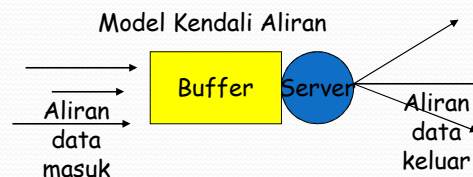


Flow Control

Kendali Aliran (Flow control)

- Fungsi lain yang diperlukan dalam mentransmisikan data di suatu link adalah kendali aliran
- Dibutuhkan terutama jika aliran data dari yang cepat ke yang lambat, dimana aliran data harus diatur agar penerima tidak overflow
- Mengatur aliran dengan cara:
 - Start - stop
 - Besarnya aliran



Dua Jenis Kendali Aliran

- Start-stop
 - Aliran data diatur sesuai dengan permintaan pihak penerima, jika penerima merasa buffer penerimaannya penuh, maka ia akan mengirim sinyal **stop** ke pengirim, dan jika buffer penerimaannya kosong, ia akan mengirim sinyal start.
 - Teknik ini sederhana, relatif mudah di implementasikan
 - Teknik start-stop umum:
 - RTS,CTS
 - X-on,X-off
- Mengatur aliran
 - Aliran data diatur berdasarkan besar bandwitdh saluran saat itu, teknik ini bekerja berdasarkan feedback dari penerima yang 'mengukur' laju data yang mampu dia terima.
 - Relatif lebih rumit dari teknik start-stop
 - Contoh : (sliding) window

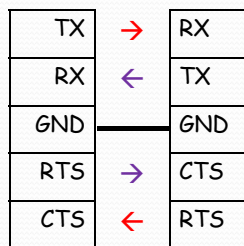
Pengguna Kendali Aliran

- Pengguna utama adalah protokol lapis datalink (RS-232, RS-..., HDLC,...)
- Untuk teknik kendali aliran yang lebih canggih diterapkan di lapis atas seperti TCP (lapis transport)

Kendali Aliran di RS-232

- Terdapat dua jenis kendali aliran yang bisa diterapkan di sistem komunikasi RS-232, yaitu teknik hardware dan teknik software
- **RTS - CTS (hardware)**, digunakan saluran tambahan untuk mengkomunikasikan informasi kendali aliran, dirancang untuk berkomunikasi dengan modem yang lebih lambat dari interface RS-232.

Koneksi fisik



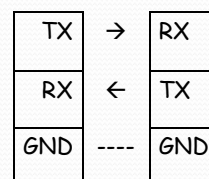
Pertukaran sinyal

- RTS
- Jika dijawab CTS maka TX jika tidak tunggu

Kendali Aliran di RS-232

- **Software (X-on, X-off)**, digunakan karakter-karakter tertentu untuk bertukar informasi kendali aliran
- Lebih sedikit membutuhkan koneksi fisik (2 kabel untuk satu arah komunikasi, 3 kabel untuk dua arah)
- **Algoritma kerja disisi pengirim**
 - Tunggu X-ON
 - Kirim TX
 - Jika mendapat X-OFF, berhenti kirim
- **Algoritma kerja disisi penerima**
 - Periksa buffer penerimaan
 - Jika kosong kirim X-ON, jika penuh kirim X-OFF

