

Systeme II

3. Die Datensicherungsschicht

Christian Schindelhauer

Technische Fakultät

Rechnernetze und Telematik

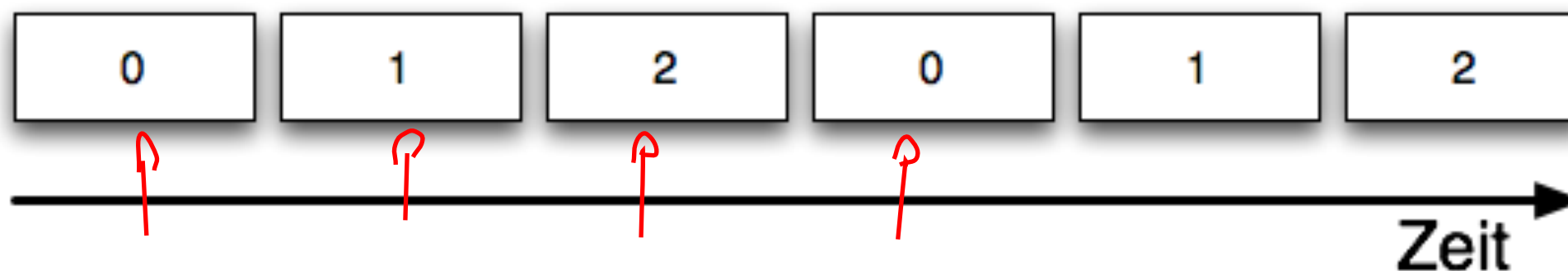
Albert-Ludwigs-Universität Freiburg

Version 31.05.2017

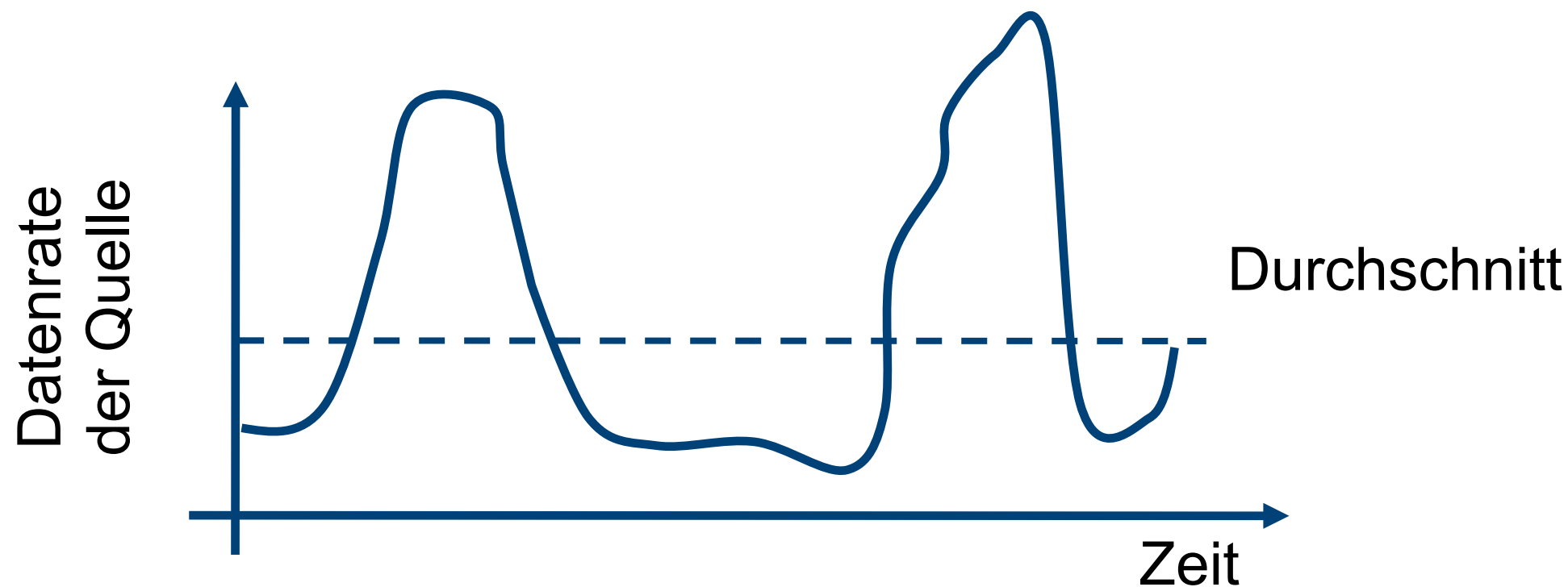
- Statisches Multiplexen
- Dynamische Kanalbelegung
 - Kollisionsbasierte Protokolle
 - Kollisionsfreie Protokolle (contention-free)
 - Protokolle mit beschränktem Wettbewerb (limited contention)


TDMA

- Gegeben sei eine einzelne Leitung (Ressource)
- Mehreren Kommunikations-verbindungen werden feste Zeiträume/Kanäle (slots/channels) zugewiesen
 - Oder: Feste Frequenzbänder werden ihnen zugewiesen
- Feste Datenraten und entsprechenden Anteilen am Kanal
 - Quellen lasten die Leitung aus



- Problem: Verkehrsspitzen (bursty traffic)
 - Definition: Großer Unterschied zwischen Spitze und Durchschnitt
 - In Rechnernetzwerken: Spitze/Durchschnitt = 1000/1 nicht ungewöhnlich



- Statisches Multiplexen
 - Dynamische Kanalbelegung
 - Kollisionsbasierte Protokolle
 - Kollisionsfreie Protokolle (contention-free)
 - Protokolle mit beschränktem Wettbewerb (limited contention)
- 

- Statisches Multiplexing ist nicht geeignet für Datenverbindung mit Spitzen
- Alternative: Zuweisung des Slots/Kanals an die Verbindung mit dem größten Bedarf
 - Dynamische Medium-Belegung
 - statt fester
- Der Mediumzugriff wird organisiert:
 - Mediumszugriff-Protokoll (Medium Access Control protocol - MAC)

■ Stationsmodell (terminal model)

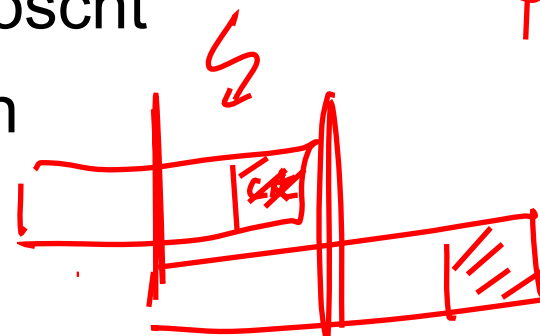
- N unabhängige Stationen möchten eine Leitung/
Ressource teilen
- Mögliches Lastmodell:
 - Wahrscheinlichkeit, dass ein Paket im Intervall der
Länge Δt erzeugt wird ist $\lambda \Delta t$ für eine Konstante λ

■ Eine Leitung/Kanal

- für alle Stationen
- Keine weitere Verbindungen möglich

■ Collision assumption

- Nur ein einfacher Frame kann auf dem Kanal
übertragen werden
- Zwei (oder mehr) sich zeitlich überschneidende
Frames kollidieren und werden gelöscht
- Noch nicht einmal Teile kommen an



$$\lambda : \# \text{Paket}_t$$

$$N \rightarrow \infty$$

$$\Delta t$$



$$P : \text{W'keit in}$$

Paket zu senden

$$P[X=k]$$

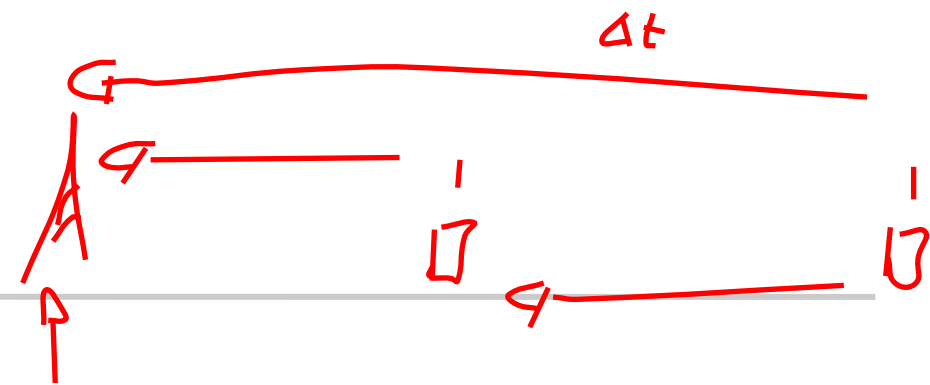
$$P[X=1] = p$$

$$P[X=0] = 1-p$$

$$E[X] = \sum_{k=0}^{\infty} k \cdot P[X=k]$$

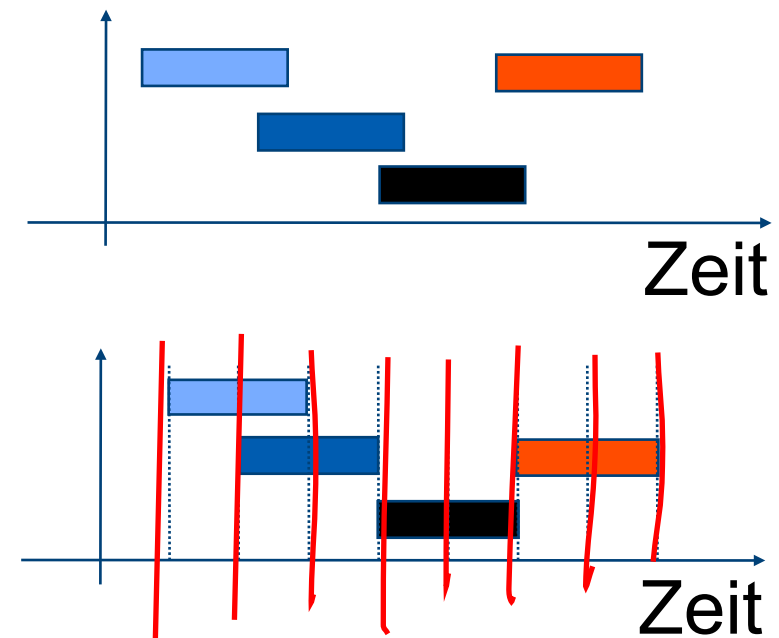
$$P[X=2] = \frac{1}{6}$$

$$\sum k \cdot P[X=k]$$



Zeitmodelle

- Kontinuierlich
 - Übertragungen können jeder Zeit beginnen (keine zentrale Uhr)
- Diskret (Slotted time)
 - Die Zeitachse ist in Abschnitte (slots) unterteilt
 - Übertragungen können nur an Abschnittsgrenzen starten
 - Slots können leer (idle), erfolgreich (mit Übertragung) sein oder eine Kollision beinhalten



Träger-Messung (Carrier Sensing)

- Stationen können erkennen ob der Kanal momentan von anderen Stationen verwendet wird
 - Nicht notwendigerweise zuverlässig

- Methoden zur Bewertung der Effizienz einer Kanalzuweisung

↻ Durchsatz (throughput)

- Anzahl Pakete pro Zeiteinheit
- Besonders bei großer Last wichtig

↻ Verzögerung (delay)

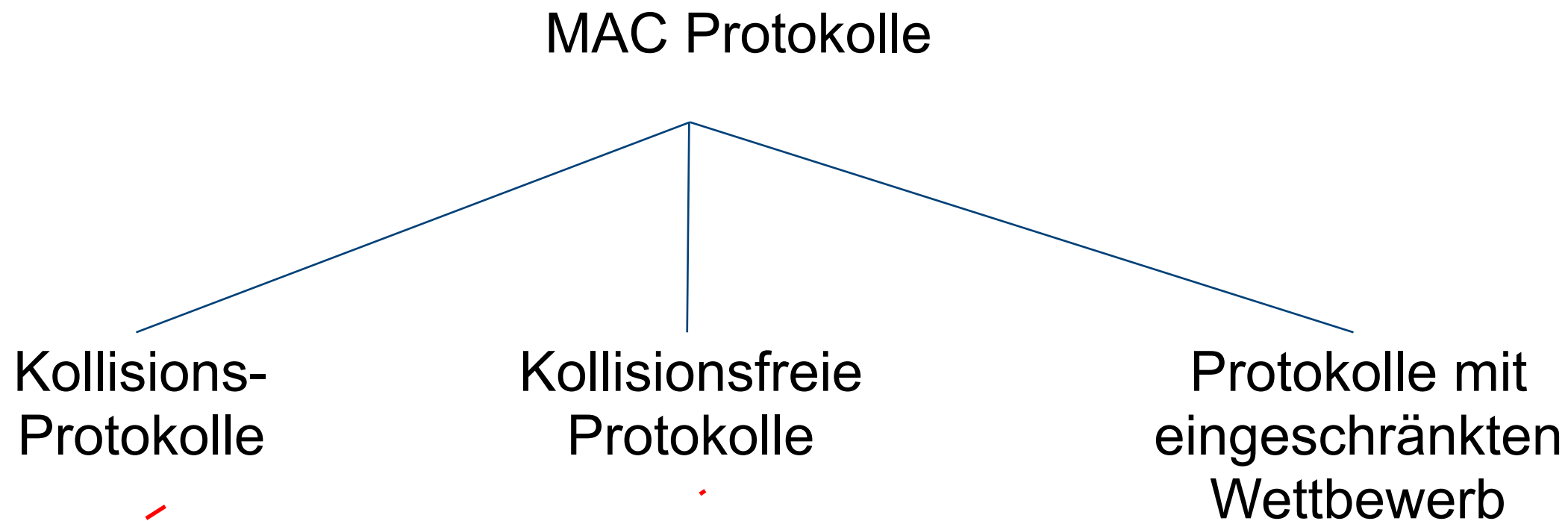
- Zeit für den Transport eines Pakets
- Muss bei geringer Last gut sein

↻ Gerechtigkeit (fairness)

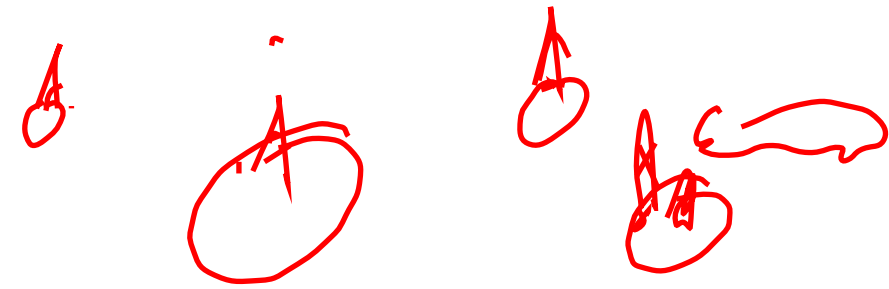
↪ QoS

- Gleichbehandlung aller Stationen
- Fairer Anteil am Durchsatz und bei Delay

- Unterscheidung: Erlaubt das Protokoll Kollisionen?
 - Als Systemscheidung
 - Die unbedingte Kollisionsvermeidung kann zu Effizienzeinbußen führen



