paket kann bei der ersten Übertragung irrtümlich akzeptiert werden oder nach einer erneuten Anforderung. Ein Paket ist entweder (i) korrekt (mit Wahrscheinlichkeit P_c) oder (ii) weist einen oder mehrere Fehler auf. Im Fall (ii) enthält das Paket mit Wahrscheinlichkeit P_r ein detektierbares Fehlermuster (wird daher neu angefordert), oder mit Wahrscheinlichkeit P_e ein nicht detektierbares (wird also fälschlicherweise akzeptiert). Daher gilt für ein übertragenes Paket: $P_c + P_r + P_e = 1$.

Für einen fehlerfreien Rückkanal erhält man (vgl. Theoretische Aufgabe T1):

$$P(E) = \frac{P_e}{1 - P_r}. \quad (2.3)$$

Unter der Voraussetzung, dass sämtliche Fehlermuster vom Fehlererkennungscode erfasst werden, garantiert ein ARQ-Protokoll absolut fehlerfreie Übertragung ($P_e = 0$ und damit $P(E) = 0$).

Der Nachteil der ARQ-basierten Fehlerkontrolle liegt darin begründet, dass der Durchsatz nicht konstant ist, je nach Übertragungsqualität die Verzögerungszeiten und Pufferlängen sehr hohe Werte annehmen können.

2.4.2 Rückkanal fehlerbehaftet

Da der Rückkanal ebenfalls fehlerbehaftet sein und somit ein ACK in ein NACK verfälscht werden kann und umgekehrt [5], untersuchen wir die Fehlerwahrscheinlichkeit für diesen allgemeinen Fall. Wir gehen hier davon aus, dass ein ACK oder NACK durch ein binäres Flag dargestellt werden.

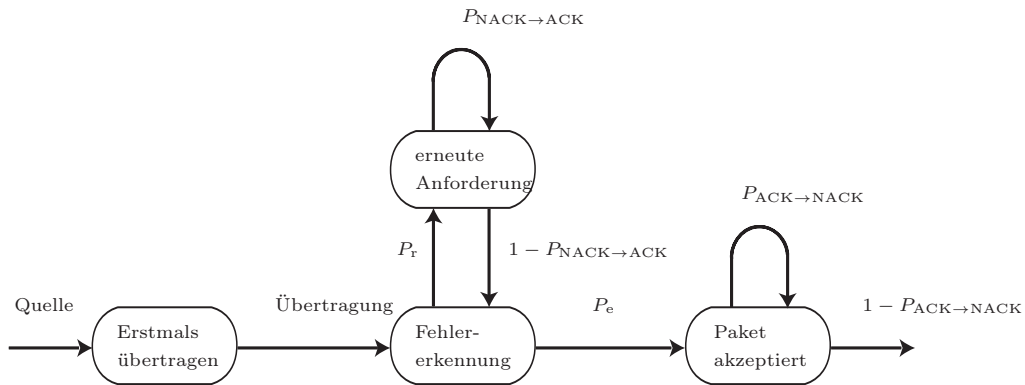


Abbildung 2.4: Zustandsdiagramm eines ARQ-Systems im Falle eines fehlerbehafteten Rückkanals und einen undetektierbaren Fehlermuster am Ausgang [3].

Nach der erstmaligen Übertragung des Pakets tritt mit der Wahrscheinlichkeit P_r der Fall ein, dass das Paket ein detektierbares Fehlermuster enthält und demzufolge neu angefordert wird. Diese erneute Anforderung (NACK) gelangt entweder fehlerfrei zum Sender oder wird mit Wahrscheinlichkeit $P_{NACK \rightarrow ACK}$ in ein ACK verfälscht. Die allgemeine Übertragungsfunktion des Zustandsdiagramms lautet [3]:

$$P(E) = \frac{P_e(1 - P_{ACK \rightarrow NACK})(1 - P_{NACK \rightarrow ACK})}{(1 - P_{NACK \rightarrow ACK} - P_r(1 - P_{ACK \rightarrow NACK}))(1 - P_{ACK \rightarrow NACK})} = \frac{P_e}{1 - P_r} \quad (2.4)$$

wobei $P_{ACK \rightarrow NACK} = P_{NACK \rightarrow ACK}$ angenommen wird. Somit sehen wir, dass die Paketfehlerrate nicht von der Qualität des Rückkanals abhängt, und auch nicht von dem ARQ Protokoll im Einsatz. Diese Unabhängigkeiten gelten jedoch nicht für den Durchsatz.

2.5 Durchsatz

In der Folge werden drei grundsätzliche ARQ-Verfahren aufgrund ihres Durchsatzes diskutiert. Es handelt sich dabei um Stop-and-Wait-ARQ (SW-ARQ), Go-Back-N-ARQ (GBN-ARQ) und Selective-Repeat-ARQ (SR-ARQ).

