

# Comparativo de protocolos na Camada de Enlace

Gabriel Martins de Miranda e Guilherme Neves

**Abstract**—Análise de eficiência entre os protocolos ARQ da camada de enlace.

**Index Terms**—Stop-and-Wait ARQ, Go-Back N ARQ, Selective Repeat ARQ

## I. INTRODUÇÃO

A camada de enlace é uma das sete camadas do modelo OSI, e também a quarta do modelo TCP/IP. Ela detecta e, opcionalmente, corrige erros que possam acontecer na camada física. É responsável por transmitir e receber quadros e também pelo controle de fluxo, sendo que opera em velocidades próximas à da luz. Na rede ethernet, cada placa de rede possui um endereço físico, que deve ser único na rede, chamado MAC.

O serviço básico da camada de enlace é mover um datagrama de camada de rede de um nó (roteador ou hospedeiro) até um nó adjacente. Todos os seus protocolos operam encapsulando um datagrama dentro de um quadro antes de transmiti-lo. Diferentes protocolos oferecem diferentes serviços de acesso ao enlace, de entrega (confiabilidade, detecção/correção de erros), de controle de fluxo e de transmissão (full-duplex ou half-duplex). Tais diferenças são devidas à vasta variedade de tipos de enlaces sobre os quais os protocolos de enlace devem operar. Um enlace ponto a ponto simples tem um único remetente e um único receptor comunicando-se por um único fio. Um enlace de acesso múltiplo é compartilhado por muitos remetentes e receptores, o que faz com que um protocolo para um canal de acesso múltiplo tenha um protocolo para coordenar o acesso ao enlace. Nos casos de ATM e MPLS, o enlace que conecta dois nós adjacentes pode, na realidade, constituir uma rede em si e por si próprio.

Dentre os princípios subjacentes à comunicação por enlace, tem-se técnicas de detecção e correção de erros, protocolos de acesso múltiplo, endereçamento de camada de enlace e a construção de redes locais ampliadas por hubs e comutadores. Para a detecção/correção de erros, é possível a adição de bits ao cabeçalho de um quadro para detectar e algumas vezes corrigir erros de mudança de bits que podem ocorrer quando o quadro é transmitido, seja por esquemas simples de paridade ou de soma de verificação, seja por esquemas mais robustos de verificação de redundância cíclica. Já em se tratando de protocolos de acesso múltiplo, três abordagens são amplamente utilizadas para coordenar o acesso a um canal broadcast, sendo elas por divisão de canal (TDM, FDM, CDMA), por acesso aleatório (protocolos ALOHA e CSMA) e através de revezamento (polling e passagem de permissão). Alguns de seus protocolos são o Ethernet e o PPP. No caso de protocolos de detecção e correção de erros da camada de enlace, temos os métodos de pedido automático de repetição (ARQ), que serão explicitados nas seções seguintes.

## II. FUNDAMENTAÇÃO TEÓRICA

### • Confiabilidade

Uma boa definição para entrega confiável é quando um dado é aceito no fim de um enlace na mesma ordem transmitida no outro fim, sem perda e sem duplicações. Isto implica em quatro restrições: sem perda (pelo menos uma cópia de cada PDU é mandado), sem duplicações (não mais de uma cópia de um PDU é enviado), entrega FIFO (PDUs entregues na ordem original) e o PDU deve ser entregue em um período confiável. Para suportar confiabilidade, o protocolo implementa um procedimento de recuperação de erros. Esta confiabilidade deve ser provida em várias camadas do modelo OSI, alguns exemplos de protocolos são: Camada de enlace (HDLC), transporte (TCP) e aplicação (TFTP). Alguns dos maiores problemas encontrados pelos protocolos são: corrupção de bits dentro do canal, falhas de temporização resultando em decodificações errôneas nos bits recebidos, erros de software na implementação dos protocolos e tamanho de buffer insuficiente nos equipamentos.

### • Stop-and-Wait-ARQ

Técnica mais simples de confiabilidade. Transmite um Protocol Data Unit (PDU) de informação e então espera por uma resposta. O receptor intercepta o PDU e então envia um Acknowledgement (ACK) PDU se o dado é recebido corretamente, ou um Negative Acknowledgement (NACK) se o dado não foi recebido. Na prática, o receptor pode não estar apto a identificar se um PDU foi recebido, e o transmissor terá de implementar um temporizador para lidar com a condição de o receptor não responder. Sob transmissão comum, o emissor receberá um ACK para o dado e então começa a transmissão do próximo bloco de dados. Por um longo enlace, o emissor pode ter de esperar um tempo apreciável pela resposta. Enquanto espera, o temporizador é dito em estado "ocioso" e é incapaz de enviar dados.