Koneksi fisik



Sliding window

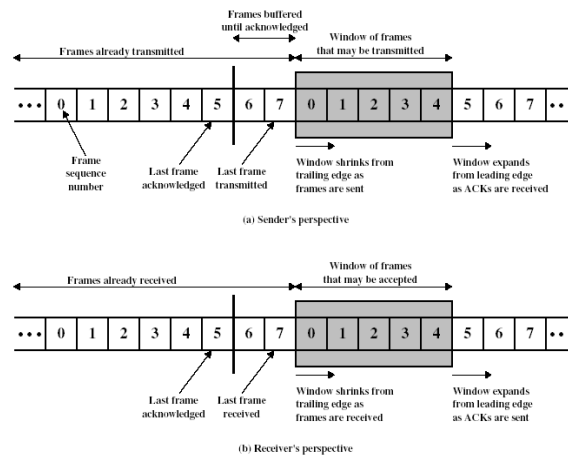
- Teknik kendali aliran start-stop mempunyai kelemahan trafik yang terjadi menjadi diskrit (bisa juga bursty), menyebabkan naiknya peluang kongesti di jaringan, tidak cocok untuk komunikasi jarak jauh (melalui banyak link).
- Dikembangkan teknik pengendalian aliran yang lebih adaptif sesuai dengan kondisi jalur transmisi yang dilewati, sehingga data dapat ditransmisikan dengan jumlah yang 'cukup' tidak berlebih dan tidak kurang. Teknik ini meningkatkan efisiensi bandwidth yang pada ujungnya akan mengurangi terjadinya kongesti jaringan.
- Salah satu teknik yang sejak awal dibuatnya protokol internet adalah teknik sliding windows

Sliding window

- Window = angka jumlah pengiriman paket saat ini
- Window = 3 → satu kali kirim maksimum 3 paket
- Cara kerja:
 - Penerima akan menetapkan jumlah window terimanya berdasarkan tingkat keberhasilan penerimaan paket, kebijakan yang ditetapkan oleh lapis aplikasi, dll
 - Pengirim kemudian akan mengirim paket sesuai dengan jumlah window yang ditetapkan penerima
- Pada TCP besarnya windows di'ikutkan' ke paket arah pengirim dari pihak penerima → tidak perlu paket khusus, meningkatkan efisiensi transmisi

SLIDING WINDOW

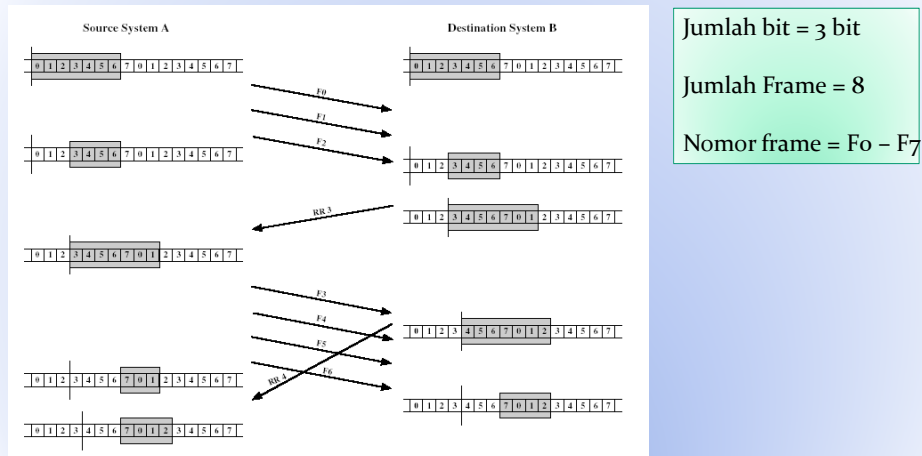
- ❖ Pada sliding window, stasiun pengirim boleh mengirim n frame tanpa menunggu acknowledgment.
- ❖ Secara umum, untuk field k-bit diperoleh range nomor urut 0 hingga 2^k-1 , dan frame dinomori modulo 2^k .
- ❖ Pada gambar berikut, asumsi nomor urut 3 bit dengan range dari 0 ke 7.