2.5.1 Stop-and-Wait-ARQ

Das Stop-and-Wait (SW)-ARQ-Protokoll stellt das einfachste ARQ-Verfahren dar. Der Sender schickt ein Paket an den Empfänger und wartet dann auf eine Bestätigung. Während dieser Wartezeit werden keine weiteren Pakete gesendet. Im Falle einer erfolgreichen Übertragung erhält der Sender nach der Umlaufzeit T_{rt} eine positive Bestätigung (ACK), worauf er das entsprechende Paket im Sendepuffer löscht und das nächste Paket schickt (Abbildung 2.5). Falls die Übertragung fehlerhaft war, erhält der Sender eine negative Bestätigung (NACK) und sendet darauf das gleiche Paket nochmals.

In einem realen System kann auch der Rückkopplungskanal Übertragungsfehler aufweisen. Um eine Blockierung des Protokollablaufes zu vermeiden, muss der Sender nach Ablauf einer Auszeit T_0 (englisch: Time-Out) die Übertragungswiederholung einleiten, auch wenn bis dahin keine gültige

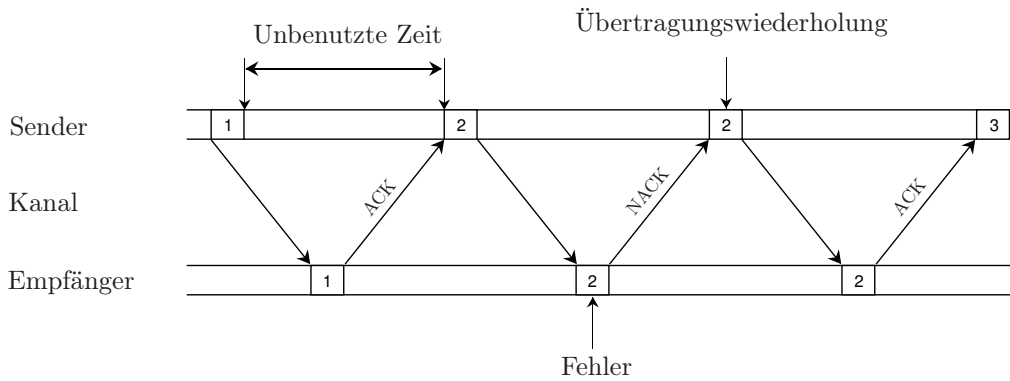


Abbildung 2.5: Ablauf des SW-ARQ-Protokolls

Bestätigungsmeldung angekommen ist. Die Dauer der Auszeit T_0 muss mindestens so lang sein wie die Umlaufzeit T_{rt} .

Bei Systemen mit langer Umlaufzeit T_{rt} entstehen relativ grosse Zeitanteile, in denen die Übertragungsstrecke unbenutzt ist, weil der Sender auf die Bestätigung des zuletzt übertragenen Pakets wartet.

Unter idealisierenden Annahmen lässt sich der Durchsatz eines ARQ-Protokolls analytisch berechnen [6]. Dazu wird ein fehlerfreier Rückkopplungskanal ($P_{ACK \rightarrow NACK} = P_{NACK \rightarrow ACK} = 0$) vorausgesetzt. Ferner basieren die Berechnungen auf einem zeitinvarianten Kanalmodell mit konstanter Wahrscheinlichkeit P_r für eine Wiederholübertragung (im folgenden als Paketfehlerwahrscheinlichkeit P_r bezeichnet).

Die Pakete haben die konstante zeitliche Dauer T_p . Wir nehmen an, dass $T_{rt} = N \cdot T_p$, wobei $N \in \mathbb{N}^*$. Die Anzahl Pakete N , welche während der Umlaufzeit T_{rt} übertragen werden können, beträgt somit

$$N = T_{rt}/T_p. \quad (2.5)$$

Die Umlaufzeit wird häufig auch durch die Anzahl Pakete N ausgedrückt, welche während der Zeit T_{rt} übertragen werden können. Bei einer Paketfehlerwahrscheinlichkeit P_r beträgt die Wahrscheinlichkeit, dass ein Paket k -mal übertragen werden muss: $P_r^{k-1} \cdot (1 - P_r)$, ($k = 1, 2, \dots$). Beim Einsatz von SW-ARQ werden pro Übertragungsversuch N Zeitschlitze gebraucht. Bei k Übertragungsversuchen werden deshalb $N_{slot} = k \cdot N$ Zeitschlitze gebraucht. Mit (2.2) ergibt sich folgender Durchsatz für SW-ARQ (vgl. Theoretische Aufgabe T2):

$$\eta = \frac{1 - P_r}{N}. \quad (2.6)$$

Aus (2.6) geht hervor, dass bei SW-ARQ der Durchsatz umgekehrt proportional ist zur Variablen N bzw. zur Umlaufzeit T_{rt} . Dieses Verfahren ist deshalb nicht geeignet für Systeme mit grosser Umlaufzeit wie z.B. Satellitensysteme. Der Durchsatz ist klein, falls die Umlaufzeit $T_{rt} \gg T_p$ gross ist.