System mit Kollisionen: **Contention System**

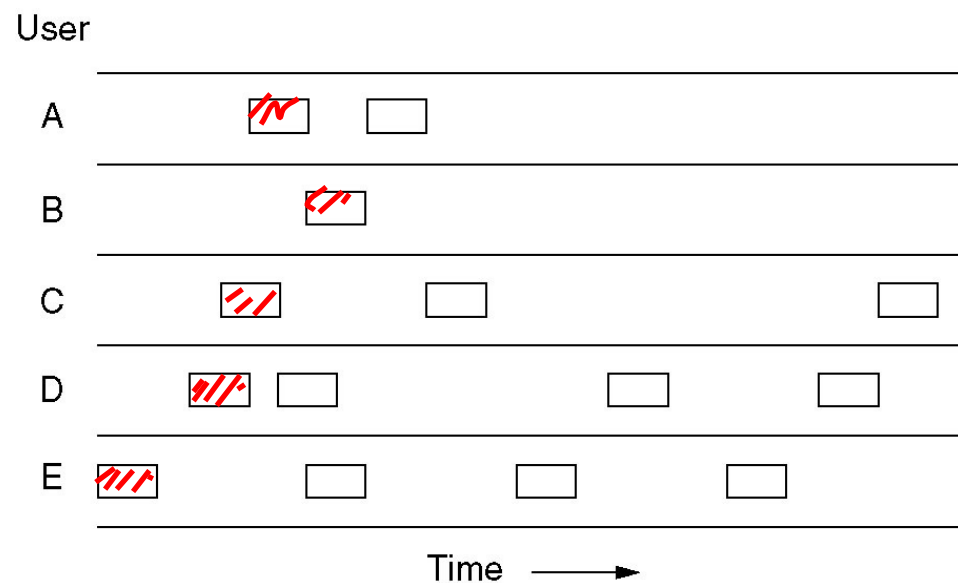
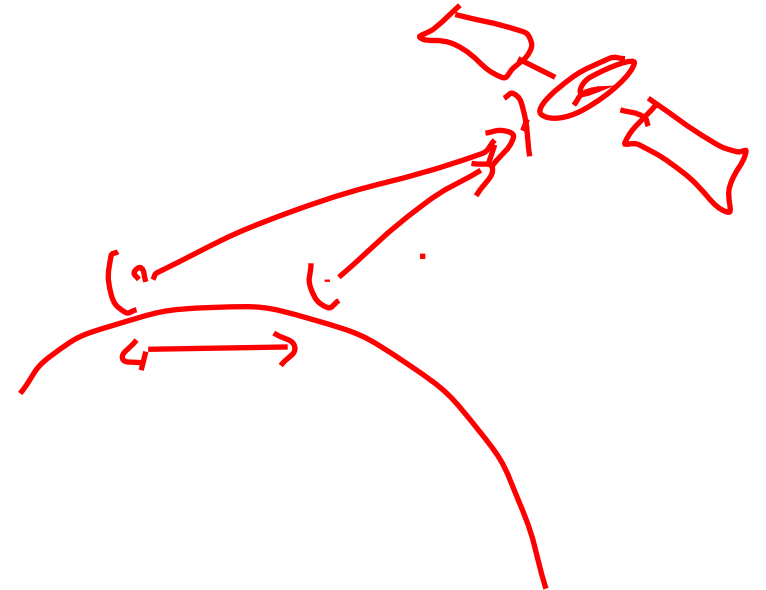


■ Algorithmus

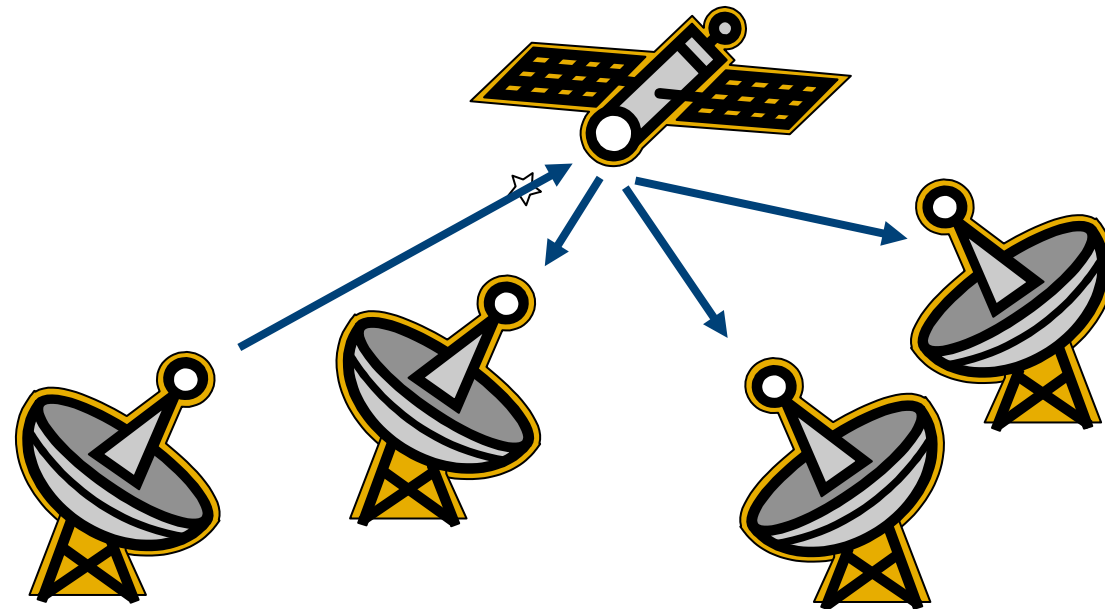
- Sobald ein Paket vorhanden ist, wird es gesendet

■ Ursprung

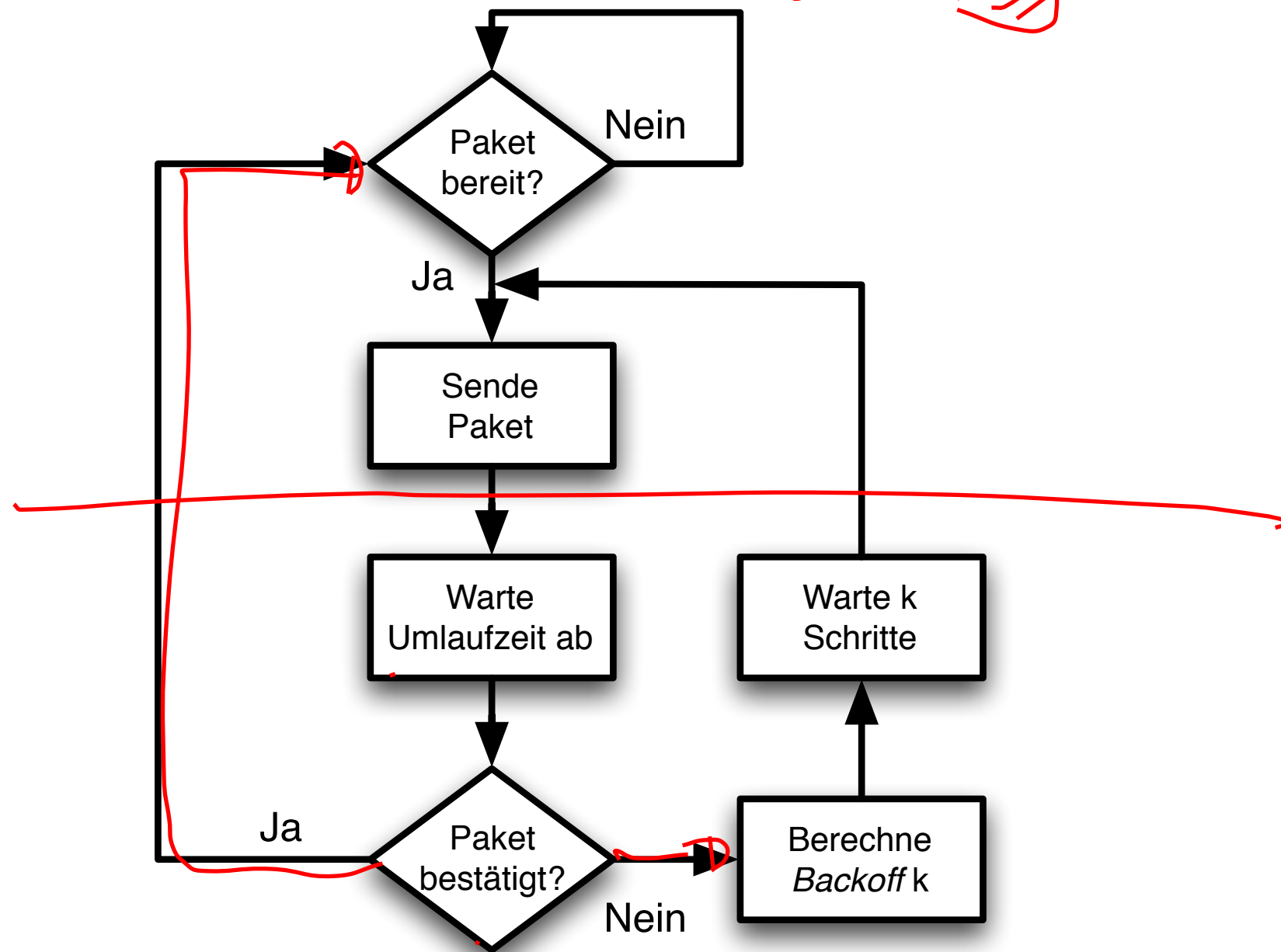
- 1985⁷⁰ by Abrahmson et al., University of Hawaii
- Ziel: Verwendung in ~~Satelliten~~-Verbindung



Pakete werden zu beliebigen Zeiten übertragen



ALOHA – Analyse



- Vorteile
 - Einfach
 - Keine Koordination notwendig
- Nachteile
 - Kollisionen
 - Sender überprüft den Kanalzustand nicht
 - Sender hat keine direkte Methode den Sende-Erfolg zu erfahren
 - Bestätigungen sind notwendig
 - Diese können auch kollidieren

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$

- Betrachte Poisson-Prozess zur Erzeugung von Paketen
 - Entsteht durch “unendlich” viele Stationen, die sich gleich verhalten
 - Zeit zwischen zwei Sende-Versuchen ist exponentiell verteilt
 - Sei G der Erwartungswert der Übertragungsversuche pro Paketlänge
 - Alle Pakete haben gleiche Länge
 - Dann gilt

$$P[k \text{ Versuche}] = \frac{G^k}{k!} e^{-G}$$

$$E[X] = \sum_{k=0}^{\infty} k \cdot \frac{G^k}{k!} e^{-G} = G$$

- Um eine erfolgreiche Übertragung zu erhalten, darf keine Kollision mit einem anderen Paket erfolgen
- Wie lautet die Wahrscheinlichkeit für eine solche Übertragung?

n Sender $\overbrace{\hspace{2cm}}^p$

$$P[k \text{ Pakete im Intervall}] = \binom{n}{k} \cdot (1-p)^{n-k} \cdot p^k$$

$$\begin{aligned} k=0 &: p^0 (1-p)^n \binom{n}{0} \\ k=1 &: p (1-p)^{n-1} \binom{n}{1} \\ k=2 &: p^2 (1-p)^{n-2} \binom{n}{2} \end{aligned}$$

$$E[X_n] = \sum_{k=0}^n k \cdot P[X=k] = E[X_1 + X_1 + \dots + X_1] = E[n \cdot X_1] = n \cdot E[X_1] = p \cdot n$$

$$p = \frac{\lambda}{n}$$

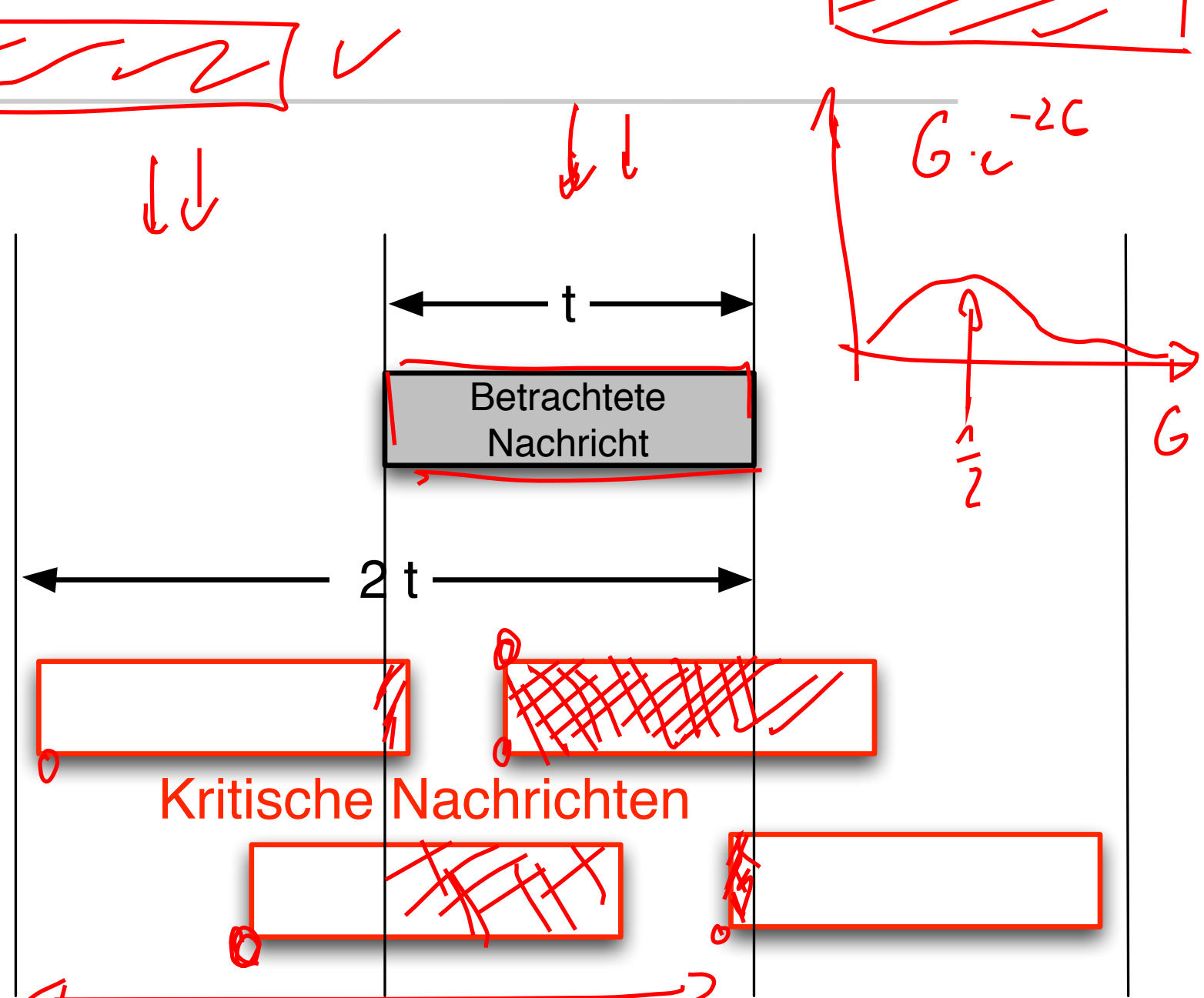
$$\lim_{n \rightarrow \infty} \binom{n}{k} \cdot \left(\frac{\lambda}{n}\right)^k \cdot \left(1 - \frac{\lambda}{n}\right)^{n-k} = \frac{\lambda^k}{k!} \cdot \frac{n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot (n-k+1)}{n \cdot n \cdot n \cdot \dots \cdot n} \cdot \left(1 - \frac{\lambda}{n}\right)^{\frac{n-k}{\lambda} \cdot \lambda}$$

$$= \frac{1}{k!} \cdot \lambda^k \cdot e^{-\lambda} \stackrel{!}{=} \text{Poisson}$$

$$\left(1 - \frac{\lambda}{n}\right)^{\frac{n-k}{\lambda}} = \left(1 - \frac{1}{x}\right)^x \rightarrow \frac{1}{e}$$

