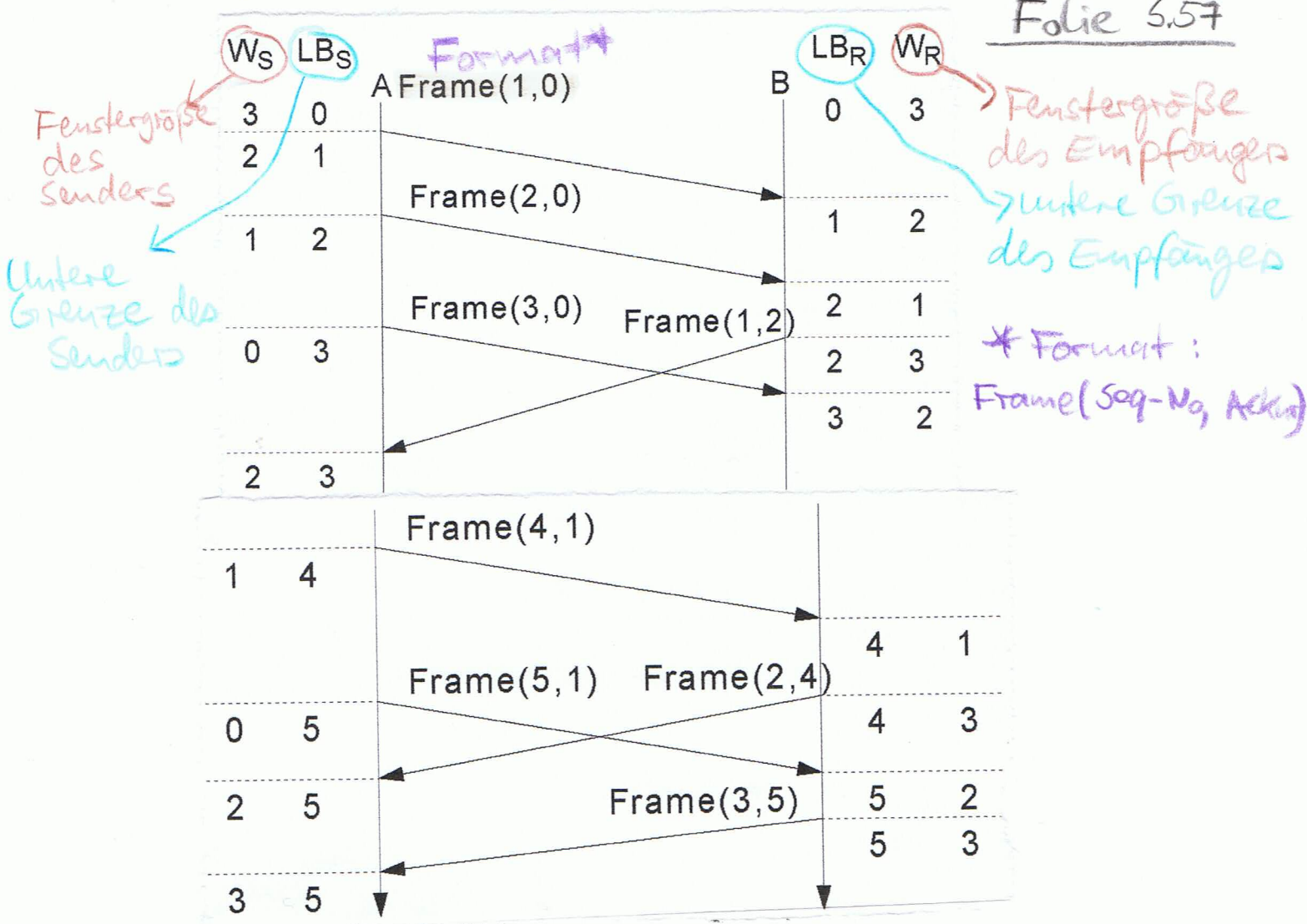


aus Studienplatzbewerbungen, bei denen man eine frankierte Postkarte optional beifügen konnte - das wäre das Acknowledgement und wie wir wissen, kann auch die Postkarte, die die Uni zur Bestätigung schickt, verloren gehen).

