

# Workshop Simulation

## *TCPSim – Ein Simulator für TCP ohne Zufallsgenerator*

*Thorsten Müller und Jörg Schüler*



**VIERLING**

radioplan

# Gliederung

- **Motivation**
  - Vom Verhalten von TCP zur Performance
- **Aufbau des Simulators**
  - Elemente der Übertragungsstrecke
  - Ereignisgesteuerter Ablauf
  - TCP-Varianten
- **Ergebnisse**
  - Ergebnisanalyse
  - Untersuchungsbeispiel
- **Zusammenfassung**

# Motivation

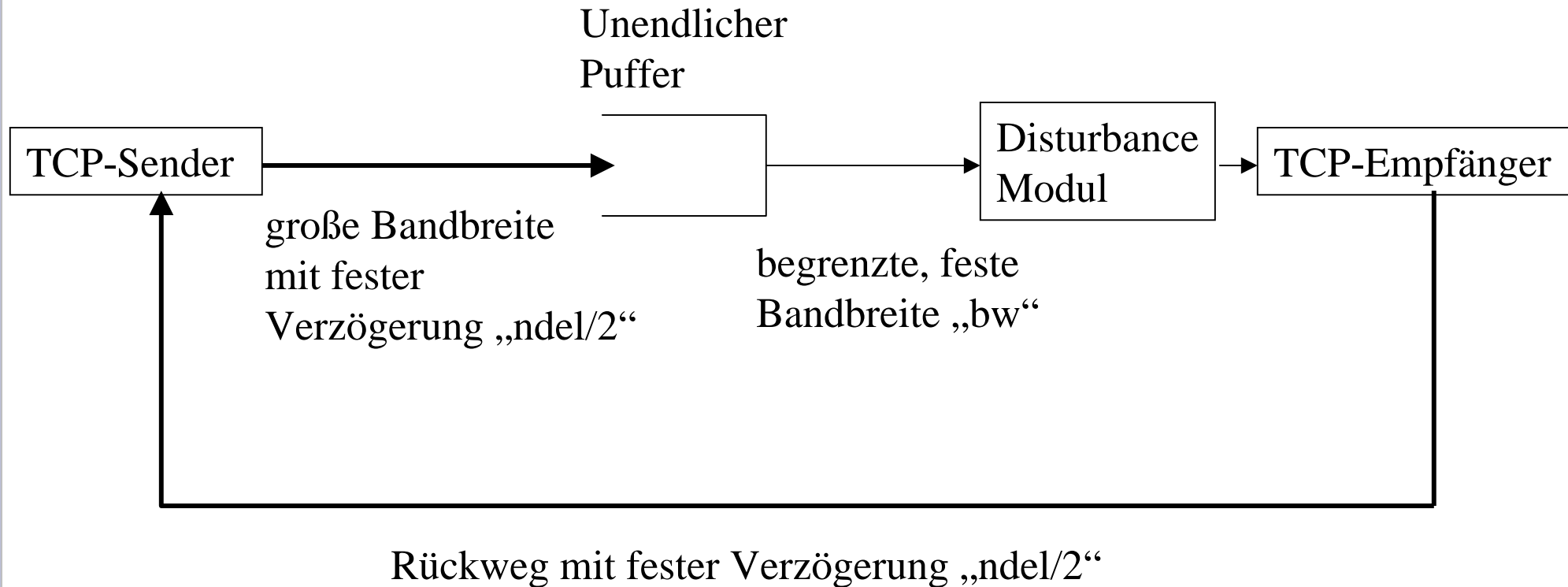
# Motivation

- Aussagen über die Performance von TCP
- Ableiten der Performance aus analytische Beschreibungen des Verhaltens von TCP
- Untersuchung der Reaktionen von TCP auf
  - Segmentverlusten
  - Segmentvertauschungen
  - Segmentverzögerungen.
- Schritte zur analytischen Beschreibung:
  - Segmentgenaue Untersuchungen von verschiedenen TCP-Varianten
  - Verschiedene Fehlerursachen und Fehlerbilder
- Verständnis für die Algorithmen von TCP
  - Empfangspuffer