ALOHA – Effizienz

- Ein Paket X wird gestört, wenn
 - ein Paket kurz vor X startet
 - wenn ein Paket kurz vor dem Ende von X startet
- Das Paket wird erfolgreich übertragen, wenn in einem Zeitraum von zwei Paketen kein (anderes) Paket übertragen wird
- Durchsatz:
 - $S(G) = G e^{-2G}$
 - Optimal für $G=1/2$, $S=1/e$



$$P[x=0]$$

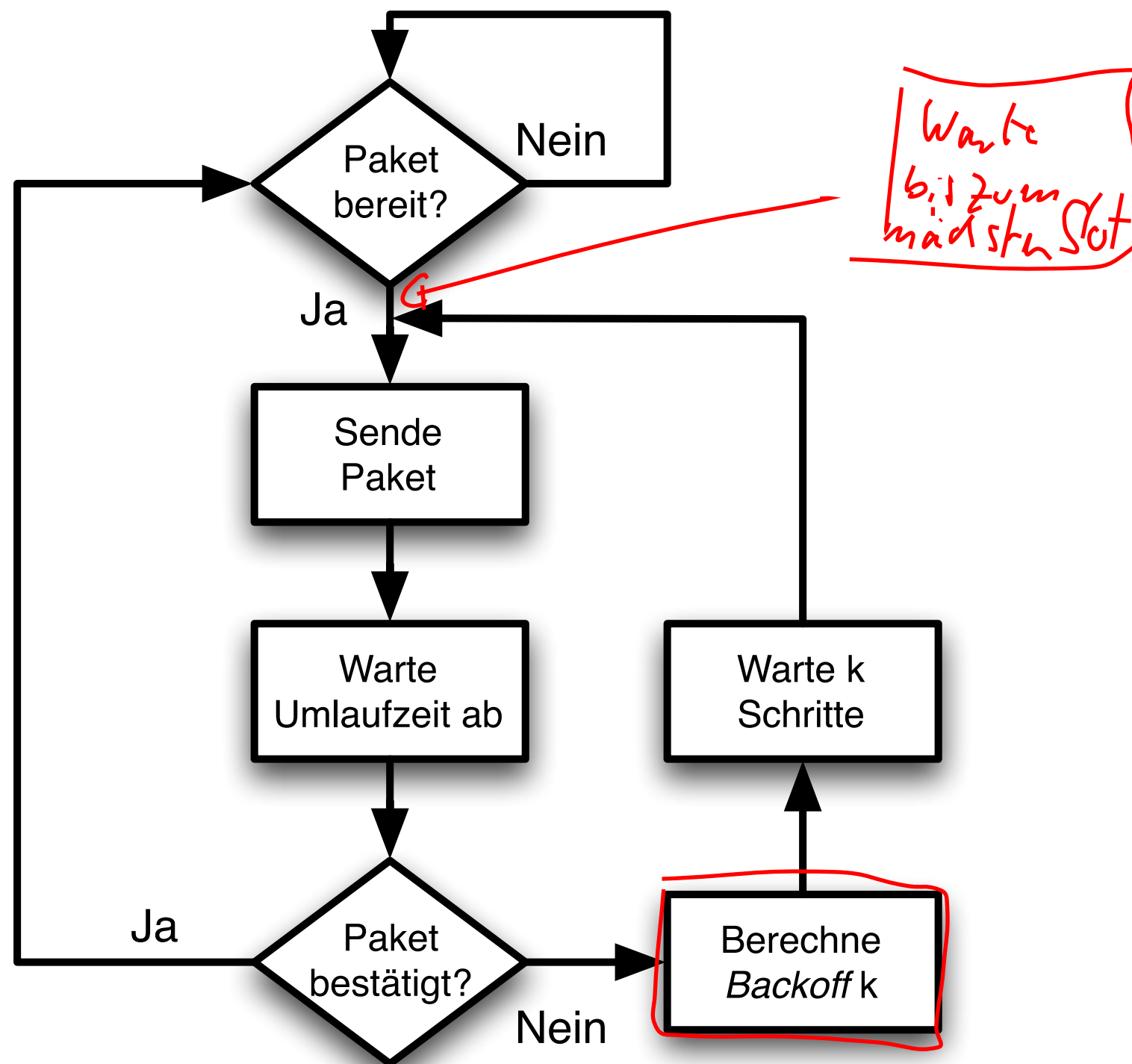
$$P[x=1]$$

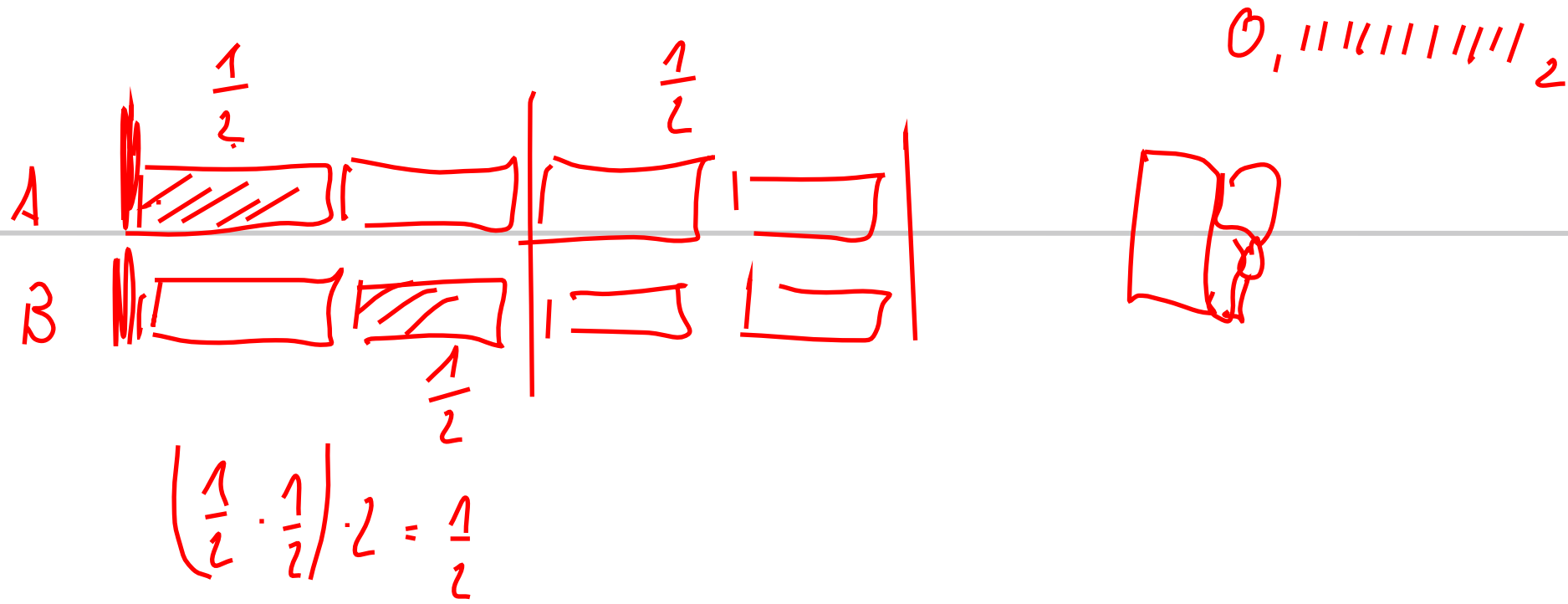
$$\frac{G^0}{0!} \cdot e^{-G}$$

$$\frac{G^1}{1} \cdot e^{-G}$$

$$= G \cdot e^{-2G}$$

Slotted ALOHA



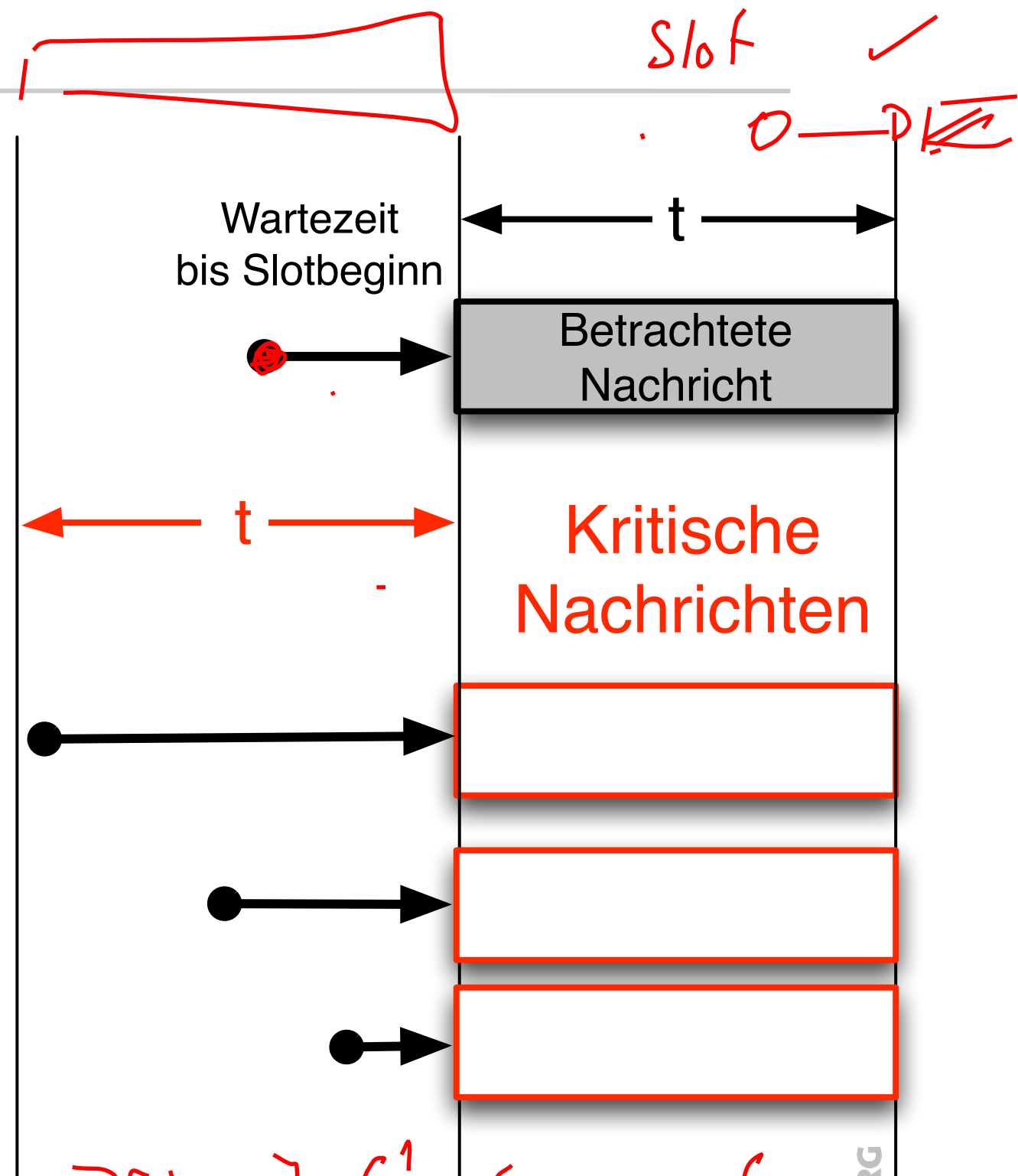


$$1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \frac{1}{8} + \dots = 2$$

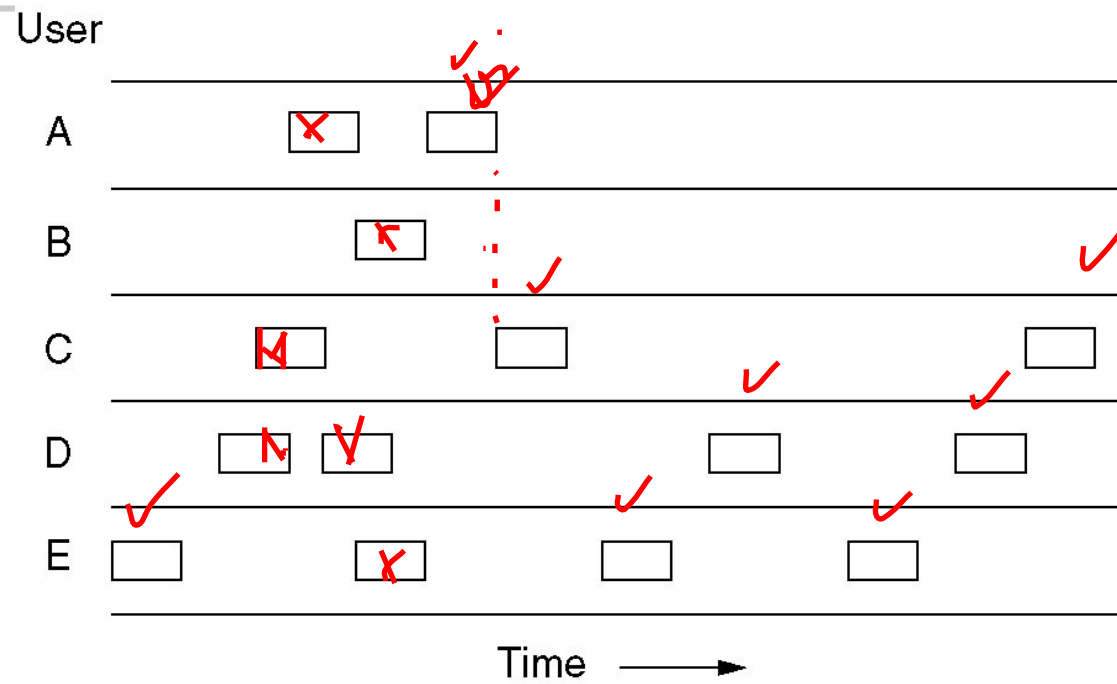
$$= \frac{1}{2} + \frac{1}{4} - \frac{1}{4} + \frac{1}{8} - \frac{1}{8} + \frac{1}{16} - \frac{1}{16} + \frac{1}{32} - \frac{1}{32} + \frac{1}{64} - \frac{1}{64} + \frac{1}{128} - \frac{1}{128} + \frac{1}{256} - \frac{1}{256} + \frac{1}{512} - \frac{1}{512} + \frac{1}{1024} - \frac{1}{1024} + \frac{1}{2048} - \frac{1}{2048} + \frac{1}{4096} - \frac{1}{4096} + \frac{1}{8192} - \frac{1}{8192} + \frac{1}{16384} - \frac{1}{16384} + \frac{1}{32768} - \frac{1}{32768} + \frac{1}{65536} - \frac{1}{65536} + \frac{1}{131072} - \frac{1}{131072} + \frac{1}{262144} - \frac{1}{262144} + \frac{1}{524288} - \frac{1}{524288} + \frac{1}{1048576} - \frac{1}{1048576} + \frac{1}{2097152} - \frac{1}{2097152} + \frac{1}{4194304} - \frac{1}{4194304} + \frac{1}{8388608} - \frac{1}{8388608} + \frac{1}{16777216} - \frac{1}{16777216} + \frac{1}{33554432} - \frac{1}{33554432} + \frac{1}{67108864} - \frac{1}{67108864} + \frac{1}{134217728} - \frac{1}{134217728} + \frac{1}{268435456} - \frac{1}{268435456} + \frac{1}{536870912} - \frac{1}{536870912} + \frac{1}{1073741824} - \frac{1}{1073741824} + \frac{1}{2147483648} - \frac{1}{2147483648} + \frac{1}{4294967296} - \frac{1}{4294967296} + \frac{1}{8589934592} - \frac{1}{8589934592} + \frac{1}{17179869184} - \frac{1}{17179869184} + \frac{1}{34359738368} - \frac{1}{34359738368} + \frac{1}{68719476736} - \frac{1}{68719476736} + \frac{1}{137438953472} - \frac{1}{137438953472} + \frac{1}{274877906944} - \frac{1}{274877906944} + \frac{1}{549755813888} - \frac{1}{549755813888} + \frac{1}{1099511627776} - \frac{1}{1099511627776} + \frac{1}{2199023255552} - \frac{1}{2199023255552} + \frac{1}{4398046511104} - \frac{1}{4398046511104} + \frac{1}{8796093022208} - \frac{1}{8796093022208} + \frac{1}{17592186044416} - \frac{1}{17592186044416} + \frac{1}{35184372088832} - \frac{1}{35184372088832} + \frac{1}{70368744177664} - \frac{1}{70368744177664} + \frac{1}{140737488355328} - \frac{1}{140737488355328} + \frac{1}{281474976710656} - \frac{1}{281474976710656} + \frac{1}{562949953421312} - \frac{1}{562949953421312} + \frac{1}{1125899906842624} - \frac{1}{1125899906842624} + \frac{1}{2251799813685248} - \frac{1}{2251799813685248} + \frac{1}{4503599627370496} - \frac{1}{4503599627370496} + \frac{1}{9007199254740992} - \frac{1}{9007199254740992} + \frac{1}{18014398509481984} - \frac{1}{18014398509481984} + \frac{1}{36028797018963968} - \frac{1}{36028797018963968} + \frac{1}{72057594037927936} - \frac{1}{72057594037927936} + \frac{1}{144115188075855872} - \frac{1}{144115188075855872} + \frac{1}{288230376151711744} - \frac{1}{288230376151711744} + \frac{1}{576460752303423488} - \frac{1}{576460752303423488} + \frac{1}{1152921504606846976} - \frac{1}{1152921504606846976} + \frac{1}{2305843009213693952} - \frac{1}{2305843009213693952} + \frac{1}{4611686018427387904} - \frac{1}{4611686018427387904} + \frac{1}{9223372036854775808} - \frac{1}{9223372036854775808} + \frac{1}{18446744073709551616} - \frac{1}{18446744073709551616} + \frac{1}{36893488147419103232} - \frac{1}{36893488147419103232} + \frac{1}{73786976294838206464} - \frac{1}{73786976294838206464} + \frac{1}{147573952589676412928} - \frac{1}{147573952589676412928} + \frac{1}{295147905179352825856} - \frac{1}{295147905179352825856} + \frac{1}{590295810358705651712} - \frac{1}{590295810358705651712} + \frac{1}{1180591620717411303424} - \frac{1}{1180591620717411303424} + \frac{1}{2361183241434822606848} - \frac{1}{2361183241434822606848} + \frac{1}{4722366482869645213696} - \frac{1}{4722366482869645213696} + \frac{1}{9444732965739290427392} - \frac{1}{9444732965739290427392} + \frac{1}{18889465931478580854784} - \frac{1}{18889465931478580854784} + \frac{1}{37778931862957161709568} - \frac{1}{37778931862957161709568} + \frac{1}{75557863725914323419136} - \frac{1}{75557863725914323419136} + \frac{1}{151115727451828646838272} - \frac{1}{151115727451828646838272} + \frac{1}{302231454903657293676544} - \frac{1}{302231454903657293676544} + \frac{1}{604462909807314587353088} - \frac{1}{604462909807314587353088} + \frac{1}{1208925819614629174706176} - \frac{1}{1208925819614629174706176} + \frac{1}{2417851639229258349412352} - \frac{1}{2417851639229258349412352} + \frac{1}{4835703278458516698824704} - \frac{1}{4835703278458516698824704} + \frac{1}{9671406556917033397649408} - \frac{1}{9671406556917033397649408} + \frac{1}{19342813113834066795298816} - \frac{1}{19342813113834066795298816} + \frac{1}{38685626227668133590597632} - \frac{1}{38685626227668133590597632} + \frac{1}{77371252455336267181195264} - \frac{1}{77371252455336267181195264} + \frac{1}{154742504910672534362390528} - \frac{1}{154742504910672534362390528} + \frac{1}{309485009821345068724781056} - \frac{1}{309485009821345068724781056} + \frac{1}{618970019642690137449562112} - \frac{1}{618970019642690137449562112} + \frac{1}{1237940039285380274899124224} - \frac{1}{1237940039285380274899124224} + \frac{1}{2475880078570760549798248448} - \frac{1}{2475880078570760549798248448} + \frac{1}{4951760157141521099596496896} - \frac{1}{4951760157141521099596496896} + \frac{1}{9903520314283042199192993792} - \frac{1}{9903520314283042199192993792} + \frac{1}{19807040628566084398385987584} - \frac{1}{19807040628566084398385987584}$$