Doch Kimmigende ist ja nicht blöd, sie schreibt einfach in den nächsten Brief, den sie an Egon sendet, rein, dass sie seinen Brief erhalten hat und spart somit das explizite Versenden einer Bestätigung (im realen Fall sogar 45 Cent für die "Freimachung" der Postkarte).

So gesehen packt sie die Bestätigung "Huckepack" auf den von ihr gesendeten Brief.

Folie 5.57



Spielen wir das Beispiel aus den Folien mal durch: A schickt etwas an B, sagen wir Anton schickt etwas an Birte.

Das Format dessen, was er schickt, sieht so aus:

Frame (Sequenznummer, Acknowledgementnummer)

Das erste, was in der Klammer steht, zeigt an, der wievielte Brief es ist, den Anton schickt (im übertragenen Sinne: Rahmen 1).

Noch hat Anton keinen Brief von Birte erhalten, also braucht er auch noch nichts zu bestätigen, deshalb steht bei Acknowledgementnummer die 0. Also wird folgendes losgeschickt: FRAME(1,0)

Dann schickt Anton den zweiten Brief, ohne bereits eine Antwort von Birte bekommen zu haben, also FRAME(2,0).

Außerdem schickt er noch einen dritten, also FRAME(3,0).

BIRTE:

Als Birte den zweiten Brief von Anton bekommt, bricht sie das Schweigen 😊 und antwortet.

In ihrem Antwortschreiben erwähnt sie: "Lieber Anton, ich habe deine beiden Briefe erhalten..." - in anderen Worten: Sie bestätigt gleichzeitig beide erhaltenen Briefe.

Der Brief, den sie an Anton schickt, ist der erste Brief von ihr (deswegen an erster Stelle eine 1) und sie bestätigt ~~zu~~ den ersten und zweiten (man spricht dabei von kumulativer, also aufaddierter, Bestätigung) Brief von Anton, also schickt sie, im übertragenen Sinn, den

FRAME(1,2).

Und so geht's die ganze Zeit weiter...

GO-Back-N-Verfahren (Folie 5.60)

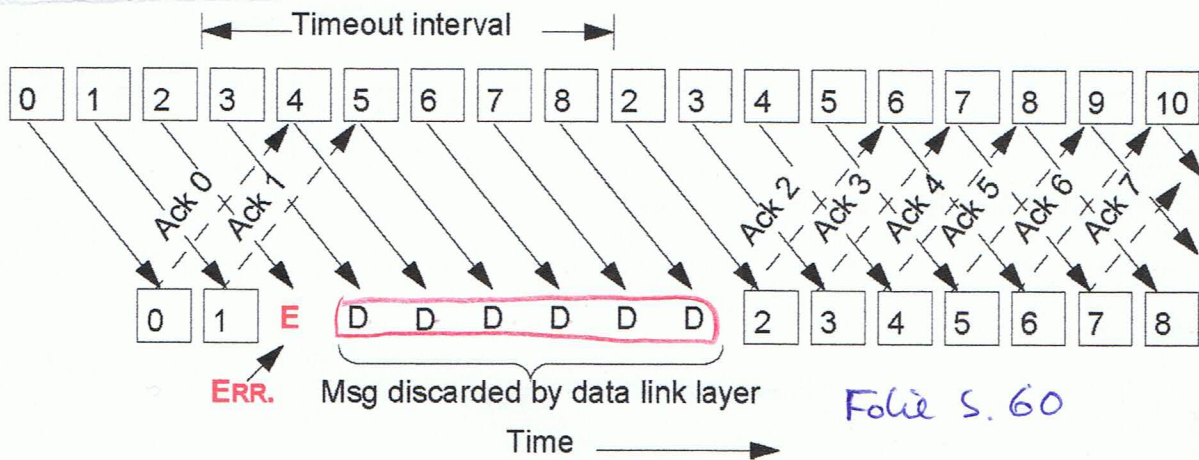
Bereits oben beim "Huckepack-Verfahren" hatten wir es mit einer "Duplex-Übertragung", also einer Übertragung in beide Richtungen (auch gleichzeitig) zu tun.