SLIDING WINDOW

- ❖ Karena frame yang berada dalam window pengirim bisa hilang atau rusak, pengirim harus tetap menyimpan frame tersebut dalam memorinya sebagai antisipasi kemungkinan retransmisi.
- ❖ Piggybacking → teknik penumpangan balasan pada frame data untuk komunikasi 2 arah (menghemat kapasitas komunikasi).
- ❖ Sending window: jumlah deretan frame maksimum yang dapat dikirim pada suatu saat
- ❖ Receiving window: jumlah frame maksimum yang dapat diterima

SLIDING WINDOW



SLIDING WINDOW

- ❖ Asumsi: field nomor urut 3-bit dan ukuran window maksimum 7 frame.
- ❖ Mula-mula A dan B mengindikasikan bahwa A akan mengirim 7 frame, dimulai dengan frame 0 (F_0)
- ❖ Setelah transmit 3 frame (F_0, F_1, F_2) tanpa ack, A telah mengurangi window-nya menjadi 4 frame dan tetap menyimpan kopi dari ketiga frame yang baru dikirim.
- ❖ Window ini berarti A masih boleh mengirim 4 frame lagi, dimulai dari frame 3.
- ❖ Kemudian B mengirim RR3 (receive ready), yang berarti “saya telah menerima sampai frame 2 dan siap menerima 7 frame berikutnya yang dimulai dari nomor 3”
- ❖ Dengan ack ini, A mendapat ijin untuk mengirim 7 frame, serta A dapat menghapus/menghilangkan frame 0, 1, dan 2 dari buffer
- ❖ A melanjutkan pengiriman frame 3, 4, 5, dan 6.
- ❖ dst.

Besarnya window

- Untuk setiap algoritma ARQ yang telah dipelajari, ukuran window yang sesuai adalah:

ARQ	Window Kirim	Window Terima
Idle RQ	1	1
Selective Repeat	N	N
Go Back N	N	1

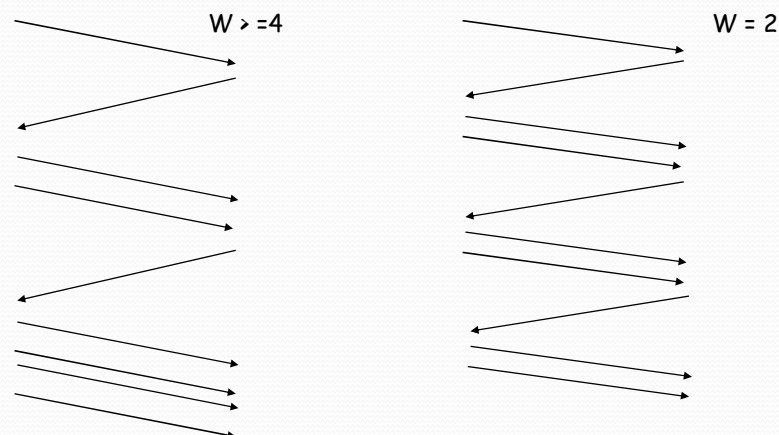
Implementasi Windows di TCP

Source port											
Destination port											
Sequence number											
Acknowledge number											
Header length		Reserved				URG	ACK	PSH	RST	SEQ	FIN
Windows											
Checksum CRC-16											
Urgent pointer											
Options											
						Padding					

Implementasi Windows di TCP

- Disediakan 16 bit untuk windows (dari 0 sd 64k).
- Untuk TCP awal, windows dimulai dari 1, kemudian naik dua kalinya untuk setiap tahap pengiriman sampai maksimum yang ditetapkan penerima, jika terjadi kegagalan penerimaan sebelum mencapai maksimal tersebut, windows akan di set kembali menjadi 1.
- Disebut mekanisme *Slow-Start TCP* yang tentu saja untuk kondisi jaringan saat ini dianggap terlampau berhati-hati, sehingga dikembangkan berbagai mekanisme TCP lain untuk memperbaikinya: TCP-Reno, TCP-Vegas, dll