# Aufbau des Simulators

# Aufbau des Simulators

- Elemente der Übertragungsstrecke



# Aufbau des Simulators

- Modellierungsebene: Schicht 4 (TCP)
- Ereignisgesteuerte Ablaufsteuerung
  - Simulation in virtueller Zeit
  - Unabhängig von der Rechnerleistung
- Kein zufallsgesteuerter Simulator sondern Protokoll-Emulator
- Programmiersprache ANSI-C (Portabilität)
- Ausgabe in ASCII-Tracefiles (vgl. ns-2)
- Zusätze: perl- & shell-Skripte

# Aufbau des Simulators

- Implementierte TCP-Sender-Varianten
  - OldTahoe (RFC 793, RFC 1122)
  - Tahoe (RFC 1122)
  - Reno (RFC 2581)
  - NewReno impatient (RFC 3782)
  - NewReno slow-but-steady (RFC 3782)
- Genaue Implementierung von
  - TCP-Empfänger (RFC 2581)
  - RTO-Timer (RFC 2988)

# Aufbau des Simulators

- **Parametrisierung über ASCII-Dateien:**
  - definitions.ini: Parameter für den Simulator
  - errorevents.ini: Art und Anzahl der Störungen
- **Ausgabe: Tracefile im ASCII-Format**
  - Grafische Darstellung mit Skripten und Gnuplot

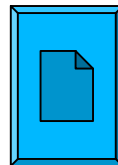
# Ergebnisse

# Ergebnisse

```
# Duration of simulation [ms]:          500000
# Size of receiver buffer [Bytes]:     65535
# Initial TCP ssthresh value [Bytes]:  1448
# TCP maximum segment size [Bytes]:   1448
# TCP clock granularity [ms]:         100
# Static network delay [ms]:          200
# Bandwidth of bottleneck link [bit/s]: 64000
# Trace Pattern:                       11111111
#
#          # sent packets | received acks | discard comments | timeout comments |
#          # fast retransmission comments | fast recovery comments |
#          # fast recovery full/part ack | markov_comments
# TCP Version:  2          # 1 == OldTahoe, 2 == Tahoe, 3 == Reno
#          #          # 4 == NewReno-impatient, 5 == NewReno-slow-but-steady
#
# Target window size [packets] and loss pattern [packet numbers starting at target]:
# 15 :  1 10 11 12
```

# Ergebnisse

#	Simtime	Seq/Ack	No	Rec Wnd	T_stamp	Cong Wnd	SSthresh	RTOval	RTOexp	QueLen	DAcks
0	<--	0 :0	45	:65535	0	1 :1448	1 :1448	3000	0	0 :0	0
0	-->	1 :1			0	2 :2896	1 :1448	3000	3000	0 :0	0
0	-->	2 :1449			0	2 :2896	1 :1448	3000	3000	0 :0	0
381	<--	1 :1448	45	:65535	0	2 :2896	1 :1448	3000	3000	1 :1448	0
381	-->	3 :2897			381	2 :3620	1 :1448	1141	1522	1 :1448	0
562	<--	2 :2896	45	:65535	0	2 :3620	1 :1448	1141	1522	0 :0	0
562	-->	4 :4345			562	2 :4199	1 :1448	1151	1713	0 :0	0
762	<--	3 :4344	45	:65535	381	2 :4199	1 :1448	1151	1713	1 :1448	0
762	-->	5 :5793			762	3 :4698	1 :1448	980	1742	1 :1448	0
762	-->	6 :7241			762	3 :4698	1 :1448	980	1742	1 :1448	0
943	<--	4 :5792	45	:65535	562	3 :4698	1 :1448	980	1742	0 :0	0
943	-->	7 :8689			943	3 :5144	1 :1448	849	1792	0 :0	0