Slotted ALOHA – Effizienz

- Ein Paket X wird gestört, wenn
 - ein Paket kurz vor X startet
 - wenn ein Paket kurz vor dem Ende von X startet
- Das Paket wird erfolgreich übertragen, wenn in einem Zeitraum von zwei Paketen kein (anderes) Paket übertragen wird



$$P[X=1] = \frac{G^1}{1!} e^{-G} = G \cdot e^{-G}$$



Aloha

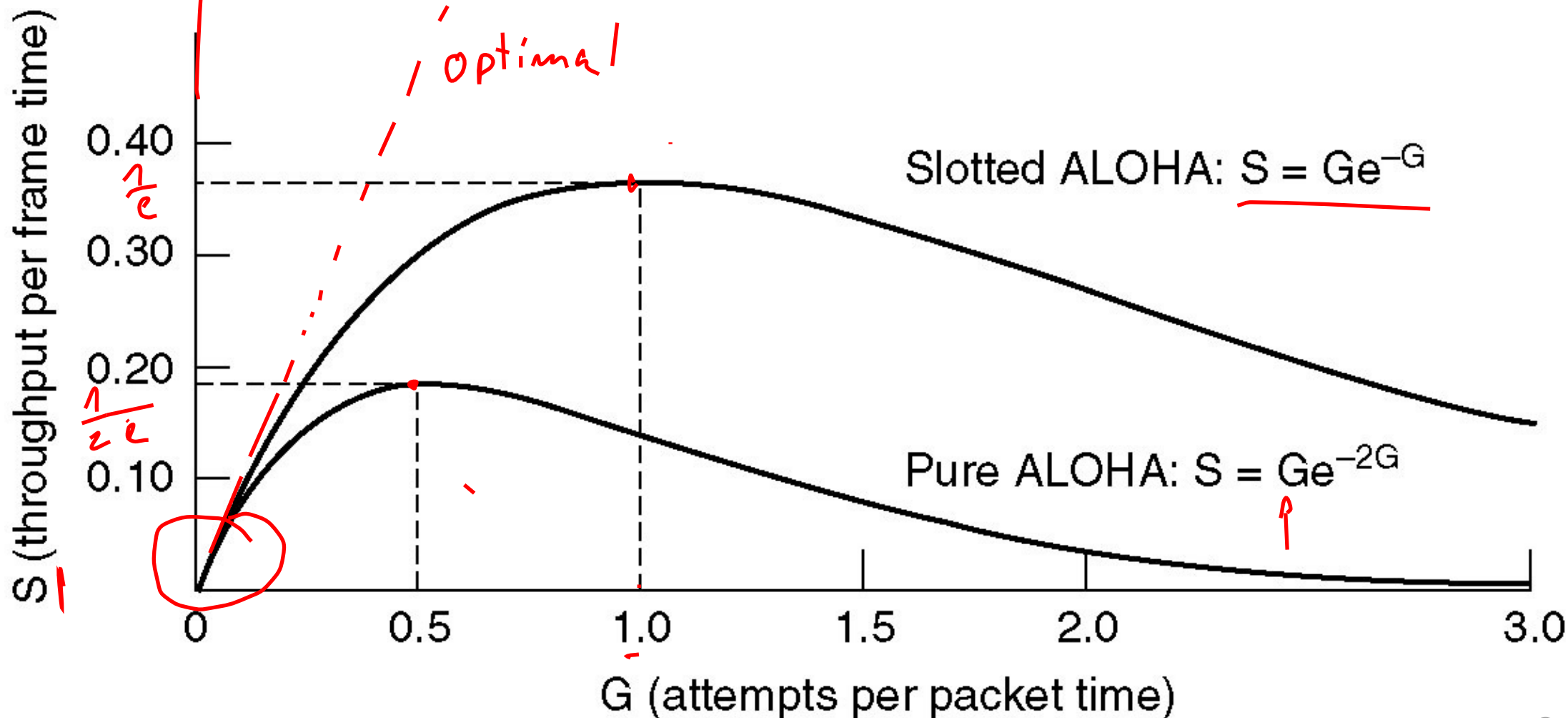
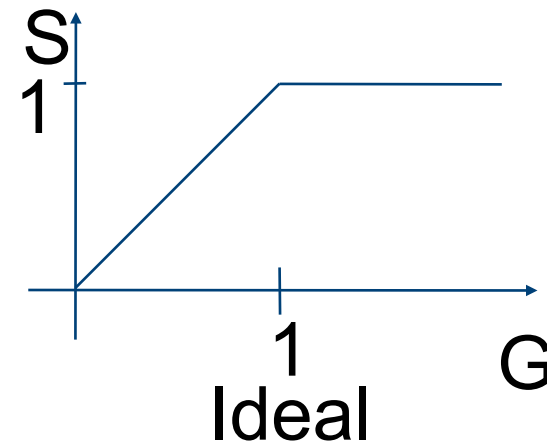


Slotted
Aloha

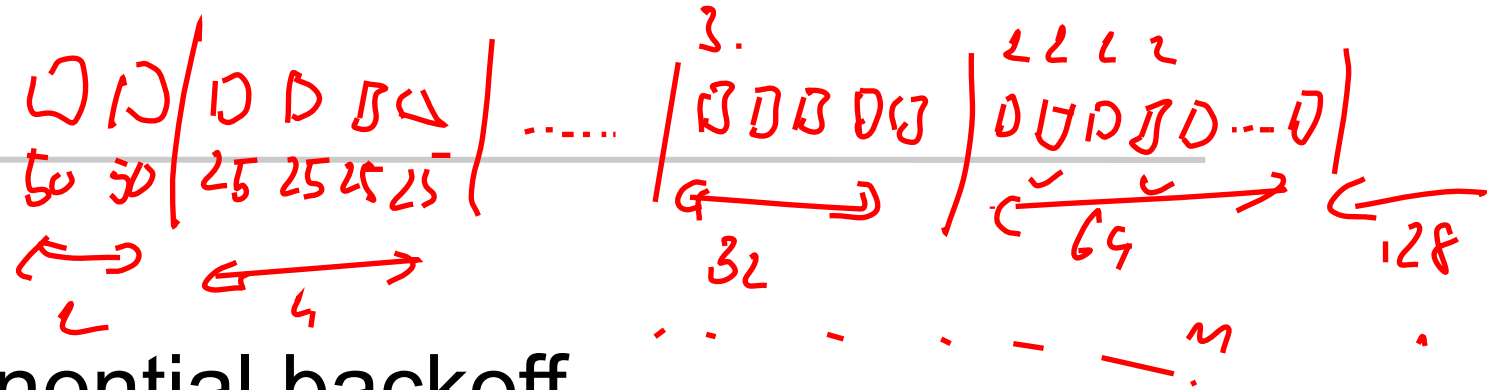
- ALOHAs Problem:
 - Lange Verwundbarkeit eines Pakets
- Reduktion durch Verwendung von Zeitscheiben (Slots)
 - Synchronisation wird vorausgesetzt
- Ergebnis:
 - Verwundbarkeit wird halbiert
 - Durchsatz:
 - $S(G) = Ge^{-G}$
 - Optimal für $G=1$, $S=1/e$

Durchsatz in Abhängigkeit der Last

- Für (slotted) ALOHA ist eine geschlossene Darstellung in Abhängigkeit von G möglich
- Kein gutes Protokoll
 - Durchsatz bricht zusammen, wenn die Last zunimmt



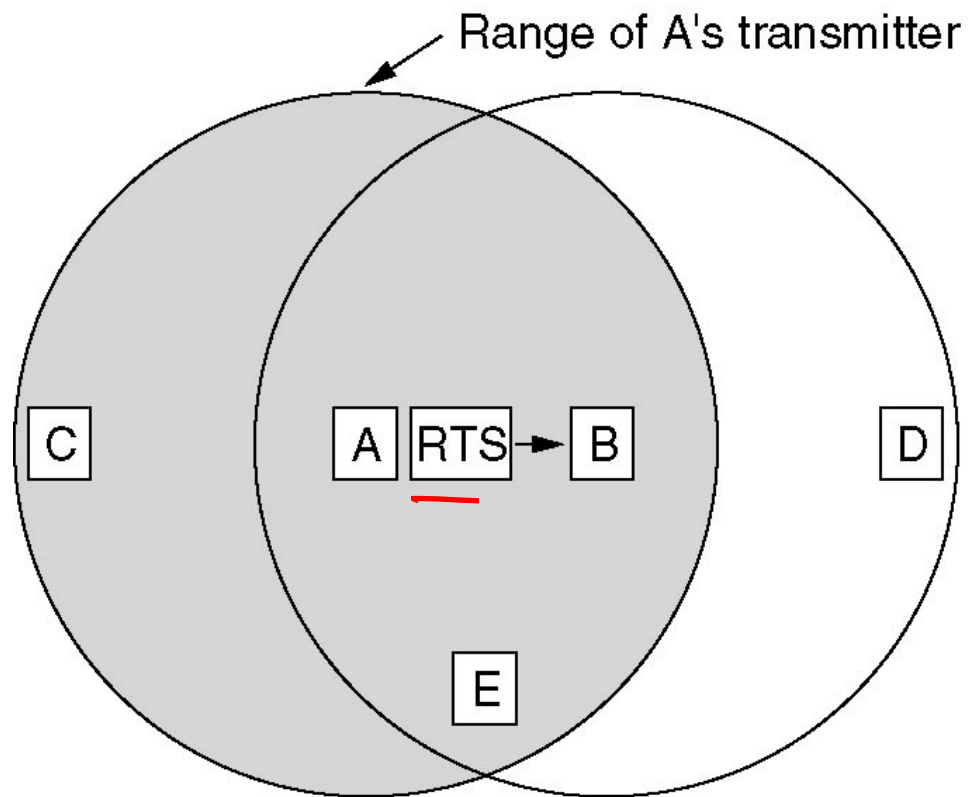
Bestimmung der Warte-Zeit



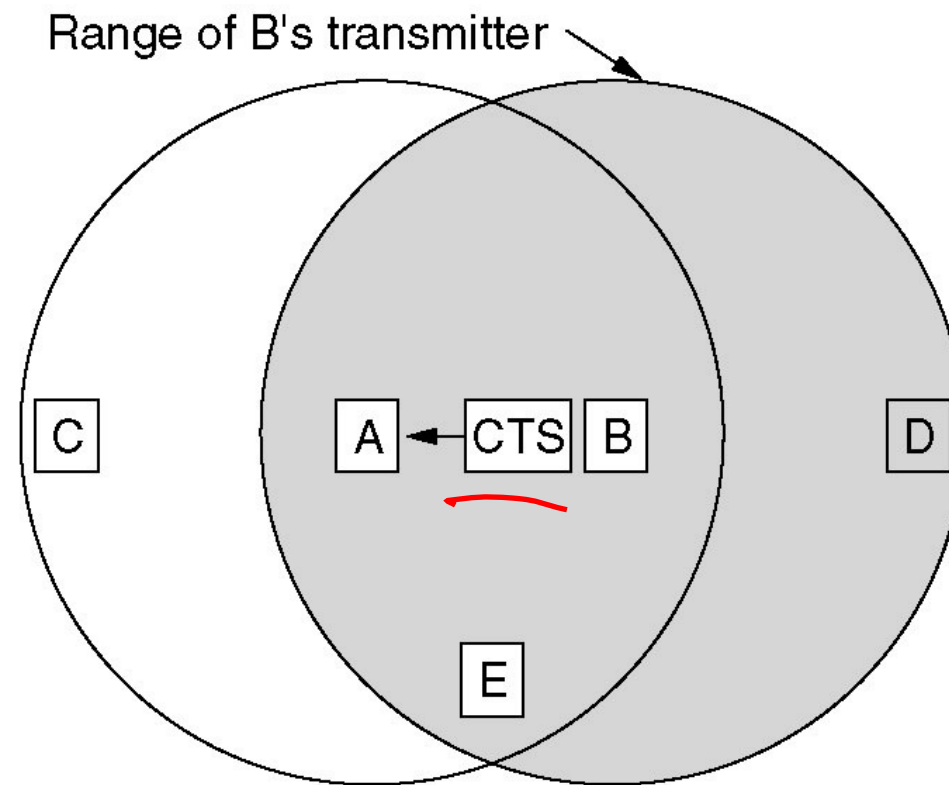
- Nach der Kollision:
- Algorithmus binary exponential backoff
 - $k := 2$
 - Solange Kollision beim letzten Senden
 - Wähle t gleichwahrscheinlich zufällig aus $\{0, \dots, k-1\}$
 - Warte t Zeit-Slots
 - Sende Nachricht (Abbruch bei Collision Detection)
 - $k := 2k$
- Algorithmus
 - passt Wartezeit dynamisch an die Anzahl beteiligter Stationen an
 - sorgt für gleichmäßige Auslastung des Kanals
 - ist fair (auf lange Sicht)

