

---

# TCP

## 誤り制御とフロー制御

コンテンツ配信技術7

菊池浩明

# トランスポート層の役割

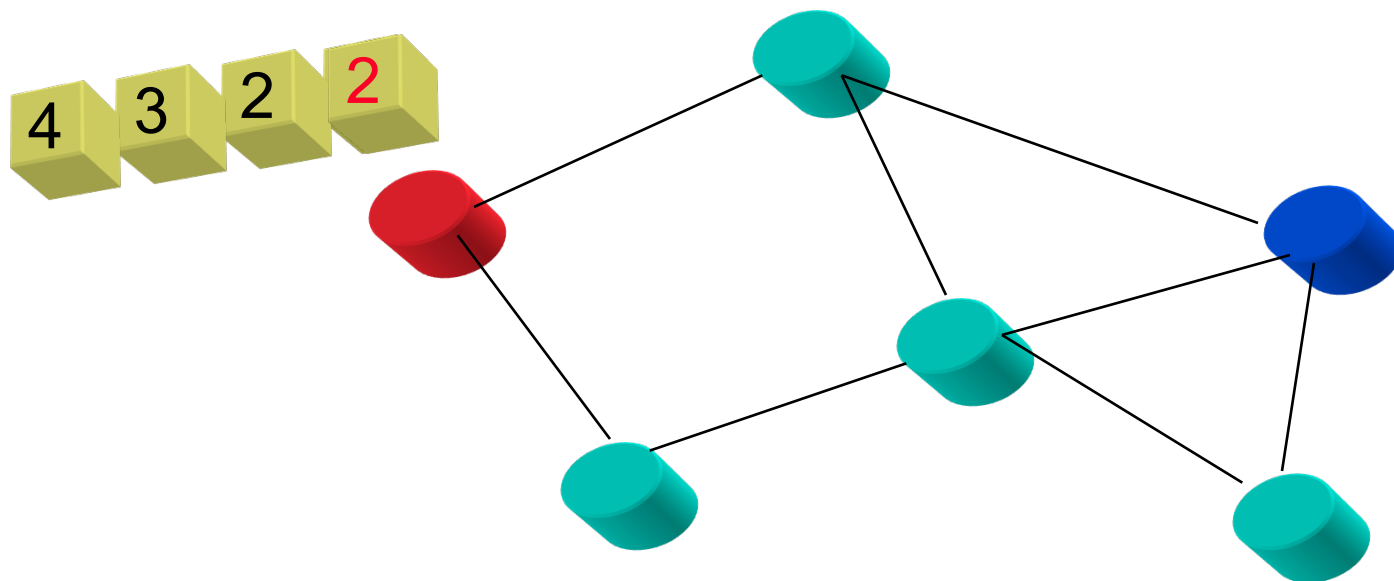
---

- ネットワークで高信頼サービスを提供すること
  - 制御
  - 順序制御
  - フロー制御

# パケット通信の問題

---

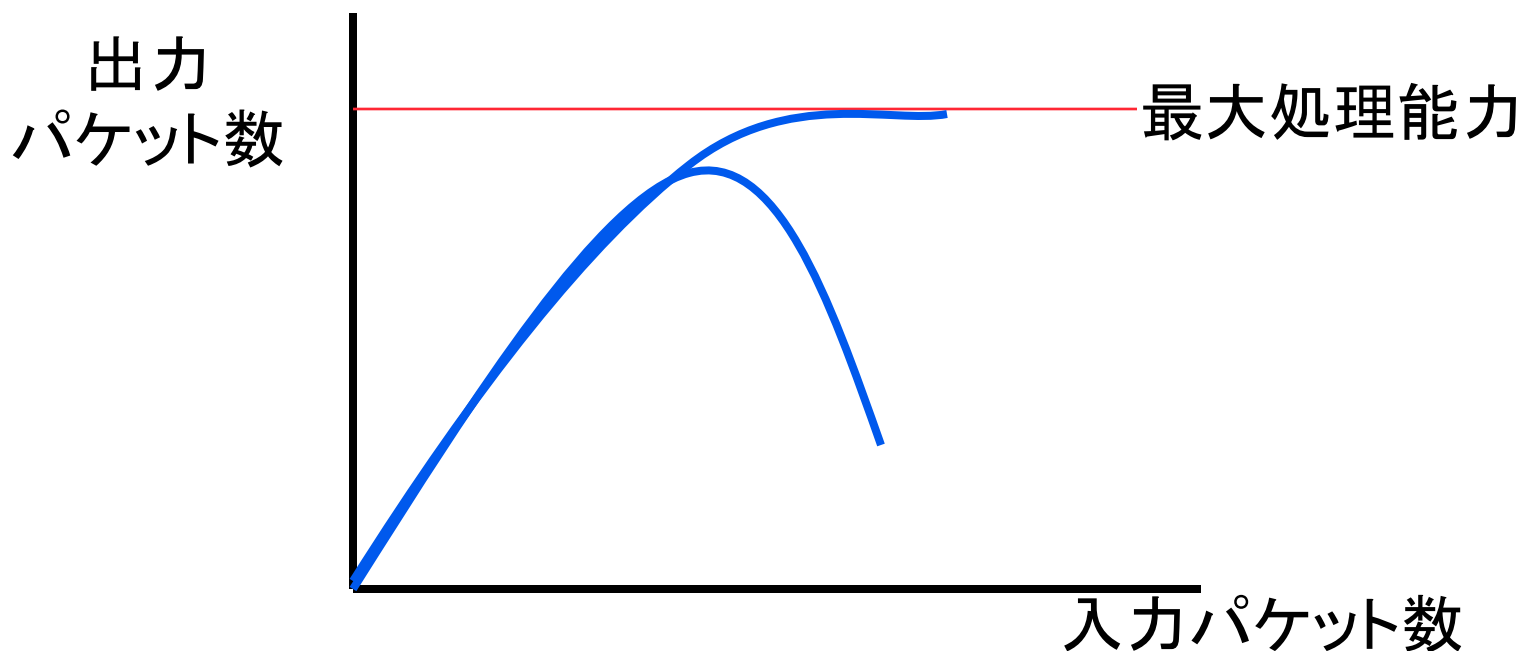
- パケット損失 → 誤り制御
- 到着順序 → 順序制御



# 輻輳制御

## ■ \_\_\_\_\_ (Congestion)

- パケット数の増加が限界に近づいて急激に性能が悪化する状態 → **輻輳**



# Internet

---

- \_\_\_\_\_による相互接続ネットワーク
  - The Internet
- TCP/IP
  - **TCP**: Transmission Control Protocol
    - » 伝送制御プロトコル
    - » RFC793
  - IP: Internet Protocol