Der Vorteil von SW-ARQ ist seine Einfachheit.

Im Falle eines fehlerhaften Rückkanals, wobei ($P_{ACK \rightarrow NACK} = P_{NACK \rightarrow ACK} = P_{Rückkanal}$) angenommen wurde, schreiben wir [3]:

$$\eta_{SW} = (1 - P_{Rückkanal}) \cdot \frac{1 - P_r}{N}. \quad (2.7)$$

2.5.2 Go-Back-N-ARQ

Der Durchsatz η von SW-ARQ ist deshalb so gering, weil der Übertragungskanal während der Umlaufzeit T_{rt} unbenutzt bleibt. Bei GBN-ARQ wird die Umlaufzeit genutzt, indem der Sender bis zu N Pakete übertragen kann, ohne eine Bestätigung für das erste Paket erhalten zu haben. Falls er nach Ablauf der Umlaufzeit eine positive Bestätigung (ACK) für das zuerst gesendete Paket erhält, kann die Übertragung ohne Unterbruch fortgesetzt werden (Abbildung 2.6). Dadurch gibt es keine unbenutzten Zeitabschnitte mehr wie beim SW-ARQ. Erhält der Empfänger ein fehlerhaftes Paket, verwirft er dieses und alle nachfolgenden. Eine negative Bestätigung (NACK) wird so lange an den Sender zurückgeschickt, bis das gewünschte Paket angekommen ist. Erhält der Sender eine negative Bestätigung, so werden alle N Pakete ab dem ersten unbestätigten nochmals übertragen.

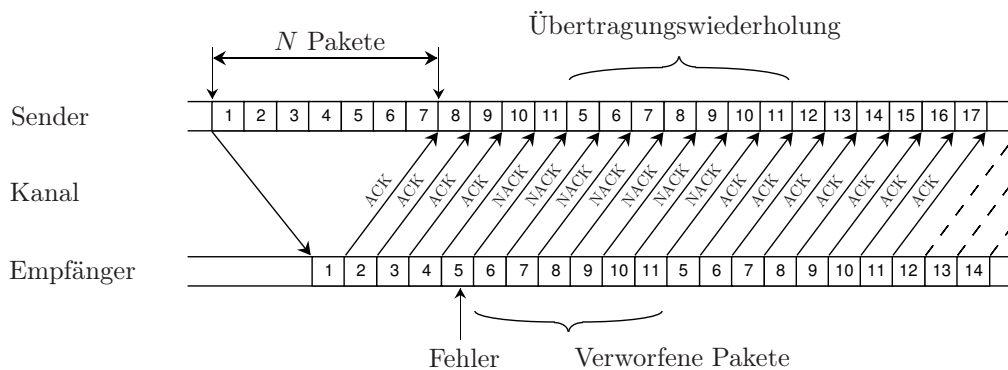


Abbildung 2.6: Ablauf des GBN-ARQ-Protokolls (from [3]).

Die Umlaufzeit T_{rt} definiert ein Übertragungsfenster, welches diejenigen Pakete enthält, die aufgrund der zuletzt erhaltenen Bestätigung gesendet werden dürfen. Abbildung 2.7 veranschaulicht das Übertragungsfenster.

Die Pakete müssen durch Sequenznummern SN beim Go-Back-N Protokoll gekennzeichnet werden, damit sie voneinander unterschieden werden. Bei SW-ARQ war eine Markierung nicht notwendig, da die Pakete nacheinander bestätigt werden und ihre Reihenfolge aufgrund des natürlichen und einfachen Aufbaus des Protokolls beibehalten.

Das erste noch nicht bestätigte Paket bestimmt den Anfang des Fensters. Falls zum Beispiel das Paket mit der Sequenznummer 0 das zuletzt bestätigte Paket darstellt (Zeiger ACK in Abbildung 2.7 a), so beginnt das Übertragungsfenster beim Paket mit Sequenznummer 1. Der mit TX bezeichnete Zeiger verweist auf das nächste zu übertragende Paket. Das Fenster hat die Länge N . Der Sender kann somit die nächsten N Pakete ab dem ersten unbestätigten Paket übertragen, ohne eine weitere Bestätigung abzuwarten. Sobald eine neue Bestätigungsmeldung beim Sender eintrifft (Abbildung 2.7 b), werden der ACK-Zeiger und mit ihm das Übertragungsfenster entsprechend verschoben.