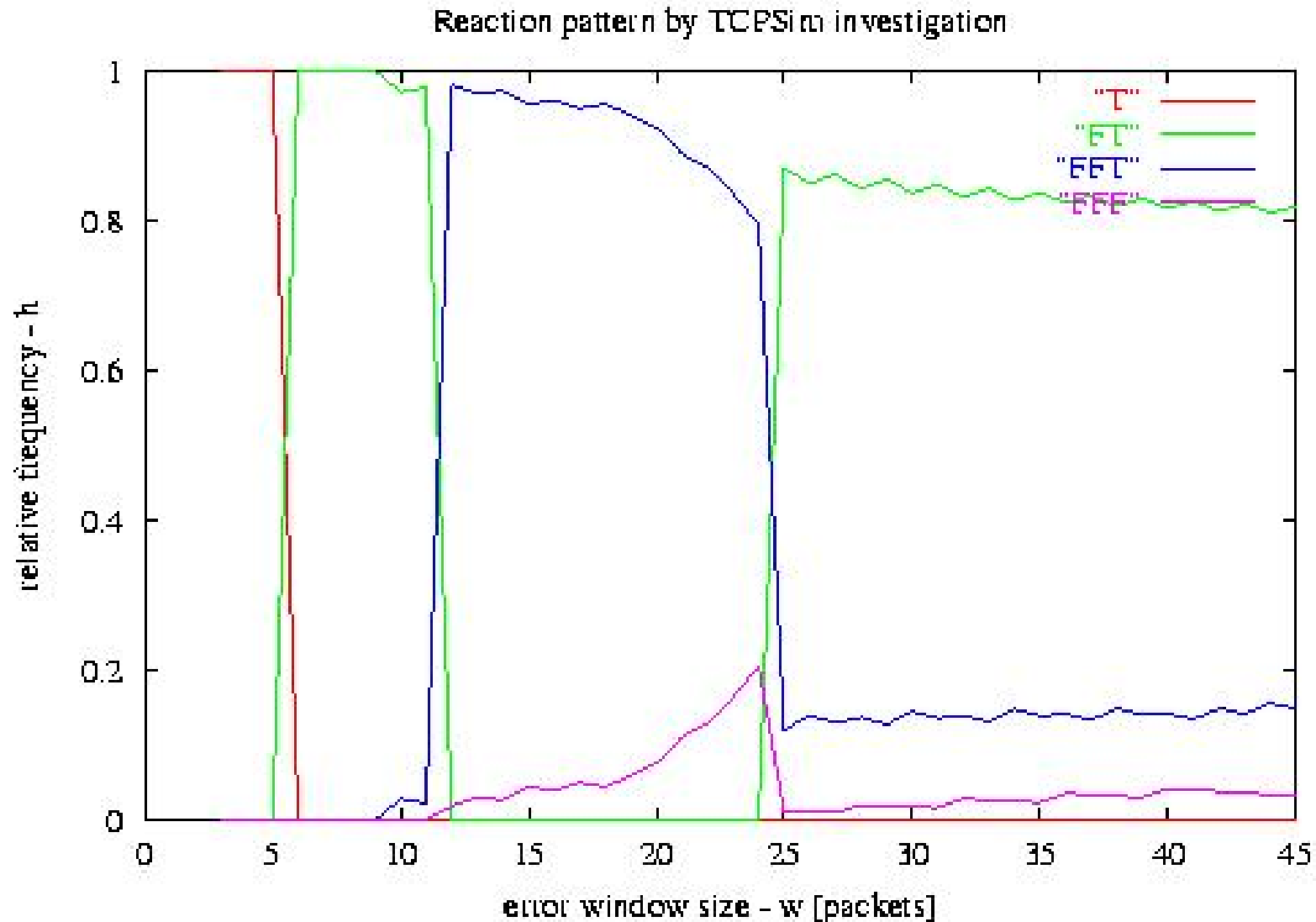


# Spurious Fast Retransmissions

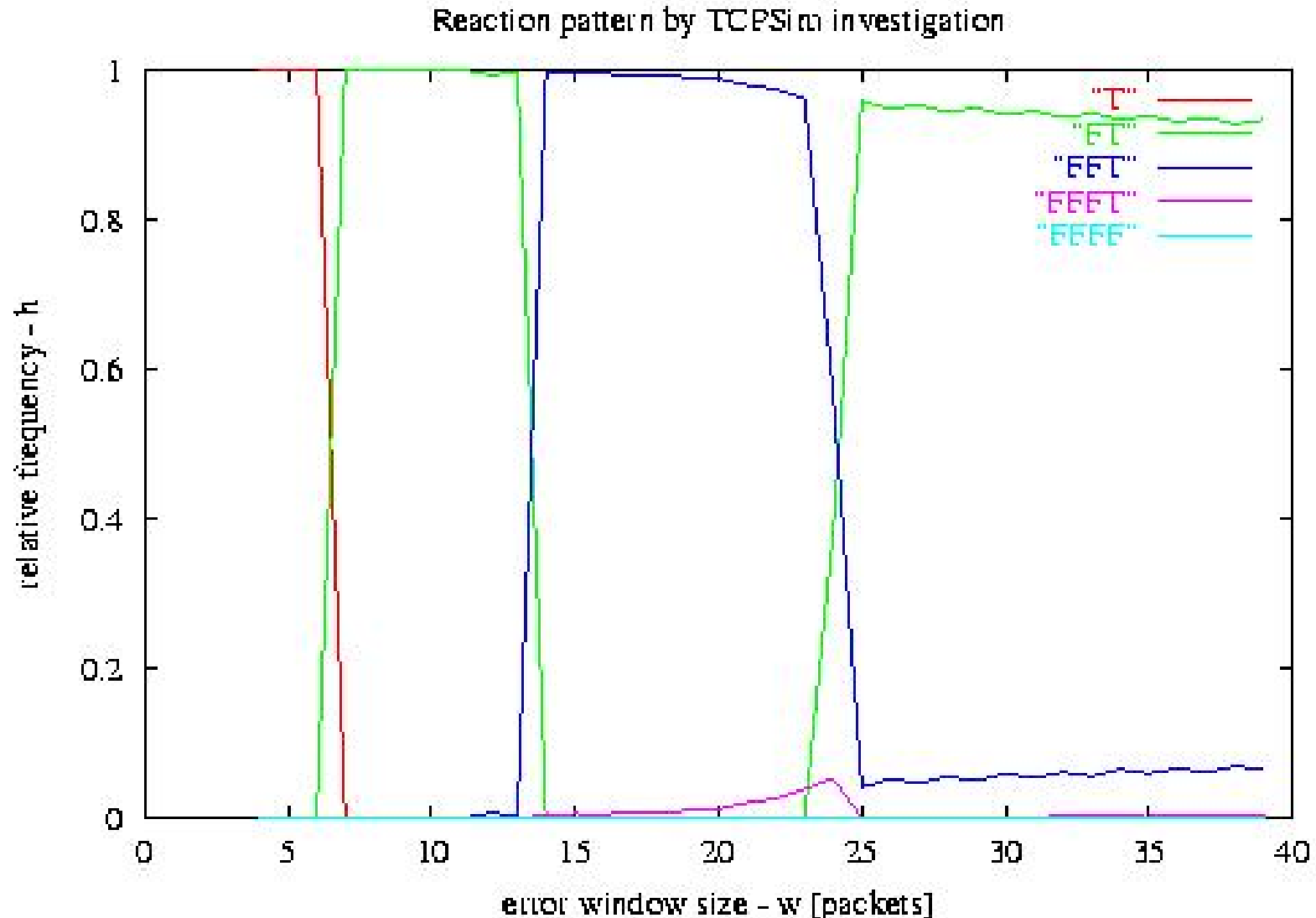
Auftreten:

- TCP-Tahoe
- Verlorene Segmente:  
124, 133, 134, 135
- Als verloren detektierte Segmente:  
124, 139, 144, 147
- Grund: Segmente werden im Empfangspuffer gespeichert und verursachen duplicate acknowledgements!

# Relative Frequency of detection with 3 drops



# Relative Frequency of detection with 4 drops



# Zusammenfassung

- Kleines, schnelles Analysetool (33kB)
- Genaue Implementierung der relevanten RFCs
- Berücksichtigung wichtiger Segmentstörungen
  - Segmentverluste
  - Segmentvertauschungen
  - Segmentverzögerungen
- Segmentgenaue Beobachtung von TCP
- Detaillierte Verhaltensanalyse von TCP
- Unterstützung einer analytischen Modellierung des Verhaltens von TCP

# Ausblick

- Aussagen über die Performance folgen aus der Analyse
- Beschreibung von Zuständen mit Störungen in einer TCP-Markov-Ketten-Modell

Herzlichen Dank für Ihre Aufmerksamkeit!

Haben Sie Fragen?

# RTO Timer

- For stable RTT values in the longrun the value of RTO tends to the RTT value leading to fast Timeouts for some segments
- This is prevented by several mechanism (RFC 2988)
  - The variation of the measured RTT values are set to a minimum
  - The RTO value itself has a minimum value
  - RTO timer is set back every time a segment leaves the sender
  - Only one RTO timer for each connection so only the previously sent packet is measured with the “right” RTO value