PRISM

TCP Performance over Multilink Long Delay Wireless Networks

Said Zaghloul

Candidate for MS in Computer Engineering

Committee

Dr. Victor Frost

Dr. Gary Minden

Dr. David Petr

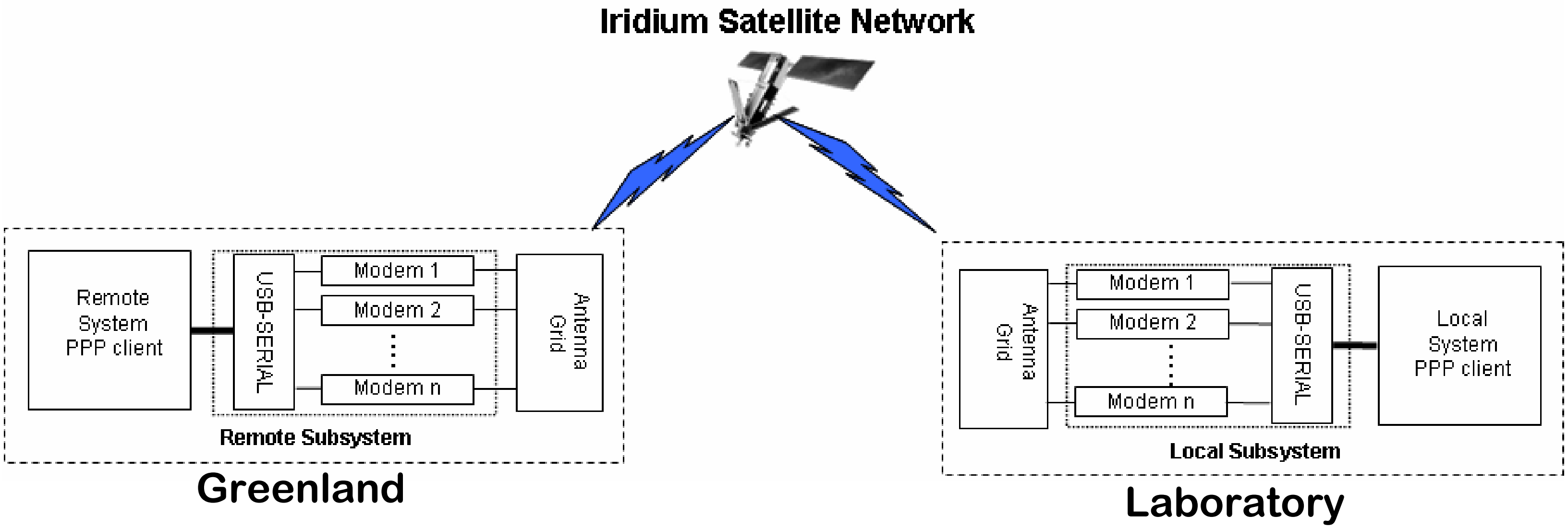
Outline

- Objectives
- Overview of the Iridium satellite system
- MLPPP and TCP overview
- Call drops
- Modeling TCP performance over MLPPP
- Model validation and extension
- Link management software
- ARQ Modeling
- Conclusions
- Future work

Research Objectives

- To develop and experimentally validate a model to predict the file transfer time for large files using TCP over multilink Point-to-Point systems
- To study the effect of several parameters on TCP performance using the developed model. The parameters of interest are:
    - Packet loss rate
    - Timeout value
- To develop an efficient and user friendly link management software for the Multilink-Iridium System
- To investigate the effect of the ARQ (Automatic Repeat Request) on the RTT in terms of physical layer parameters

The Iridium Satellite System

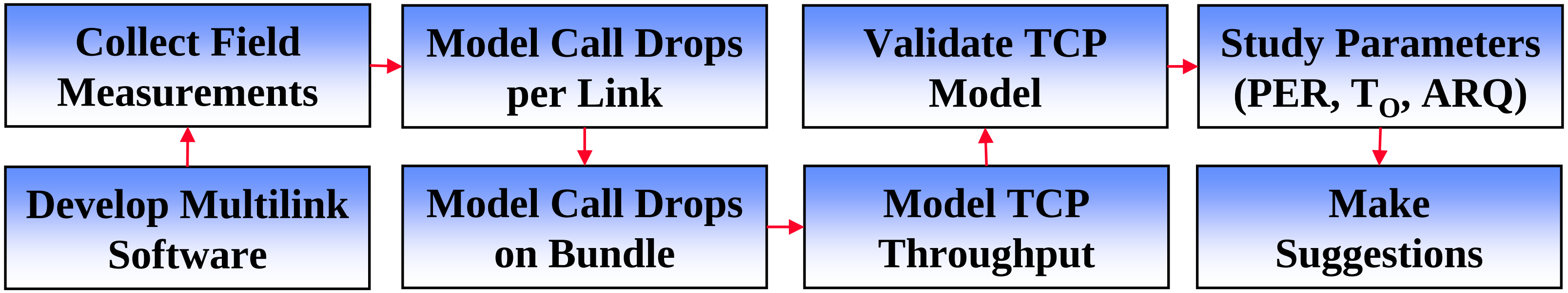


- ➢ MLPPP was used to satisfy the applications’ requirements
- ➢ System was tested successfully in Greenland in 2004 with 8 modems and throughput of 18.6 kbps
- ➢ TCP over MLPPP performance model was developed and experimentally validated using the Multilink-Iridium system

The Iridium Satellite System

Research Roadmap

- TCP throughput is mainly affected by packet losses which might be due to congestion, due to wireless errors, etc
- All available models discuss TCP performance over a single link
- No models for TCP performance over multiple circuit switched links
- Individual links in the MLPPP bundle may experience call drops
- A call drop affects TCP throughput
- Need to predict TCP throughput taking call drops into account



Develop Multilink Software

Collect Field Measurements

Model Call Drops per Link

Model Call Drops on Bundle

Model TCP Throughput

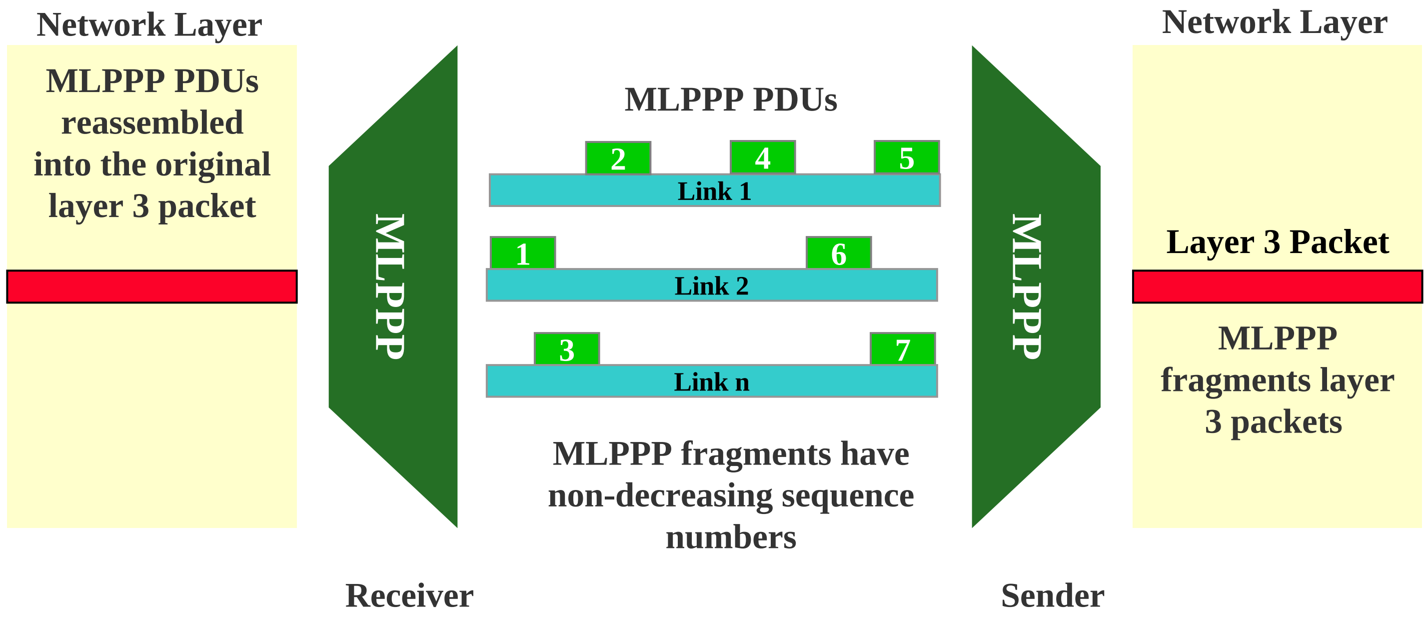
Validate TCP Model

Study Parameters (PER, T_O, ARQ)

Make Suggestions

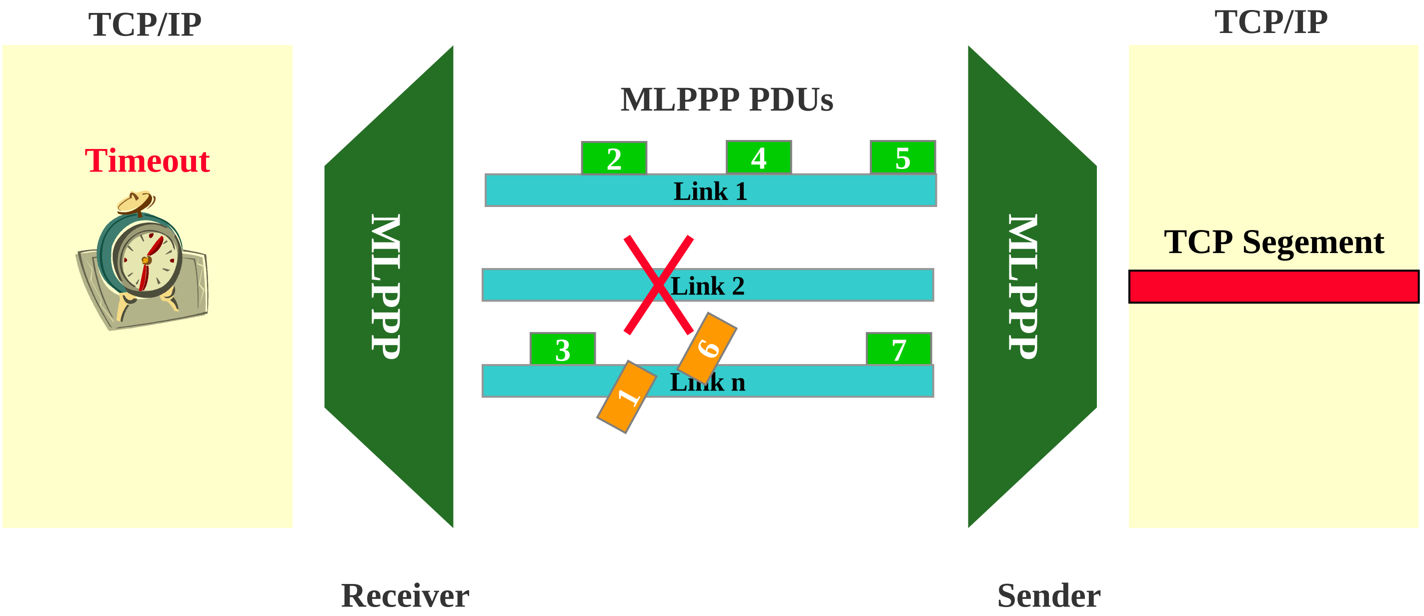
Multi-Link Point-to-Point Protocol

- Multilink option and MRRU are negotiated when establishing the connection.
- Packets may be fragmented.



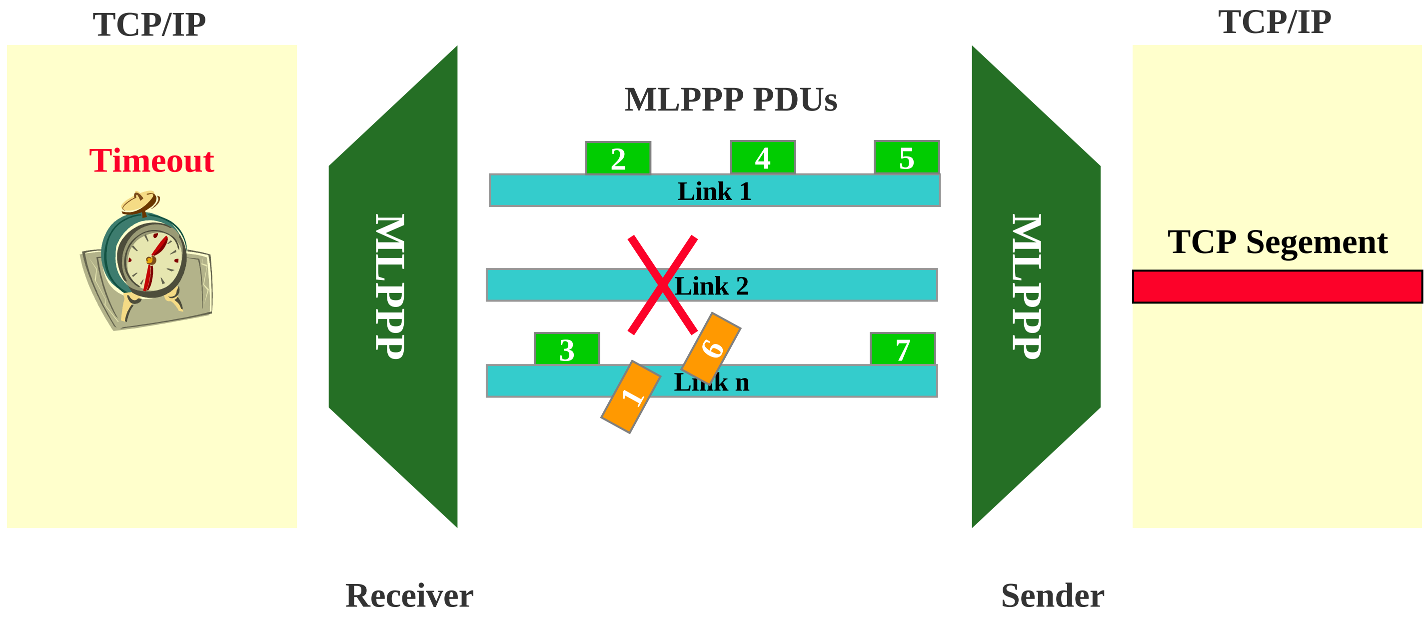
Multi-Link Point-to-Point Protocol

- When links go down, TCP times out
- To model TCP transfer time, it is important to model call drops (i.e., timeouts!)