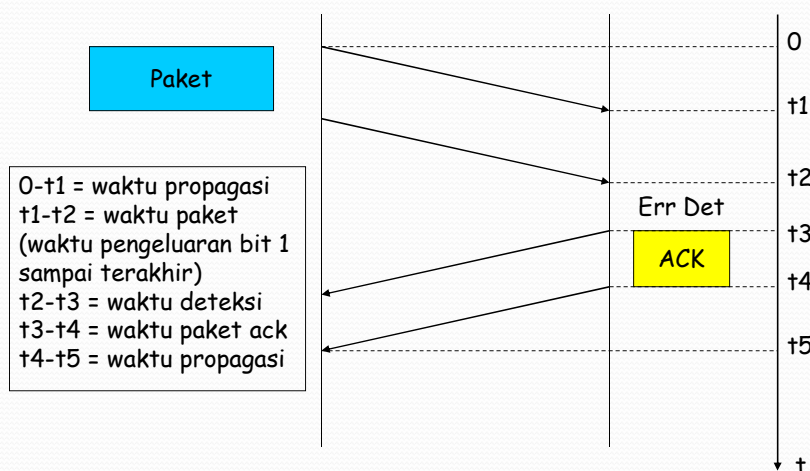
Pengaruh Ukuran Windows Terhadap Proses Pengiriman Paket



Perhitungan Waktu Transmisi Paket

- Suatu transmisi data di link memerlukan waktu.
- Penggunaan ARQ menyebabkan waktu transmisi adalah sama dengan waktu dari mulai paket dikirim sampai dengan waktu diterimanya ACK oleh pengirim
- Komponen waktu transmisi bisa dihitung dengan penyerhanaan sebagai berikut:

Perhitungan Waktu Transmisi Paket



Perhitungan Waktu Transmisi Paket

- Waktu propagasi = waktu yang diperlukan untuk 1 bit menempuh jarak pengirim-penerima →

$$t_{pro} = \text{jarak} / \text{kecepatan}$$

- Waktu paket = waktu yang diperlukan untuk mengeluarkan semua bit pada paket tersebut →

$$t_{pac} = \text{panjang paket (bit)} / \text{bitrate}$$

Perhitungan Waktu Transmisi Paket

- Waktu deteksi = waktu yang dibutuhkan oleh penerima untuk menentukan paket yang diterima benar atau salah →

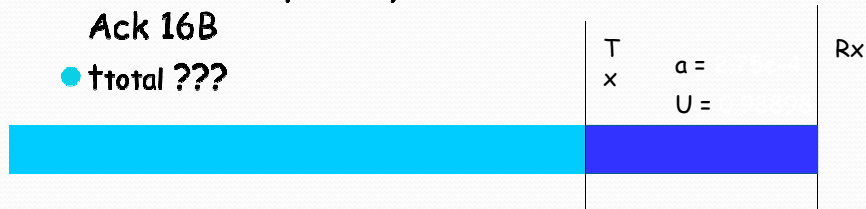
$$t_{det} = f(\text{metoda, kecepatan komputer}) \text{ undefined (semakin hari semakin cepat) (kelas ns)}$$

- Waktu paket ack → $t_{ack} = \text{panjang paket ack} / \text{bitrate}$

$$t_{total} = 2t_{pro} + t_{pac} + t_{det} + t_{ack}$$

Contoh kasus:

- Interface E : 10 Mbps,
Paket 1512B, 100m,
Ack 16B
- t_{total} ???



t_{pro}

t_{pak}

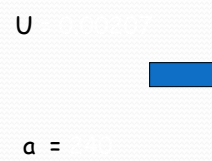
t_{ack}

t_{det} 0.00000001 s

$t_{total} =$

Contoh kasus lain:

- Satelit :
 - Geosat : 36000km
 - E1 : 2Mbps
 - Paket : 1000b, Ack : 40b
 - T_{total} ???