Abbildung 2.7 c zeigt den Fall, dass der TX-Zeiger das Ende des Übertragungsfensters erreicht hat. Dies kann die Folge eines Übertragungsfehlers im ersten Paket des Fensters sein. In diesem Fall schickt der Empfänger wiederholt dieselbe Bestätigung, d.h. der ACK-Zeiger und das Fenster können nicht verschoben werden. Die in Abbildung 2.7 c dargestellte Situation kann auch eintreten, falls alle Pakete korrekt übertragen worden sind, die Bestätigungsmeldungen infolge von Übertragungsfehlern im Rückkopplungskanal aber nicht zum Sender gelangen. Der Sender unterbricht

nun die Übertragung und der TX-Zeiger wird an den Anfang des Fensters zurückgesetzt. Die Übertragungswiederholung beginnt dann mit dem ersten unbestätigten Paket.

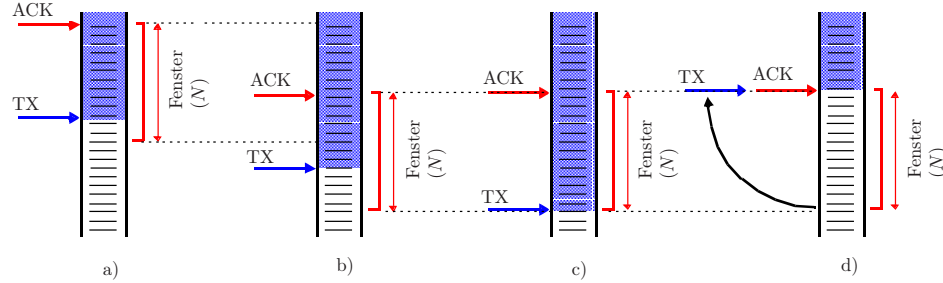


Abbildung 2.7: Übertragungsfenster des GBN-ARQ-Protokolls.

Wie bereits bei SW-ARQ lässt sich auch bei GBN-ARQ unter vereinfachenden Annahmen der Durchsatz analytisch berechnen. Auch hier beträgt die Wahrscheinlichkeit, dass ein Paket genau k -mal übertragen werden muss, $P_r^{k-1} \cdot (1 - P_r)$ ($k = 1, 2, \dots$). Im Falle von SW-ARQ beanspruchte jeder Übertragungsversuch N Zeitschlitze. Bei GBN-ARQ hingegen müssen pro Übertragungswiederholung N Zeitschlitze aufgewendet werden. Der letzte, schliesslich erfolgreiche Übertragungsversuch braucht aber nur noch einen Zeitschlitz, weil danach bereits mit der Übertragung neuer Pakete fortgefahren werden kann.

Bei k Übertragungsversuchen werden deshalb $N_{\text{slot}} = (k - 1) \cdot N + 1$ Zeitschlitze gebraucht. Mit (2.2) ergibt sich folgender Durchsatz für GBN-ARQ (vgl. Theoretische Aufgabe T2):

$$\eta = \frac{1 - P_r}{1 + P_r \cdot (N - 1)}. \quad (2.8)$$

Bei SW-ARQ ist der Durchsatz gemäss (2.6) umgekehrt proportional zur durch N ausgedrückten Umlaufzeit. Bei GBN-ARQ wird die Variable N gemäss (2.8) mit P_r gewichtet. Dies bedeutet, dass die Umlaufzeit bei kleinen Paketfehlerwahrscheinlichkeiten P_r praktisch keinen Einfluss auf den Durchsatz hat. Dies geht auch aus der Beschreibung des Protokollablaufes hervor, bei der festgestellt wurde, dass der Sender im Falle fehlerfreier Übertragung ohne Unterbruch fortfahren kann. Lediglich bei einer Übertragungswiederholung müssen sämtliche während einer Umlaufzeit gesendeten Pakete nochmals übertragen werden.

Im Falle eines fehlerhaften Rückkanals, wobei ($P_{\text{ACK} \rightarrow \text{NACK}} = P_{\text{NACK} \rightarrow \text{ACK}} = P_{\text{Rückkanal}}$) angenommen wurde, schreiben wir [3]:

$$\eta_{\text{GBN}} = \frac{(1 - P_{\text{Rückkanal}})^2 (1 - P_r)}{1 - P_{\text{Rückkanal}} + (N - 1)(P_{\text{Rückkanal}} + P_r - P_{\text{Rückkanal}}^2 - 3P_r P_{\text{Rückkanal}} + 2P_r P_{\text{Rückkanal}}^2)}. \quad (2.9)$$

2.5.3 Selective-Repeat-ARQ

Der Einfluss der Umlaufzeit T_t auf den Durchsatz η ist bei GBN-ARQ deutlich kleiner als bei SW-ARQ. Aus (2.8) ist jedoch ersichtlich, dass die Abhängigkeit des Durchsatzes von der Umlaufzeit mit steigender Paketfehlerwahrscheinlichkeit P_r zunimmt. Dies liegt darin begründet, dass im Falle

eines Übertragungsfehlers sämtliche während einer Umlaufzeit gesendeten Pakete nochmals übertragen werden müssen, auch wenn ihre Erstübertragung fehlerfrei war.