Multi-Link Point-to-Point Protocol

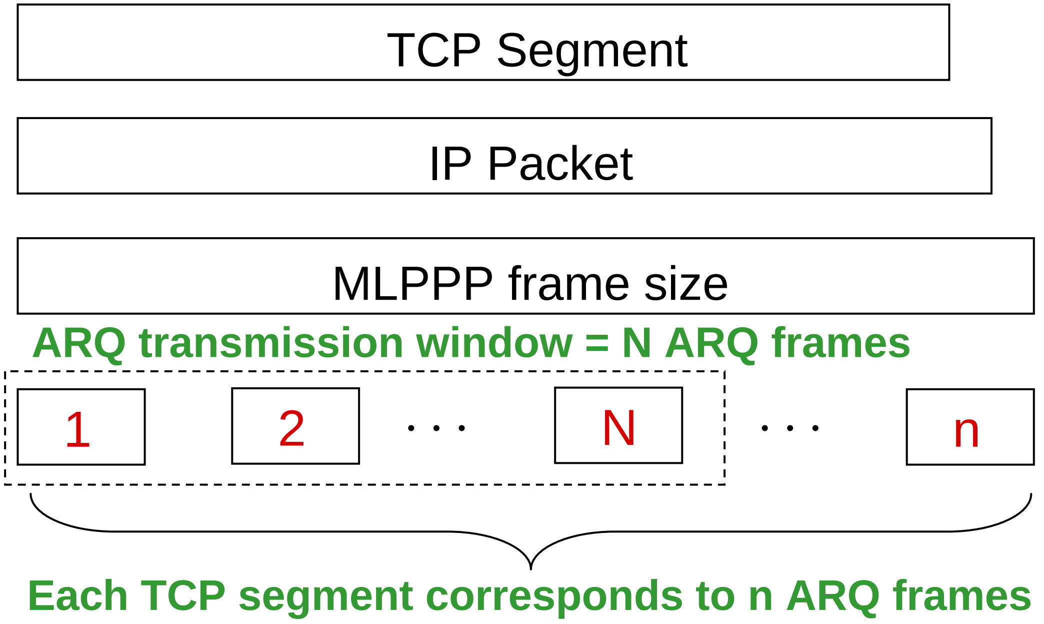
File Transfer Time over MLPPP



TCP Over Wireless Channels

- Performance is very sensitive to PER
  - TCP performs poorly if it is run directly over wireless links [10% losses degrades Reno’s performance by (50-75%)]
- Physical layer ARQ provides a reliable channel for TCP
  - ARQ types:
    stop-and-wait, Go-Back-N, and Selective ARQs
- Call Drops Effect in MLPPP

Go-Back-N ARQ



TCP Segment

IP Packet

MLPPP frame size

ARQ transmission window = N ARQ frames

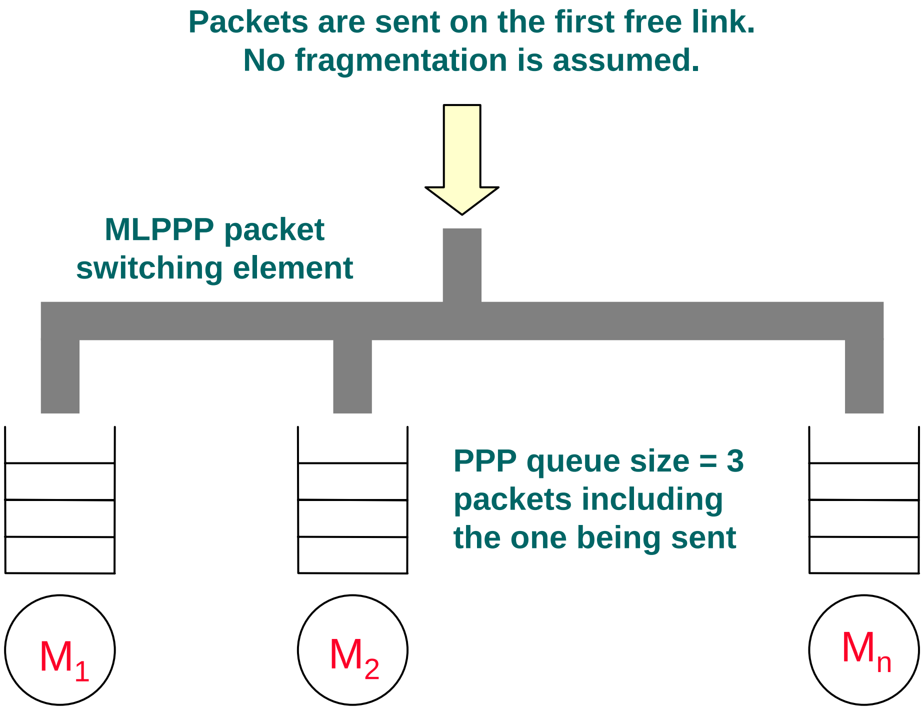
1 2 . . . N . . . n

Each TCP segment corresponds to n ARQ frames

- Out of order packets or single/multiple frame losses result in retransmission of the whole N frame window
- If the maximum number of retransmissions is exceeded, the ARQ frame is dropped

Effect of MLPPP Queuing

- Packets are not fragmented. Each is sent on 2.4 kbps. Transmit time = 5 sec, 2 T_p = 1 s
- Average of 1 second is added due to ARQ retransmissions
- RTT = (5+1)*3 + 1 = 19 sec
- Observed average RTT = 20 sec



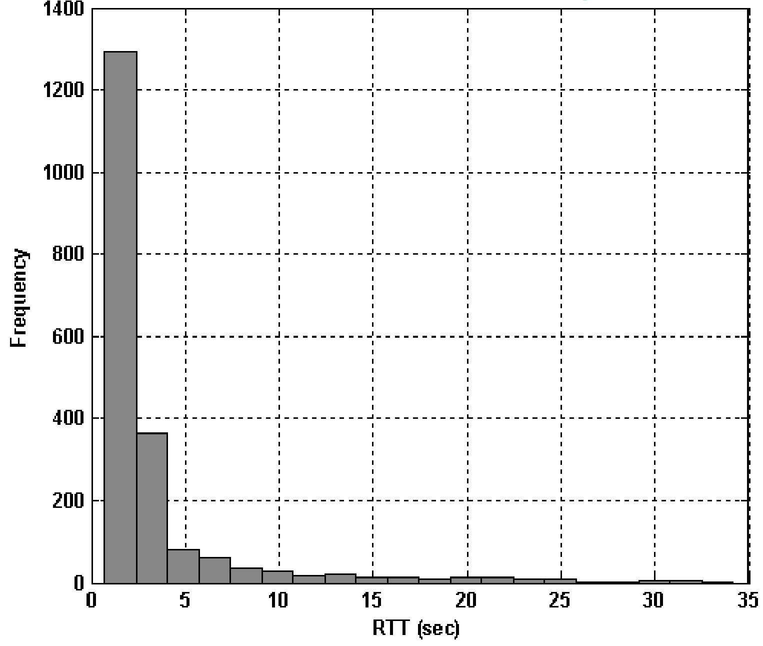
Packets are sent on the first free link.
No fragmentation is assumed.

MLPPP packet switching element

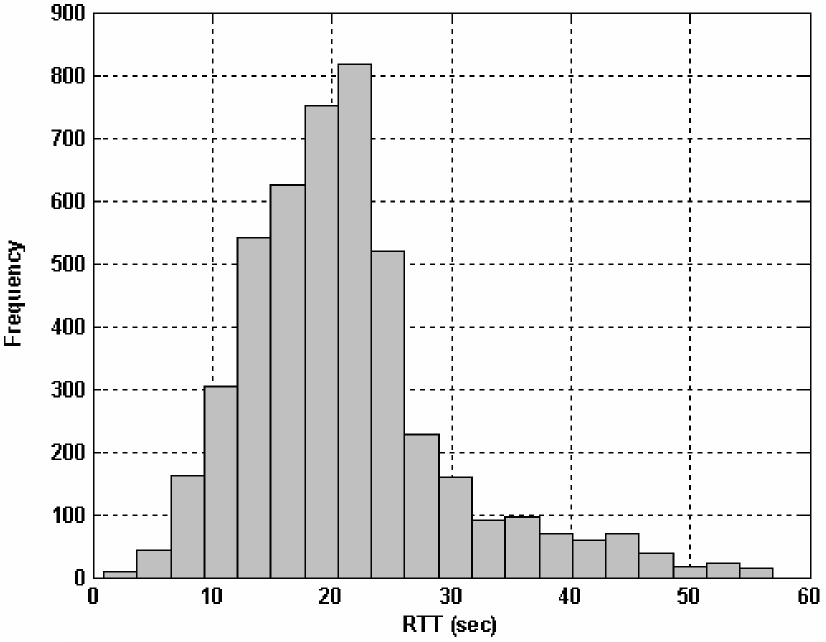
PPP queue size = 3 packets including the one being sent

Effect of MLPPP Queuing

PING size = 100 bytes

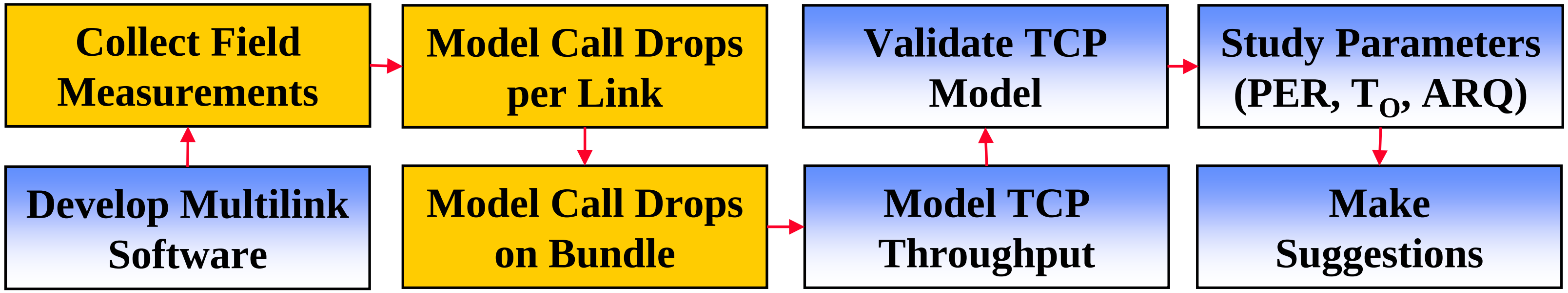


TCP – MSS = 1448 bytes



- PING: Mean =3.3 sec, Standard Deviation = 4.8 sec
- TCP: Mean = 21 sec, Standard Deviation = 8.7 sec

Next …



Collect Field Measurements

Model Call Drops per Link

Validate TCP Model

Study Parameters (PER, T_O, ARQ)

Develop Multilink Software

Model Call Drops on Bundle

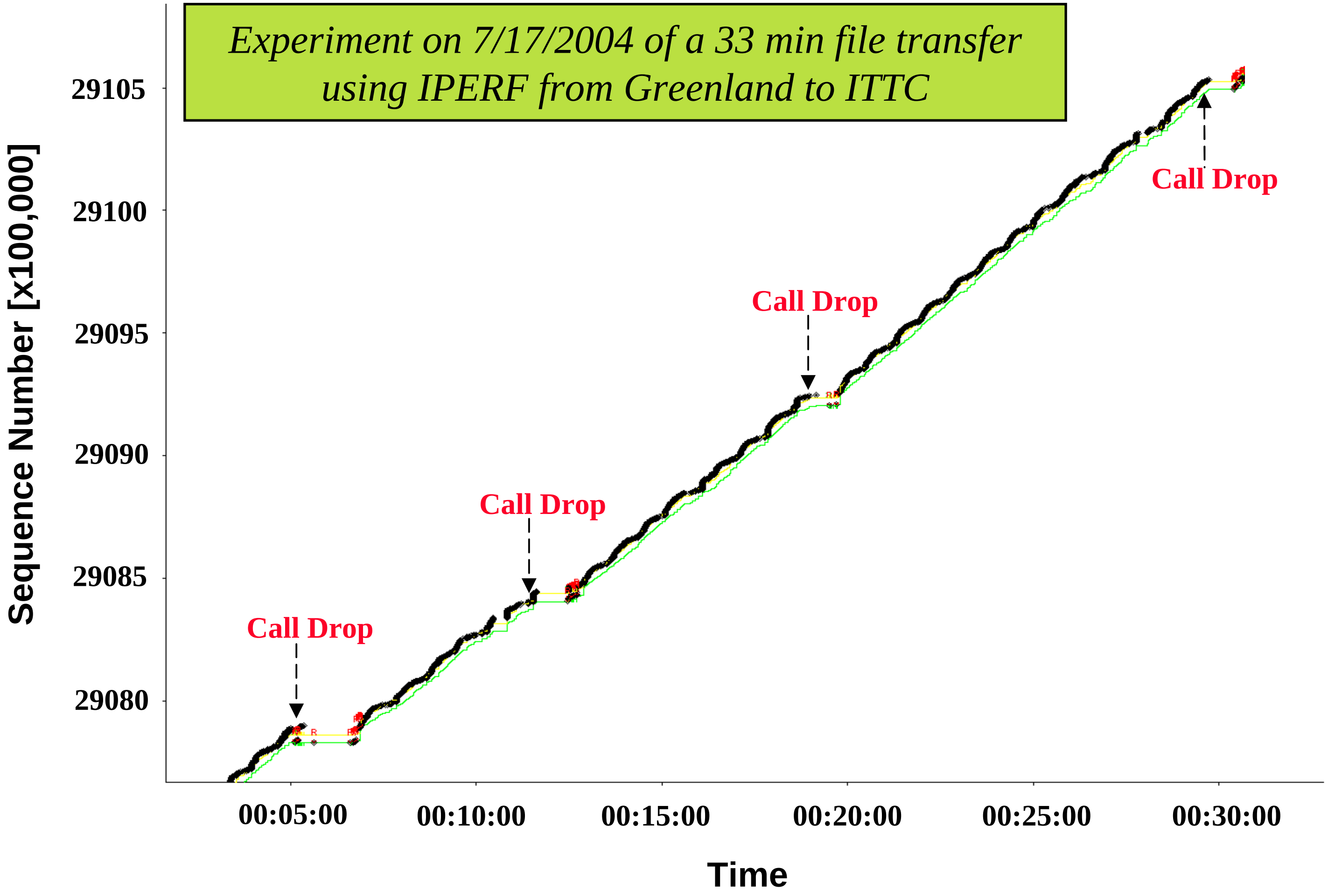
Model TCP Throughput

Make Suggestions

Call Drops - Introduction

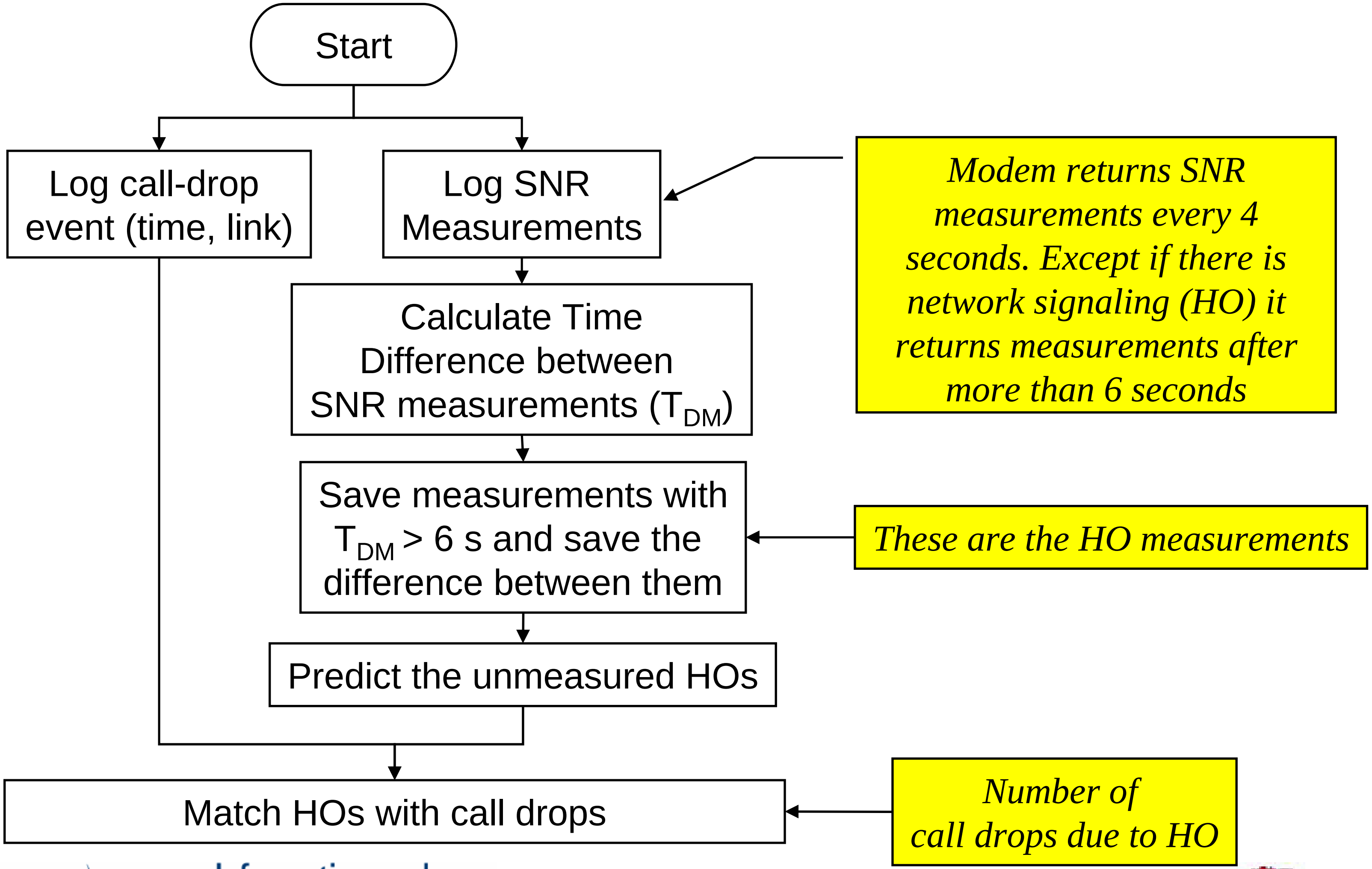
- A call drop is the event of losing an established connection suddenly
- Connections are automatically re-established
- It was observed that a call drop results in TCP timeouts
- Various reasons that might lead to call drops,
  - Low signal level
  - Failure of the inter-satellite handovers
- Two hypotheses to test
  - Call drops are periodic (due to handovers)
  - Random ?

