

- Aula 4 -**CAMADA DE ENLACE DE DADOS****1. INTRODUÇÃO**

A Camada de Enlace de dados é a camada que lida com quadros, grupo de bits transmitidos pela rede. Ela depende da camada Física para enviar e receber os bits. Esta camada assegura que os quadros enviados pela rede serão recebidos com o devido tratamento de erros eventuais, que podem implicar em retransmissão do quadro.

A camada de enlace não é realmente parte do modelo TCP/IP, mas é o método usado para passar quadros da camada de rede de um dispositivo para a camada de internet de outro. Esse processo pode ser controlado tanto em software (*Driver*) para a placa de rede quanto em firmware ou *chipsets* especializados. Esses irão executar as funções da camada de enlace de dados como adicionar um header de pacote para prepará-lo para transmissão, então de fato transmitir o quadro através da camada física. Do outro lado, a camada de enlace irá receber quadros de dados, retirar os cabeçalhos adicionados e encaminhar os pacotes recebidos para a camada de internet. Esta camada não é orientada à conexão, se comunica pelos datagramas (pacotes de dados).

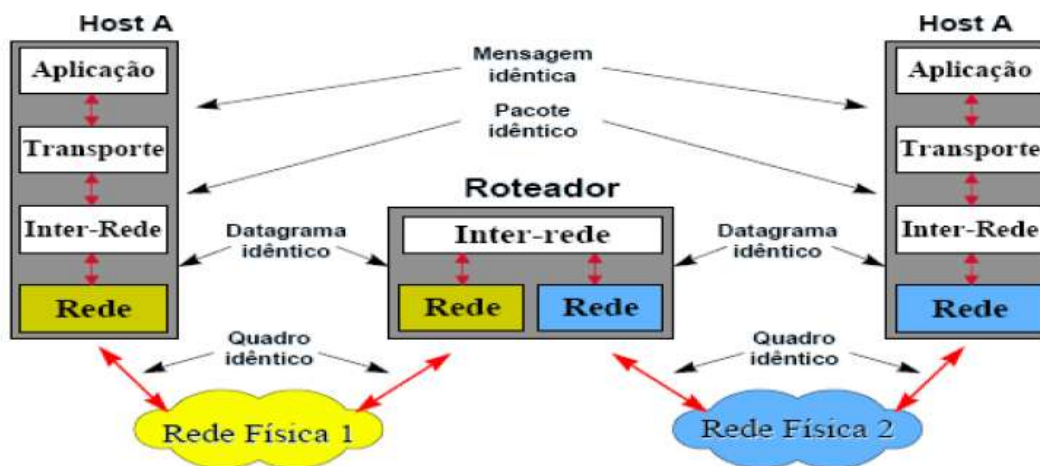
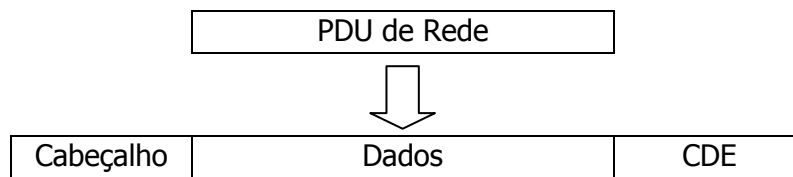


Figura 1 - Camada de Enlace

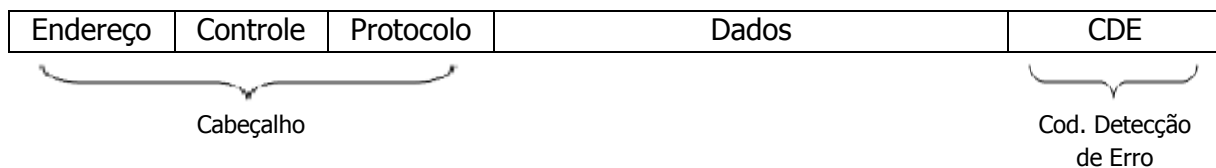
É função da camada de enlace de dados garantir a comunicação entre dispositivos adjacentes. Enquanto a camada física trabalha com bits, a camada de enlace de dados trabalha com blocos de bits, denominados **quadros** ou **frames**. Cabe a esta camada criar e interpretar corretamente os quadros, detectar possíveis erros e, quando necessário corrigi-los. Deve ainda controlar o fluxo de quadros entre um dispositivo e outro de forma a não sobrecarregá-lo com um volume excessivo de dados.

Na maioria dos protocolos de enlace os quadros são formados por três estruturas básicas: o cabeçalho, os dados e o CDE.

**Figura 2 - Estrutura de um quadro**

O cabeçalho possui informações de controle para que haja a comunicação horizontal entre as camadas de enlace da origem e do destino. O **cabeçalho** é formado por diversos campos, cada um com uma função específica no protocolo. O campo de **dados** encapsula o PDU (*Protocol Data Unit*) de redes passando pela camada de rede. Finalmente, o código de detecção de erro (**CDE**) tem a função de controlar erros na camada de enlace.