# TCPとUDP

---

## ■ TCP

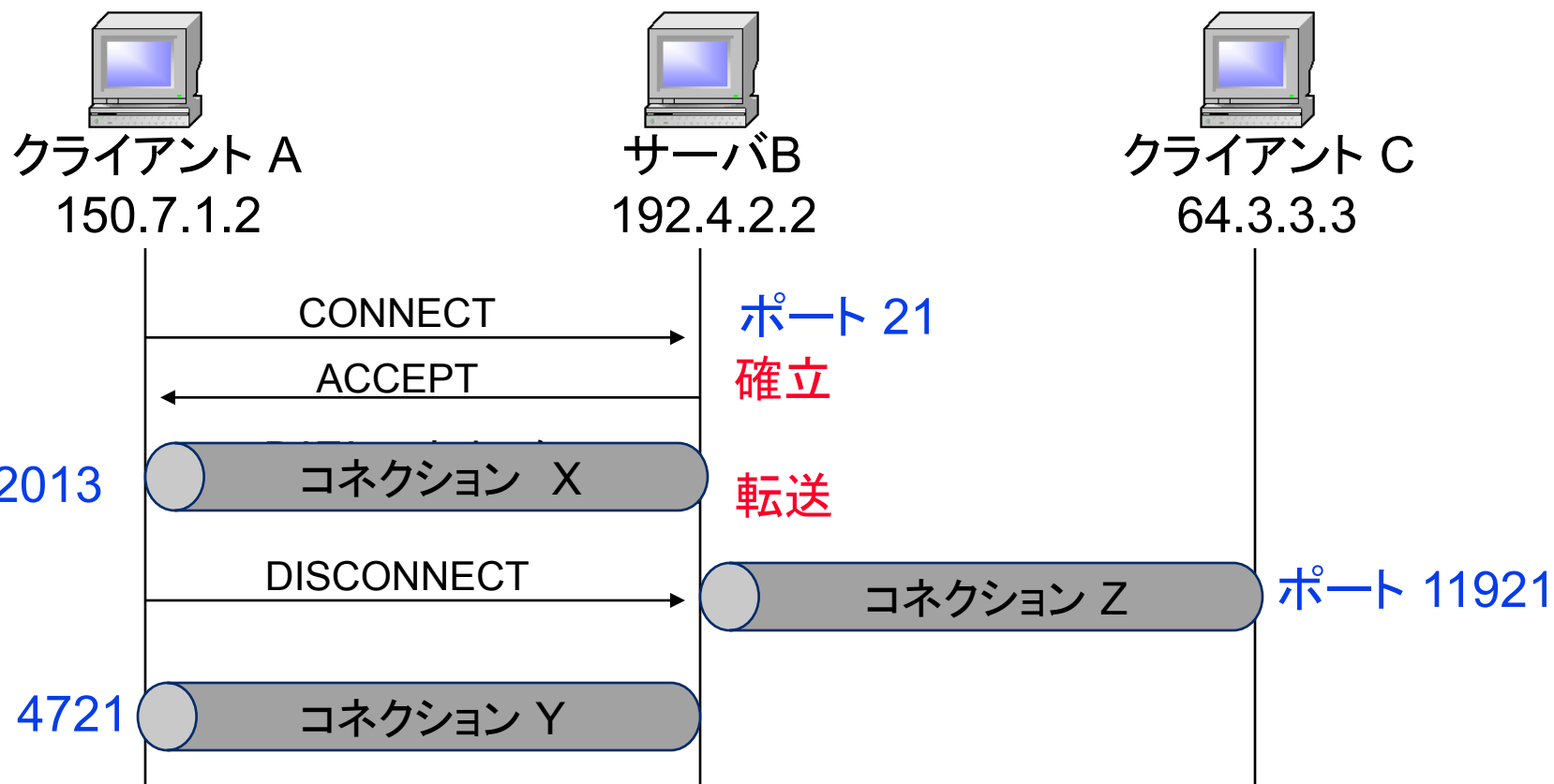
- Transmission Control Protocol
- **信頼性**のあるストリーム
  - » 再送と通信順序の保証
- コネクション・バーチャルサーキット

## ■ UDP

- \_\_\_\_\_ Datagram Protocol
- コネクションレス

トランスポート層		UDP
ネットワーク層	IP	
データリンク層	Ethernet / FDDI	

# コネクション



X の識別子: **エンドポイント** (150.7.1.2, **2013**) と (192.4.2.2, **21**)

Y の識別子: **エンドポイント** (150.7.1.2, **4721**) と (192.4.2.2, **21**)