Call Drops - Measurements



Experiment on 7/17/2004 of a 33 min file transfer using IPERF from Greenland to ITTC

Call Drops - Handovers



Start

Log call-drop event (time, link)

Log SNR Measurements

Calculate Time Difference between SNR measurements (T_{DM})

Save measurements with T_{DM} > 6 s and save the difference between them

Predict the unmeasured HOs

Match HOs with call drops

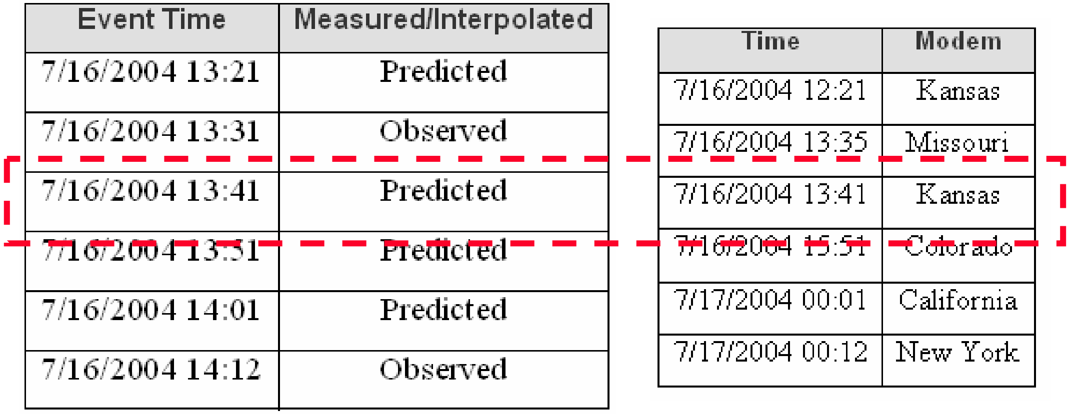
Modem returns SNR measurements every 4 seconds. Except if there is network signaling (HO) it returns measurements after more than 6 seconds

These are the HO measurements

Number of call drops due to HO

Call Drops - Handovers

- Only 10% of call drops were due to handovers! The call drop process is NOT a periodic process
- SNR relationships with call drops was hard to obtain
- Call drops are modeled by observing the call drop event itself rather than observing its potential causes



Event Time	Measured/Interpolated
7/16/2004 13:21	Predicted
7/16/2004 13:31	Observed
7/16/2004 13:41	Predicted
7/16/2004 13:51	Predicted
7/16/2004 14:01	Predicted
7/16/2004 14:12	Observed

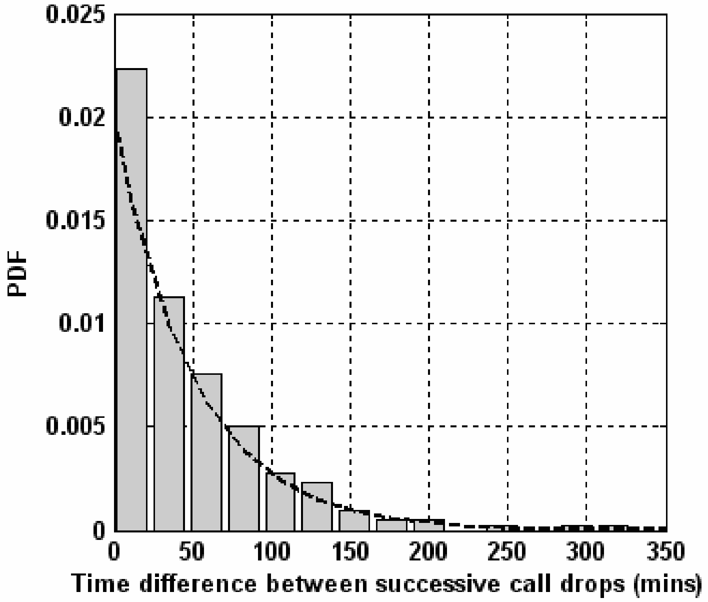
Time	Modem
7/16/2004 12:21	Kansas
7/16/2004 13:35	Missouri
7/16/2004 13:41	Kansas
7/16/2004 15:51	Colorado
7/17/2004 00:01	California
7/17/2004 00:12	New York

Greenland – Kansas Connection

Call Drops - Distribution

- 394 call-drop measurements were collected in the field
- The Inter Call drop Time Difference (ICTD) follows an exponential distribution
- The single link ICTD is a Poisson random process with a rate β
- A N Link bundle's ICTD is a Poisson process with a dropping rate of:

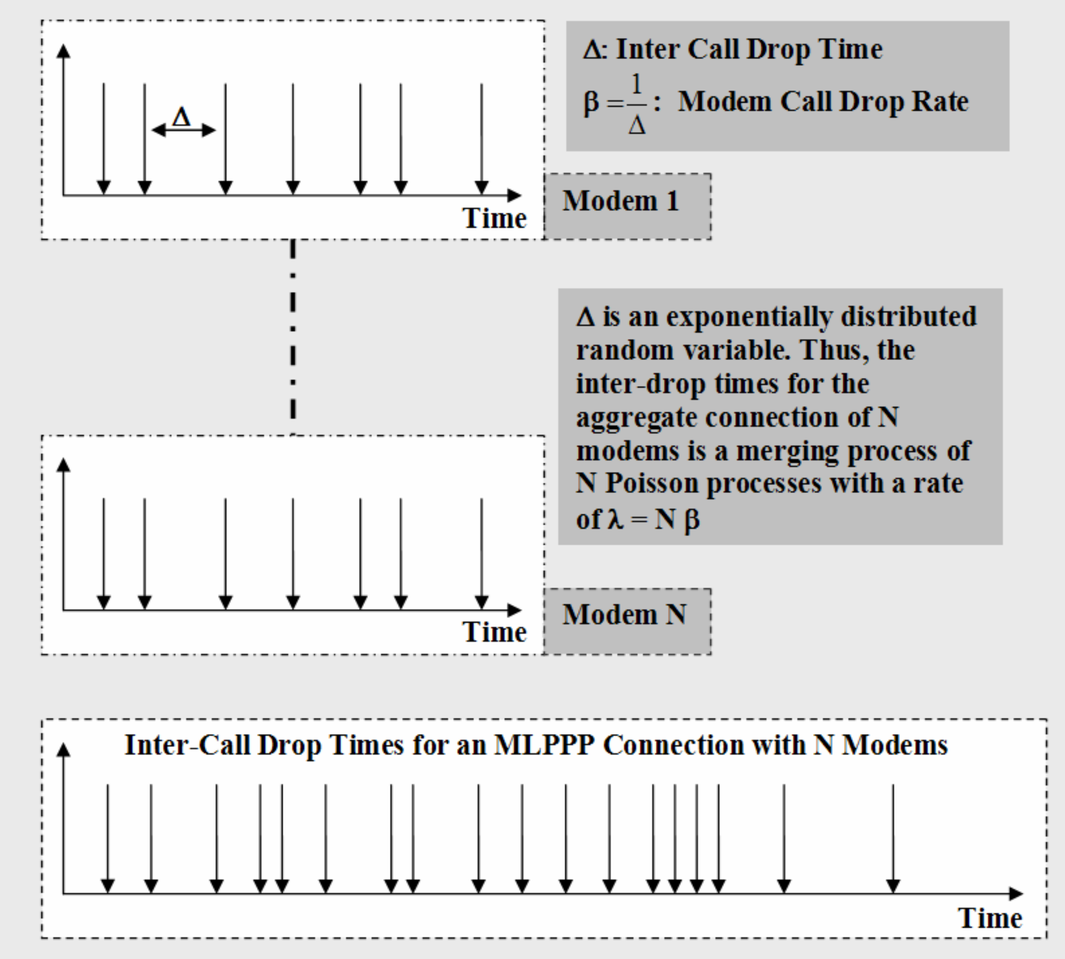
$$\lambda = N\,\beta$$



ICTD PDF based on Greenland–Kansas measurements. Estimated exponential distribution (0.02exp(-0.02t)) passes the chi-square goodness-of-fit test (5% significance level and 14 bins)

Call Drops - Distribution

- KS-Greenland call dropping rate per link is 1/50 min^{-1}
- KS-KS call dropping rate per link is 1/52 min^{-1}



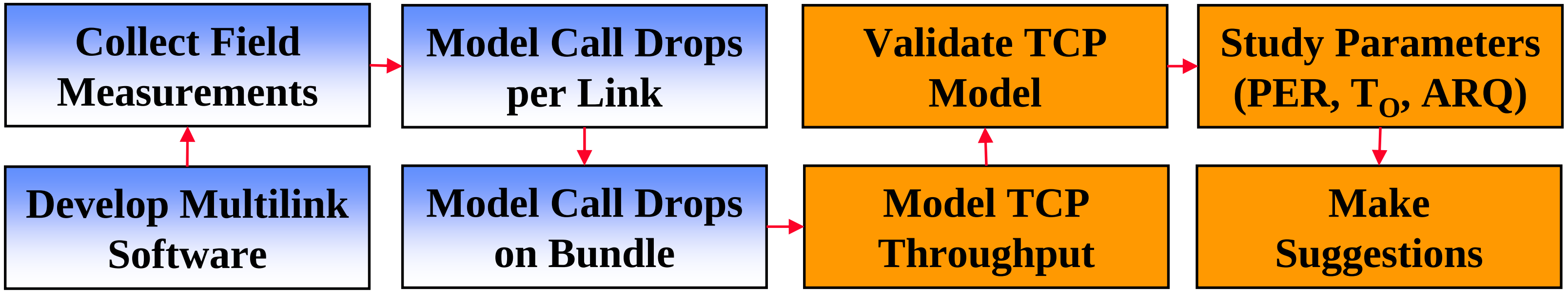
Δ: Inter Call Drop Time

$\beta = \frac{1}{\Delta}$: Modem Call Drop Rate

Δ is an exponentially distributed random variable. Thus, the inter-drop times for the aggregate connection of N modems is a merging process of N Poisson processes with a rate of $\lambda = N\beta$

Inter-Call Drop Times for an MLPPP Connection with N Modems

Next …



Collect Field Measurements

Model Call Drops per Link

Validate TCP Model

Study Parameters (PER, T_O, ARQ)

Develop Multilink Software

Model Call Drops on Bundle

Model TCP Throughput

Make Suggestions

TCP Performance Model

- TCP transfer latency for f_s bytes given the MSS is

$$T_d = \left\lceil \frac{f_s}{MSS} \right\rceil B \quad \text{(sec)}$$

- To estimate TCP throughput (B) in packets/sec:
  a. Evaluate the throughput if no timeouts take place
  b. Extend the no timeout throughput using the empirical call drops PDF to include timeouts
- Main Assumptions
  - Packet losses are due to ARQ failures (no timeouts)
  - Timeouts are caused by call drops only

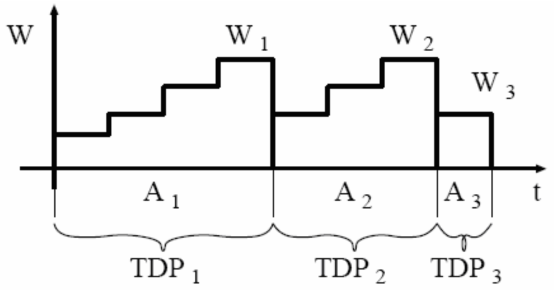
TCP Performance Model

a. Evaluate the throughput if no timeouts take place

$$B_{NT}=\frac{E\{Y\}}{E\{A\}}$$

Number of Packets sent in a TDP period

The length of the TDP period



Packet Error Rate (PER)

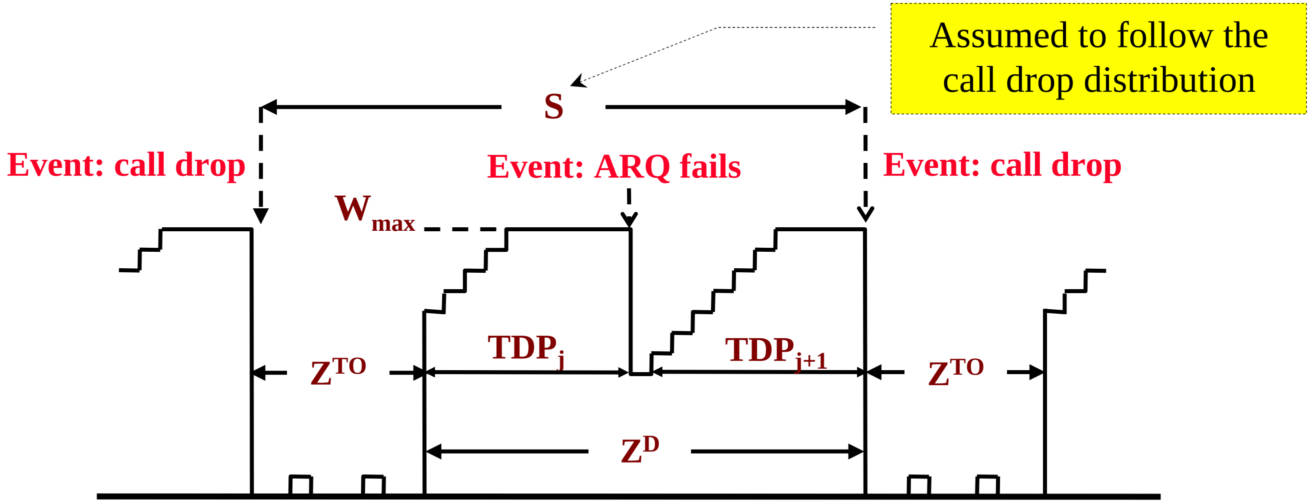
$$B_{NT}=\begin{cases}\dfrac{\dfrac{1-p}{p}+\dfrac{2+b}{3b}+\sqrt{\dfrac{8(1-p)}{3bp}+\left(\dfrac{2+b}{3b}\right)^{2}}}{RTT\left(\dfrac{2+b}{6}+\sqrt{\dfrac{2b(1-p)}{3p}+\left(\dfrac{2+b}{6}\right)^{2}}+1\right)} & E\{W_u\}<W_{\max}\\[2ex] \dfrac{\dfrac{1-p}{p}+W_{\max}}{RTT\left(\dfrac{b}{8}W_{\max}+\dfrac{1-p}{p.W_{\max}}+2\right)} & E\{W_u\}\geq W_{\max}\end{cases}$$

The average unconstrained window size is given as

$$E\{W_u\}=\frac{2+b}{3b}+\sqrt{\frac{8(1-p)}{3bp}+\left(\frac{2+b}{3b}\right)^{2}}$$

TCP Performance Model

b. Extend the no timeout throughput using the call drops PDF to include timeouts



$$B = \left[1 - \lambda T_0 \frac{1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + +32p^6}{1-p}\right] B_{NT} + \frac{\lambda}{1-p}$$