Bei GO-Back-N ist dies auch der Fall.

Dieses Verfahren ist schnell erklärt.

Stellen wir uns vor, der Sender schickt Rahmen 1, 2, 3, 4 und 5 direkt nacheinander an den Empfänger (was beim Duplex ja möglich ist, da die Bestätigungen auf einer anderen Leitung gesendet werden).

Nehmen wir an, Rahmen 2 hat einen Fehler.



Folie 5.60

Die Bestätigungen für Rahmen 0 und Rahmen 1 schickt der Empfänger los. Da Rahmen 2 einen Fehler hat, schickt der Empfänger die Bestätigung für Rahmen 2 nicht los. Wenn der Timer für Rahmen 2 beim Sender abläuft, weiß der Sender, dass Rahmen 2 entweder auf dem Weg zum Empfänger verloren gegangen, der Rahmen fehlerhaft angekommen oder die Bestätigung verloren gegangen ist. Da er nicht wissen kann, ob ~~der Fehler~~ das Problem beim Hin- oder Rückweg passiert ist, schickt er Rahmen 2 nochmal. Bis zur erneuten Übertragung von Rahmen 2 hat

er ja inzwischen schon \mathbb{P} die Rahmen
 #3 bis 8 geschickt. Doch diese Rahmen hat
 der Empfänger jeweils nicht angenommen, da er
 ster auf Rahmen 2 (allgemein auf Rahmen N),
 deswegen "Go-Back- N ") gewartet hat.
 Da der Empfänger alle auf den Rahmen 2
 empfangenen Rahmen verworfen hat, muss der
 Sender diese nochmal neu schicken - wie wir
 sehen: Er tut es auch! (i)

Vorteil von Go-Back- N :

- ziemlich einfaches Protokoll

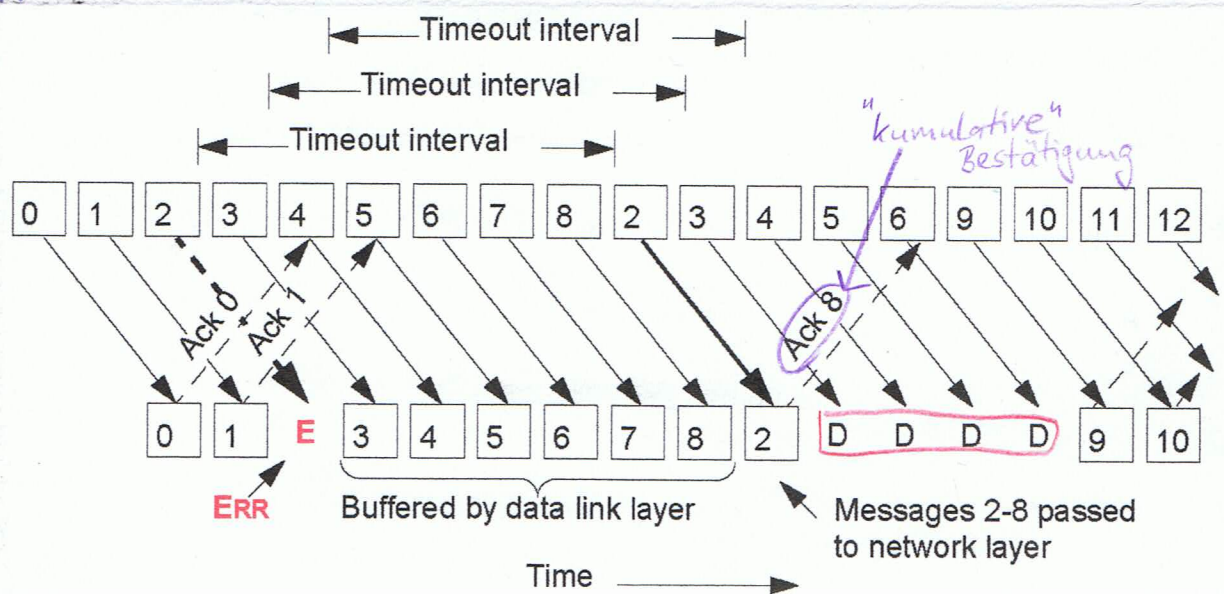
Nachteil von Go-Back- N :

- Ein Fehler führt zu einer ~~schlechten Kanalauslastung~~, schlechten Durchsatz, da ja einige Nachfolgerahmen einfach verworfen werden...