# Well-known port

---

- 1024までの共通ポート番号
  - サーバ側で利用
  - RFC 1700 Assigned Numbers

ポート番号	サービス
21	ftp
23	telnet
25	smtp
	http
110	pop3

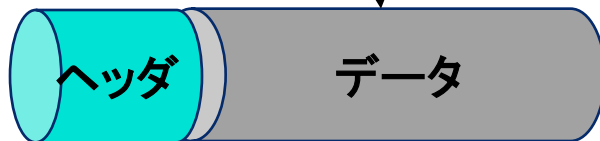


# ストリームとセグメント

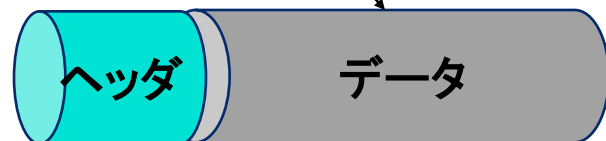
## ■ ストリーム



## ■ セグメント



シーケンス番号 SEQ=4000



SEQ=4100

# TCPとUDP

---

性質	TCP	UDP
	コネクション	
_____・フロー・ 順序制御	あり	なし
コネクションの 確立	必要	不要
オーバーヘッド (負担)	大	小
応用例	ssh, sftp, http	DNS, ntp, realaudio