Bei SR-ARQ hingegen legt der Empfänger alle erfolgreich erhaltenen Pakete in einem Empfangspuffer ab. Falls einzelne Pakete fehlerhaft empfangen worden sind, fordert er diese selektiv nochmals beim Sender an, indem er für jedes fehlende Paket eine negative Bestätigung (NACK) zurückschickt (Abbildung 2.8). Sobald diese Pakete nach wiederholter Übertragung erfolgreich empfangen worden sind, fügt sie der Empfänger an den entsprechenden Stellen in den Empfangspuffer ein. Es ist sofort ersichtlich, dass bei SR-ARQ wirklich nur diejenigen Pakete nochmals übertragen werden, welche bei ihrer Erstübertragung einen Fehler aufwiesen.

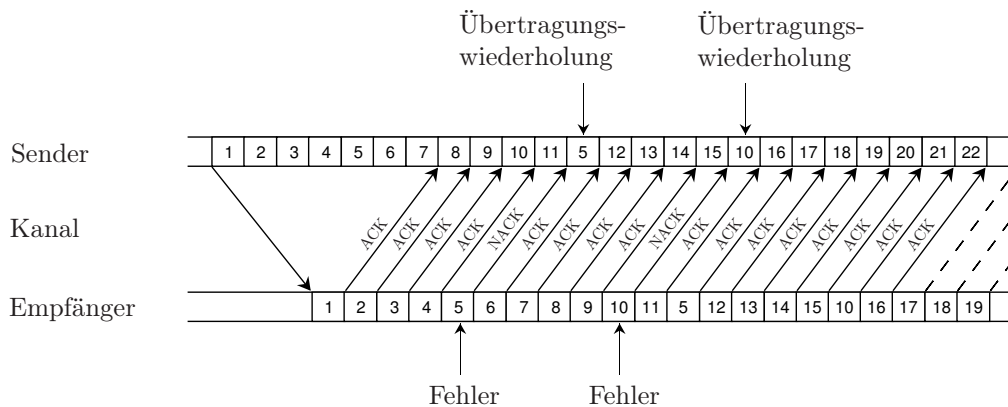


Abbildung 2.8: Ablauf des SR-ARQ-Protokolls.

Die analytische Berechnung des Durchsatzes von SR-ARQ gestaltet sich wesentlich einfacher als bei SW- und GBN-ARQ, weil die Umlaufzeit T_{rt} nicht mehr in Erscheinung tritt. Jeder Übertragungsversuch benötigt genau einen Zeitschlitz.

Der Durchsatz von SR-ARQ wird dann nach (2.2):

$$\eta = 1 - P_r. \quad (2.10)$$

Im Falle eines fehlerbehafteten Rückkanals ($(P_{ACK \rightarrow NACK} = P_{NACK \rightarrow ACK} = P_{Rückkanal})$) schreiben wir für den Durchsatz [3]:

$$\eta_{SR} = \frac{(1 - P_r) \cdot (1 - P_{ACK \rightarrow NACK})}{1 - P_r \cdot P_{ACK \rightarrow NACK}}. \quad (2.11)$$

2.5.4 Vergleich

Die gemäss (2.6), (2.8) und (2.10) berechneten Werte für den Durchsatz von SW-, GBN- und SR-ARQ sind in Abbildung 2.9 in Abhängigkeit der Paketfehlerwahrscheinlichkeit P_e dargestellt. Für die durch die Variable N ausgedrückte Umlaufzeit wurde dabei der Wert $N = 20$ gewählt.

Aus Abbildung 2.9 geht hervor, dass der Durchsatz von SW-ARQ deutlich kleiner ist als derjenige von GBN- und SR-ARQ. Ausserdem ist ersichtlich, dass SR-ARQ für Paketfehlerwahrscheinlichkeiten P_r welche den Wert 10^{-3} überschreiten, einen wesentlich höheren Durchsatz aufweist als GBN-ARQ. Gerade in drahtlosen Netzen (z.B. Mobilfunknetze) liegt die Paketfehlerwahrscheinlichkeit typischerweise etwa in diesem Bereich. Bei solchen Anwendungen kommen deshalb vorzugsweise SR-ARQ-Protokolle zum Einsatz. Die Implementierung eines SR-ARQ-Protokolls führt

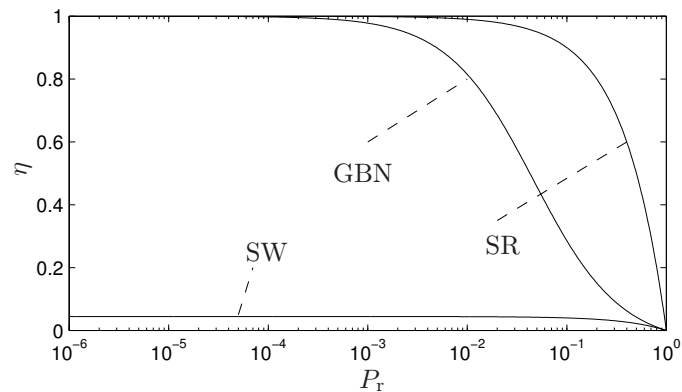


Abbildung 2.9: Durchsatz in Abhängigkeit der Paketfehlerwahrscheinlichkeit P_r für SW-, GBN- und SR-ARQ.

aber zu deutlich grösseren Kosten infolge der erhöhten Komplexität, welche aus der benötigten Zwischenspeicherung und der Sortierung der Pakete im Empfänger resultiert.