A figura abaixo apresenta o quadro do protocolo PPP (*Point-to-Point Protocol*), muito usado em conexões discadas e dedicadas. A figura representa os campos de protocolo e o tamanho de cada campo em bytes. Como é possível observar, os campos podem ter tamanhos diferentes e até mesmo tamanhos variáveis.

**Figura 3 - Quadro PPP**

Um quadro pode ser formado por uma seqüência de caracteres ou por uma seqüência de bits. No primeiro caso é formado por um conjunto de números inteiros formando os bytes a cada 8 caracteres. No segundo caso é transmitida uma seqüência de bits, não existindo uma relação entre o número de bits que compõem o quadro e o tamanho do caractere utilizado. Nesta seara existem vários protocolos. Alguns orientados a bit como o HDLC (*High-level Data Link Control*) ou orientado a caracteres como o BSC (*Binary Synchronous Control*). O protocolo PPP pode trabalhar tanto com um como com outro.

2. SERVIÇOS OFERECIDOS A CAMADA DE REDE

O serviço que deve ser ofertado a camada superior, a camada de rede, é de transferir dados da camada de rede da máquina de origem até a camada de rede da máquina de destino. Este serviço pode ser ofertado da seguinte forma:

- Serviços sem Conexão e sem Confirmação
- Serviço sem conexão e com Confirmação
- Serviço com Conexão e com Confirmação

2.1. Serviços sem Conexão e sem Confirmação

Neste caso a máquina de origem envia dados independentes a máquina de destino. Não há confirmação de recebimento, não há estabelecimento de conexão lógica, não nenhuma tentativa de detectar ou recuperar quadros danificados ou perdidos e é largamente usada em redes com taxa de erros muito baixas e recuperação em uma camada acima. É também muito usada em Redes Locais e em aplicações de tempo real.

2.2. Serviço sem conexão e com Confirmação

O serviço sem conexão e com confirmação não estabelece conexão lógica. O receptor confirma o quadro recebido para o emissor e retransmite quadros não confirmados após um tempo predeterminado sem confirmação. São utilizados em canais não confiáveis e em conexões sem fio.

2.3. Serviço com Conexão e com Confirmação

Este tipo de serviço estabelece conexão lógica entre o emissor e o receptor. Os quadros são numerados e a entrega é garantida. É garantida ainda a não duplicidade, a entrega na ordem correta.

Existem três fases no processo de transmissão:

- Estabelecimento da Conexão
- Transferência dos dados
- Encerramento da Conexão

3. ENQUADRAMENTO

Como a camada de enlace trabalha com quadros, o receptor deve ser capaz de identificar o início e o final de cada bloco transmitido. Essa função é chamada de enquadramento ou *Framing*.

Abaixo é apresentada uma seqüência de bits transmitida pela camada física, a qual a camada de enlace de dados identificou, através do mecanismo de enquadramento, três quadros:

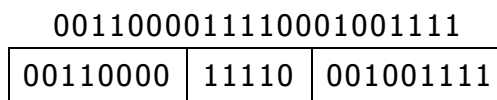


Figura 4 – Enquadramento

A maioria dos protocolos de enlace utiliza um *flag* para identificar os limites de cada quadro, que pode ser um caractere ou uma seqüência de bits especiais. Abaixo é apresentado um quadro delimitado no início e no final pelo flag 01111110. Esse esquema de enquadramento é implementado nos protocolos PPP e HDLC.

Flag 01111110	Quadro	Flag 01111110
------------------	--------	------------------

Flag 01111110	Endereço	Controle	Protocolo	Dados	CDE	Flag 01111110
------------------	----------	----------	-----------	-------	-----	------------------

Figura 5 – Delimitadores

O grande problema desta solução é quando a informação transmitida incorrer na mesma representação do *flag*. A solução para este problema são as técnicas de **byte stuffing** e **bit stuffing**. Ambas as técnicas consistem na utilização de um *scape*. Veja:

Quadro Original			Quadro Transmitido		
FI	FF ... FI ... FF ... CECE	FF	FI	CEFF ... CEFI ... CEFF ... CECECECE	FF

Figura 6 - Byte Stuffing

No *Bit Stuffing* a funcionalidade é parecida, porém para quebrar a igualdade a cada 5 bits 1 é inserido um 0. Ao ser recebido pelo receptor esse 0 é retirado e se aparecer um 0 após uma seqüência de 5 bits 1, significa que a seqüência é uma informação.

Quadro Original			Quadro Transmitido		
01111110	0111111010110	01111110	01111110	011111 0 1010110	01111110
Quadro Recebido			Quadro Original		
01111110	011111 0 1010110	01111110	01111110	0111111010110	01111110

Figura 7 - Bit Stuffing

Além dessas soluções, existem outras alternativas que levam em consideração o tamanho do quadro.