Multiple Access with Collision Avoidance

MACA

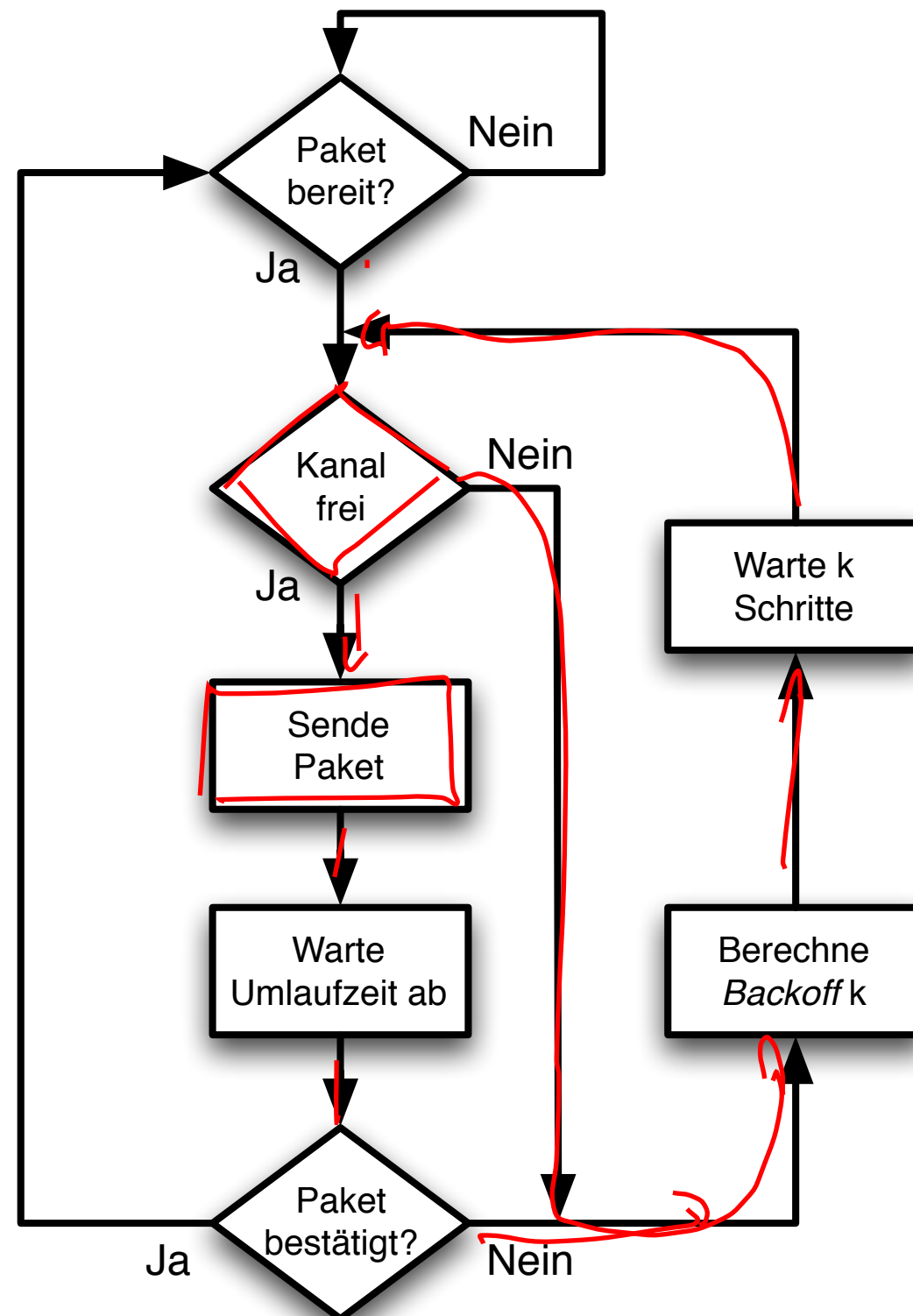


(a)

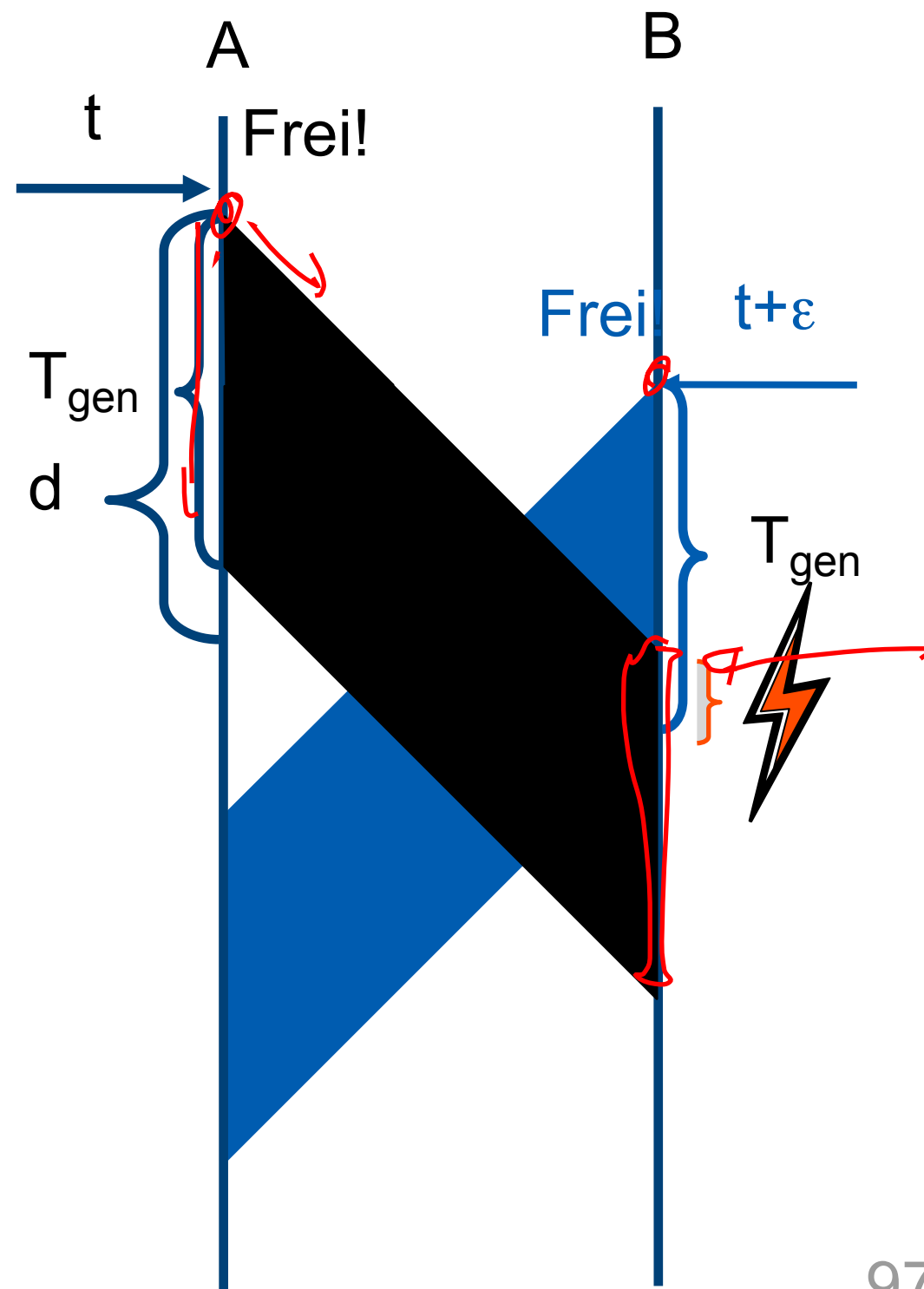


(b)

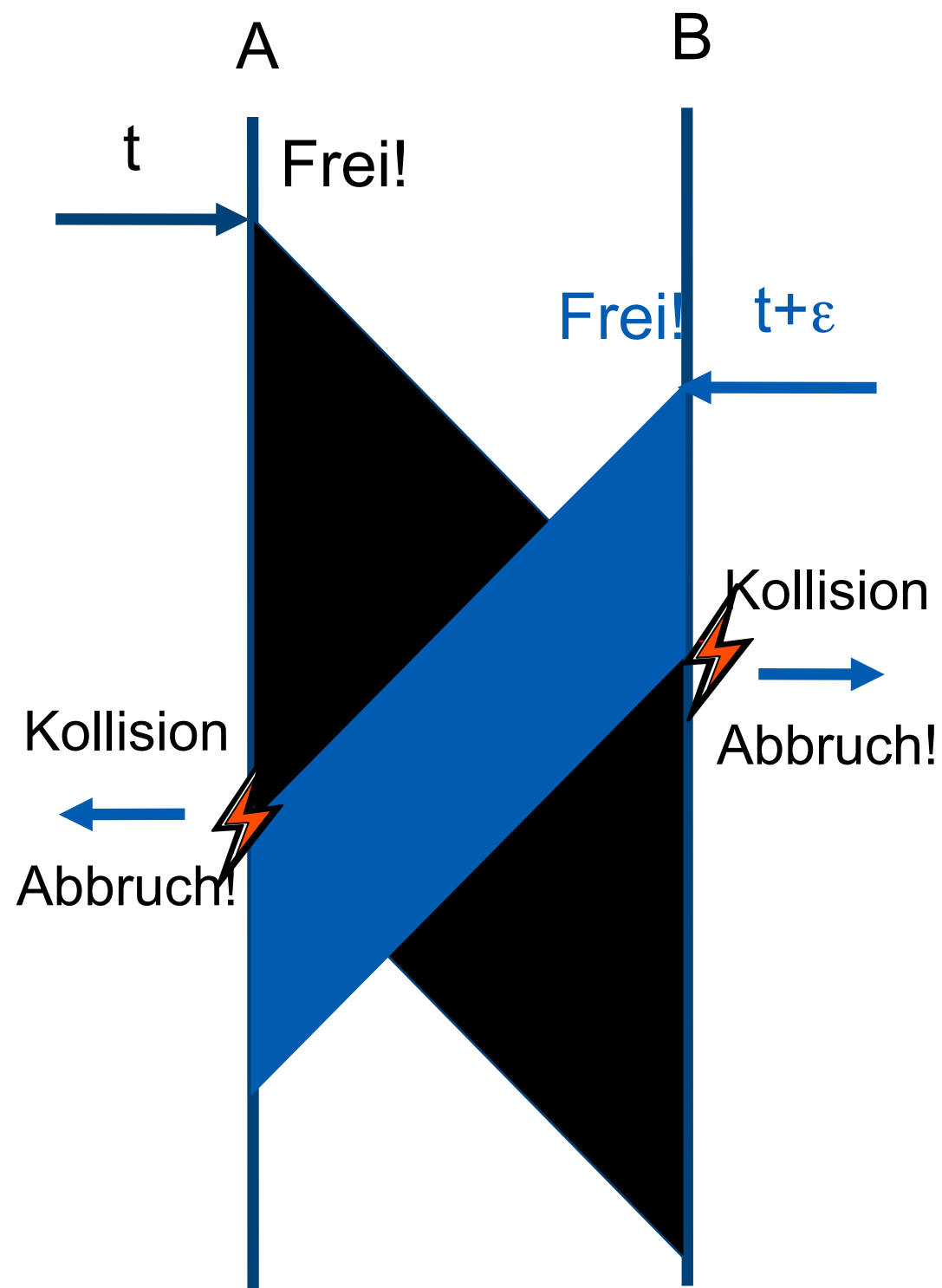
- (a) A sendet Request to Send (RTS) an B.
- (b) B antwortet mit Clear to Send (CTS) an A.



- Carrier Sense Multiple Access:
 - Erst senden wenn der Kanal frei ist
- CSMA-Problem:
 - Übertragungszeit d (propagation delay)
- Zwei Stationen
 - starten Senden zu den Zeitpunkten t und $t+\epsilon$ mit $\epsilon < d$
 - sehen jeweils einen freien Kanal
- Zweite Station
 - verursacht dann eine Kollision

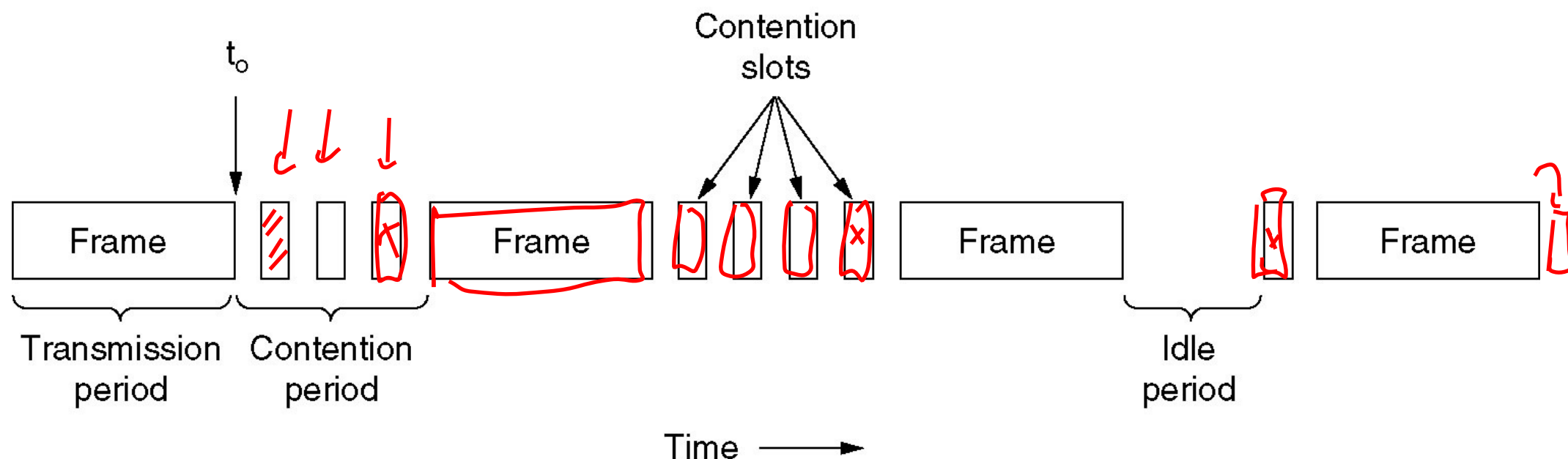


- Falls Kollisionserkennung (collision detection) möglich ist,
 - dann beendet der spätere Sender seine Übertragung
 - Zeitverschwendung wird reduziert, da mindestens eine Nachricht (die erste) übertragen wird
- Fähigkeit der Kollisionserkennung hängt von der Bitübertragungsschicht ab
- CSMA/CD – Carrier Sense Multiple Access/Collision Detection
- Collision Detection
 - setzt gleichzeitiges Abhören des Kanals nach Kollisionen voraus
 - Ist das was auf dem Kanal geschieht, identisch zu der eigenen Nachricht?



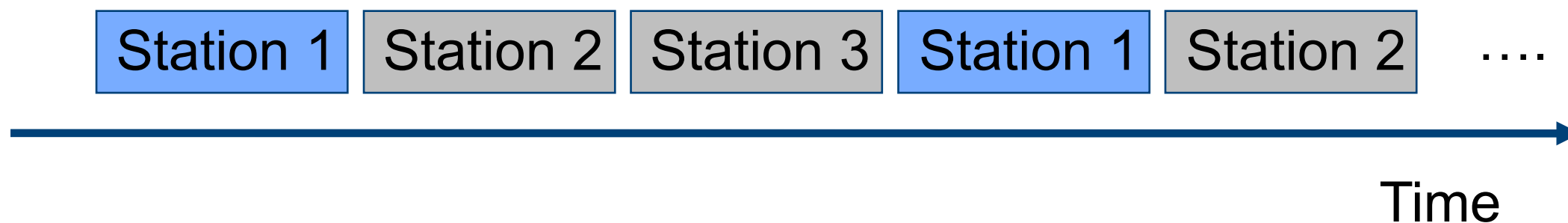
Phasen in CSMA/CD

- Leer-Phase (IDLE)
 - Keine Station sendet einen Frame
 - Wettbewerbsphase (Contention Period)
 - Kollisionen entstehen, Übertragungen werden abgebrochen
 - Übertragungsphase (Transmission Period)
 - Keine Kollision, effektiver Teil des Protokolls
- ! Es gibt nur Wettbewerbs-, Übertragungsphasen und Leer-Phasen

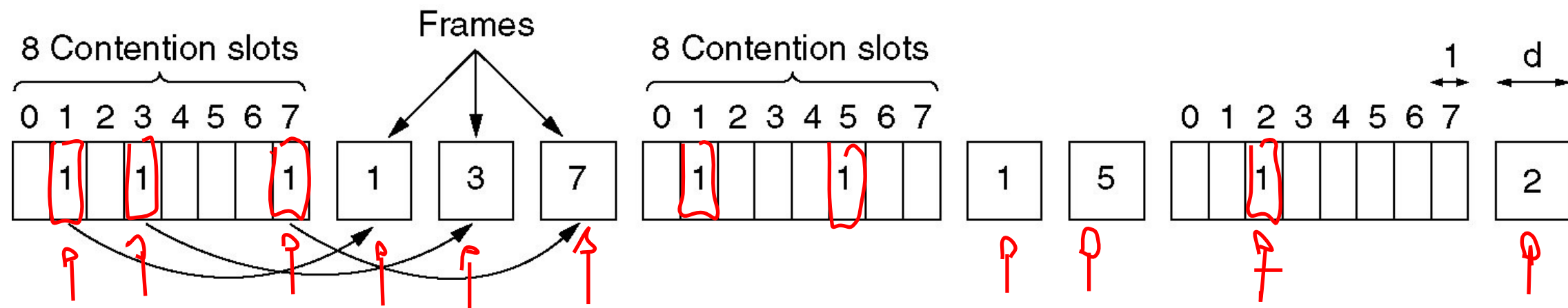


- Statisches Multiplexen
- Dynamische Kanalbelegung
 - Kollisionsbasierte Protokolle
 - Kollisionsfreie Protokolle (contention-free)
 - Protokolle mit beschränktem Wettbewerb (limited contention)

- Einfaches Beispiel: Statisches Zeit-Multiplexen (TDMA)
 - Jeder Station wird ein fester Zeit-Slot in einem sich wiederholenden Zeitschema zugewiesen
- Nachteile bekannt und diskutiert
- Gibt es dynamische kollisionsfreie Protokoll?