t_{pro} s

t_{pak} s

t_{det} 0.00000002 s

t_{ack} s

t_{total} s

Rumus Hasil Penyederhanaan

- Didapatkan dari kedua contoh kasus, waktu total transmisi paket didominasi oleh waktu paket atau waktu propagasi tergantung dari jarak transmisi, sedangkan waktu deteksi sangat bergantung pada kecepatan perhitungan penerima yang cenderung semakin kesini semakin cepat, waktu ack relatif dapat diabaikan karena panjang paket ack jauh lebih kecil dari panjang paket data.
- Sehingga : $t_{total} \approx 2 t_{propagasi} + t_{paket}$

Variabel a

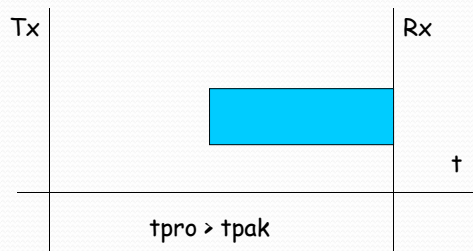
- Untuk memudahkan penulisan rumus dan memperlihatkan suatu variabel penentu hasil perhitungan utilitas link, maka dibuatlah variabel a
- Dengan: $a = t_{pro}/t_{pak}$

Kasus $a = 1$



- $a = 1$, menyatakan gejala fisik saluran akan penuh oleh paket, dalam arti bit pertama mulai diterima saat bit terakhir dikirim
- Terjadi jika waktu untuk menghasilkan paket sama persis dengan waktu propagasi

Kasus $a > 1$



- $a > 1$, menyatakan gejala fisik saluran akan sebagian kosong, dalam arti paket telah selesai dihasilkan saat bit pertama diterima
- Terjadi jika waktu untuk menghasilkan paket lebih kecil dari waktu propagasi

Kasus $a < 1$



- $a < 1$, menyatakan gejala fisik saluran akan penuh oleh paket lebih lama dari waktu propagasinya
- Terjadi jika waktu untuk menghasilkan paket lebih lama dari waktu propagasi

Utilitas Link

- Penggunaan ARQ menyebabkan tidak mungkin saluran fisik digunakan 100% (utilitas link = 1) untuk mengirim paket data, terdapat waktu yang terbuang (tidak dipakai mengirim data) untuk pengiriman ARQ dan karena variabel a pada link tersebut
- Definisi: utilitas link = waktu efektif (yang betul-betul digunakan) paket dari *waktu total* pengiriman paket
- Atau : rasio perbandingan waktu paket dengan waktu total pengiriman paket
 - $U = t_{pak} / t_{total}$
 - IdleRQ(tanpa error) $\rightarrow U = 1 / (2a+1) \rightarrow U_{maks} = 1$, jika $a \ll$

Pengiriman Ulang Paket

- Kapanakah pengiriman paket harus diulang?
 - Jika tidak sampai (1)
 - Jika ada error bit (2)
- Kasus (1) susah dianalisis, kasus ke (2) lebih mudah → ada ilmu statistik utk analisis
- Pengiriman ulang utamanya disebabkan oleh parameter Bit Error Rate (BER) pada saluran tersebut.
- BER adalah *besaran statistik* yang menyatakan peluang error bit di suatu saluran transmisi, disebabkan oleh banyak faktor, baik faktor alam maupun faktor perangkat yang berpengaruh pada kondisi saluran

Parameter BER

- BER = $1e-3$ berarti terjadi *rata-rata* 1 bit error untuk setiap 1000 bit
- 1 bit atau >1bit error akan menyebabkan paket tersebut dinyatakan error
 - Yang menyebabkan paket error bisa 1,2,3,atau lebih
 - $Pe(\text{paket}) = Pe(1) + Pe(2) + Pe(3) + \dots Pe(n)$
 $= \underline{1 - Pe(0)}$

Binomial

- Perhitungan peluang error paket $P_e(k)$ (peluang error k bit dari paket n bit, jika diketahui peluang error bit p) menggunakan binomial

$$P(k) = \binom{n}{k} p^k (1-p)^{n-k}$$

$$\text{dengan } \binom{n}{k} = n! / k!(n-k)!$$

Contoh Perhitungan BER

- Berapakah peluang error paket 1000 bit jika diketahui $BER = 10^{-5}$??