4. CONTROLE DE ERROS

Qualquer transmissão está sujeita a problemas, como ruídos e atenuação, e a camada de enlace de dados tem a função de realizar o tratamento dos possíveis erros.

O controle de erro envolve duas etapas:

- Detecção dos possíveis erros nos dados transmitidos;
- Correção dos erros encontrados.

O mecanismo de detecção de erro é semelhante ao esquema do dígito verificador largamente utilizado em códigos de barra, no CPF, etc. O dígito verificador é gerado a partir dos números que compõem os números antecessores, utilizando-se de uma função previamente definida.

A detecção de erros é feita através de informações de controle que são enviadas juntamente com os dados transmitidos. Antes de enviar uma mensagem o transmissor utiliza

uma função para gerar um código de detecção de erro (CDE) a partir da mensagem a ser enviada, de forma a gerar uma espécie de dígito verificador.

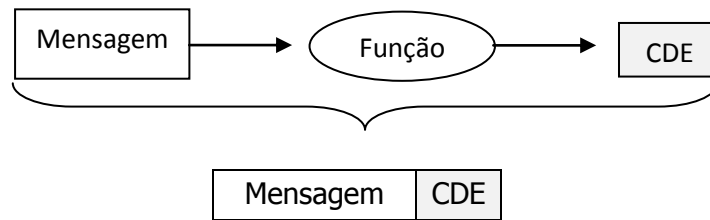


Figura 8 - Geração do código de detecção de erro

O destinatário ao receber a mensagem recalcula o código de detecção de erro (CDE) e o compara com o código recebido. Se o código calculado no destino for igual ao transmitido, a mensagem está íntegra, caso contrário houve algum erro na transmissão.

Controlar erros significa garantir que a informação que chegou ao destino é confiável. Isso pode implicar em descarte das informações erradas.

Existem duas técnicas largamente utilizadas para a detecção de erros. São elas:

- Bit de Paridade
- Verificação de Redundância Cíclica (CRC)

4.1. Bit de Paridade

Existem duas formas de implementá-la:

- Paridade simples
- Paridade múltipla.

A **paridade simples** consiste em adicionar um bit ao final de cada caractere transmitido, de modo que, com esse bit, o total de bits 1 seja par (paridade par) ou ímpar (paridade ímpar).

Caractere	Paridade Par	Paridade Ímpar
1011010	10110100	10110101
0000001	00000011	00000010

Este esquema permite identificar problema em apenas um bit do caractere. Se, por exemplo, o caractere 1011010 tiver dois de seus bits alterados, resultando em 1111000, o bit de paridade não permitirá identificar o problema. Este esquema deve ser utilizado somente em transmissões de baixa velocidade ou que apresentam poucos erros.

A **paridade múltipla** é uma melhoria da paridade simples. Além do bit de paridade adicionado ao final de cada caractere, é adicionado outro bit para um bloco de caracteres transmitido.

4.2. Verificação de Redundância Cíclica (CRC)

Esta é uma das técnicas mais utilizadas. Também conhecida como Código Polinomial já que é possível ver a seqüência de bits a ser enviada como um polinômio cujos coeficientes são os valores de 0 e 1 da seqüência de bits, sendo as operações na seqüência de bits interpretadas como aritmética polinomial.

O calculo do CRC baseia-se em uma propriedade da divisão que diz:

Se o resto for subtraído do dividendo e novamente dividido pelo mesmo divisor, o resto da divisão será zero.

$$\begin{array}{r} 210278 \quad | \quad 10941 \\ \underline{2399} \quad \rightarrow \text{resto} \end{array}$$

$$\begin{array}{r} (210278-2399) \quad | \quad 10941 \\ \underline{\quad\quad\quad} \quad \rightarrow \text{resto} \\ \text{zero} \end{array}$$

Figura 9 - Cálculo CRC

Este mesmo raciocínio é empregado no cálculo do CRC:

- O dado a ser transmitido é o dividendo
- O divisor é um número predefinido.
- O resto obtido dessa divisão, chamado de seqüência de verificação do quadro ou FCS (*Frame Check Sequence*), é incorporado ao dado e então transmitido.
- No destino é realizado o mesmo cálculo. Se o resto dessa divisão for zero, os dados enviados estão corretos, caso contrário houve erro na transmissão.

A figura a seguir mostra como ficaria utilizando-se de uma seqüência em binário.