2.6 Anwendung in UMTS (Universal Mobile Telecommunications System)

ARQ Protokolle treten auf drei verschiedenen Schichten des OSI Systems vom globalen Universal Mobile Telecommunications System (UMTS) auf, nämlich der physikalischen Schicht, der Radio Link Control (RLC) Schicht (Teil der Sicherungsschicht) und der Transportschicht:

- In der physikalischen Schicht wird neben einem ARQ Protokoll ein Kanalkodierer eingesetzt (hybrid ARQ),
- Auf der Transportschicht setzt TCP Selective-Repeat (SR)-ARQ ein (Abschnitt 2.5.3),
- Die RLC Schicht setzt Go-Back-N ein, um die Datenübertragung zu garantieren.

Kapitel 3

Theoretische Aufgaben

- T1. Beweisen Sie den Ausdruck für die Paketfehlerrate $P(E)$ im Falle eines fehlerfreien Rückkanals, wie in Gleichung (2.3) gegeben. (Annahme: Verlustloses Netz, jedes Paket kann unendlich oft wiederholt werden)
- T2. Beweisen Sie den Ausdruck für den Durchsatz vom SW-ARQ (Gleichung (2.6)), GBN-ARQ (Gleichung (2.8)) und SR-ARQ (Gleichung (2.10)).

Wir betrachten einen Datenlink via Satellit zwischen zwei Bodenstationen mit hohem Durchsatz (Distanz Bodenstationen zum Satelliten: 37600km). Wenn wir die Dauer von ACK/NACK, sowie die Verarbeitungszeiten vernachlässigen, ist die Umlaufzeit 4 mal so gross wie die Übertragungszeit vom Sender zum Satelliten. Die maximale Symbolrate soll $42.2 \cdot 10^6$ QPSK Symbole pro Sekunde sein.

- T3. Geben Sie die Umlaufzeit T_{rt} für das System an.
- T4. Geben Sie die Paketdauer bei einer konstanten Packetlänge von 204 byte an.
- T5. Geben Sie die Anzahl der Pakete pro Umlauf an.

Kapitel 4

Praktische Aufgaben

Ihre Tardis Logindaten werden für das Einloggen auf dem PC benötigt. Speichern Sie eine Kopie der Dateien auf Ihrem Tardis-Account und verwenden Sie diese für die anstehenden Aufgaben.

```
...> tar -xvf /home/iktprak/KT42/KT42.tar
...> cd KT42
```

Starten Sie MATLAB™ und dann die Simulationsumgebung `arq_sim`.

```
...> matlab &
...> arq_sim
```

4.1 Aufgabe 1

Starten Sie **Aufgabe 1** im Menu **File**. Diese Aufgabe führt Sie durch die verschiedene Blöcke eines Übertragungssystems. Ihre Aufgabe wird darin bestehen den MATLAB™ Code der verschiedenen Kommunikationsblöcke zu ergänzen. Das System besteht aus einem Sender, einem Kanal und einem Empfänger, welche jeweils in einem separaten Panel im MATLAB™Fenster dargestellt sind. Im **Sender** Panel können Sie das Bit-Muster wählen, welches übertragen werden soll. Der Sender betrachtet das von Ihnen ausgewählte Muster als eine Sequenz von Bits, die nacheinander mit einem BPSK oder QPSK Alphabet (Abbildung 4.1) übertragen werden. Das Alphabet (oder die Modulationsart) wird im **Eigenschaften** Panel ausgewählt. Die generierten Symbole werden dann über einen Kanal übertragen. Es sind zwei Kanalmodelle vorhanden: Das Additive White Gaussian Noise (AWGN) Kanalmodell und das (fast) fading Kanalmodell. Das Kanalmodell kann im **Kanal**-Panel im Feld **Charakteristik** ausgewählt werden. Wenn x das Input Symbol (BPSK oder QPSK), h den Kanal, n das Rauschen am Empfänger und y das Empfangssignal darstellt, können wir die Übertragung wie folgt modellieren:

$$y = h \cdot x + n$$

Die Energie von x ist im Fall eines BPSK oder QPSK Alphabets mit $E_s = |x|^2$ gegeben, weil alle Symbole die gleiche Energie aufweisen. Wir verwenden eine per-Bit Energie Normierung, so dass die Energie per Bit durch E_b gegeben ist. Somit erhalten wir für BPSK Symbole $E_s = E_b$.