- Probleme von TDMA
 - Wenn eine Station nichts zu senden hat, dann wird der Kanal nicht genutzt
- Reservierungssystem: Bit-map protocol
 - Kurze statische Reservierung-Slots zur Ankündigung
 - Müssen von jeder Station empfangen werden



- Verhalten bei geringer Last
 - Falls keine Pakete verschickt werden, wird der (leere) Wettbewerbs-Slot wiederholt
 - Eine Station muss auf seinen Wettbewerbs-Slot warten
 - Erzeugt gewisse Verzögerung (delay)
- Verhalten bei hoher Last
 - Datenpakete dominieren die Kanalbelegung
 - Datenpakete sind länger als die Contention-Slots
 - Overhead ist vernachlässigbar
 - Guter und stabiler Durchsatz
- Bitmap ist ein Carrier-Sense Protokoll!

- Statisches Multiplexen
- Dynamische Kanalbelegung
 - Kollisionsbasierte Protokolle
 - Kollisionsfreie Protokolle (contention-free)
 - Protokolle mit beschränktem Wettbewerb (limited contention)

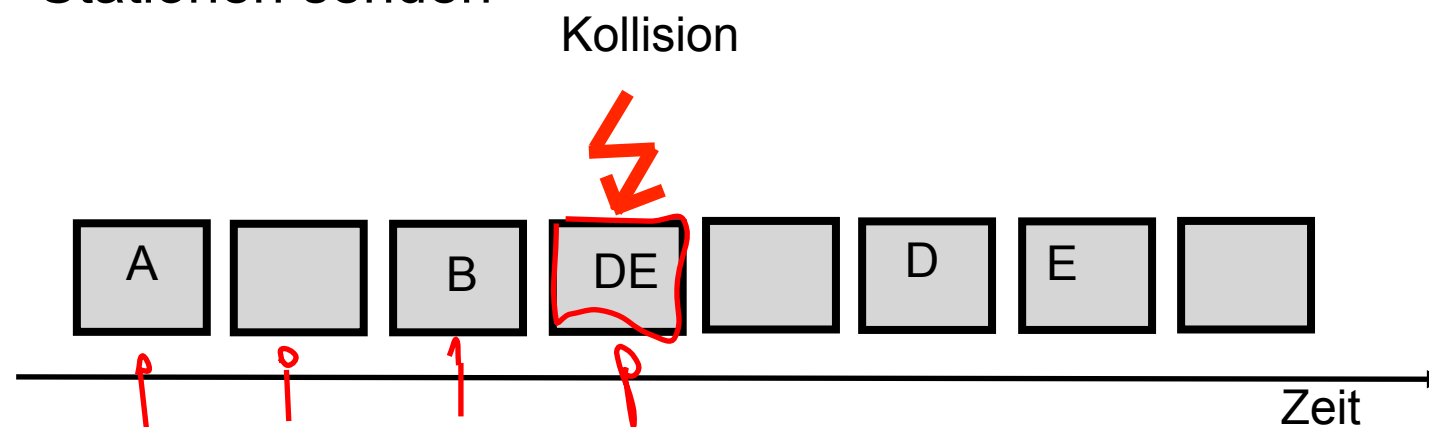
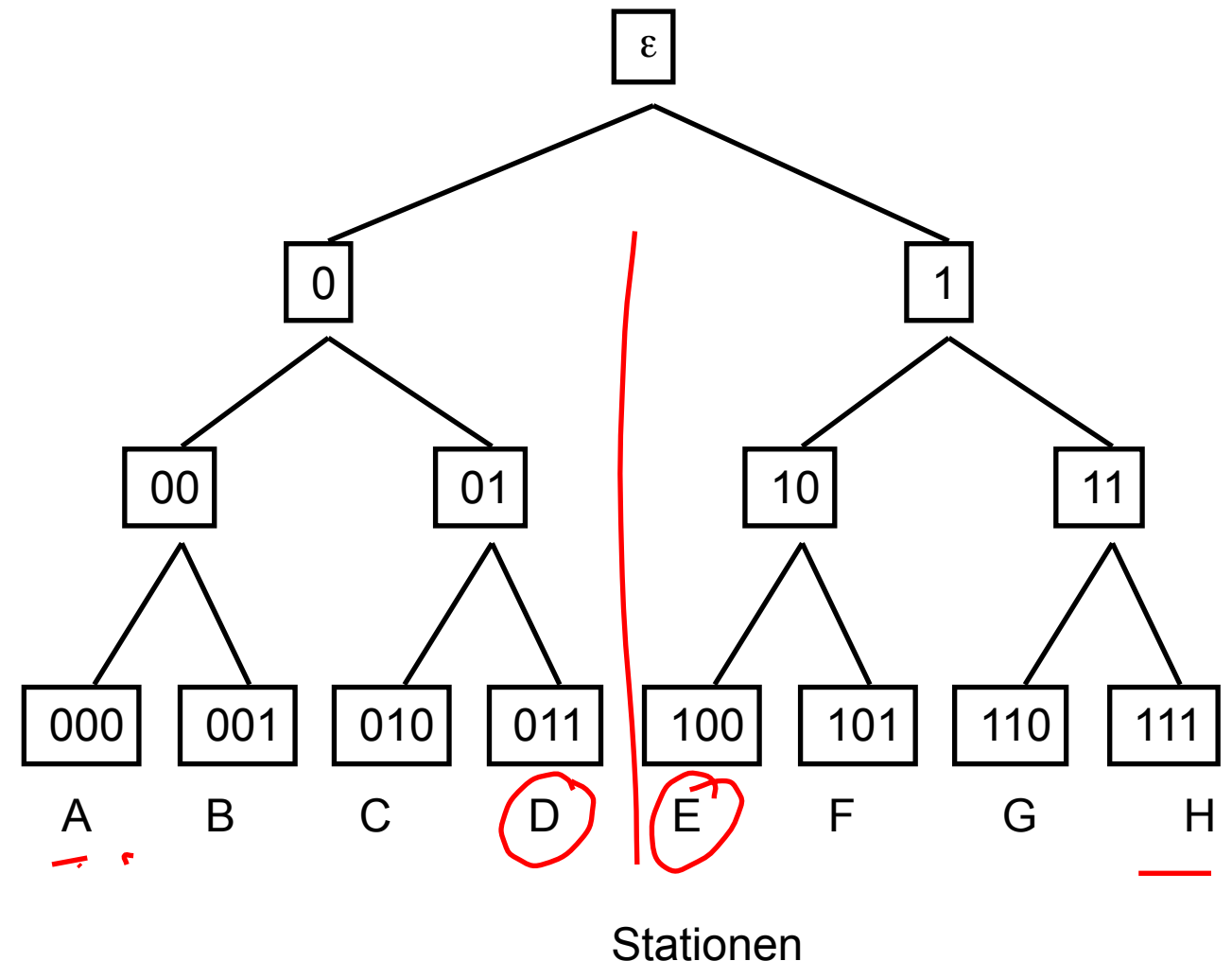
■ Ziel

- geringe Verzögerung bei kleiner Last
 - wie Kollisionsprotokolle
- hoher Durchsatz bei großer Last
 - wie kollisionsfreie Protokolle

■ Idee

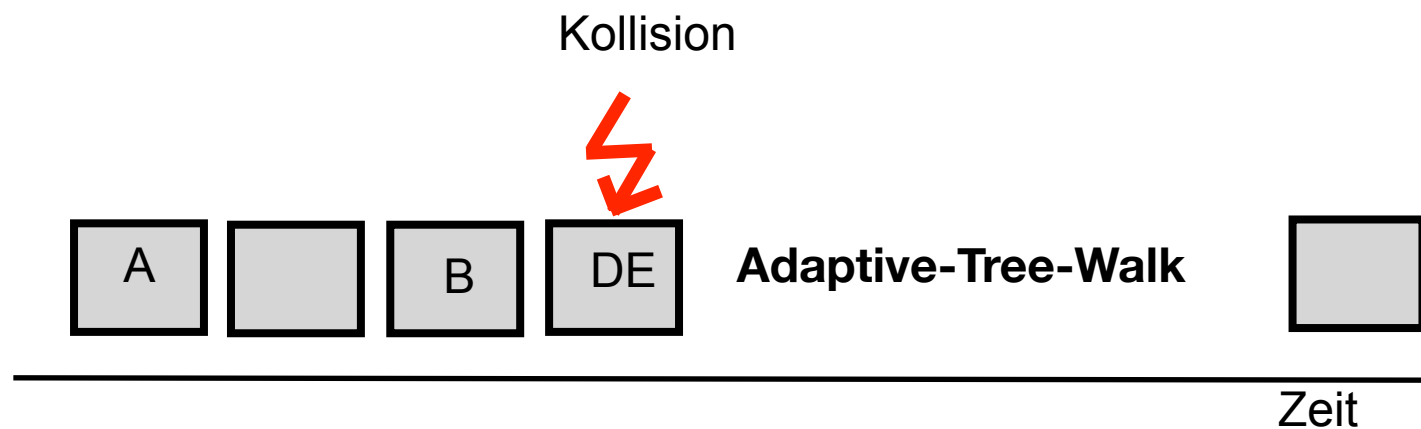
- Anpassung des Wettbewerb-Slots (contention slot) an die Anzahl der teilnehmenden Stationen
- Mehrere Stationen müssen sich dann diese Slots teilen

- Adaptives Baumprotokoll (adaptive tree walk)
- Ausgangspunkt:
 - Binäre, eindeutige Präsentation aller Knoten (ID)
 - Dargestellt in einem Baum
 - Synchronisiertes Protokoll
 - Drei Typen können unterschieden werden:
 - Keine Station sendet
 - Genau eine Station sendet
 - Kollision: mindestens zwei Stationen senden



■ Basis-Algorithmus

- Jeder Algorithmus sendet sofort (slotted Aloha)
- Falls eine Kollision auftritt,
 - akzeptiert keine Station mehr neue Paket aus der Vermittlungsschicht
 - Führe Adaptive-Tree-Walk(ϵ) aus

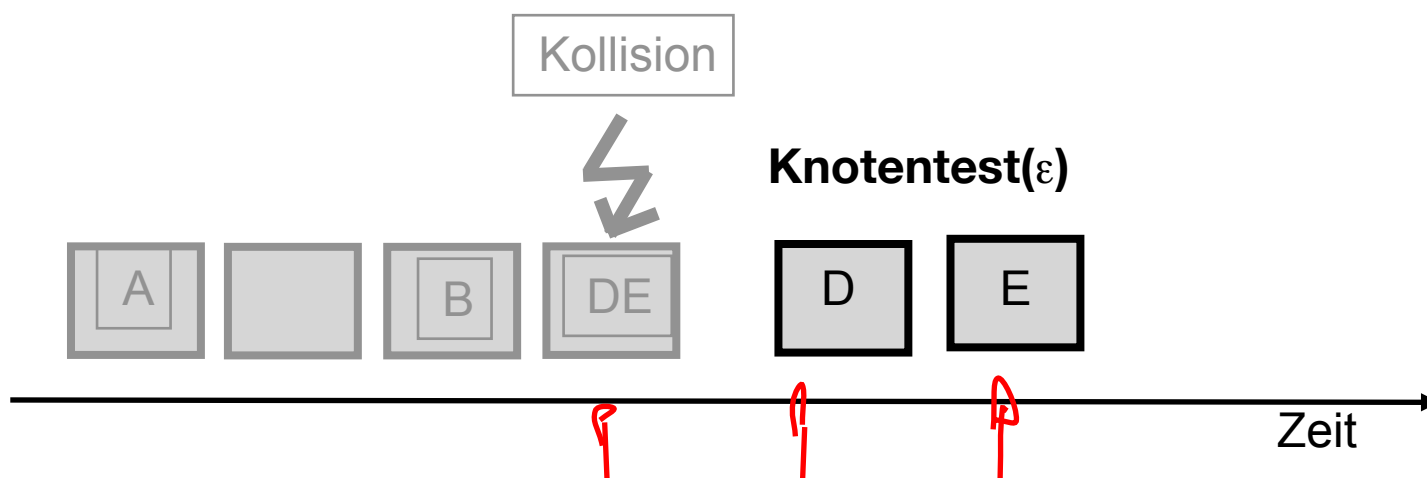
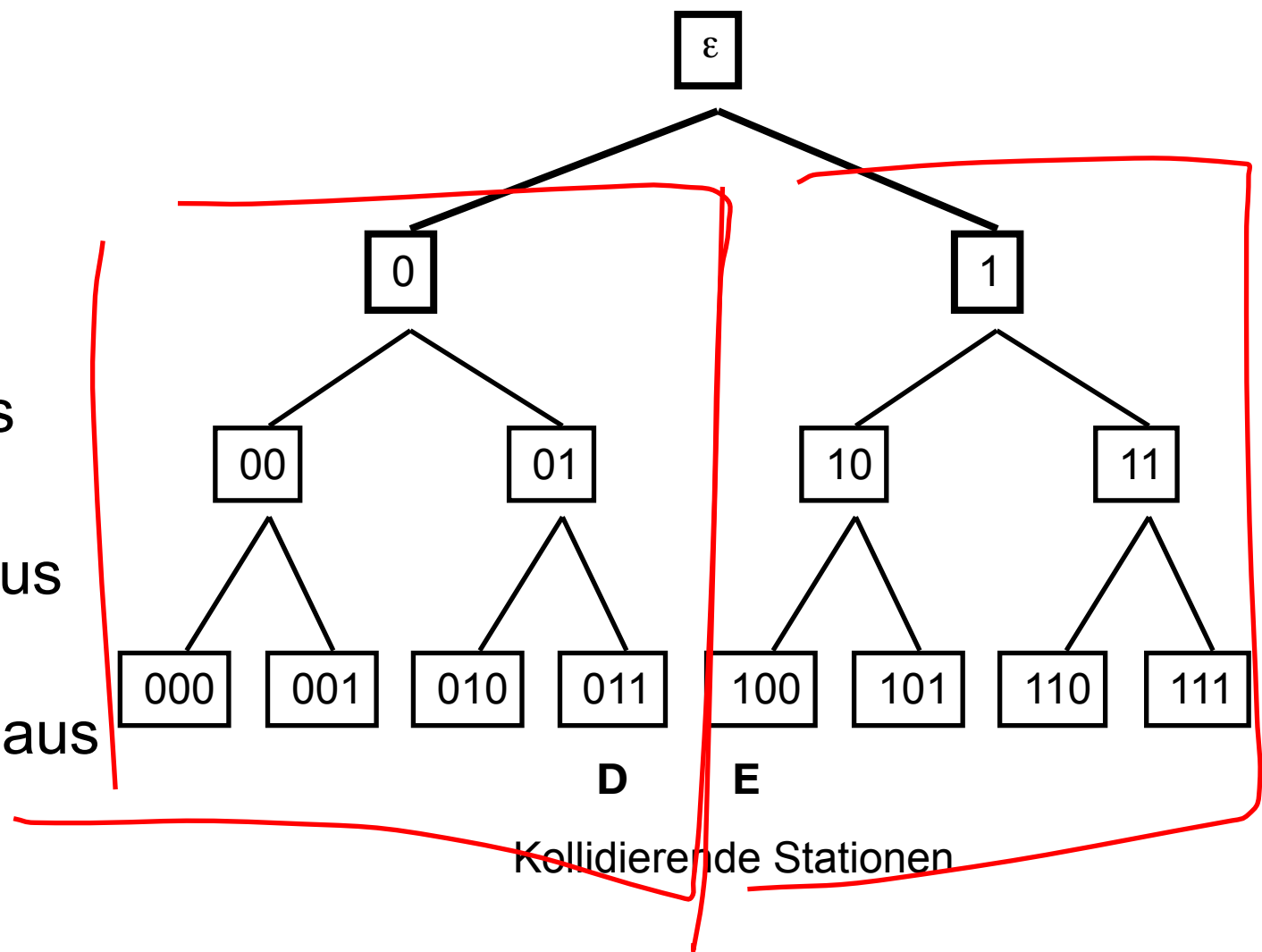