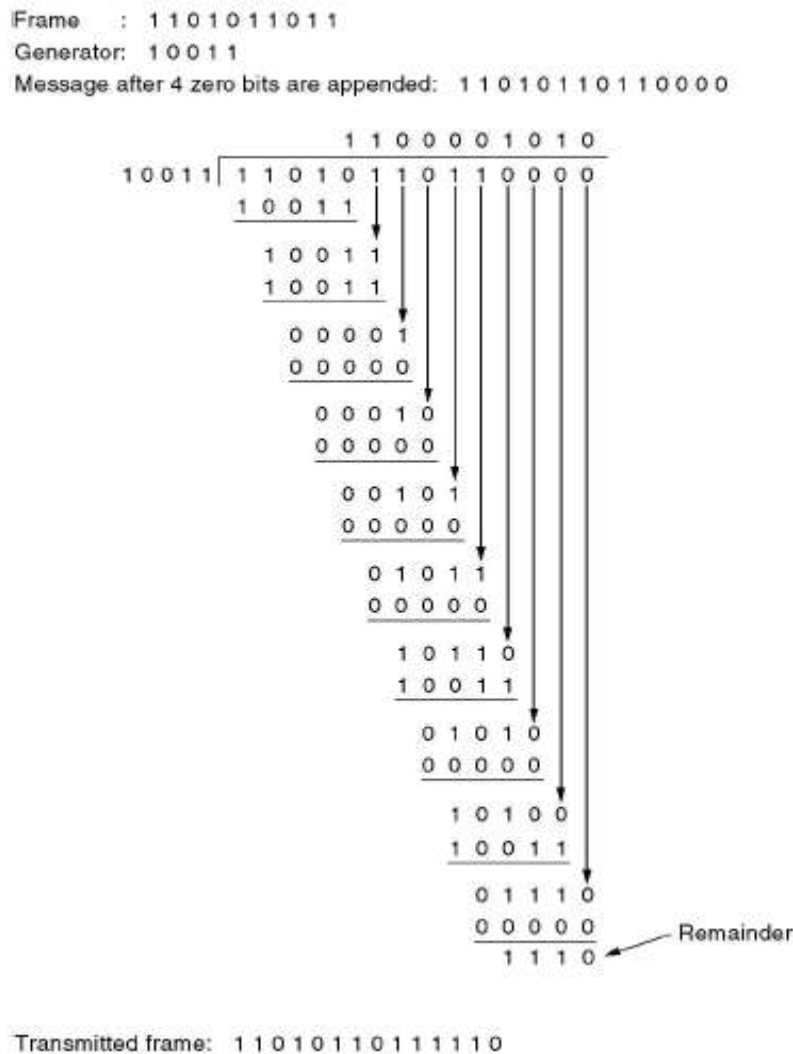


Figura 10 - Cálculo do CRC em binário

5. CORREÇÃO DE ERRO

A detecção de erro forma a primeira parte da função de controle de erros, no entanto nem sempre a camada de enlace de dados implementa algum mecanismo de correção de erros. Por outro lado, em meios onde a taxa de erros é muito grande, como em redes sem fio, o controle de erro torna-se indispensável nesta camada.

Sempre quando há uma transmissão com sucesso o receptor confirma a recepção do quadro utilizando um *ACK*(*ACKnowledgement*). O *ACK* funciona para o transmissor como um reconhecimento de que o destinatário recebeu corretamente o quadro enviado.

Quando um quadro não chega ao destino, o receptor não tem nada a fazer a não ser aguardar que o transmissor resolva o problema. Neste caso, o transmissor mantém um temporizador para cada quadro enviado e, caso não chegue um *ACK* em certo intervalo de tempo, ocorre o *timeout* e o quadro é retransmitido. Esse esquema é conhecido como **retransmissão por *timeout***.

O ACK pode ser implementado de duas formas diferentes: como um quadro especial ou fazendo parte do cabeçalho de enlace.

No caso do quadro chegar ao destino com erro, existe outras estratégias:

- Descartar o quadro recebido e aguardar que ocorra o timeout para que haja a retransmissão;
- Enviar um aviso ao transmissor indicando que houve erro no quadro e que este deve ser retransmitido;
- FEC (*Forward Error Correction*), que implementa a correção de erro no destino.

Os mecanismos que utilizam o reconhecimento e retransmissão de quadro como mecanismos para correção de erro são chamados genericamente de **protocolos ARQ(Automatic Repeat reQuest)**.

Para a correção de erros é muito comum encontrar o Código de Hamming. Basicamente o Código de Hamming emprega o *bit-stuffing* para assegurar que eventuais erros sejam detectados. Esses bits inseridos dentro do bloco de dados vão detectar com alguma eficiência, eventuais erros no envio de dados, bem como em alguns casos, corrigir mediante a posição do bit com erro, determinado pelo receptor.

O código de Hamming tem infinitas variantes, no entanto existe universalidade na representação da designação.

5. CONTROLE DE FLUXO

O controle de fluxo está no equilíbrio entre a taxa de transmissão e de recepção de quadros. O protocolo deve manter regras bem definidas sobre quando um transmissor pode enviar o quadro seguinte. O controle de fluxo permite que o dispositivo transmissor regule o volume de dados enviados de forma a não a gerar um *overflow* (transbordamento) no receptor. Caso contrário, o destino terá que descartar os dados transmitidos, obrigando a retransmissão dos mesmos.