- Bisa dihitung sebagai

$$\bullet Pe(\text{paket}) = 1 - Pe(0)$$

$$= 1 - 1.1.(1-BER)^{1000}$$

$$= 1 - 0.99004978$$

$$= 0.00995$$

$$\approx 0.01 \text{ (Setiap 100 paket rata-rata 1 paket error)}$$

- Setara : $Pe(\text{paket}) = BER * \text{panjang paket} \rightarrow$
syarat $\text{abs}(\text{pangkat BER}) \gg \text{panjang paket}$

Contoh Lain Perhitungan BER

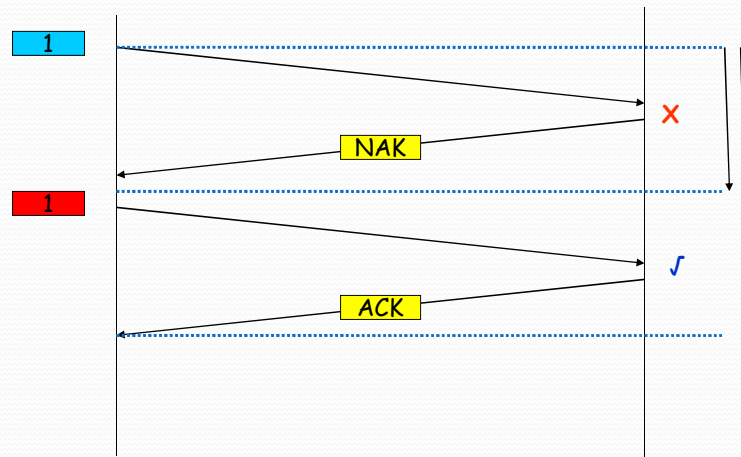
- Berapakah peluang error paket 1000 bit jika diketahui : $BER = 10^{-5}$ dan error yang terjadi untuk setiap paket maksimum 3 bit??
- Bisa dihitung sebagai
 - $Pe(\text{paket}) = Pe(1) + Pe(2) + Pe(3)$
 - $= 9.9e-3 + 4.94e-5 + 1.64e-7$
 - $= 9.95e-3$

Hubungan $Pe(\text{paket})$ Dengan ARQ

- $Pe(\text{paket})$ menentukan berapa kali pengiriman ulang !!
- $Pe(\text{paket}) = 1/2 \rightarrow N$ (rata-rata banyaknya pengiriman suatu paket agar diterima dengan benar) = 2
- $Pe(\text{paket}) = 1/3 \rightarrow N = 3/2$
- $Pe(\text{paket}) = 1/4 \rightarrow N = 4/3$

$$N = 1 / (1 - Pe)$$

ARQ : Idle RQ



Pengaruh Ke Parameter U

- Dikarenakan ada pengiriman ulang sebanyak N kali, maka rumus perhitungan parameter U mengalami revisi menjadi:

$$U = 1 / (N(1+2a))$$

atau

$$U = (1-P_e) / (1+2a)$$

Contoh Perhitungan U

■ Satelit :

- Geosat : 36000km
- E1 : 2Mbps
- Paket : 1000b, Ack : 40b
- BER = $10e-5$



t _{pro}	0.12 s	a = 240	U(tanpa error) = 0.00207
t _{pak}	0.0005 s	U = (1-Pe) / (1+2a)	
t _{det}	20 ns	= (1-10e-5*1000)/(1+480)	
t _{ack}	0.00002 s	= 0,99/481	
t _{total}	0.24052002 s	= 0.00205821	