Selective Repeat

Um diesen Nachteil zu beheben (oder zumindest zu verbessern) reicht es, wenn man das Go-Back- N -Verfahren etwas verändert: Man baut einfach einen Zwischenspeicher, den sogenannten Puffer (engl.: "buffer") beim Empfänger ein, der die nachfolgenden Rahmen speichert. Sobald der ~~verlorene~~ ^{fehlerhafte} Rahmen korrekt eintrifft, wird die höchste Rahmennummer, die angekommen ist, kumulativ bestätigt.

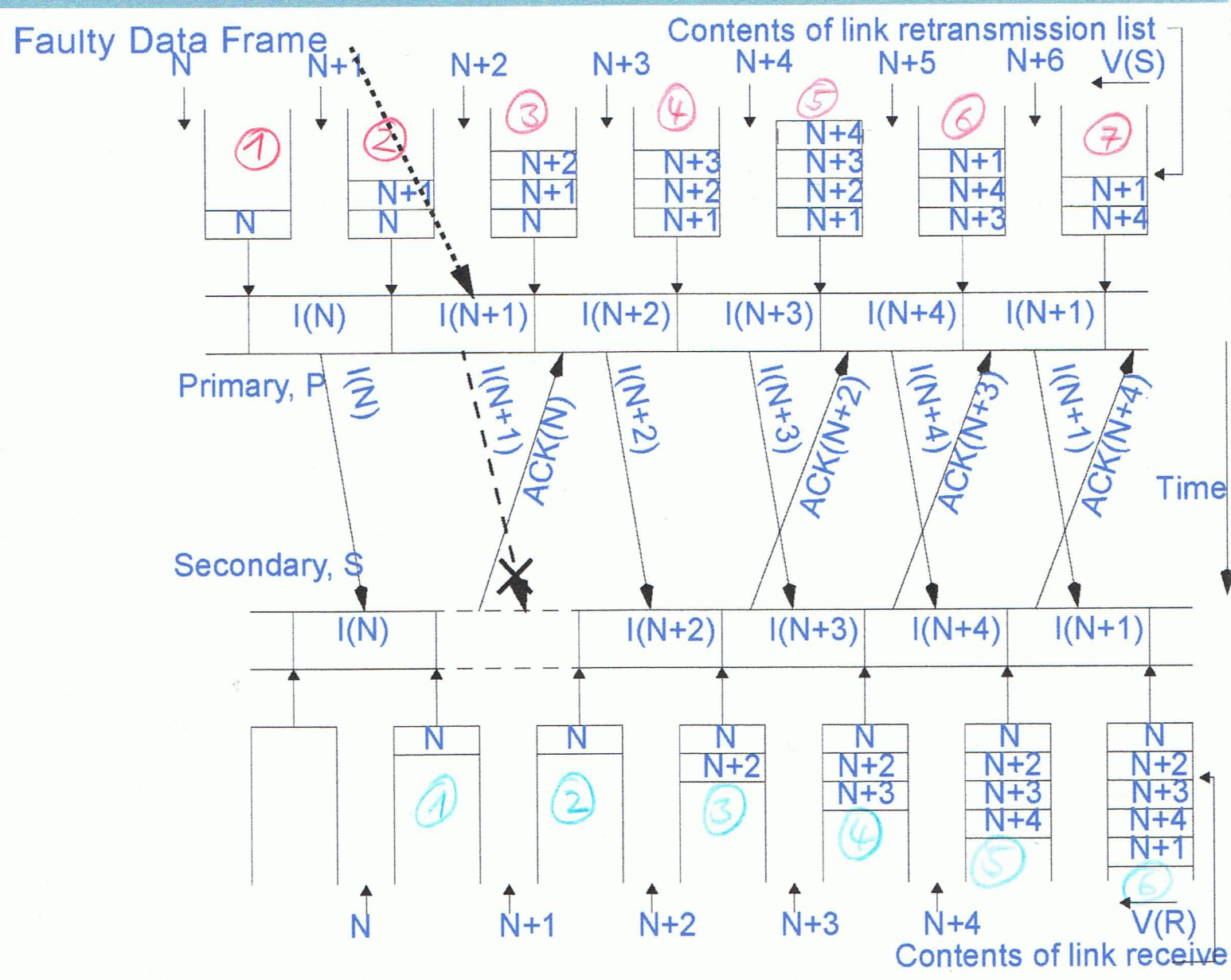
Die Anzahl der verworfenen Rahmen wird somit vermindert...