Uma técnica de controle de fluxo bastante utilizada pelos protocolos da camada de enlace é fazer com que o receptor envie mensagens para o transmissor informando se está apto ou não a receber novos quadros. Exemplificando, imagine um dispositivo A que envia cinco quadros para o dispositivo B. O dispositivo B recebe os quadros, verifica que o seu buffer está cheio e, em seguida, envia uma mensagem para A informando o problema. Após receber a mensagem, o dispositivo A para de enviar novos quadros e aguarda até que o dispositivo B autorize novamente o envio. O dispositivo B processa os quadros recebidos e informa ao dispositivo A, que inicia a transmissão de novos frames.

Assim, o controle de fluxo é geralmente resolvido com o receptor informando ao transmissor sua capacidade de recepção de dados, ou seja, o tamanho do seu buffer de recepção. Dessa maneira, o transmissor pode regular o volume de dados enviados de forma a não sobrecarregar o destinatário.

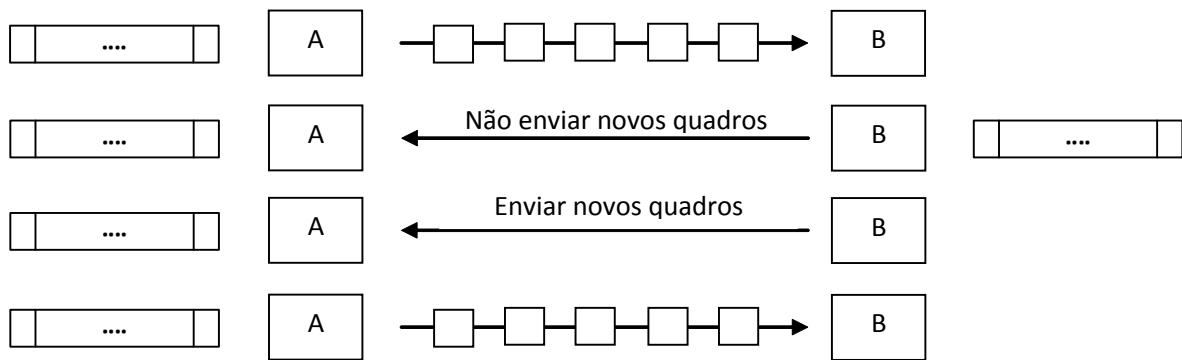


Figura 11 – Controle de fluxo

6. PROTOCOLOS

6.1. Protocolo Simplex sem restrição

É um protocolo muito simples, onde os dados são transmitidos somente em um sentido. As camadas de rede do transmissor e do receptor estão sempre prontas à espera de informações. O tempo de processamento pode ser ignorado, o espaço em buffer é infinito e o canal de comunicação entre as camadas de enlace de dados nunca é danificado e nem perde quadros. Em suma, é um protocolo imaginário ou "utópico".

O protocolo consiste em dois procedimentos distintos, um que envia e outro que recebe a informação. Neste caso, não são usados números de seqüência ou de confirmação.

A parte referente aos dados é repassada à camada de rede, e a camada de enlace de dados volta a esperar pelo próximo quadro, ficando efetivamente em suspenso até a chegada de outro quadro.

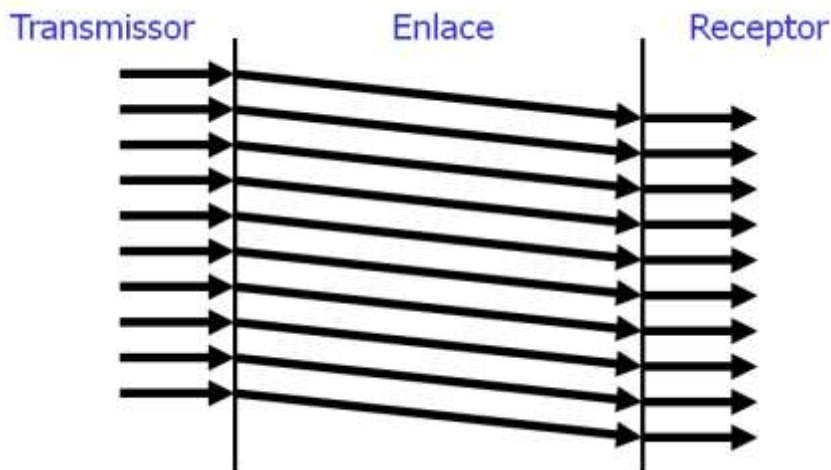


Figura 12 - Protocolo Simplex sem restrições

6.2. Um protocolo simplex *stop-and-wait*

O principal problema que este protocolo veio lidar foi impedir que o transmissor inunde o receptor com dados, mais rapidamente do que este é capaz de processá-los. Em determinadas circunstâncias talvez seja possível para o transmissor simplesmente inserir um retardo no protocolo 1, a fim de reduzir sua velocidade e impedi-lo de sobrecarregar o receptor.