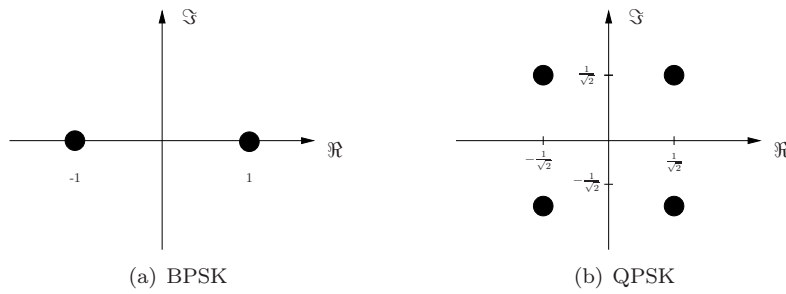


Abbildung 4.1: Zur Verfügung stehende Symbolalphabete

Für die QPSK Symbole gilt $E_s = 2E_b$, weil ein Symbol zwei Bits enthält.

Im Falle eines AWGN Kanalmodells wird $h = 1$ angenommen. Für den Fall des fast fading Kanalmodells, wird für h die Realisierung eines komplexen Gauss-Prozesses mit Mittelwert 0 und Varianz 1 genommen. (Notation: $\mathcal{CN}(0, 1)$). Dabei beträgt die Varianz $1/2$ für den Realteil und $1/2$ für den Imaginärteil. Der Wert von h ändert sich bei jedem Symbol (fast fading), aber die Verteilung bleibt unverändert über die Zeit (Stationarität).

Das Rauschen n ist die Realisierung eines komplexen normalen Prozesses mit Mittelwert 0 und Varianz $\frac{N_0}{2}$ für den realen und imaginären Teil. Sie können das signal to noise ratio (SNR) vom System durch den Parameter $\frac{E_b}{N_0}$ im Panel **Eigenschaften** bestimmen.

Der Empfänger muss aus y den Wert von x schätzen, dabei basiert seine Entscheidung auf einer in unserem Simulationsmodell (zu jeder Zeit) als perfekt angenommenen Kanalkennntnis. Das bedeutet, dass der Empfänger die aktuelle Realisierung vom Kanal h kennt. Er schätzt das Sendesymbol mit einem Maximum Likelihood Entscheider und kann als optimaler Empfänger angesehen werden. Die für diese Schätzung verwendeten Empfangssignale werden in einem separaten Fenster angezeigt, dabei wird der Einfluss des Kanals berücksichtigt. Vom geschätzten Symbol des Empfängers wird eine Schätzung für die übertragenen Bits gemacht. Im Empfänger-Panel sehen Sie das geschätzte Muster, nachdem Sie den Knopf **transmit** gedrückt haben, der Sender die 100 Bits aus dem Sendemuster übertragen hat. Die Bitfehlerwahrscheinlichkeit findet sich im Panel **Eigenschaften** und kann mit einem Klick auf den Knopf **Plot** auf einem BER/SNR Diagramm (SNR in $\frac{E_b}{N_0}$ gemessen) zusammen mit einer Referenzkurve, die die theoretische Bitfehlerwahrscheinlichkeit angibt, wenn die Sendebits gleichwahrscheinlich sind (was hier nicht für alle möglichen Muster der Fall ist), dargestellt werden.

- P1. Schauen Sie sich die Funktion `modulator` im Ordner `transmitter` an. Ergänzen Sie die Subfunktion `map_bit2symbol`. Die Definitionen der Variablen und der Prototyp der Funktion befinden sich im Ordner `students`.
- P2. Schauen Sie sich die Funktion `receiver` im Ordner `receiver` an. Ergänzen Sie die Subfunktionen `maximum_likelihood_decoder` und `map_symbol2bit`. Die Definitionen der Variablen und der Prototyp der Funktion befinden sich im Ordner `students`.

Jetzt ist die Simulationsumgebung fertig. Simulieren Sie verschiedene Übertragungen mit Werten für $\frac{E_b}{N_0}$ zwischen -10 dB und $+4$ dB.

- P3. Warum sind Werte für $\frac{E_b}{N_0}$, die grösser als $+4$ dB sind, bei einem AWGN Kanalmodell, nicht mehr sinnvoll? was sollte man ändern?

- P4. Sehen Sie einen Unterschied zwischen BPSK und QPSK? Falls ja, weshalb?
- P5. Vergleichen Sie für eine gegebene Modulationsart die Steigung der Kurve eines AWGN-Kanalmodells mit der eines fading Kanalmodells. Was für eine Steigung würden Sie erwarten, falls Sie die Simulation mit höheren $\frac{E_b}{N_0}$ Werten laufen lassen würden?

4.2 Aufgabe 2

Unter dem Menu **File** starten Sie jetzt **Aufgabe 2**.