Results and Extensions

- Model Validation
  - Compare the model’s predictions with results of various transfers carried out in Greenland and in the lab
  - Increase the call drop rate over the base (i.e., Iridium) using a software modules
- Model Parameters
  - Study the effect of increased packet drop rate (i.e., increased ARQ failures).
  - Study the effect of the timeout value

Model Validation

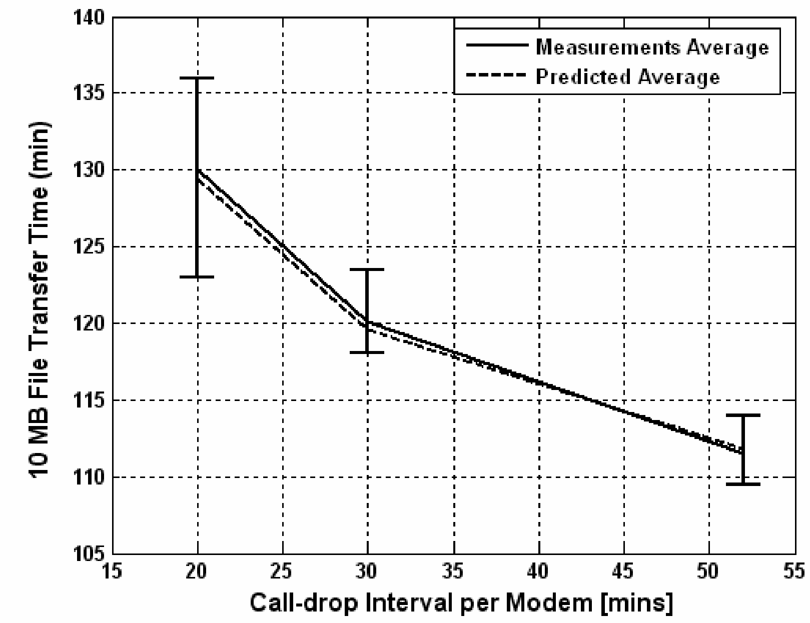
File Transfers from Greenland to the University of Kansas (Summer 2004),
T_0 = 60s, p = 5E-4, β = 1/50 min^{-1}, MSS = 1448, RTT = 19s, Wmax= 47.9KB

File Size (MB)	**1.38**	**5.62**	**20.6**	**35.7**
Measured Transfer Time (min)	11	46	180	315
Predicted Transfer Time (min)	12	51	187	324
Error %	13	11	4	3

Number of Links	**3**	**4**	**5**	**6**	**7**	**8**
File Size (MBytes)	4.82	0.85	1.91	1.39	3.40	1.40
W_{max} (KBytes)	16.1	22.0	28.3	34.7	41	47.9
Measured Time (min)	96	15	21	13	30	12
Prediction (min)	98.1	13.4	24.1	15.4	33.2	12.7

Case Study: Increased Dropping Rates

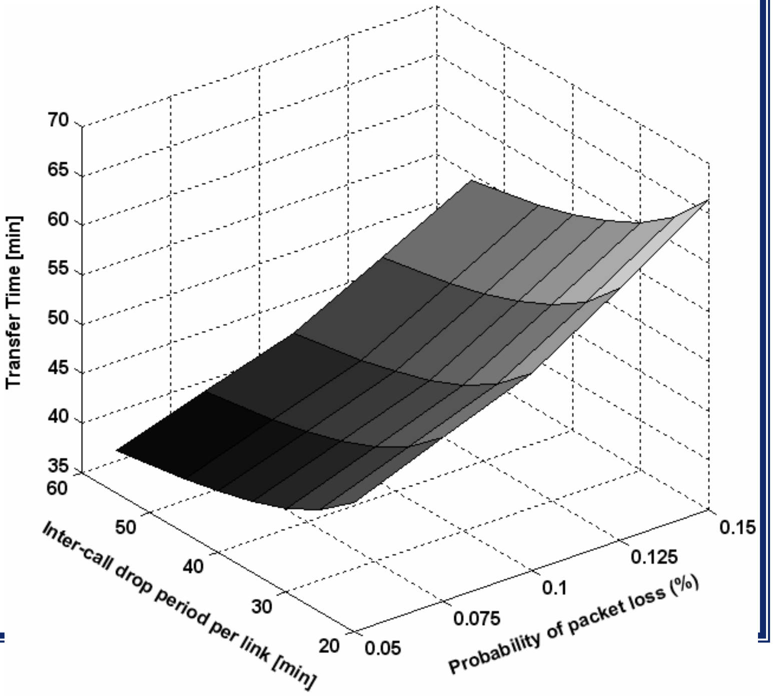
- A software module was built and added to the developed link management software
- The added module generates call drops according to a Poisson process for any given dropping rate



Case Study: Effect of Wireless Errors

- A wireless error refers to the errors that the physical layer ARQ could not handle
- Effect is amplified for low bandwidth-long delay connections (ex. Iridium)
- An efficient ARQ mechanism minimizes wireless errors

- Inmarsat GEO, bundle =4
- RTT = 0.61 s, BW = 128 kbps
- MSS=1 KB and W_{max}=40 KB
- A slight increase of the packet loss probability results in approximately 25 min increase in the transfer time

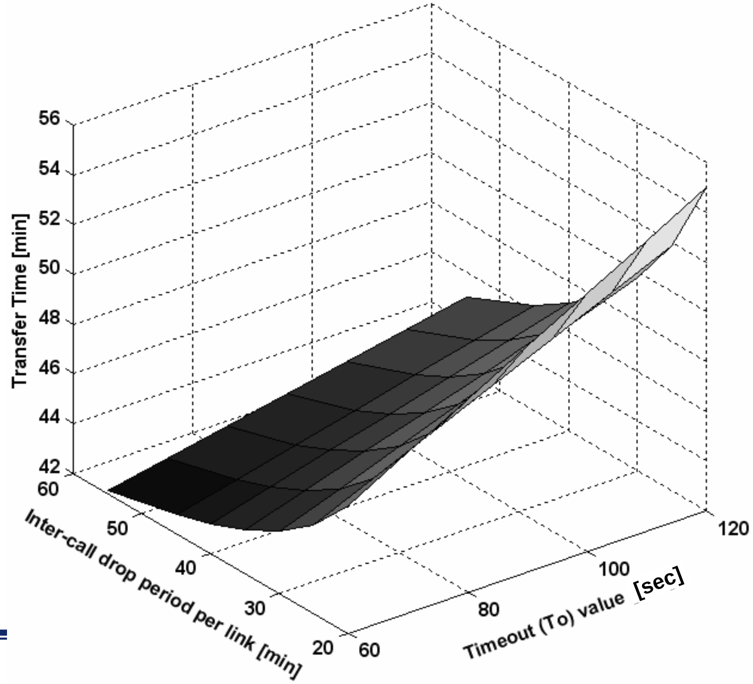


Case Study: Effect of the Timeout Value

- The timeout value determines the impact of a call drop on the performance of TCP
- Key reasons for the RTT variations in P2P connections are the physical layer ARQ (and MLPPP queuing)
- Need good ARQ and MLPPP frame distribution

- Inmarsat GEO, bundle =4
- RTT = 0.61 s, BW =128 Kbps
- MSS=1KB , Wmax=40KB
- Increased initial timeout value magnifies the effect of the call drops

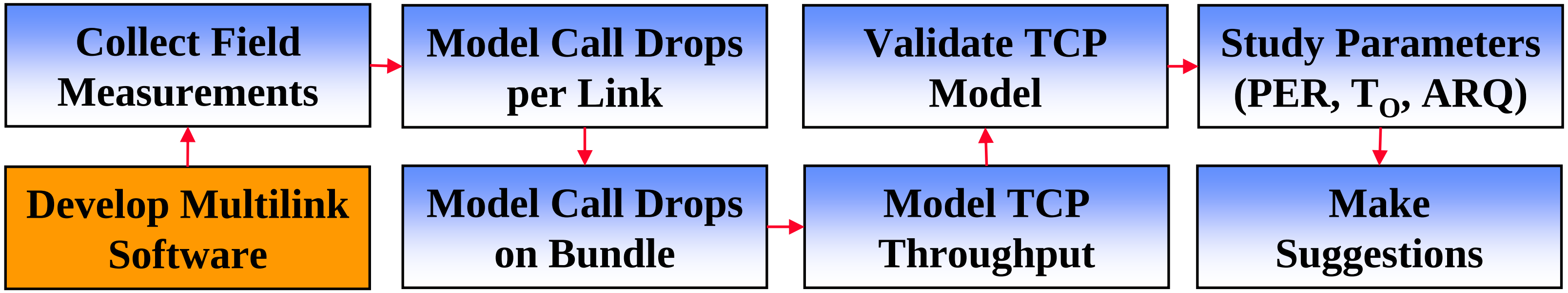
$$Timeout = Estimated\_RTT + 4 \times Deviation$$



Recommendation

- Potential advantage if a triple acknowledgement is generated once a call drop is detected
- This way call drops only cause packet losses
- TCP switches to the recovery mode instead of timing out
- Implementing this technique requires that the kernel and/or the MLPPP drivers be modified to keep track of the received TCP acknowledgements
- Since ARQ failures are very low, the probability of a loss is updated to account for call drops only

Next …



Collect Field Measurements

Model Call Drops per Link

Validate TCP Model

Study Parameters (PER, T_O, ARQ)

Develop Multilink Software

Model Call Drops on Bundle

Model TCP Throughput

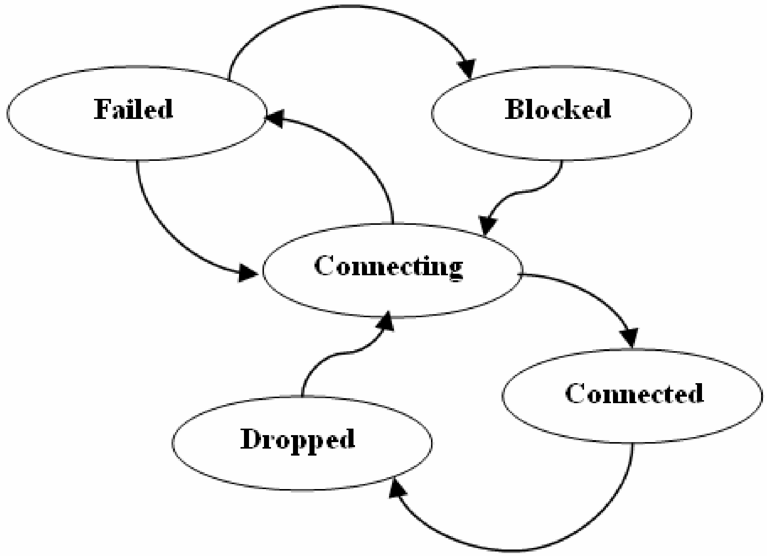
Make Suggestions

The Link Management Software

- First link management software was tested successfully in ‘03 in Greenland for a multi-Iridium link with 4 modems
- Original software lacked flexibility (Hard coded scripts)
- Number of links was doubled to 8 and requirements for a new management software were defined,
  - Should support a variable number of modems
  - Should allow connections to occur in any order
  - Should log call events in a portable format
  - Should provide a friendly user interface (non-expert users)
  - Should provide real-time monitoring of the system status
  - Should allow adding debugging and experimental modules

The Link Management Software

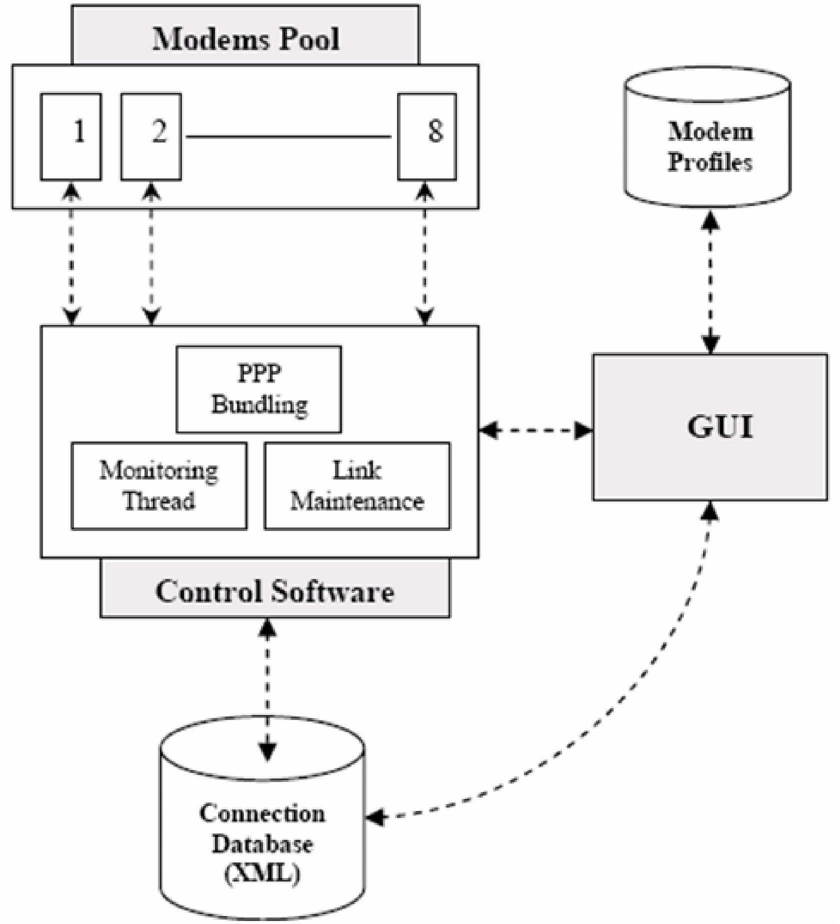
- The developed link management software is called KU Multilink Iridium Communication System (KUMICS)
- Modem states in KUMICS are connecting, failed, connected, dropped, blocked



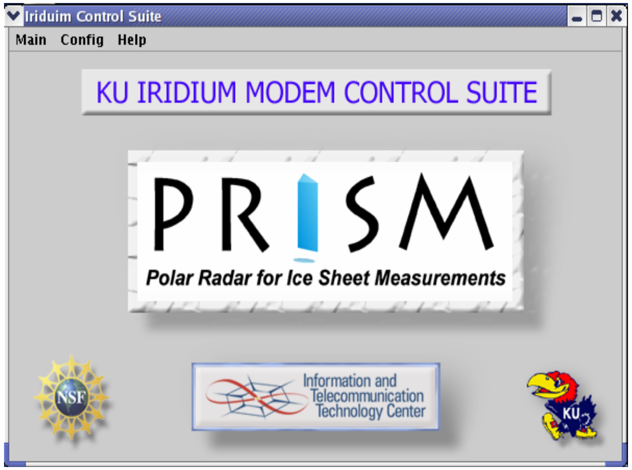
Last PPPD 2.4.3 implementation has a limitation that if the primary modem is lost then it can not be redialed until all other links are dropped

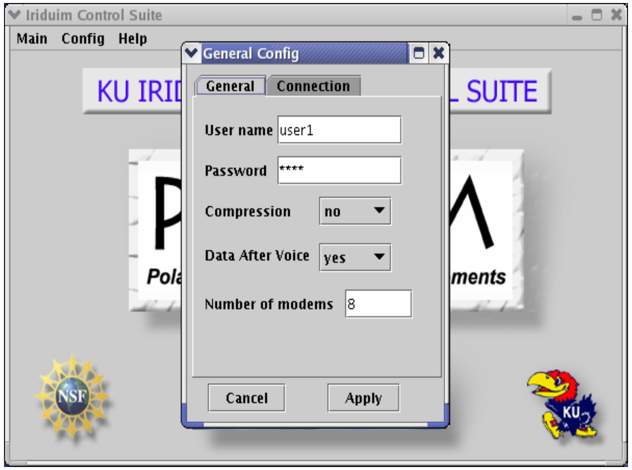
KUMICS - Architecture

- KUMICS is designed based on three primary components: GUI, Control Software, XML Database
- GUI: offers a user friendly means to configure and monitor the MLPPP connection
- Control Software: Internal logic that handles dialing, disconnecting, bringing up the modems, and updating the LOGS
- XML Database: is where call drop times and experimental measurements are stored