Uma solução mais viável é fazer o receptor enviar um *feedback* ao transmissor, ou seja, uma vez enviado um determinado quadro, outro somente será enviado após o recebimento de uma confirmação. Esta estratégia é um mecanismo, inclusive de controle de fluxo. Embora o tráfego de dados seja simplex, há fluxo de quadros em ambos os sentidos.

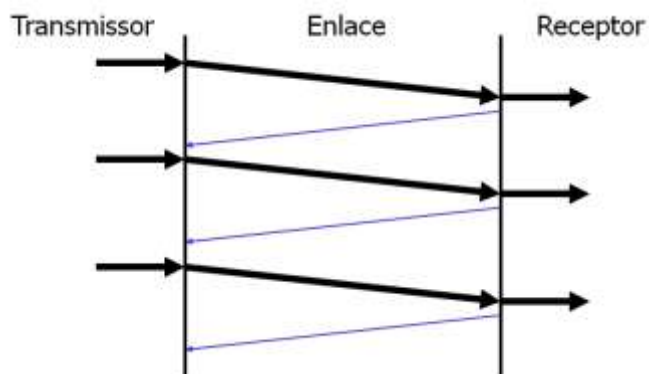


Figura 13 - Protocolo Simplex *stop-and-wait*

6.3. Um protocolo simplex para um canal com ruído

Neste protocolo considera-se uma situação normal em um canal de comunicação, no qual ocorrem erros. Os quadros podem ser danificados ou completamente perdidos. Se assim acontecer, supomos que o hardware receptor detectará essa ocorrência ao calcular o total de verificação.

Somente uma confirmação por parte do receptor não é suficiente. Caso a comunicação seja perdida, em muitos casos é necessário realizar uma retransmissão, e para isso é indispensável adicionar um número de seqüência no cabeçalho de cada quadro enviado de modo que o receptor saiba que o quadro enviado novamente é o mesmo que fora enviado anteriormente. O receptor informa caso recepção corra sem problemas e é importante notar que o número de seqüência pode ter comprimento de apenas 1 bit.

Esse tipo de protocolo tem diversas variantes, a saber:

- PAR (*Positive Acknowledgement with Retransmission*) – Confirmação Positiva com Retransmissão;
- ARQ (*Automatic Repeat reQuest*) – Solicitação de Repetição Automática



Figura 14 - Protocolo Simplex para um canal com ruído

A solução para o problema dos erros acima é utilizar uma numeração seqüencial.

6.4. protocolos de janela deslizante

O protocolo de janelas deslizantes é usado para a entrega confiável e ordenada de mensagens. É um protocolo orientado a conexão (primeiro garante que a conexão está ativa, para depois iniciar o envio das mensagens) que garante que todas as mensagens enviadas são entregues aos destinatários integralmente e na ordem correta de envio.

O receptor envia uma mensagem de confirmação de recebimento (ACK) a cada mensagem recebida. Se o transmissor não recebe o ACK de uma mensagem num tempo pré-estabelecido, ele envia a mesma mensagem novamente. O transmissor cria uma espécie de tabela, onde cada posição é uma janela, em que são gravadas todas as mensagens que foram enviadas.

A cada ACK recebido, avança uma posição da tabela para a direita (a janela "desliza"). Por questão de segurança, se o receptor receber uma mensagem com numeração fora do intervalo de numeração das janelas a mesma é descartada e se estiver na numeração, porém fora de ordem a mensagem é armazenada.

Exemplificando, imagine duas estações **A** e **B**. Tem-se que **A** representa a estação transmissora, capaz de transmitir **W** quadros de uma só vez sem receber nenhum tipo de reconhecimento do recebimento destes quadros, e **B** a estação receptora, capaz de receber até **W** quadros de uma só vez sem enviar nenhum reconhecimento. A estação **B** reconhece um quadro enviando um reconhecimento (ACK) com o índice do próximo quadro desejado. Esse reconhecimento indica que a estação **B** está pronta para receber os próximos quadros a partir do índice em questão. A estação **A** mantém uma lista com os índices dos quadros que pode enviar, enquanto a estação **B** mantém uma lista com os índices dos quadros que espera receber. Essas listas podem ser vistas como "janelas de quadros", como ilustrado na figura abaixo:

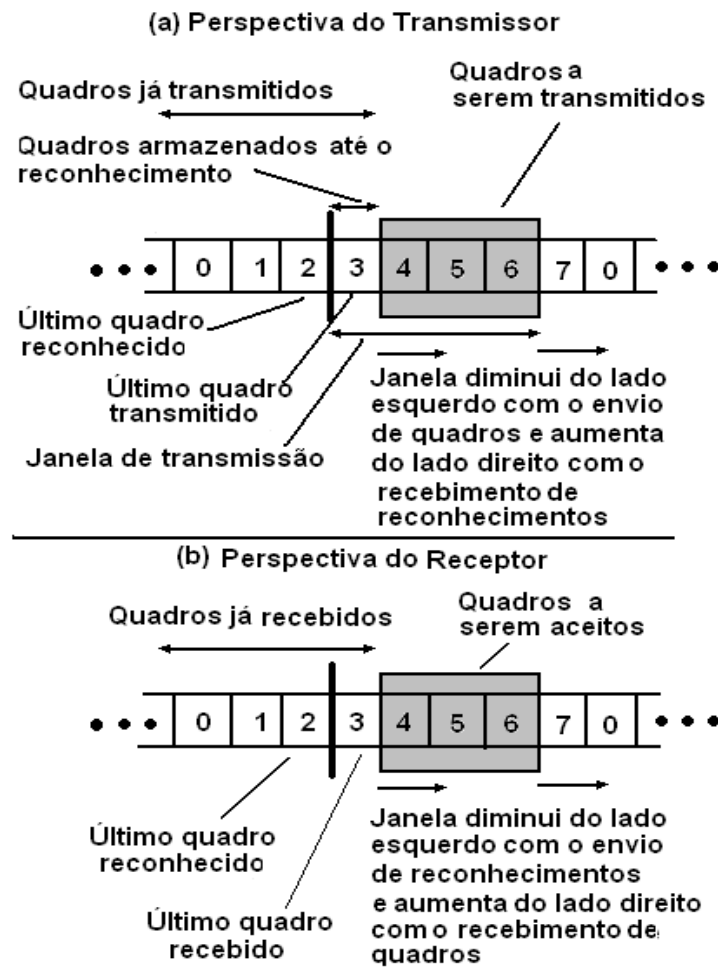


Figura 15 - Protocolo de Janelas Deslizantes

A figura mostra janelas com os índices de 0 a 7 dos referentes quadros, de maneira que os quadros subseqüentes começam a ser referenciados novamente a partir do índice 0. Stallings(2003) em seu livro explica que para um campo de número de seqüência de K-bit, resultando numa escala de número de seqüência de 2^k , o tamanho máximo possível da janela de quadros é 2^{k-1} .

No exemplo, o valor do campo de número de seqüência é 3-bit, resultando em uma escala de número de seqüência de valor 8 (índices de 0 a 7) e o tamanho máximo da janela permitido é 4. A figura 1(a) representa a janela sob a perspectiva da entidade transmissora. Neste caso, o retângulo sombreado indica os quadros a serem enviados. A cada vez que um quadro é enviado, o retângulo sombreado diminui da esquerda para a direita, e quando um reconhecimento é recebido, o retângulo cresce da esquerda para a direita. A barra vertical é utilizada para marcar o início da janela de quadros a serem transmitidos, e se move para a direita à medida que reconhecimentos vão sendo recebidos. Os índices anteriores à barra vertical indicam os quadros já transmitidos e que já receberam reconhecimento. Os índices entre a barra vertical e o retângulo sombreado representam os quadros que foram transmitidos, porém ainda não receberam reconhecimento. Esses quadros ainda não reconhecidos devem ser armazenados, no caso da necessidade de retransmissão dos mesmos.

A Figura 1(b) representa a janela sob a perspectiva da entidade receptora, cuja especificação é correspondente à janela da entidade transmissora, porém de maneira oposta. Por exemplo, enquanto na janela do transmissor a parte sombreada indica os quadros a serem enviados, na janela do receptor isso representa os quadros a serem recebidos, e assim sucessivamente.

Observe que o protocolo de Janelas Deslizantes fornece uma forma de controle de fluxo: a entidade receptora **B**, precisa estar apta a receber apenas **W** quadros de uma só vez, garantindo que não haja sobrecarga na transmissão dos dados. Além disso, existe a vantagem com relação a outros protocolos como o *Stop-And-Wait* que é a maior eficiência na utilização do meio, uma vez que não é necessário aguardar o reconhecimento de cada quadro para que o seu subsequente possa ser enviado.