Selective Repeat

Neben den beiden Go-Back-N-Versionen gibt es nun auch die Möglichkeit, ausgewählte Pakete einzeln erneut zu übertragen, deswegen heißt das Verfahren "Selective Repeat".

Faulty Data Frame



Damit wir wissen, wovon ich gleich "spreche", habe ich das Bild auf der Vorseite mit **roten** und **blauen** Nummern durchnummeriert, wobei **rot** für den Sender (Primary) und **blau** für den Empfänger steht.

Sender und Empfänger halten eine Liste über die Rahmen, die sie gesendet (Sender) bzw. empfangen (Empfänger) haben.

Wenn ein gesendeter Rahmen bestätigt wird, so wird er als abgehakt betrachtet und aus der Liste gestrichen.

Beim Empfänger werden die Rahmen, die empfangen wurden, ~~gespeichert~~ in der Liste aufgeführt.

Gehen wir das ~~Fehlbeurteilung~~ ganze jetzt mal für den Fall einer Fehlübertragung durch:

- ① Der Sender sendet Rahmen N , nimmt N also in seine Liste auf
- ① Der Empfänger erhält Rahmen N , nimmt N also in seiner Liste auf und bestätigt Rahmen N durch ACK(N).
- ② Der Sender hat noch keine Bestätigung für Rahmen N bekommen, also ist N noch in seiner Liste. Nun schickt er den nächsten Rahmen, also Rahmen $N+1$. Ab sofort stehen also N sowie $N+1$ in seiner Liste.
- ② Da Rahmen $N+1$ auf dem Weg zum Empfänger verloren geht, erhält der Empfänger diesen Rahmen nicht und somit bleibt seine Liste weiterhin bei N .
- ③ Der Sender hat noch keine Bestätigungen erhalten, um sendet er Rahmen $N+2$. Seine aktuelle Liste enthält also die Rahmen N , $N+1$ sowie $N+2$.

- ③ Da der Empfänger bisher nur Rahmen N und Rahmen $N+2$ erhalten hat, sind auch nur diese beiden Rahmen in seiner Liste (denn den Rahmen $N+2$ bestätigt er).
- ④ Mittlerweile ist die Bestätigung für den Rahmen N eingegangen. Der Sender schießt jetzt noch Rahmen $N+3$ los. Rahmen N streicht er also aus seiner Liste (ist jetzt abgehakt) und Rahmen $N+3$ packt er dazu.
- ④ Hier passiert nichts dramatisches, es wird der empfangene Rahmen $N+3$ in die Liste aufgenommen und bestätigt.
- ⑤ Rahmen $N+4$ wird losgeschickt, also in die Liste aufgenommen.
- ⑤ Rahmen $N+4$ wird empfangen, also in die Liste aufgenommen und bestätigt.
- ⑥ Die Bestätigung für Rahmen $N+2$ ist eingetroffen. Da aber keine für $N+1$ gekommen ist, weiß der Sender, dass da irgendwo ein Fehler passiert sein muss und schießt deshalb nochmal Rahmen $N+1$.
- ⑥ Jetzt hat die Übertragung geklappt und Rahmen $N+1$ wird deshalb in die Liste aufgenommen und eine Bestätigung für $N+1$ wird geschickt...