KUMICS – GUI





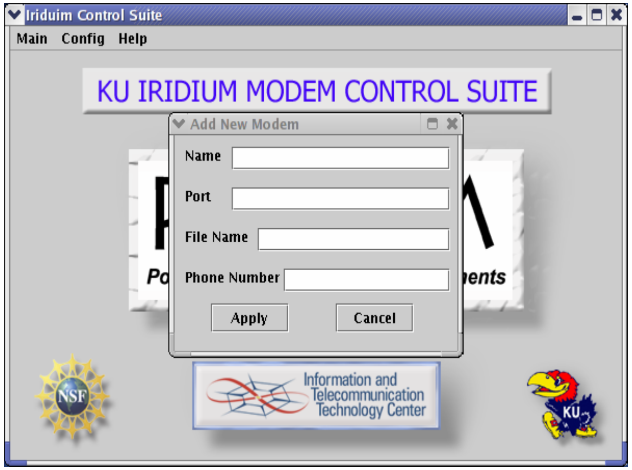
User name user1

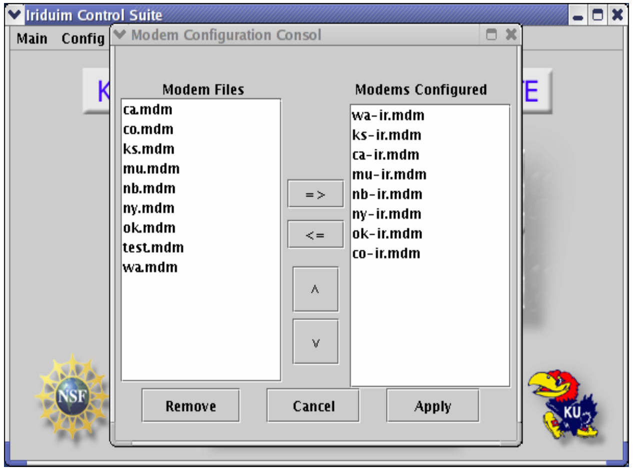
Password ****

Compression no

Data After Voice yes

Number of modems 8





Modem Files

- ca.mdm
- co.mdm
- ks.mdm
- mu.mdm
- nb.mdm
- ny.mdm
- ok.mdm
- test.mdm
- wa.mdm

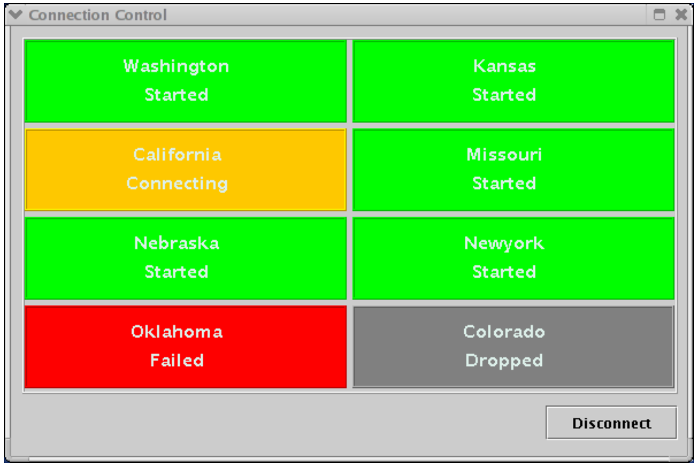
Modems Configured

- wa-ir.mdm
- ks-ir.mdm
- ca-ir.mdm
- mu-ir.mdm
- nb-ir.mdm
- ny-ir.mdm
- ok-ir.mdm
- co-ir.mdm

KUMICS – GUI

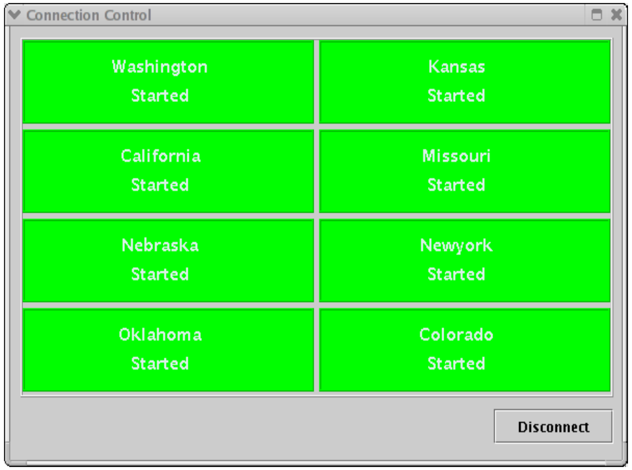
Connection Control

Washington Connecting	Kansas Stopped
California Stopped	Missouri Stopped
Nebraska Stopped	Newyork Stopped
Oklahoma Stopped	Colorado Stopped



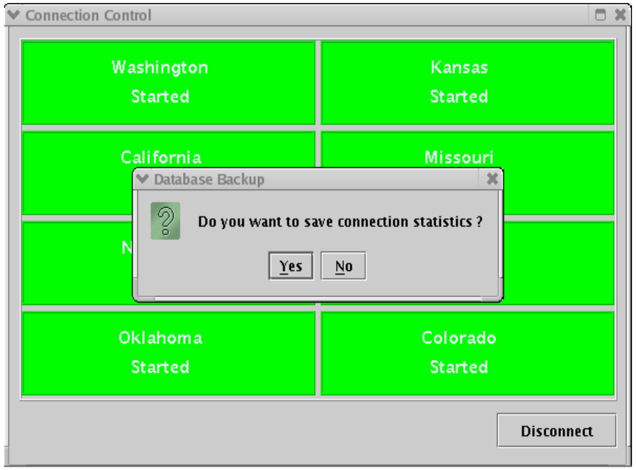
Connection Control

Washington Started	Kansas Started
California Connecting	Missouri Started
Nebraska Started	Newyork Started
Oklahoma Failed	Colorado Dropped



Connection Control

Washington Started	Kansas Started
California Started	Missouri Started
Nebraska Started	Newyork Started
Oklahoma Started	Colorado Started



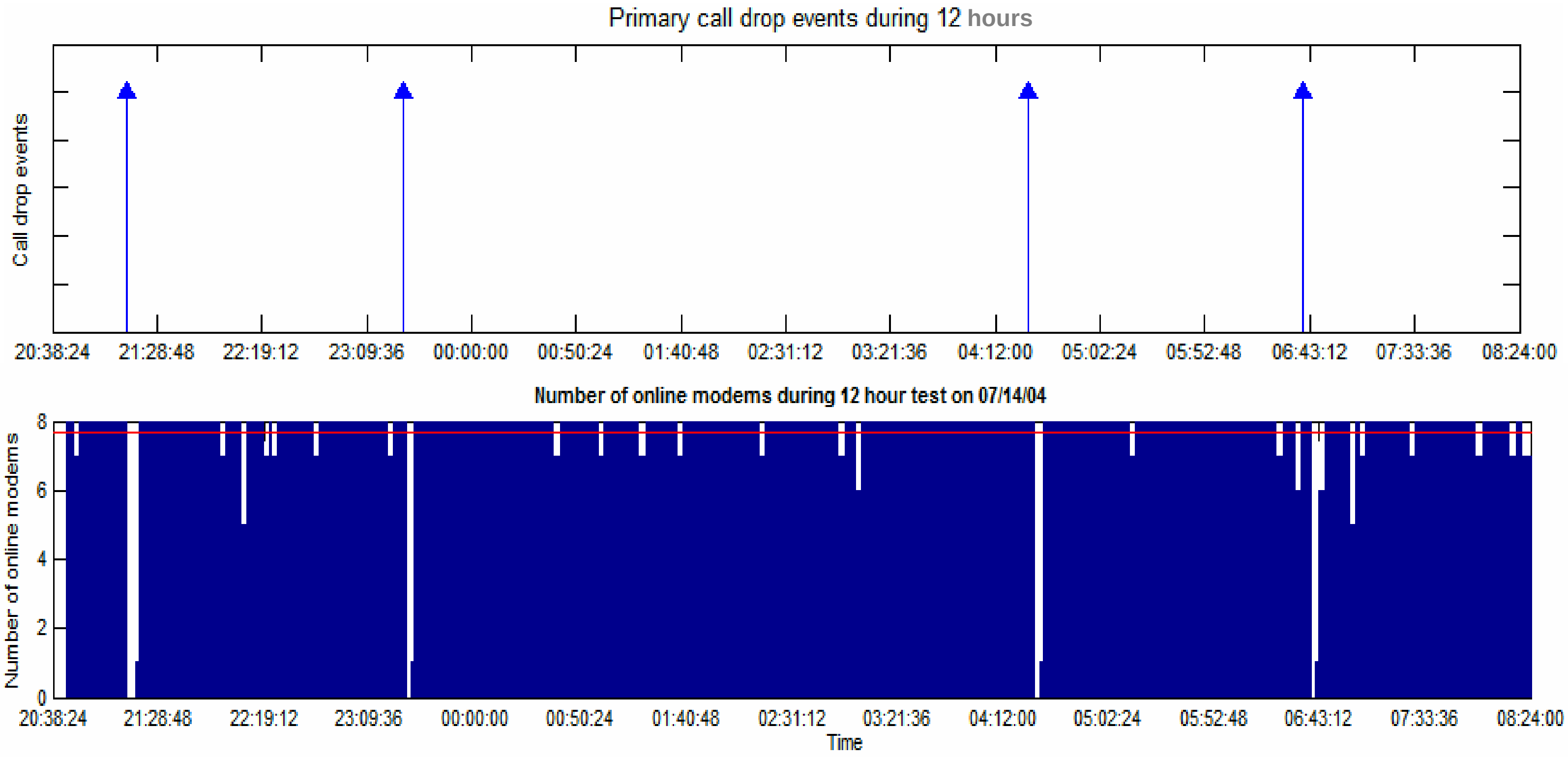
Connection Control

Washington Started	Kansas Started
California	Missouri
N	
Oklahoma Started	Colorado Started

Database Backup

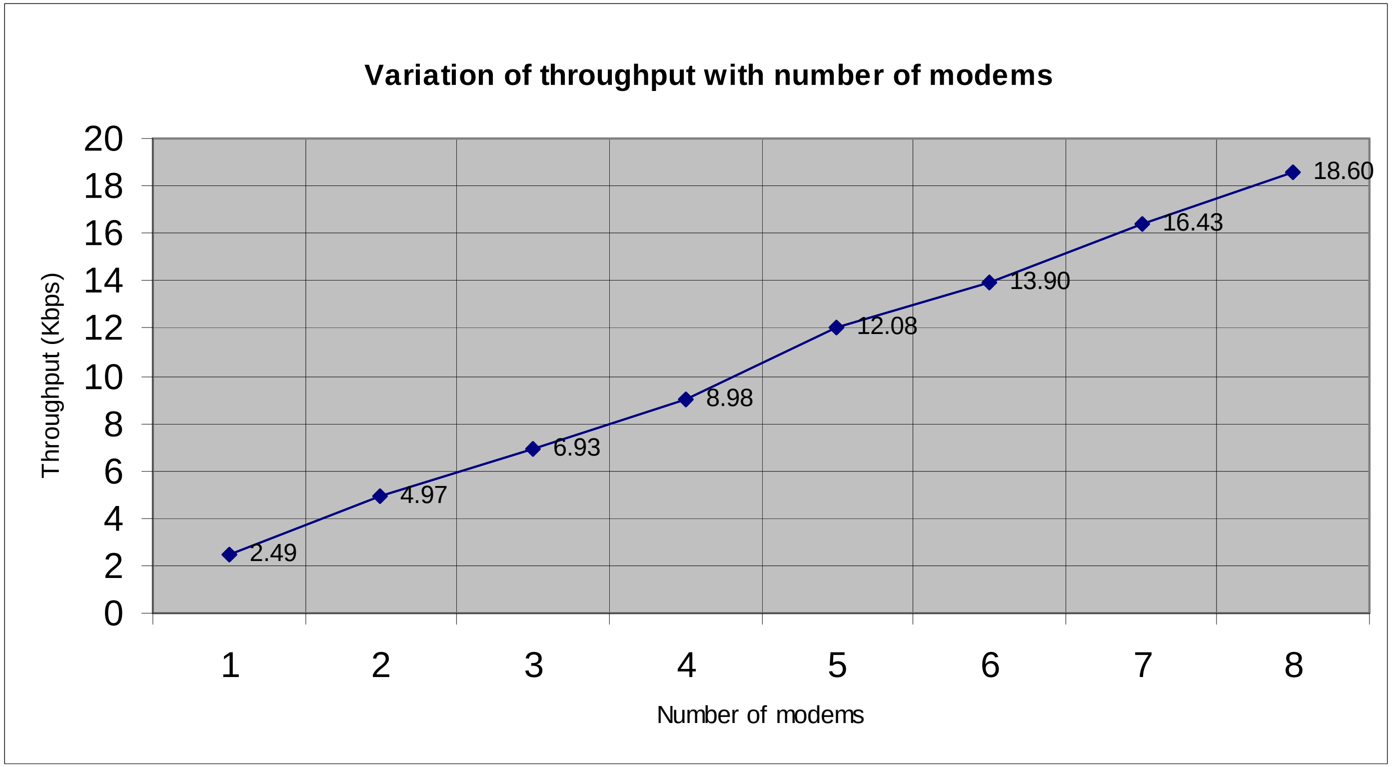
Do you want to save connection statistics ?

System Reliability Results



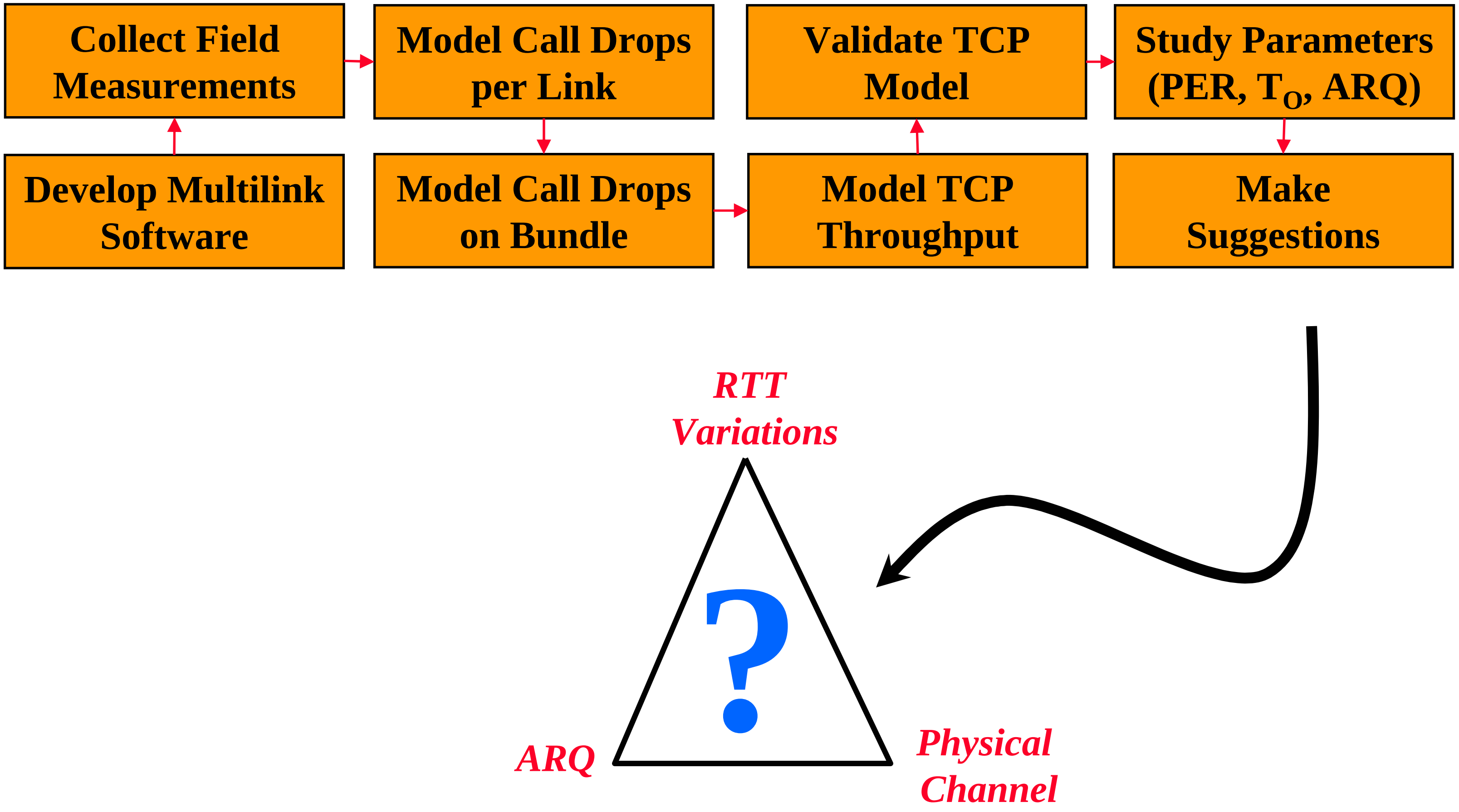
- The system delivered full throughput (i.e., when all modems are connected) 87% of the time
- The system was up with at least one modem 97% of the time