---

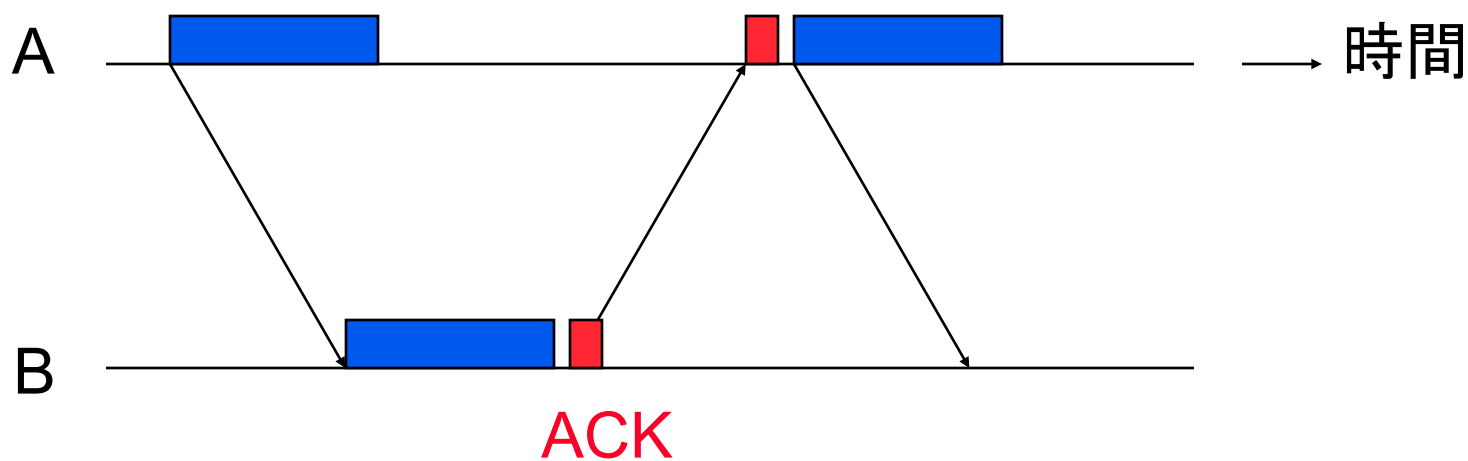
# トランスポート層の技術

Stop-and-wait  
sliding window

# 誤り制御1. Stop-and-waitプロトコル

## ■ ACK ( )

□ 到達確認情報



# フェイルセーフ(安全側に失敗する機構)

---

- ACK

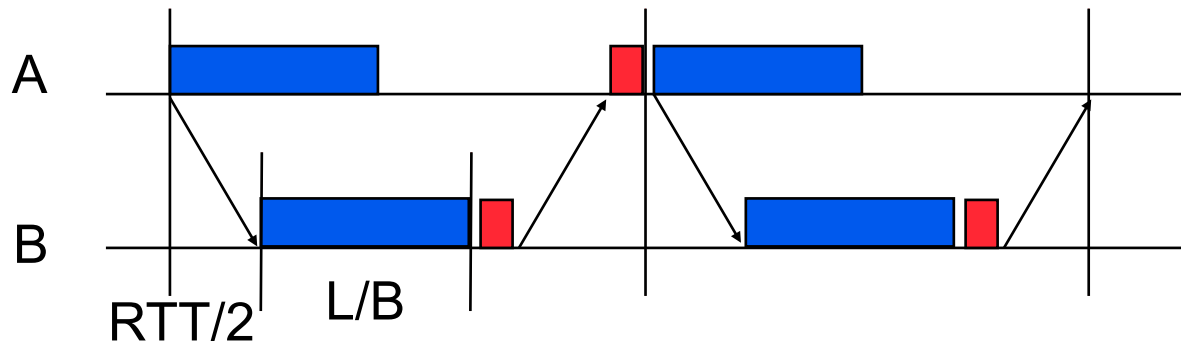
- 到達したとき送る

- NACK

- 未着の時送る

# Latency

- Latency (サービス要求にかかる遅延時間)  $E$ 
  - $E = (\text{受信完了時刻}) - (\text{送信開始時刻})$
- Stop-and-Waitプロトコル
  - 往復時間  $RTT = 20$  [ms]
  - 帯域(伝送速度)  $B = 10$  [Mbps]
  - データ長  $D = 20k$  byte
  - パケット長  $L_1 = 8000$  [bit], ACK長  $L_2 = 0$  [bit]
  - レイテンシー  $E = D/L (RTT + (L_1 + L_2)/B) = 20(20 + 0.8) = \underline{\hspace{2cm}} \text{ ms}$