In dieser Aufgabe wird ein Stop-and-Wait-ARQ Protokoll betrachtet. Die Panel sind die gleichen wie bei **Aufgabe 1**. Wenn Sie auf den Knopf **Start** drücken wird die Übertragung des im Panel **Sender** gewählten Musters gestartet. Der Sender überträgt ein Symbol und wartet mit dem nächsten Symbol bis er vom Empfänger den Index des ersten Bits des nächsten Symbols bekommt, das übertragen werden muss. Falls das Symbol vom Empfänger falsch detektiert wurde, erscheint ein rotes Feld 'Repeat Request'. In diesem Fall gibt der Empfänger nochmals den gleichen Index zurück und der Sender schickt das Symbol erneut durch den Kanal. Nach der Übertragung des gesamten Musters werden im Panel **Eigenschaften** die Bit- und Symbolfehlerwahrscheinlichkeit angezeigt.

- P6. Ergänzen Sie die Matlab Funktion `stopandwait_decision` damit ihre Ausgangsvariable der Index des ersten Bits des nächsten Symbols ist, das übertragen wird. Die Definitionen der Variablen und der Prototyp der Funktion befinden sich im Ordner **students**.
- P7. Wie ist die Bitfehlerrate bei Stop-and-Wait, wenn für die Übertragung des Musters unendlich viel Zeit zur Verfügung steht?
- P8. Simulieren Sie verschiedene Übertragungen. Wie ist die Bit- und Symbolfehlerrate für das (fast) flat fading Kanalmodell? Wie sind die Fehlerraten für das AWGN Kanalmodell? Stimmt das mit der **Aufgabe 1** überein? Erläutern Sie den Kompromis zwischen Zuverlässigkeit und Effizienz des Protokolls. Variieren Sie auch $\frac{E_b}{N_0}$.
- P9. Simulieren Sie verschiedene Übertragungen mit BPSK und QPSK. Welche Modulation zeigt die beste Rate? Hierzu nehmen Sie an, dass die Übertragung eines BPSK-Symbols gleich lang dauert wie die eines QPSK-Symbols. Variieren Sie auch $\frac{E_b}{N_0}$.

4.3 Aufgabe 3

Unter dem Menu **File** starten Sie jetzt **Aufgabe 3**.

Die Aufgabe 3 simuliert den Durchsatz von den SW, Go-Back-N (GBN) und SR-ARQ Protokollen. Das Protokoll wird im Panel **Eigenschaften** ausgewählt. In dieser Aufgabe wird das AWGN Kanalmodell verwendet und der Rückkanal ist wie vorher fehlerfrei. Mit dem Parameter **Runs** bestimmen Sie wie viele Symbole (BPSK oder QPSK) in der Simulation übertragen werden. Bei allen drei Protokollen können Sie in der Konsole **Eigenschaften** die Umlaufzeit bestimmen. Beim GBN Protokoll können Sie dazu die Fensterlänge bestimmen, und beim SR Protokoll die Länge des Sendebuffers. Die Umlaufzeit vereinfacht sich zu einer reinen Ausbreitungsverzögerung (die Verarbeitungszeit wird vernachlässigt) und ist ein Vielfaches der Symboldauer. Da die Ausbreitungsverzögerung von Sender zum Empfänger und vom Empfänger zum Sender gleich sind, muss die Umlaufzeit eine gerade Zahl sein. Bei GBN ist momentan die Simulationsumgebung beschränkt auf den Fall, in dem die Umlaufzeit grösser oder gleich ist wie die Fensterlänge.

- P10. (SW) Begründen Sie den Wert für den Durchsatz, wenn P_r sehr klein wird.
- P11. (GBN) Simulieren Sie den Fall, wo P_r klein wird, und die Umlaufzeit grösser ($>$) als die Fensterlänge ist. Begründen Sie das Durchsatzverhältnis zwischen der theoretischen Kurve (die Umlaufzeit ist gleich der Fensterlänge) und der simulierten Kurve.
- P12. (SR) Was ist der Einfluss der Bufferlänge auf dem Durchsatz?

Literaturverzeichnis

- [1] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition, 1992.
- [2] A. S. Tanenbaum. *Computer Networks*. Prentice-Hall, Englewood Cliffs, NJ, 1989.
- [3] M. Bossert and M. Breitbach. *Digitale Netze*. B.G. Teubner, Stuttgart, 1999.
- [4] Michael Philipp Schmitt. *ARQ Systems for Wireless Communications*. PhD thesis, Universität Darmstadt, Buffalo, 2002.
- [5] Stephen B. Wicker. *Error control systems for digital communication and storage*. Prentice-Hall, Englewood Cliffs, New Jersey 07632, 1995.
- [6] S. Lin, D. J. Costello, and M. J. Miller. Automatic repeat request error-control schemes. *IEEE Commun. Mag.*, 22(12):5–17, December 1984.