System Reliability Results



- The relationship between the number of modems and the throughput is linear. Maximum attained throughput with 8 modems is 18.6 kbps

Next …



Collect Field Measurements

Model Call Drops per Link

Validate TCP Model

Study Parameters (PER, T_O, ARQ)

Develop Multilink Software

Model Call Drops on Bundle

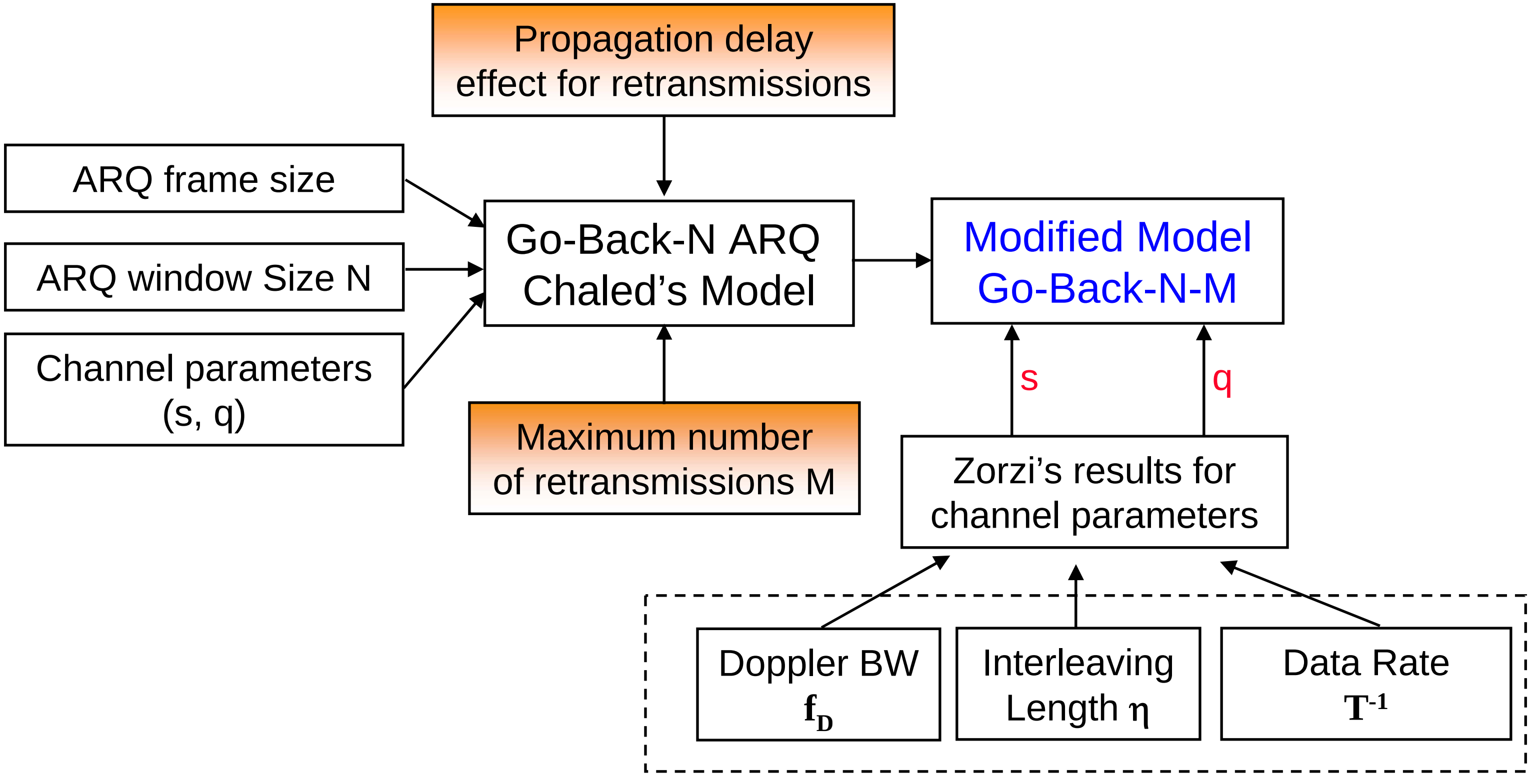
Model TCP Throughput

Make Suggestions

The effect of ARQ on RTT

- Problem
    - The RTT varies significantly when running TCP over ARQ
- Importance of Modeling RTT variations
    - May be used to detect the presence of wireless links
    - Increased RTT variations might result in timeouts
    - Need to have better understanding of RTT as a function of: speed, data rate, etc
- Research Limitations
    - Need to be validated by simulation or measurements
    - Only apply to flat fading Rayleigh channels (no multipath)

Solution Methodology



Propagation delay effect for retransmissions

ARQ frame size

ARQ window Size N

Channel parameters (s, q)

Go-Back-N ARQ Chaled's Model

Maximum number of retransmissions M

Modified Model Go-Back-N-M

s

q

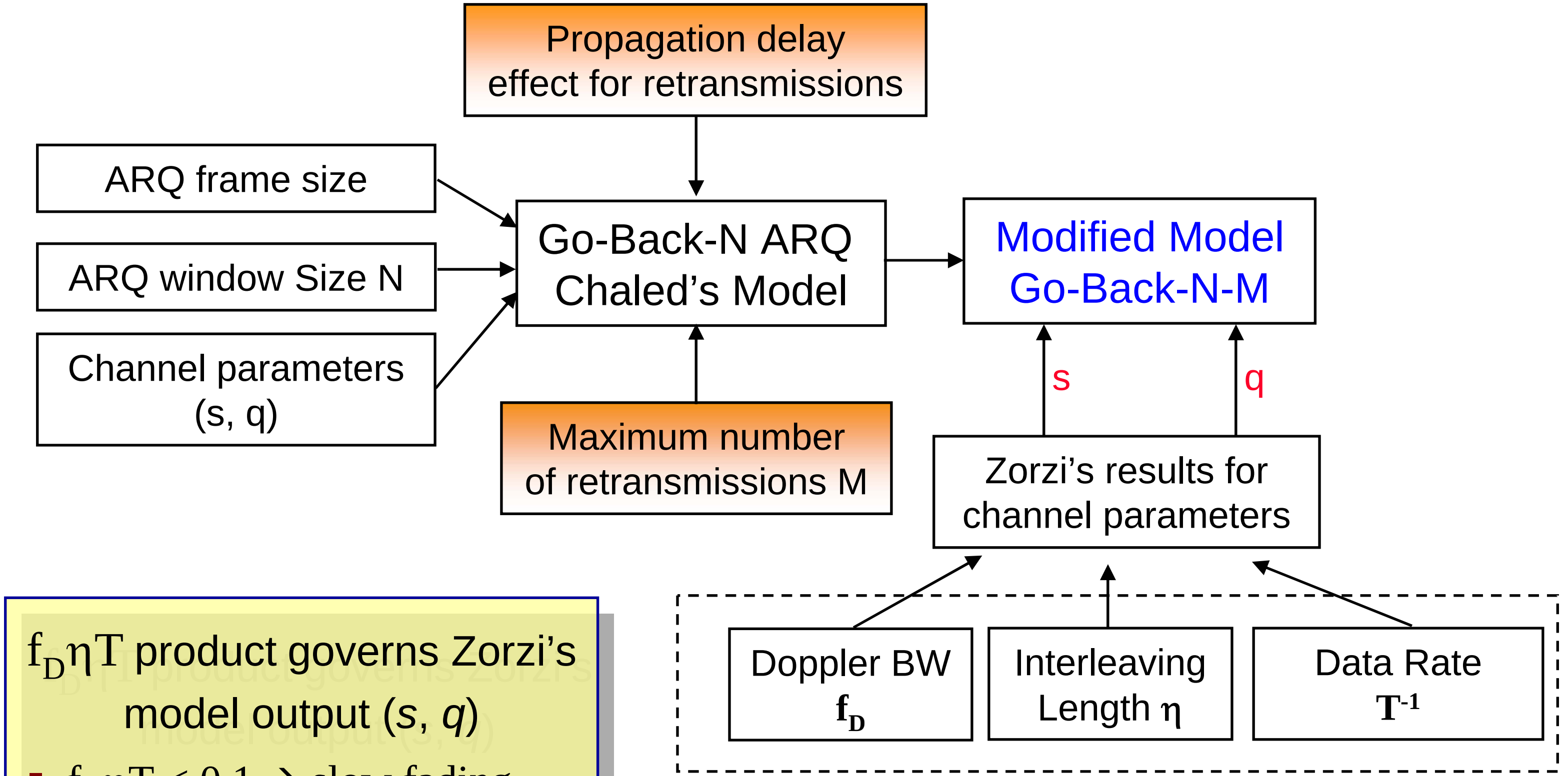
Zorzi's results for channel parameters

Doppler BW f_D

Interleaving Length η

Data Rate T^{-1}

Solution Methodology



Propagation delay effect for retransmissions

ARQ frame size

ARQ window Size N

Channel parameters (s, q)

Go-Back-N ARQ Chaled's Model

Maximum number of retransmissions M

Modified Model Go-Back-N-M

s

q

Zorzi's results for channel parameters

Doppler BW f_D

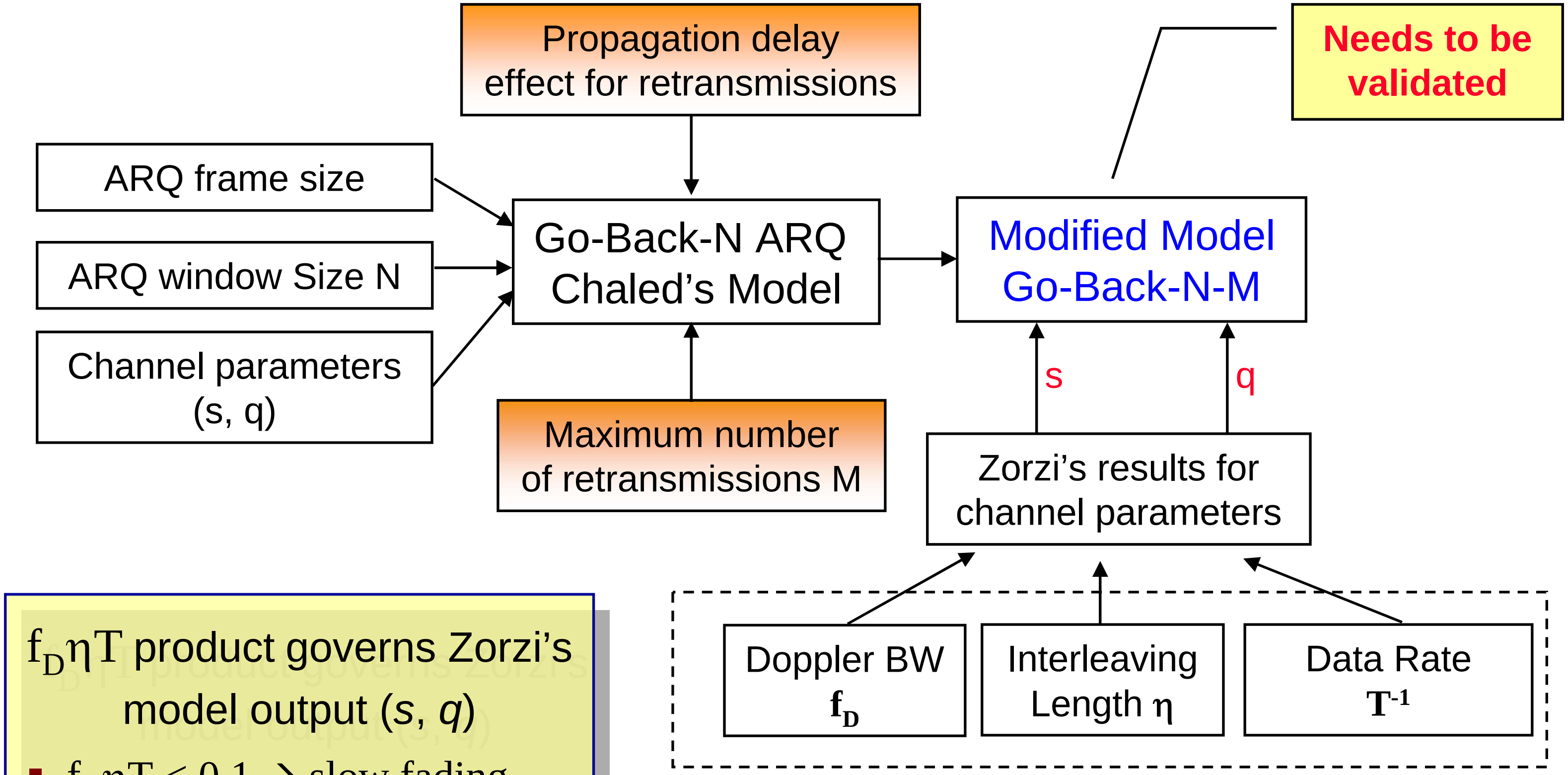
Interleaving Length η

Data Rate T^{-1}

$f_D\eta T$ product governs Zorzi's model output (*s*, *q*)

- $f_D\eta T < 0.1$ → slow fading
- $f_D\eta T > 0.2$ → fast fading

Solution Methodology



$f_D \eta T$ product governs Zorzi's model output (*s*, *q*)

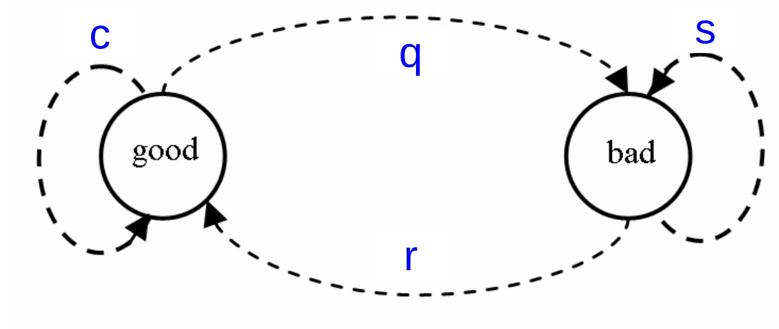
- $f_D \eta T < 0.1$ → slow fading
- $f_D \eta T > 0.2$ → fast fading

Solution Methodology

- General equation:

$$RTT_{TCP} = n\, b\, D_{ARQ} + RTT_{channel} + (N\, D_{ARQ} + RTT_{channel})E[n_{retx}]$$

- Markovian channel model:



$r + s = 1$
$c + q = 1$

- Result

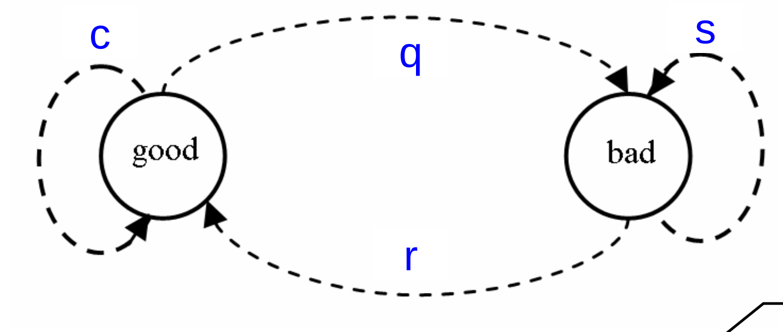
$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1 - s^{NM}\left(1 + M\,(1 - s^{N})\right)}{\left(1 - s^{NM}\right)\left(1 - s^{N}\right)}\right]\left[q(nb-1)\right]$$

Solution Methodology

- General equation:

$$RTT_{TCP} = n\ b\ D_{ARQ} + RTT_{channel} + (N\ D_{ARQ} + RTT_{channel})E[n_{retx}]$$

- Markovian channel model:



$r + s = 1$
$c + q = 1$

- Result

For high delay channels b = 1 and E{Dack} is added

$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1 - s^{NM}\left(1 + M\left(1 - s^{N}\right)\right)}{\left(1 - s^{NM}\right)\left(1 - s^{N}\right)}\right]\left[q(nb-1)\right]$$

Solution Methodology

$r = \frac{1-\varepsilon}{\varepsilon}[Q(\theta,\rho\theta)-Q(\rho\theta,\theta)]$

$q = Q(\theta,\rho\theta)-Q(\rho\theta,\theta)$

$s = 1-r$

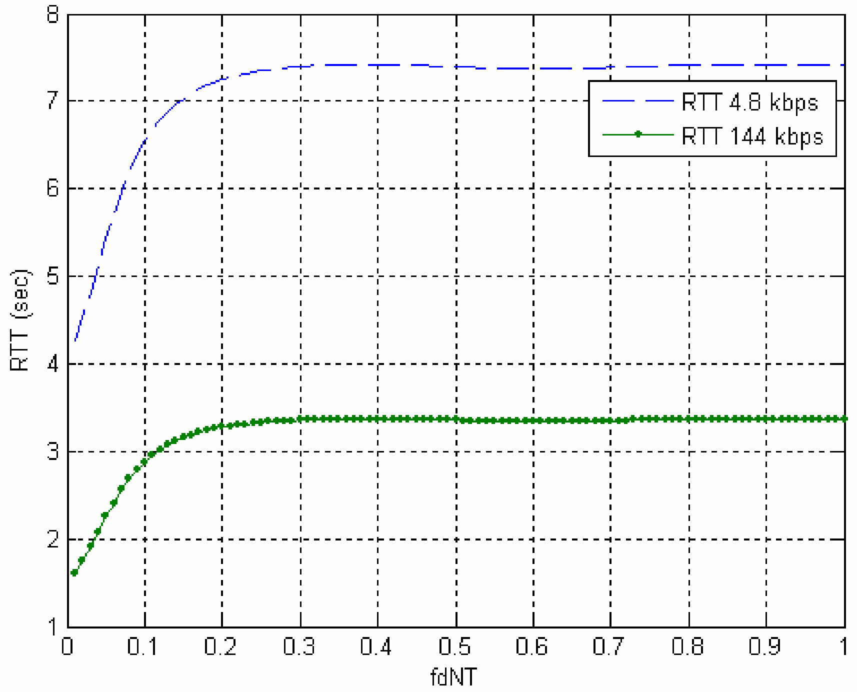
$\varepsilon = \frac{q}{q+r}$

$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1-s^{NM}\left(1+M\,(1-s^{N})\right)}{\left(1-s^{NM}\right)\left(1-s^{N}\right)}\right][q(nb-1)]$$

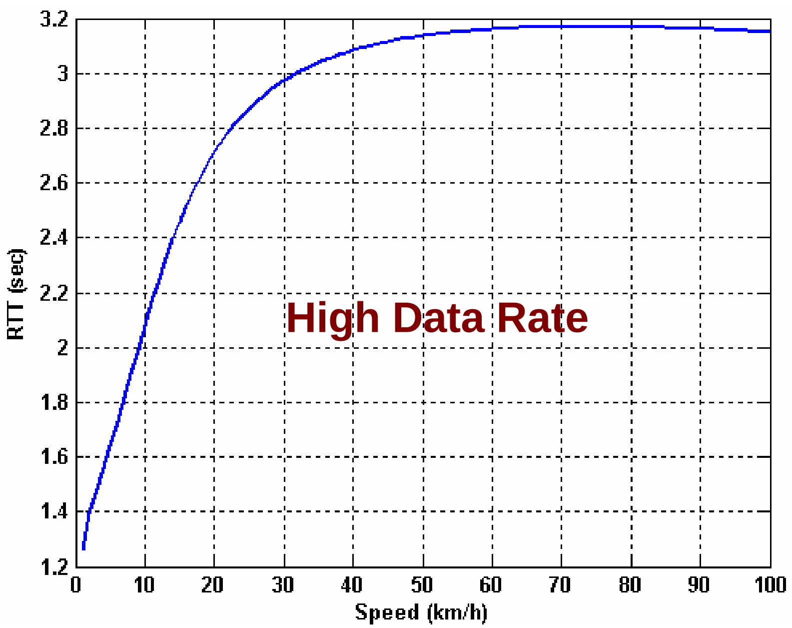
$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1-\left[1-\frac{1-\varepsilon}{\varepsilon}q\right]^{NM}\left[1+M\left(1-\left(\frac{1-\varepsilon}{\varepsilon}q\right)^{N}\right)\right]}{\left(1-\left[1-\frac{1-\varepsilon}{\varepsilon}q\right]^{NM}\right)\left(1-\left[1-\frac{1-\varepsilon}{\varepsilon}q\right]^{N}\right)}\right][q(nb-1)]$$

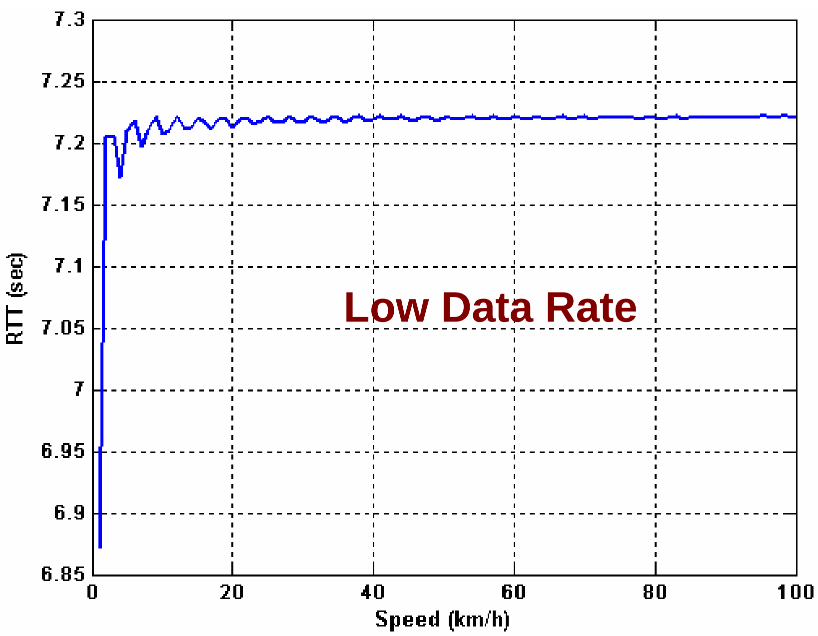
RTT as a Function of $f_D \eta T$

- If transmitting at low data rates, the difference between samples is large. This implies that the speed does not need to be large to drive $f_D \eta T$ beyond the fast fading limit (0.2)
- RTT is not a function of the mobile speed when the block errors become independent
- Remember results are valid for flat Rayleigh channels



RTT as a Function of Speed





- For η = 500, RTT almost constant for low date rates
- For η = 500, RTT varies for high data rate
- If η = 1, then the results will be opposite (i.e., constant for high data rates, variable for low data rates)

Conclusions

- A new model for predicting the file transfer time for TCP over MLPPP was developed and experimentally validated
  - Excessive call drops result in exponential increase of the TCP transfer latency
  - Efficient ARQ is important to minimize packet losses and timeout values
  - Model results suggest that OS kernel be modified to avoid timeouts in case of call drops

Conclusions

- A new Go-Back-N-M ARQ model was developed by modifying Chaled’s model and combining it with Zorzi’s results for flat fading Rayleigh channels.
- RTT variations are governed by $f_D \eta T$ product
- A user friendly link management software was developed to operate the multilink Iridium system efficiently

Future Work

- Validate the ARQ analysis either by measurements or by simulation
- Kernel modification to reduce the impact resulting from call drops (timeouts to retransmissions)
- KUMICS enhancements and upgrades to include instant alarms

Future Work

- The developed model can be extended to predict the behavior of TCP over multilink cellular systems. The new challenges here are:
  - A new call drops model has to be developed (taking handovers effect)
  - The speed of the mobile which relate to call drops in terms of (SNR, Handover failures)
  - To study the ARQ performance in cellular environments, Zorzi’s model should be modified for multi-path channels
  - The network behavior and call drops effect may be different in case of data calls such as those over EVDO or using GPRS or EDGE

Questions

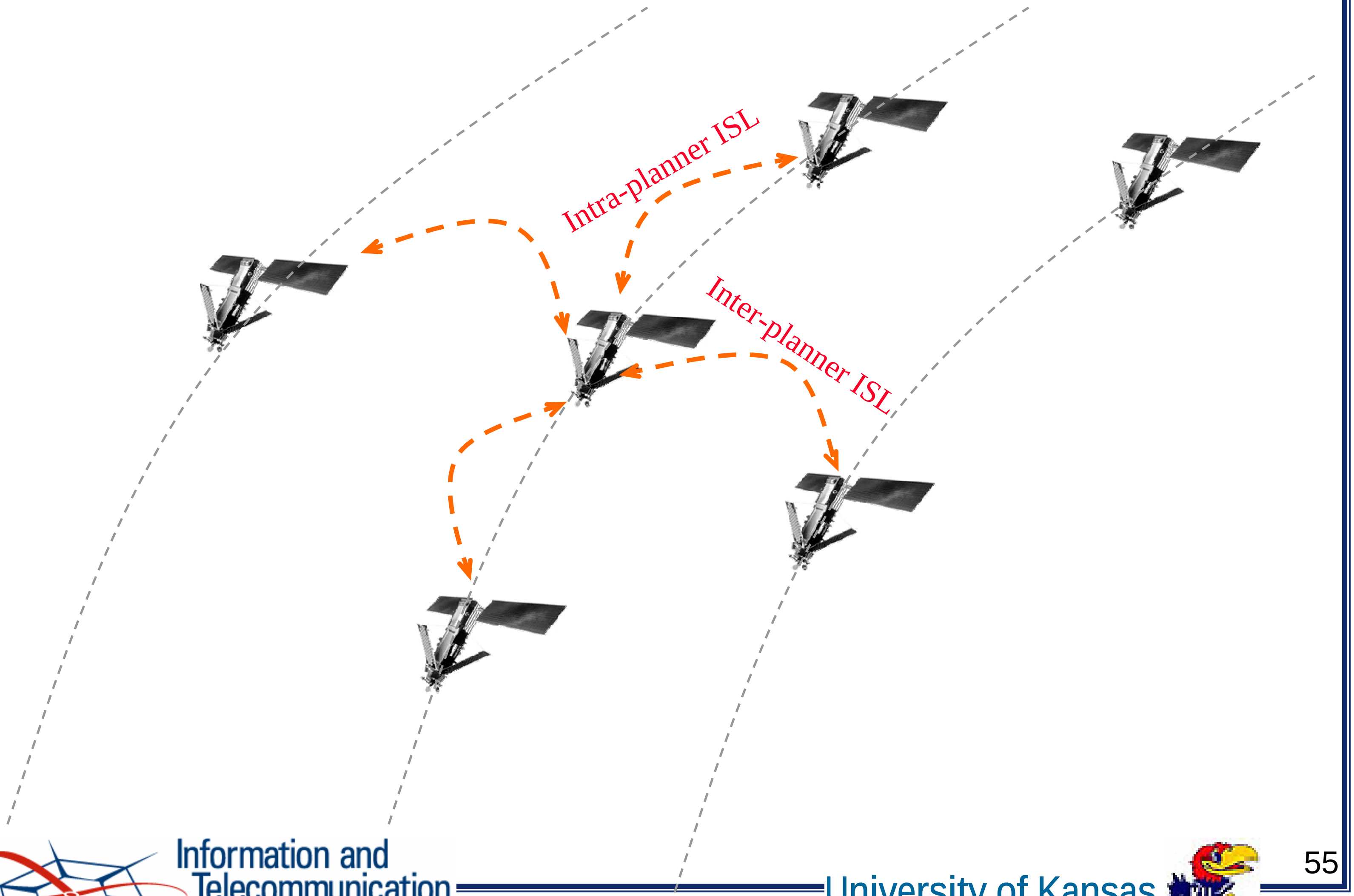
Thank you very much!

Questions ?

The Iridium Satellite System

- 66 LEO satellites (Originally were 77 satellites)
- Height is 780 km and orbital period is 100 mins
- Expected life time 7-9 years
- Satellites are divided into 6 planes [11 satellites / plane]
- The only satellite network that provides full coverage (temporal and geographic) for Polar Regions
- Designed for voice communications (2.4 kbps data rate)
- Smart onboard switching
- Gateways control connection setup and route traffic in case of PSTN calls

The Iridium Satellite System

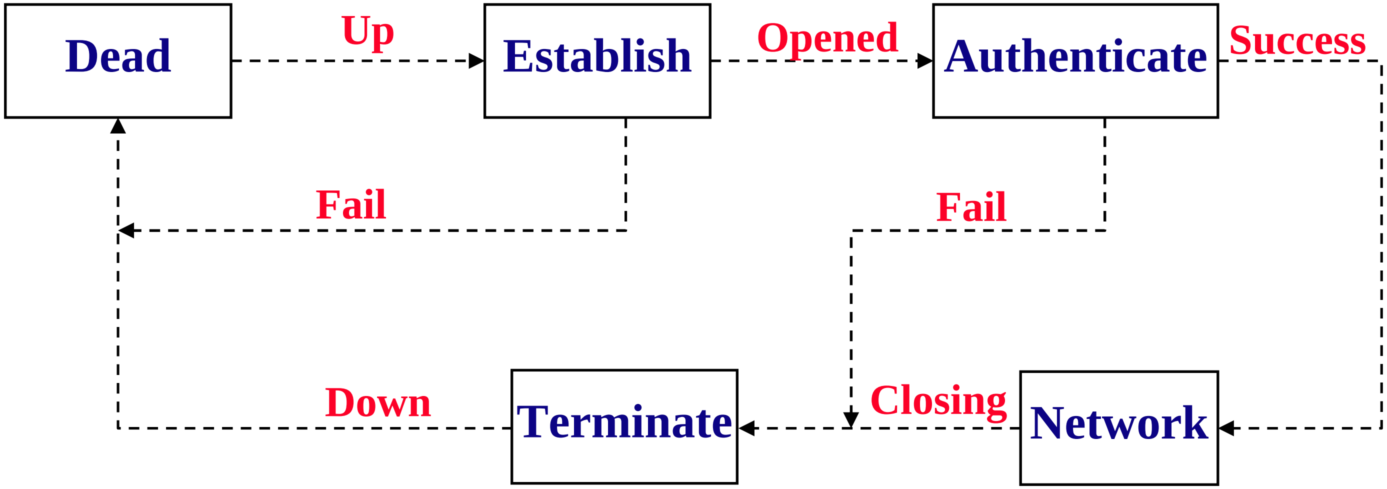


The Iridium Satellite System

Modulation Technique	QPSK
Frame Structure	FDMA/TDMA
Frequency	L-band from 1610 to 1626.5 MHz
Inter Satellite Links Frequency	Ka band from 23.18 to 23.38 GHz
Ground Segment Links	Ka band: Uplink: 29.1 – 29.3 GHz Downlink: 19.4 – 19.6 GHz
Ground-Based Digital Switches	Siemens GSM-D900
Multiple Access Technique	FDMA / TDMA
Digital Voice and Data Rate	2.4 kbps
Error Protection	FEC 3/4
Channel Bandwidth	31.5 Khz with 41.5 Khz guard bands
Inter-Satellite Handover	9-10 min