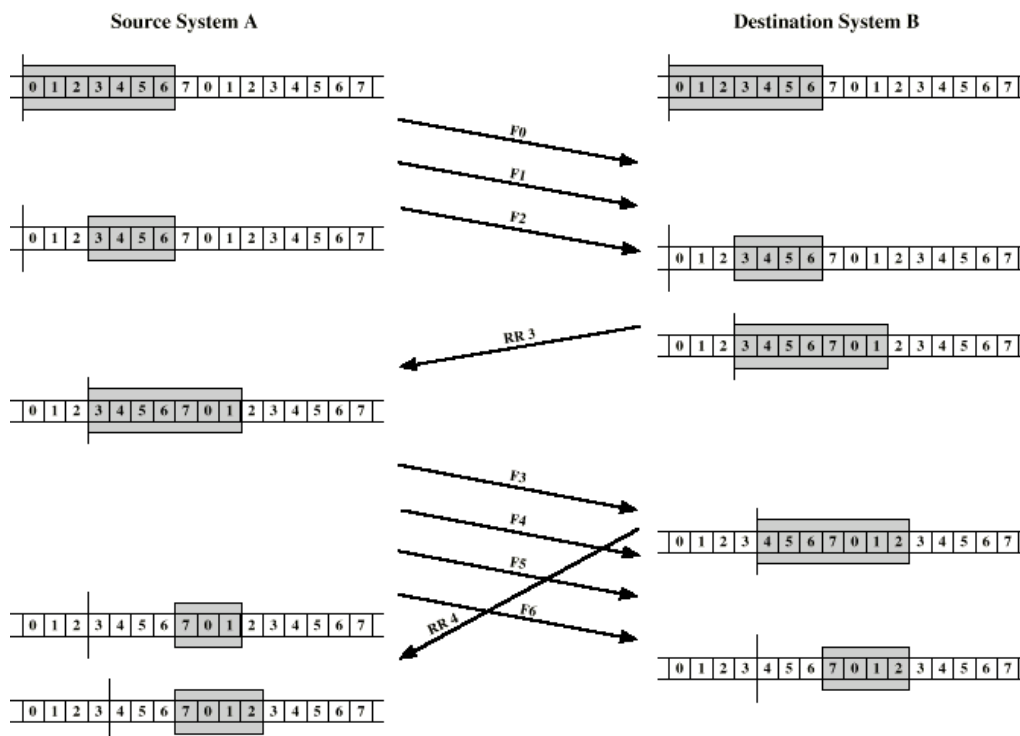


Figura 16 – Transmissão com protocolo de Janelas Deslizantes

7. A CAMADA DE ENLACE DE DADOS NA INTERNET

A Internet consiste em máquinas individuais (hosts e roteadores) e na infra-estrutura de comunicação que as conecta.

Na prática, a comunicação ponto-a-ponto é utilizada principalmente em duas situações. Na primeira, milhares de organizações têm uma LAN ou mais e um roteador ou ponte (*Bridge*). Comumente os roteadores são interconectados por uma LAN de *backbone*. Assim, todas as conexões com o mundo exterior passam por um ou mais roteadores que têm linhas privadas ponto-a-ponto. São esses roteadores e suas linhas que compõem as sub-redes de comunicação que a Internet se baseia. Os computadores domésticos estabelecem uma conexão com um determinado roteador de um provedor de serviços da Internet.

Ambas as conexões, seja ela entre roteadores ou entre computadores domésticos e roteadores, é necessário o uso de um protocolo de enlace de dados ponto-a-ponto na linha para cuidar do enquadramento, do controle de erros e de outras funções da camada de enlace de dados. O único protocolo utilizado na Internet é o PPP (*Point-to-Point Protocol*).

7.1. *Point-to-Point Protocol*

Este protocolo é utilizado para cuidar do tráfego de roteador para roteador e de usuário doméstico para ISP (Provedor de Serviços da Internet). O PPP dispõe de três recursos:

a) Um método de enquadramento que delinea de forma não-ambígua o fim de um quadro e o início do quadro seguinte. O formato do quadro também lida com a detecção de erros.

b) Um protocolo de controle de enlace usado para ativar linhas, testá-las, negociar opções e desativá-la novamente quando não forem mais necessárias. Esse protocolo é denominado LCP (*Link Control Protocol*) – Protocolo de controle de enlace. Ele admite circuitos síncronos e assíncronos, e ainda codificações orientadas a bytes e a bits.

c) Uma maneira de negociar as opções da camada de rede de modo independente do protocolo da camada de rede a ser utilizado. O método escolhido deve ter um NCP (*Network Control Protocol*) – Protocolo de Controle de Rede diferente para cada camada de rede aceita.

Considerando uma conexão entre roteadores ou computadores domésticos e ISP, uma série de pacotes e parâmetros são trocados. Uma série de pacotes LCP e NCP são enviados para configurar a camada de rede. O NCP é o responsável por atribuir o endereço IP, passando o PC a ser um host da Internet podendo enviar e receber pacotes IP. Quando o usuário termina, o NCP é utilizado para desativar a conexão e liberar o IP. O LCP encerra a conexão.

Em suma, o PPP é um mecanismo de enquadramento multiprotocolo, adequado para utilizações em modems e em linhas seriais. Ele aceita a detecção de erros, a negociação de opções, a compactação de cabeçalhos e, opcionalmente, a transmissão confiável com o uso de quadros de tipos especiais.

REFERÊNCIAS:

TANENBAUM, A. S. **Redes de computadores**. Rio de Janeiro: Campus, 2003.

Stallings, W. **Computer Networking with Internet Protocols and Technology**, 2003.