Algorithmus Knoten-Test

- für Knoten u des Baums und
- kollidierende Menge S von Station

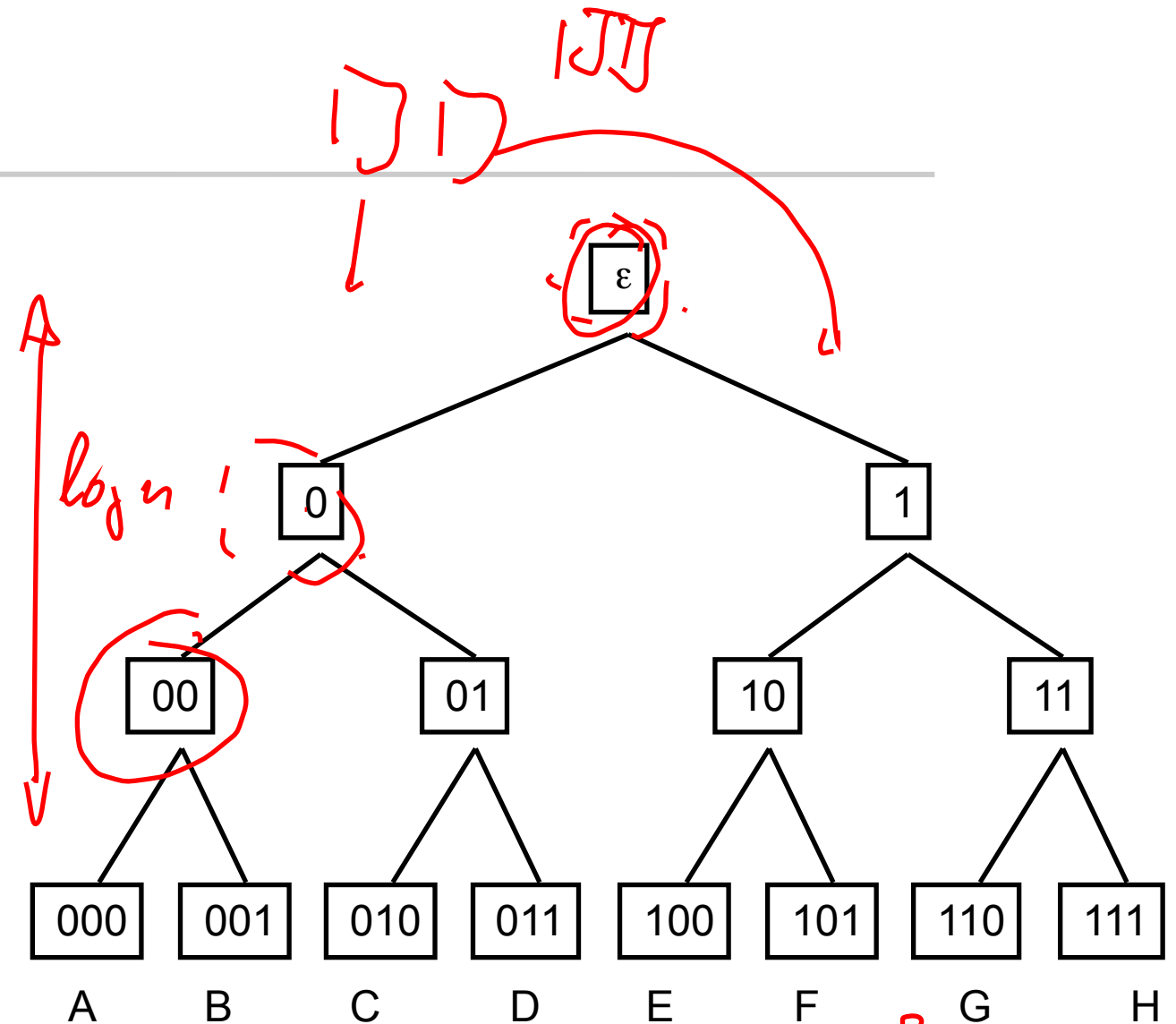
Knoten-Test(u)

- Betrachte zwei Slots pro Knoten des Baums
- Im ersten Slot senden alle Knoten aus S , die mit ID $u0$ anfangen
- Im zweiten Slot senden alle Knoten aus S , die mit ID $u1$ anfangen



Adaptives Baumprotokoll Kern-Algorithmus

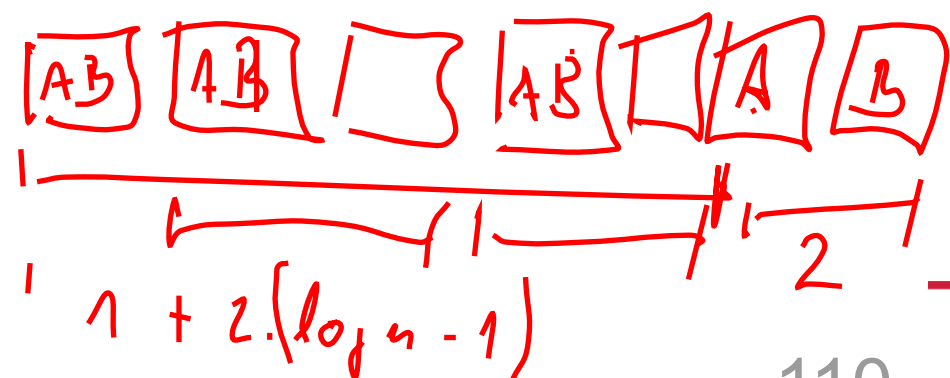
- Algorithmus Knoten-Test
 - für Knoten u des Baums und
 - kollidierende Menge S von Station
- Knoten-Test(u)
 - Betrachte zwei Slots pro Knoten des Baums
 - Im ersten Slot senden alle Knoten aus S , die mit ID $u0$ anfangen
 - Im zweiten Slot senden alle Knoten aus S , die mit ID $u1$ anfangen
- Adaptive Tree Walk(x)
 - Führe Knoten-Test(x) aus
 - Falls Kollision im ersten Slot,
 - führe Adaptive-Tree-Walk($x0$) aus
 - Falls Kollision im zweiten Slot,
 - Führe Adaptive-Tree-Walk($x1$) aus



n Teilnehmern

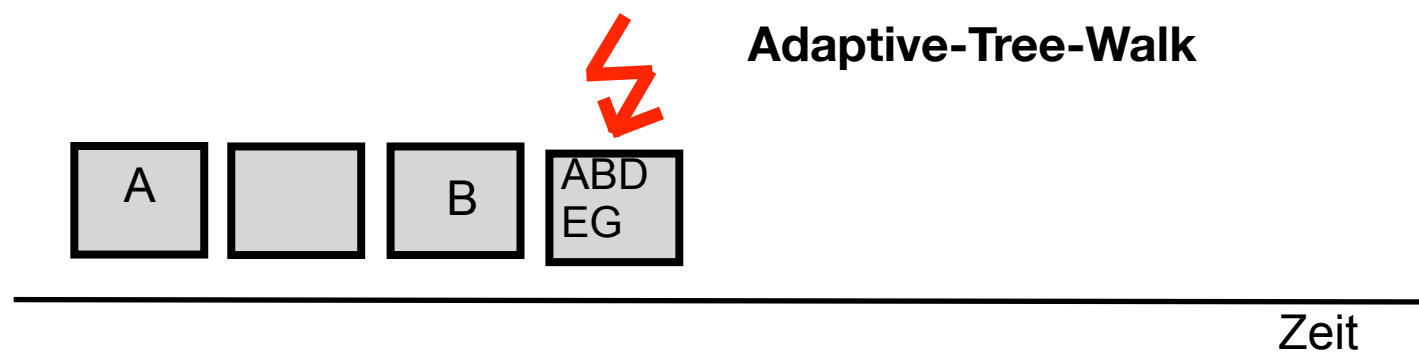
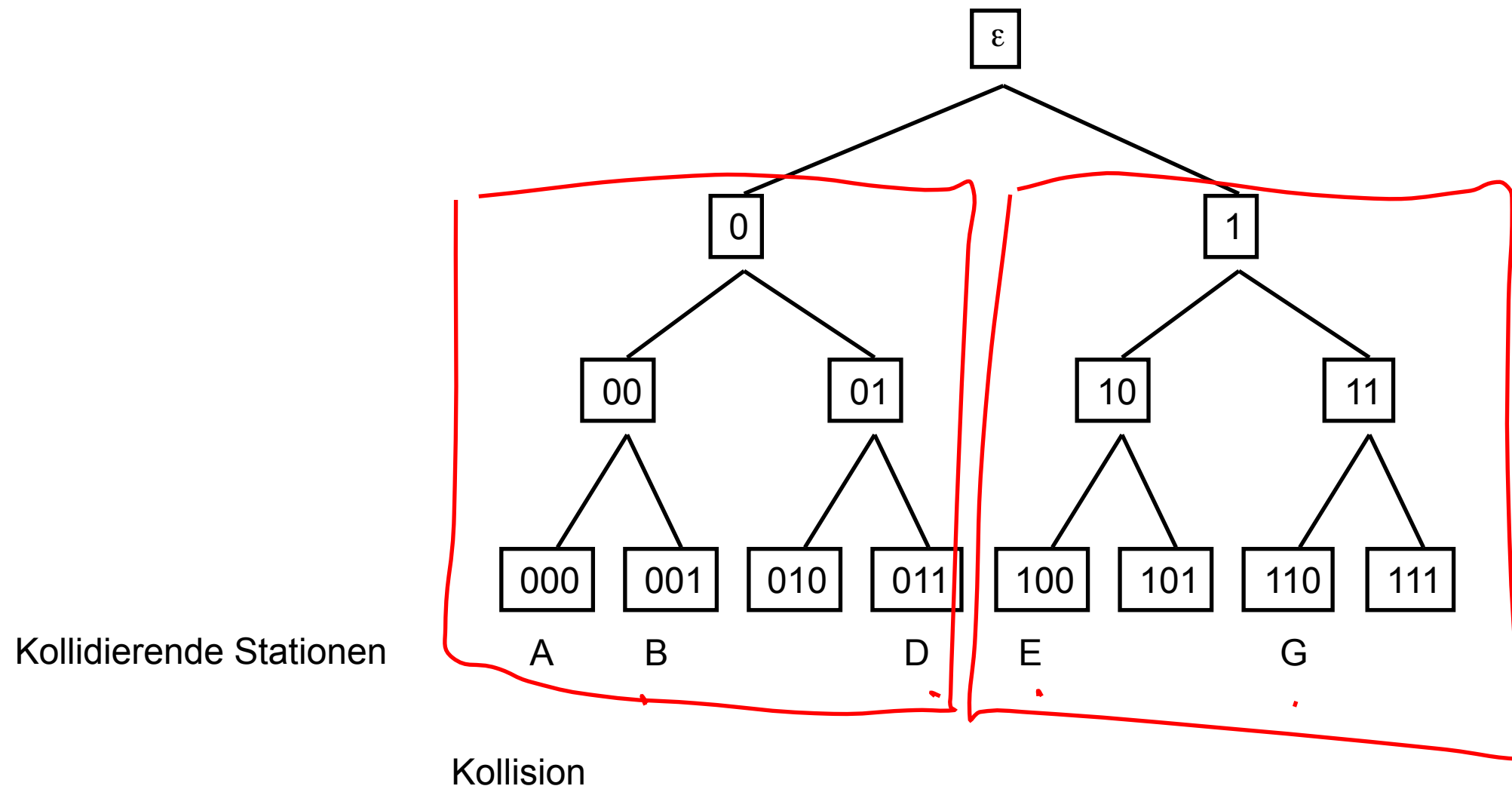
Stationen

$$2 \log n + 1$$

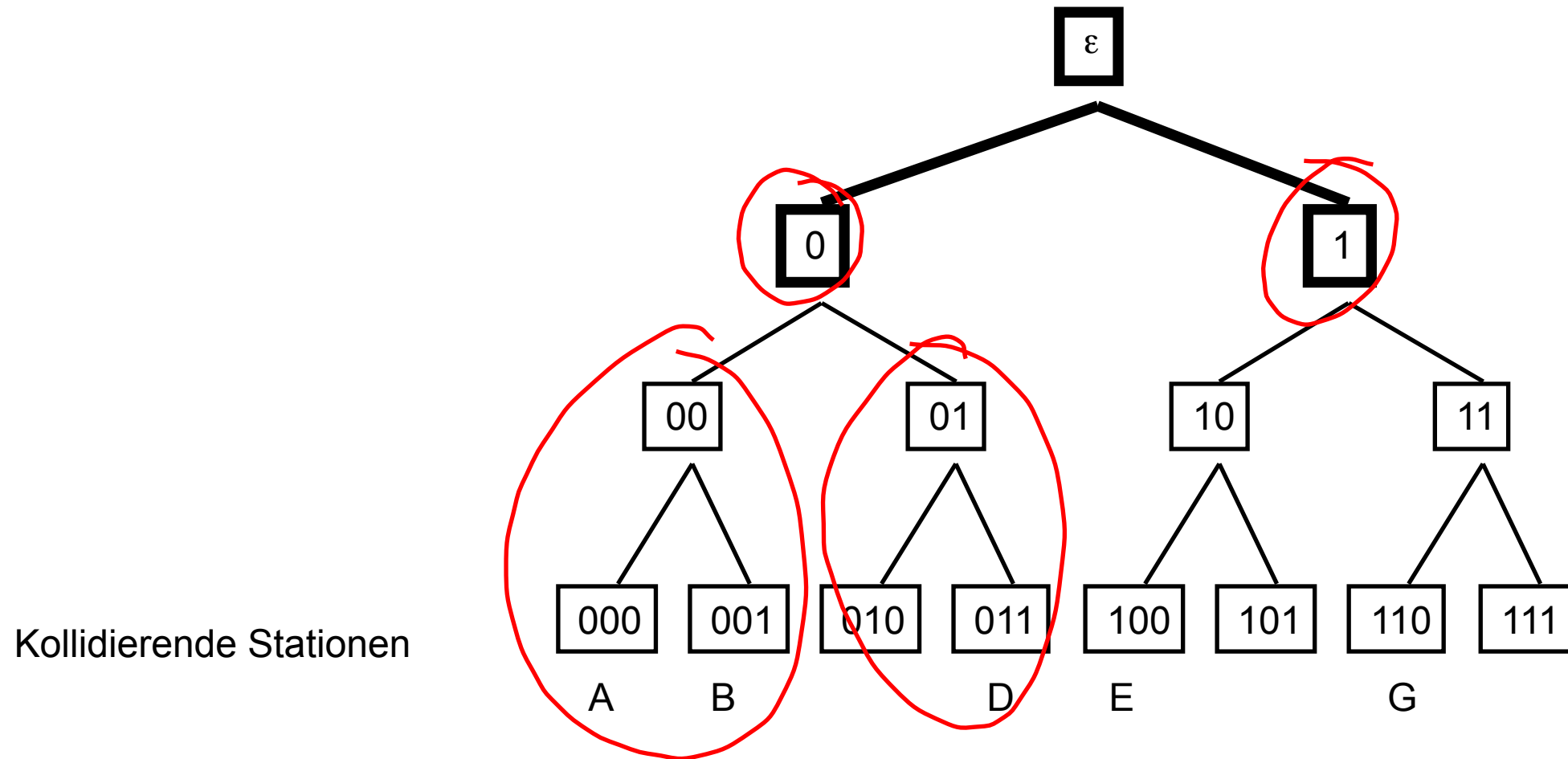


110

Adaptives Baumprotokoll Beispiel (1)

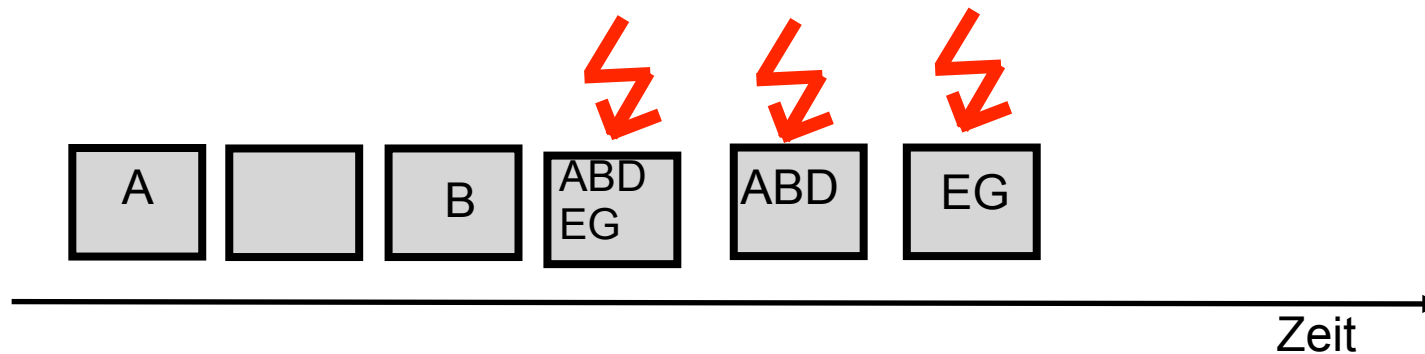


Adaptives Baumprotokoll Beispiel (2)