The Point-to-Point Protocol (PPP)

- Used to encapsulate layer 3 protocols' data in a standardized frame format and transport it over point-to-point links T1, Dial-up, CDMA2000-EVDO
- Overhead is 5-7 bytes



MLPPP Configuration Example

- Example PPP configuration parameters for Iridium

Parameter	Recommended values for Iridium
PAP-restart	10 – 15 sec
LCP-restart	10 - 15 sec
LCP-max-configure	10
LCP-echo-interval	30 sec
LCP-echo-failure	2
Connect-delay	5000 ms

TCP Performance Model

- If M represents the number of packets sent during a period S, then the throughput (B) (in the case of call drops) is given as

$$B = \frac{E\{M\}}{E\{S\}} = \frac{E\{M\}}{1/\lambda} \longrightarrow E\{M\} = E\{n\}E\{Y\} + E\{R\}$$

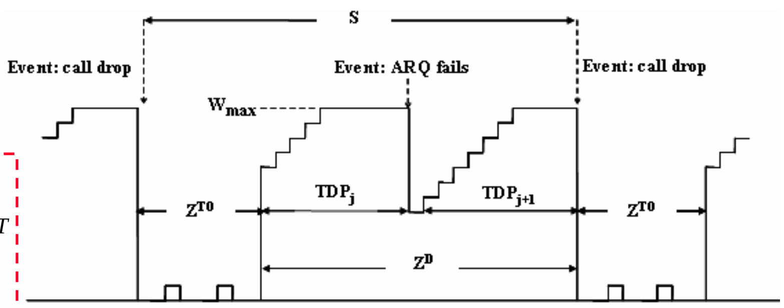
- The value of M can be expressed as the product of the number of the TDP periods (n) and the number of packets sent in each TDP period plus the number of packets sent during Z^{TO}

$$E\{M\} = E\{n\}E\{Y\} + E\{R\} \qquad E\{n\} = \frac{E\{Z^{D}\}}{E\{A\}} = \frac{E\{S\} - E\{Z^{TO}\}}{E\{A\}} = \frac{1/\lambda - E\{Z^{TO}\}}{E\{A\}}$$

- If M represents the number of packets sent during a period S, then the throughput (B) (in the case of call drops) is given as

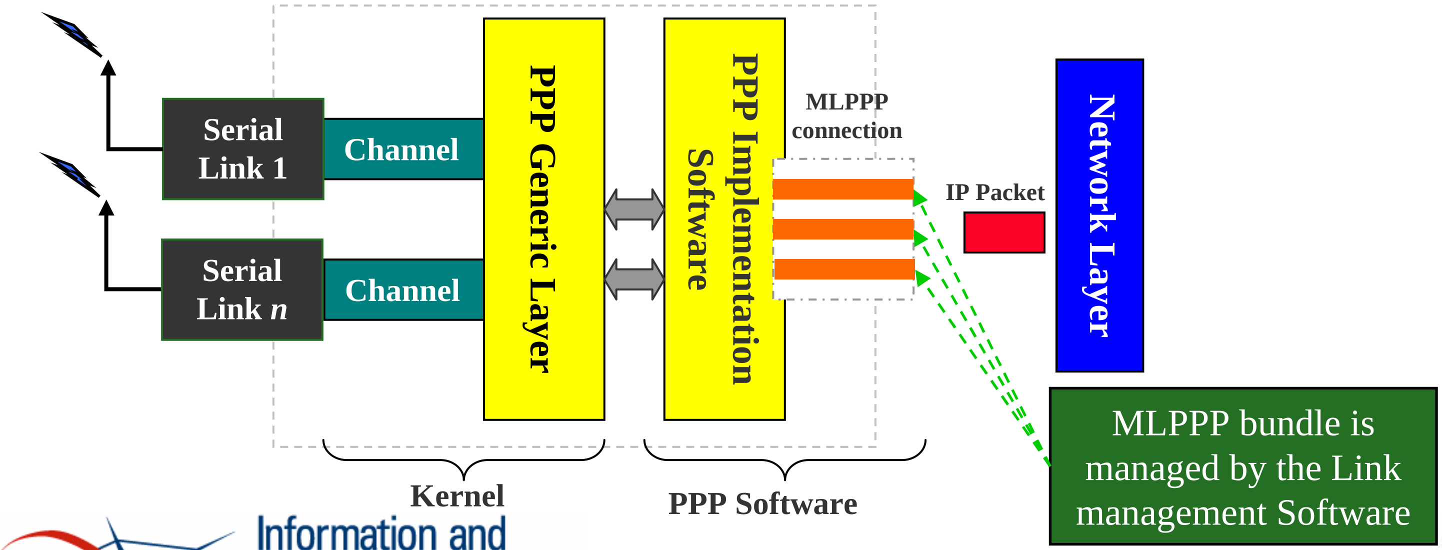
$$B = \frac{E\{n\}E\{Y\} + E\{R\}}{1/\lambda} = \frac{\frac{1/\lambda - E\{Z^{TO}\}}{E\{A\}} E\{Y\} + E\{R\}}{1/\lambda}$$

$$B = \left[1 - \lambda T_0 \frac{1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + +32p^6}{1-p}\right] B_{NT} + \frac{\lambda}{1-p}$$



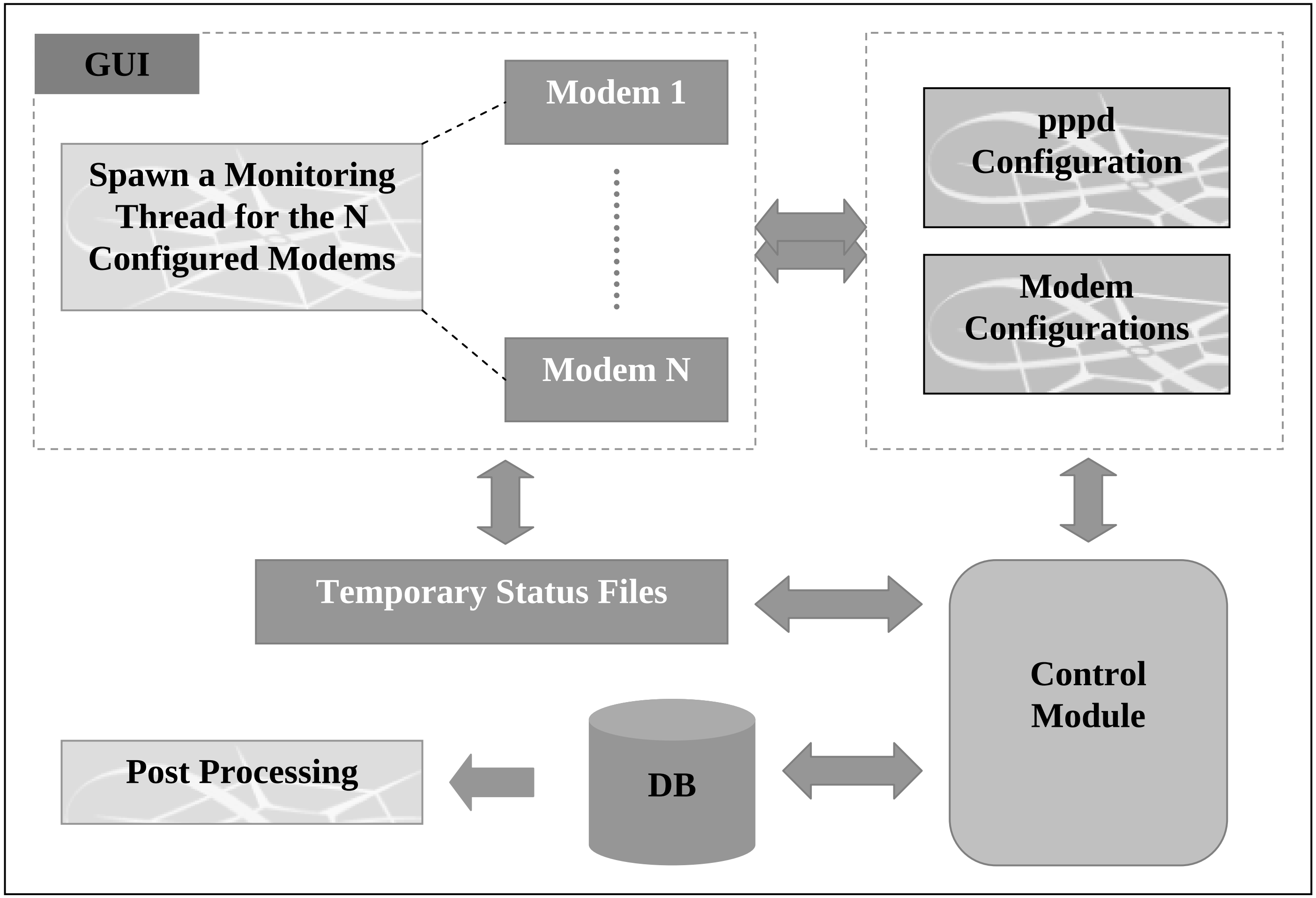
MLPPP Implementation

- Linux kernel provides a generic layer interface for basic PPP services
- PPP life cycle is implemented on an additional package (example, PPPD 2.4.1)
- MLPPP link management software uses PPPD 2.4.1 to handle call drop events



MLPPP bundle is managed by the Link management Software

KUMICS – Architecture



Aspects of TCP Operation

- Delayed ACK Timer
  Windows (100 – 200 ms), Unix (0 – 200 ms)
- Timeout Value
  Considers the variance of the RTT samples
- Performance is very sensitive to PER
  - TCP performs poorly if it is run directly over wireless links [10% losses degrades Reno's performance by (50-75%)]
  - Physical layer ARQ provides a reliable channel for TCP
  - ARQ types:
    stop-and-wait, Go-Back-N, and Selective ARQs
- Call Drops Effect in MLPPP

TCP Overview

- Many models capture TCP performance as a function of various parameters: (PER, periodic HO, congestions)
- TCP models help develop optimized protocol designs, such as, TCP over Satellite and CATNIP
- Available TCP models fall into two catagories:
  - Long Transfer Models (FTP): example Padhye’s model, relatively simpler analytical expressions
  - Short Transfer Models (Web browsing): example Sikdar’s Model, more complex because it has to consider various factors: slow start, delayed Ack timer expiration, packet loss rate

TCP Overview

- Popular versions of TCP are Tahoe, Reno, and SACK (and their variations)
- The behavior of the three versions is the same after a timeout: go to slow start.
- Different behaviors happen when having losses that do not result in timeouts.
  - Tahoe goes to slow start immediately
  - Reno can tolerate one loss per recovery period
  - SACK can handle multiple losses (half the congestion window)

Packet Loss Models

- Bernoulli (independent packet loss pattern)
  - A packet loss does not affect any of the following packets
  - Most suitable for networks that implement RED and for wireless connections running ARQ (fast fading)
- Drop-Tail (correlated loss pattern)
  - If a packet gets lost in a particular round all the following packets in the same round are lost
  - Suitable for Internet [routers use simple drop-tail queues]
- In this research, sources of packet losses are call drops and ARQ failures (negligible losses)

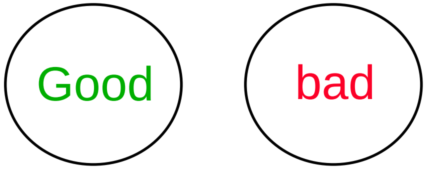
TCP Performance Model

- Objectives
  - Evaluate of the effect of call drops on TCP performance
  - Analyze TCP transfer latency as a function of PER and timeout variations
  - Investigate the effect of ARQ on TCP transfer latency as a function of physical layer parameters (*needs validation*)
- Procedure
  - Extend Padhye’s model to consider call drops
- Main Assumptions
  - Packet losses are due to ARQ failures (no timeouts)
  - Timeouts are caused by call drops only

ARQ Operation

Good

ARQ Operation



ARQ Operation

Good

bad

bad

ARQ Operation

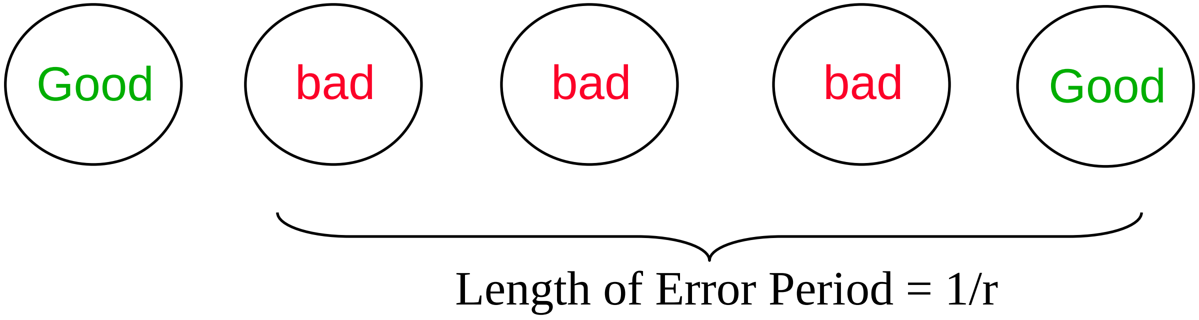
Good

bad

bad

bad

ARQ Operation



- The number of error periods in a transfer of b n ARQ frames follows a binomial distribution
- The average number of retransmissions is given by the product of the mean number of retransmissions in each error period and the mean number of error periods

$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + [ND_{ARQ} + RTT_{ARQ}]E\{U\}E\{k\}$$

ARQ Operation

- The RTT as a function of s, q is,

$$E\{RTT_{TCP}\} = nbD_{ARQ} + RTT_{ARQ} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1 - s^{NM}\left(1 + M\,(1 - s^{N})\right)}{\left(1 - s^{NM}\right)\left(1 - s^{N}\right)}\right]\left[q(nb-1)\right]$$

- Zorzi’s model for flat Rayleigh channel is used to evaluate s and q

$$F = \frac{1}{\mathrm{SNR}_{\min}}$$

$$\varepsilon = 1 - e^{-SNR_{\min}} \qquad \varepsilon = \frac{q}{q+r}$$

$$r = \frac{1-\varepsilon}{\varepsilon}\left[Q(\theta, \rho\theta) - Q(\rho\theta, \theta)\right]$$

$$\theta = \sqrt{\frac{2SNR_{\min}}{1-\rho^{2}}}$$

$$\rho = J_0^{2}(2\pi f_D \eta T)$$

$$Q(x,y) = \int_{y}^{\infty} e^{-\frac{(x^2+w^2)}{2}} I_o(xw)\,w\,dw$$

$$q = \frac{\varepsilon r}{1-\varepsilon} = Q(\theta, \rho\theta) - Q(\rho\theta, \theta)$$

$$f_D = \frac{v}{\lambda}$$

Discussion of ARQ performance

- q is high while s is low for fast fading channels
- It is known that if $f_D \eta T < 0.1$, the samples are highly correlated (slow fading channel)
- It is known that if $f_D \eta T > 0.2$, the channel samples are almost independent (fast fading channel)

The product $f_D \eta T$ is the key factor
that governs ARQ variations

- For high delay channels b = 1 and $E\{D_{ack}\}$ is added

$$E\{RTT_{TCP}\} = nD_{ARQ} + RTT_{ARQ} + E\{D_{ACK}\} + \left[RTT_{ARQ} + ND_{ARQ}\right]\left[\frac{1 - s^{NM}\left(1 + M\,(1 - s^{N})\right)}{\left(1 - s^{NM}\right)\left(1 - s^{N}\right)}\right]\left[q(n-1)\right]$$

Discussion of ARQ performance

- Block Error rate of the channel = 0.01
- Data Rate = 144 kbps
- Interleaving block size (η) = 500 bits
- Carrier Frequency = 1.616 GHz

At high speeds, the burst length approaches 1. This means that losses become independent.

The Markovian Channel Parameters as a Function of the Mobile Speed