Selective Repeat Tanpa Error

- Dengan analisis yang sama, didapatkan untuk mekanisme ARQ selective repeat utilitas jaringan menjadi:

$$U = \frac{K}{1+2a} \rightarrow K < 1+2a$$

$$U = \frac{(1+2a)}{1+2a} \approx 1 \rightarrow K \geq 1+2a$$

Selective Repeat dengan Error

- Dengan analisis yang sama, didapatkan untuk mekanisme ARQ selective repeat utilitas jaringan menjadi:

$$U = \frac{K(1 - Pe)}{1 + 2a} \rightarrow K < 1 + 2a$$

$$U = \frac{(1 + 2a)(1 - Pe)}{1 + 2a} \approx 1 - Pe \rightarrow K \geq 1 + 2a$$

Go Back N

- Sedangkan untuk Go Back N analisisnya jauh lebih rumit dikarenakan adanya pengiriman ulang paket dalam jumlah besar.
- Didapatkan utilitas link memenuhi rumus:

$$U = \frac{K(1 - Pe)}{(1 + 2a) + (1 + 2a)Pe(K - 1)}$$

$$= \frac{K(1 - Pe)}{(1 + 2a)(1 + Pe(K - 1))} \rightarrow K < 1 + 2a$$

$$U = \frac{(1 + 2a)(1 - Pe)}{(1 + 2a)(1 + Pe(2a))}$$

$$= \frac{1 - Pe}{1 + Pe(2a)} \rightarrow K \geq 1 + 2a$$

Catatan : $Pe = 1 - (1 - \text{BER})^{(n-k)}$

Soal Latihan

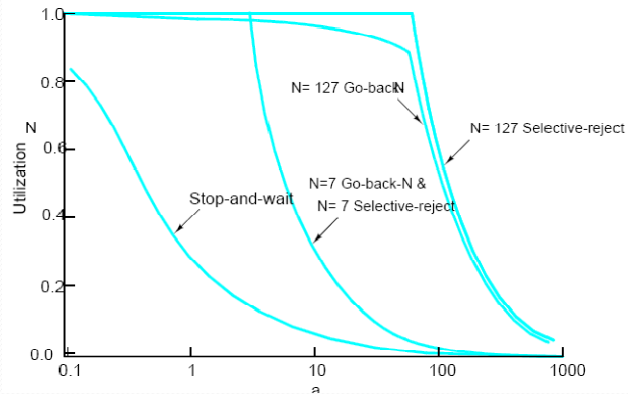
- Paket-paket 1000bit dikirim melalui datalink 100km dengan lajudata 20Mbps. Jika kecepatan propagasi link 2×10^8 m/detik dan BER 4×10^{-5} . Hitung utilisasi link, jika:
 - Idle RQ
 - Selective Repeat dengan K=10
 - Go Back N dengan K=10

Solusi

- $T_{\text{pro}} = L/V = 100 \text{ km} / (2 \times 10^5 \text{ km/s}) = 500 \text{ us}$
- $T_{\text{paket}} = P/\text{Bitrate} = 1000 \text{ bit} / 20 \text{ Mbps} = 50 \text{ us}$
- $a = t_{\text{pro}}/t_{\text{paket}} = 500 \text{ us} / 50 \text{ us} = 10$
- $P_e = 1 - P_e(0) \approx (1000 \times 4 \times 10^{-5}) \approx 0,04$
- IdleRQ : $U = 1 / N(1+2a) = (1 - P_e)/(1+2a) = 0,96 / 21 = 0,0457$
- SR : $U = K (1 - P_e)/(1+2a) = 9,6 / 21 = 0,4571$
- GBN : $U = K (1 - P_e) / (1+2a) + (1+2a)P_e(K-1) =$
 $10 (0,96) / (21 + 21(0,04(10-1))) =$
 $9,6 / (21 + 7,56) =$
 $0,3382$

Kesimpulan ARQ

Performance Comparison



- Yang paling efisien (Utilitas link lebih tinggi) = Selective Repeat
- Yang paling tidak efisien = Idle RQ
- Utilitas link = $f(\text{metoda, BER, panjang paket, jumlah paket sekali pengiriman})$