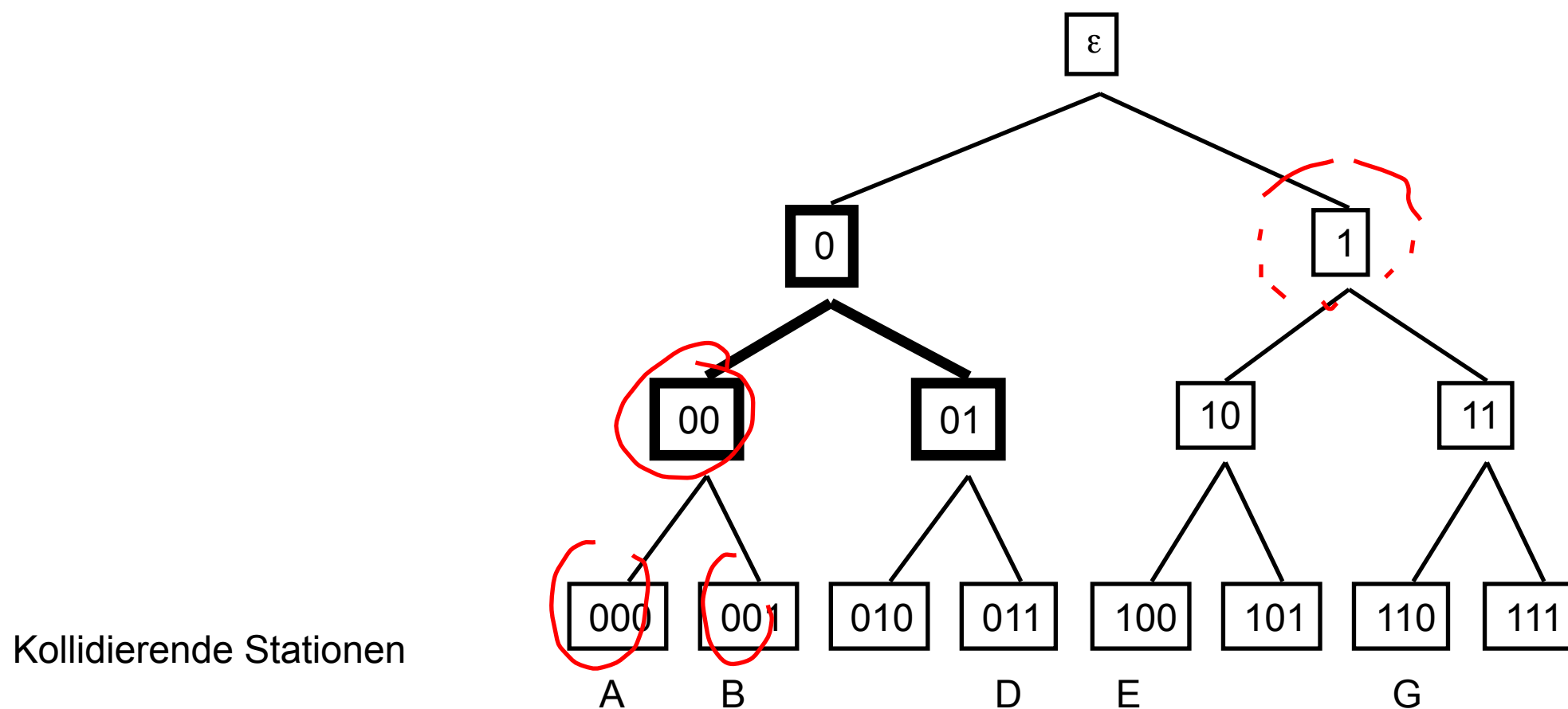


Adaptive-Tree-Walk

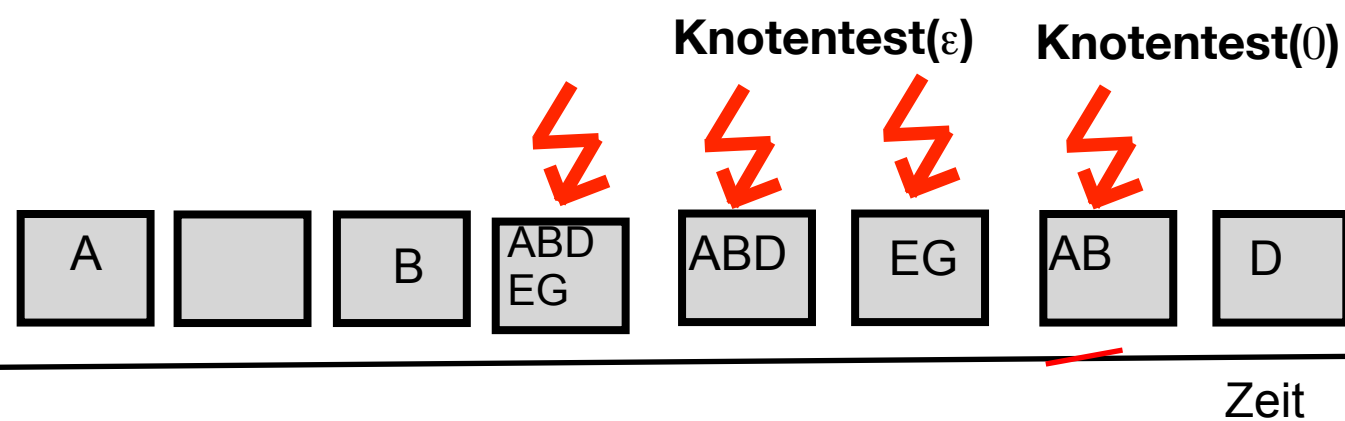
Knotentest(ϵ)



Adaptives Baumprotokoll Beispiel (3)

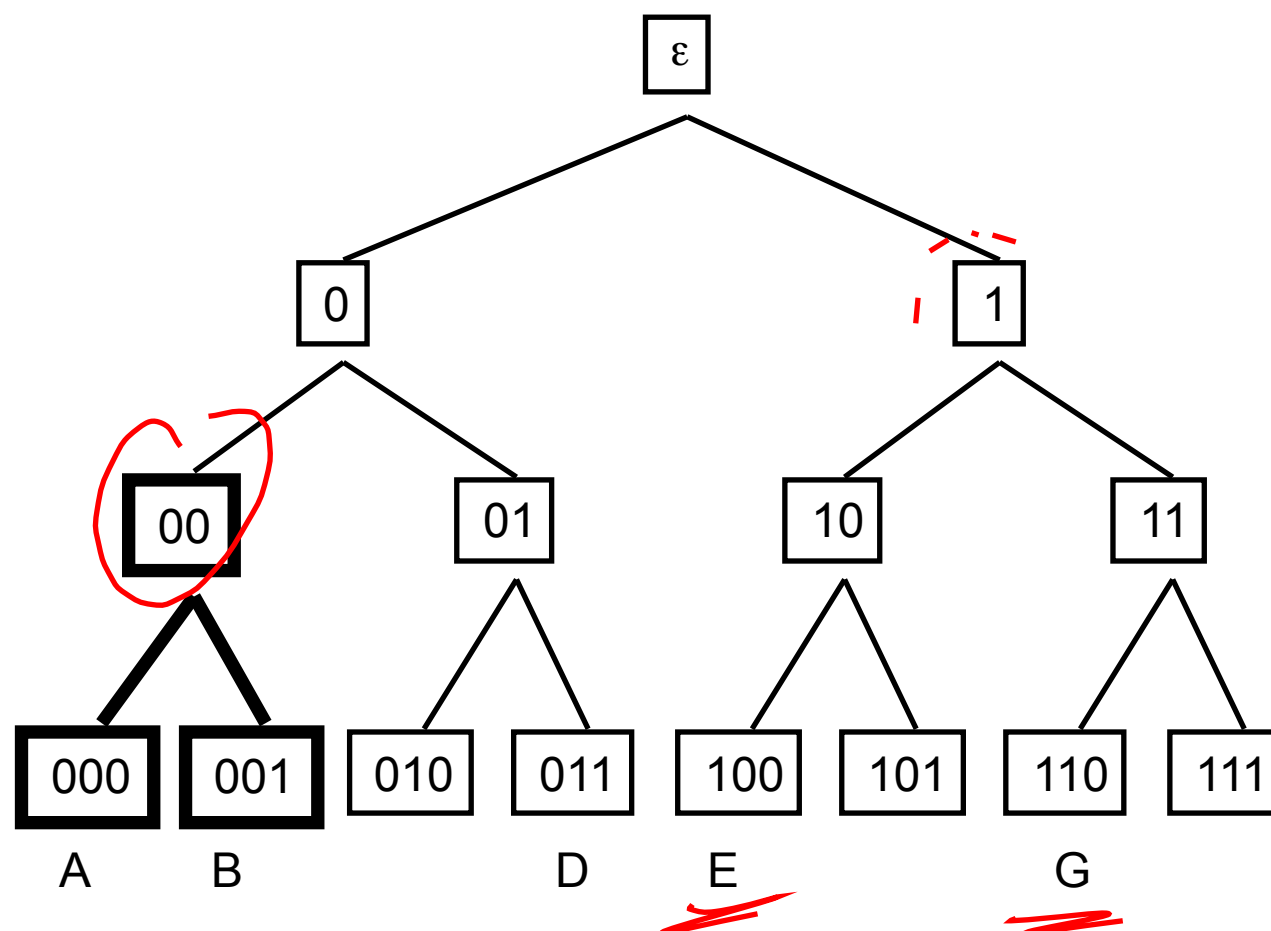


Adaptive-Tree-Walk



Adaptives Baumprotokoll

Beispiel (4)



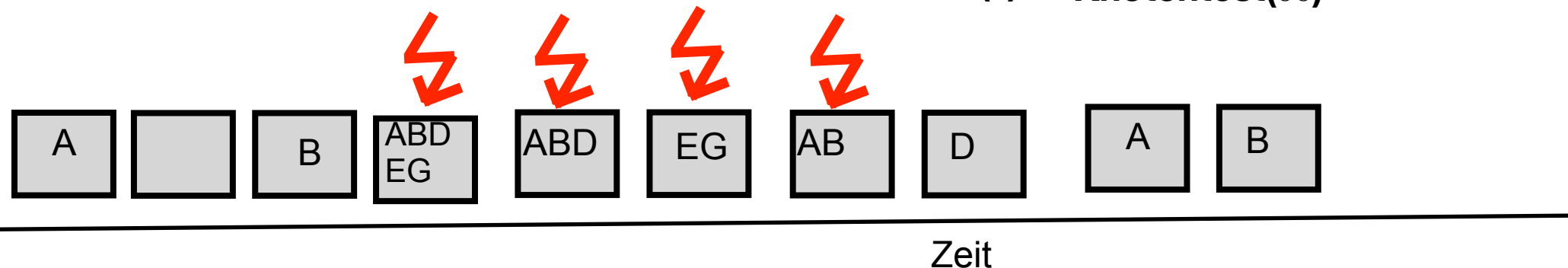
Kollidierende Stationen

Adaptive-Tree-Walk

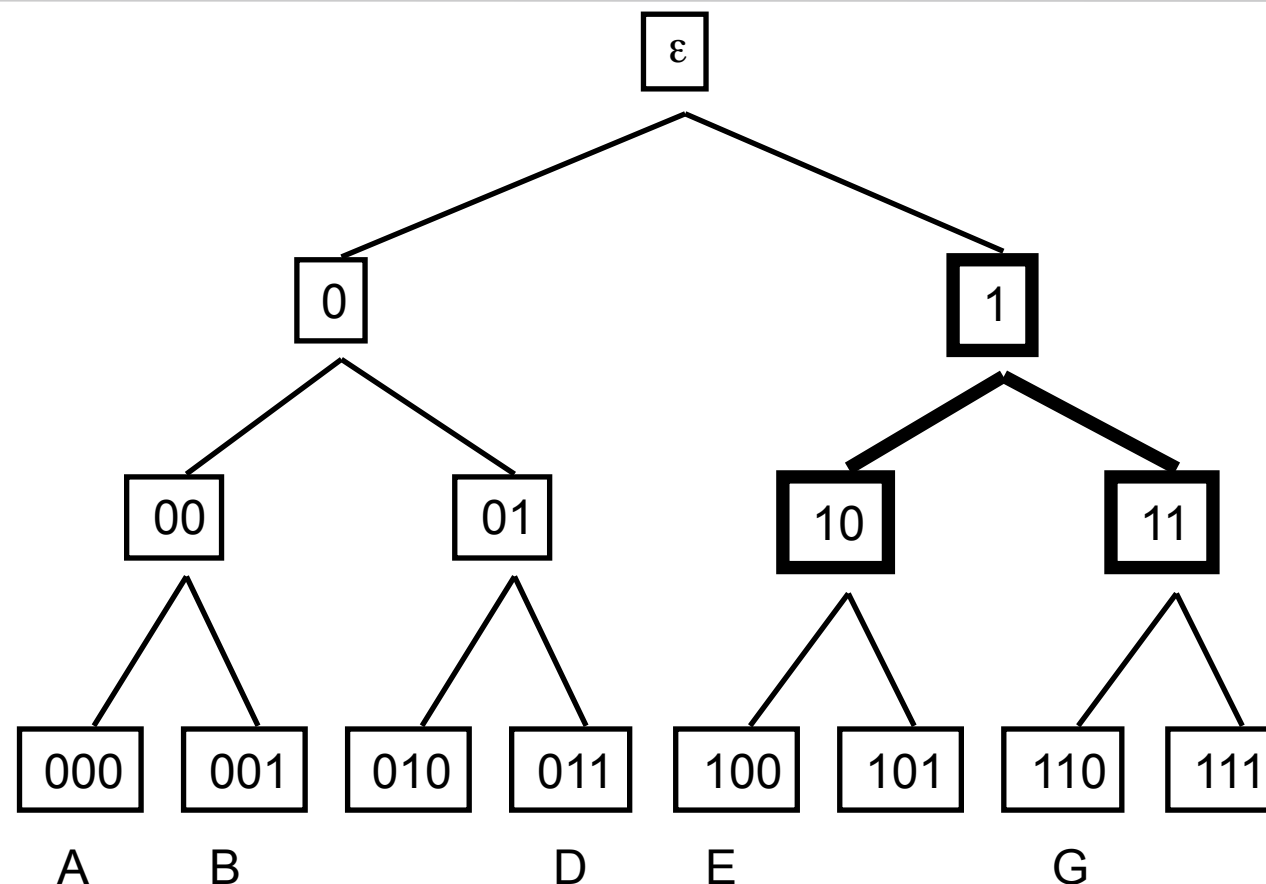
Knotentest(ϵ)

Knotentest(0)

Knotentest(00)

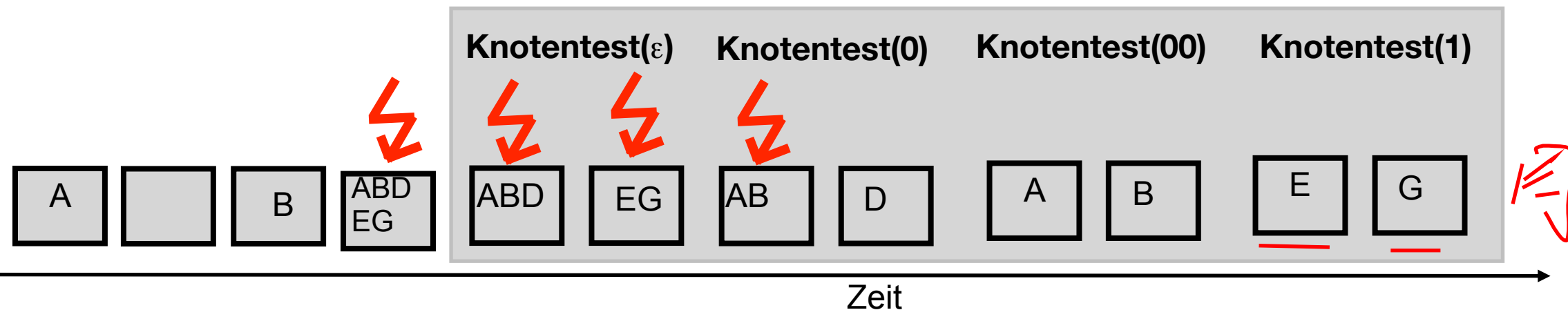


Adaptives Baumprotokoll Beispiel (5)



Kollidierende Stationen

Adaptive-Tree-Walk



Systeme II

3. Die Datensicherungsschicht

Christian Schindelhauer

Technische Fakultät

Rechnernetze und Telematik

Albert-Ludwigs-Universität Freiburg