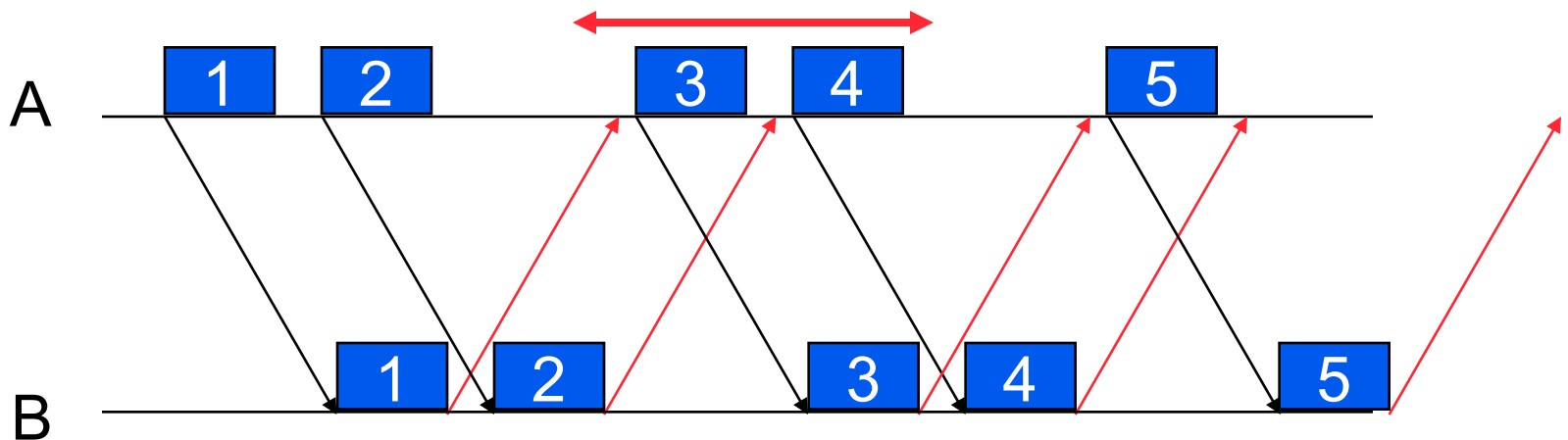
# 誤り制御2. sliding windowプロトコル

## ■ ウィンドウ (送受信バッファ) によるパイプライン

ウィンドウサイズ

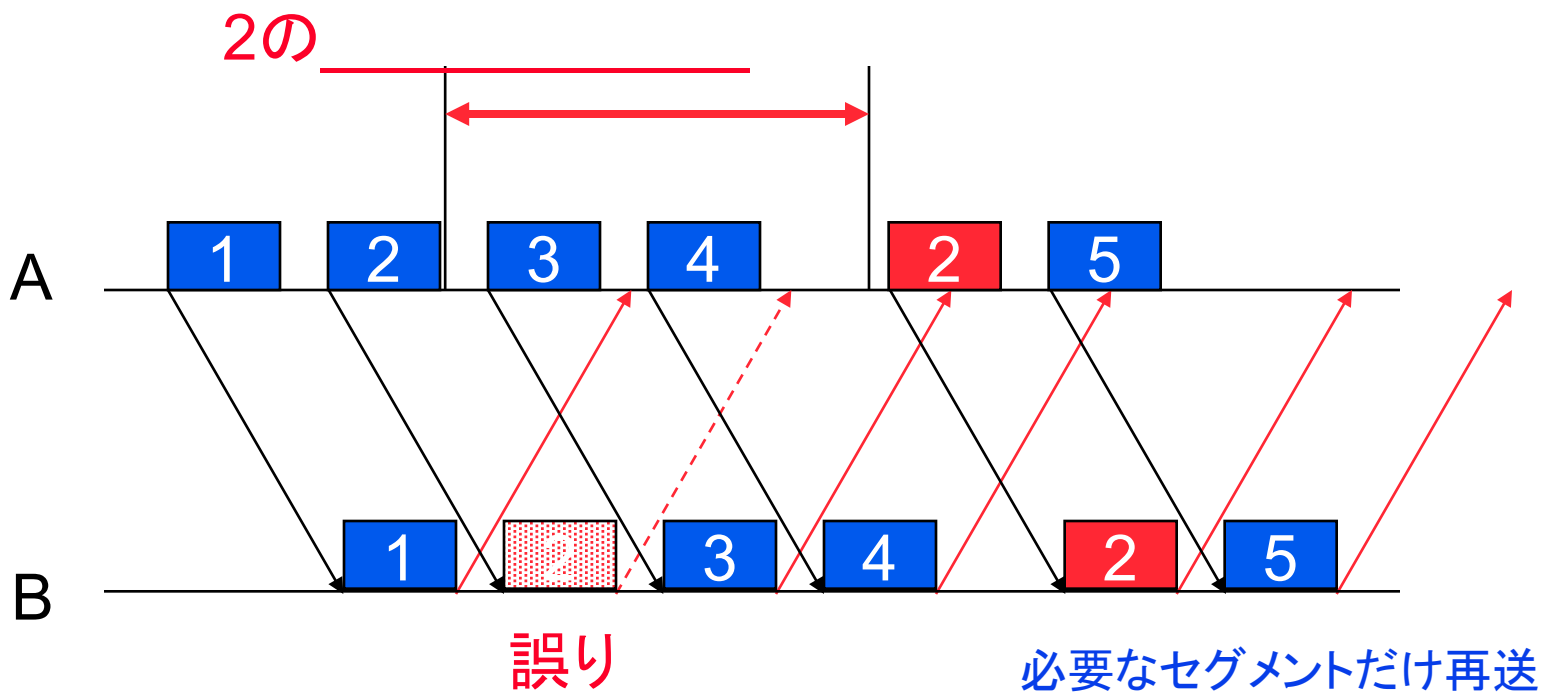
$$w=2L$$

2はACK1を待たずに送信



# 再送

## ■ 受信バッファとタイムアウト





# 固定長ウィンドウのレイテンシー

---

- 十分なウィンドウ長があるとき

- ウィンドウ長:  $W (>D = 20k)$

- $E_2 = RTT + D/B = 20 + 16 = \underline{\hspace{2cm}}$  [ms]

- 有限のウィンドウ長 $W$

- ウィンドウ長:  $W = 10k = 10L (<D = 20k)$

- 繰返し数:  $n = D/W = 2$

- パケット長  $L_1 = 8000$  [bit]

- $E_3 = n (RTT + W/B) = 2(20+8) = \underline{\hspace{2cm}}$  [ms]

# フロー制御

---

## ■ ウィンドウサイズ $W$

- 大きいほど回線利用効率が\_\_\_\_
- レイテンシー(サービス待ち時間)が小さい

W	E
0	416 ms
10k	56 ms
20k	36 ms

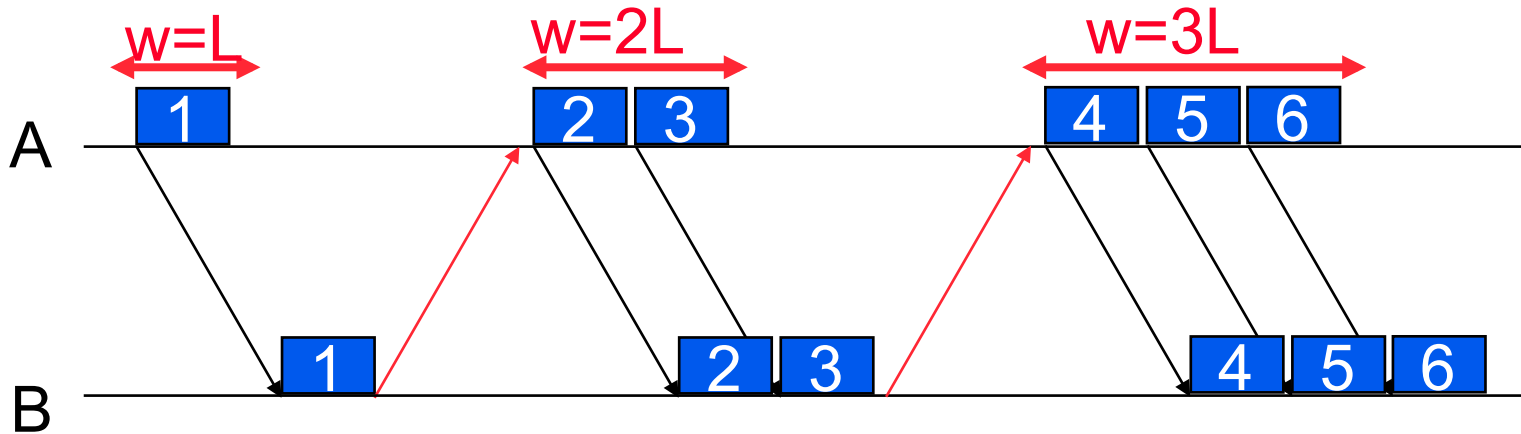
- 大きいほど\_\_\_\_\_も起きやすい
- 可変長, 最適値を得る様に更新していく

# 誤り制御3.

# プロトコル

## ■ 適応型可変長ウィンドウ

- 加法的増長:  $W = kL$  (連続した $k$ 回の通信)  
ウィンドウサイズ



- 指数的減少:  $W' = W/2$  if タイムアウトなど損失

# まとめ

---

- トランスポート層は、低信頼のネットワークで生じるパケット損失を( )制御し、受信側のバッファが溢れない様に( )制御を行う。
- 応答確認を行うパケットを( )といい、決められた時刻までにこれが届かないと( )する。送信が失敗した時にだけNACKを送るプロトコルは( )でない。
- TCPでは( )番号によって、パケットの到着順序を制御する。送受信の両方のIPアドレスと( )番号の組で、TCPの( )を識別する。( )では通信前に通信路を確立する必要がない。

# 演習

---

- $D=50\text{Mbyte}$ のデータを遅延時間  $\text{RTT}=2\text{ms}$ の宛先に送りたい.
  - 伝送速度  $B=20\text{ Mbps}$
  - パケット長  $L=1\text{kbyte}$  (パケット損失はなし, Ackの長さ=0)
  - 1. stop-and-waitプロトコルでレイテンシー  $E_1$  を求めよ.
  - 2. 固定長  $W=8L$  のスライディングウィンドウの長さは, このRTTに対して十分か?
  - 3. 2の時のレイテンシー  $E_2$ を求めよ.