Baseia-se em dois caminhos de transmissão (full duplex ou half duplex). Um pequeno delay de processamento deve ser introduzido entre a recepção do último byte PDU e a geração do ACK correspondente. Quando um PDU é perdido, o receptor não irá normalmente ser capaz de identificar a perda, o transmissor então deve ser confiável para detectar a falta de resposta.

### • Go-Back-N-ARQ

Detecta e retransmite I-frames que foram corrompidos devido à erros na camada física. Para suportá-lo, o protocolo deve numerar cada PDU que é enviado, geralmente feito usando a aritmética de módulo. O nó local deve manter um buffer de todos os PDUs que foram mandados e não receberam ACK ainda, enquanto o receptor deve guardar um registro do maior número de PDU que foi corretamente recebido. Para realizar a recuperação de frames perdidos, primeiro há o descarte dos

PDU no nó remoto do receptor. Em seguida este nó solicita a retransmissão do PDU faltante usando o controle de NACK ou REJECT. O receptor descarta todos os PDUs que não tem o número do PDU requerido. Finalmente ocorre a retransmissão do quadro perdido. O controle indica o último PDU recebido com sucesso. Isto permite instruir o emissor de onde deve começar a retransmissão de PDUs. Todos os PDUs fora de sequência são descartados.

Após o recebimento de um PDU de controle pelo nó local, o transmissor volta a janela de volta onde a unidade pendente que recebeu no buffer de NACKs. Este então retransmite seguido de todos os PDUs sucessivos a este. Se a retransmissão falhar, o PDU perdido pode ser enviado através da técnica de pooling.

- Selective-Repeat-ARQ

É o mais complexo dos que provêm recuperação de erros, entretanto o mais eficiente. É o empregado pelo protocolo TCP da camada de transporte. Assim como no Go Back N, para suportá-lo o protocolo deve numerar cada PDU que é mandado, normalmente com módulo. O nó local deve também manter um buffer de todos os PDUs que foram mandados mas que não foram reconhecidos. O receptor no nó remoto mantém um registro do PDU de maior valor que foi corretamente recebido. Este número corresponde ao último ACK que enviou. Ao contrário do Go Back N, o receptor também deve manter um buffer de quadros que foram recebidos, mas não reconhecidos.

A recuperação de um PDU corrompido ocorre em quatro etapas: primeiramente, o PDU é descartado no nó remoto do receptor. Após isto, o nó remoto requisita retransmissão do PDU faltante usando o controle de PDU (chamado Selective Reject). O receptor então armazena todos os PDUs fora de sequência em seu buffer até que o faltante seja retransmitido. O emissor recebe o pedido de retransmissão e então transmite o quadro perdido. Finalmente, o receptor recebe o faltante e todos os subsequentes são tomados em ordem do seu buffer. A retransmissão feita pelo nó remoto de PDUs corrompidos é feita através de um controle indicando o quadro faltante. Após o transmissor receber o controle, manda um único PDU do buffer de PDUs sem ACK. Daqui, o transmissor continua sua transmissão normalmente até novo pedido de controle. No Selective Repeat, PDUs perdidos também são recuperados através de pooling.

### III. IMPLEMENTAÇÃO TEÓRICA

O sistema foi modelado na linguagem C, utilizando o sistema operacional LINUX. Para emular a emissão e recepção de pacotes, tanto quanto os vários casos de erros envolvidos, funções foram usadas com estes papéis. Para todos os protocolos implementados, foi modelada as funções *Emissor*, *Receptor* e *Canal*. Todos os protocolos encaminham os bits utilizando um vetor como pacote de dados, sendo que a cada iteração este é enviado como argumento para a função receptora. Dentro do emissor temos os loops referentes ao tipo de protocolo utilizado. Para o stop and wait, foram utilizadas estruturas de repetição até que um ack fosse recebido. No go back n, uma janela deslizante de quatro pacotes de tolerância, sendo que em caso de perdas retransmite as últimas informações.

Por último, o selective repeat guarda os próximos pacotes que deve-se enviar e espera enquanto não enviar o número pré-definido.

- Stop-and-Wait-ARQ

Para o cálculo da eficiência do Stop-and-Wait foi utilizada a fórmula representada pela Figure 1, sendo  $n_f$  o número total de bits no frame,  $n_o$  o número de bits no header e no CRC,  $t_0$  o tempo total para enviar a informação,  $P_f$  a probabilidade de ocorrer um erro de transmissão e  $t_0/(1 - P_f)$  o tempo médio total gasto por frame.

$$\eta_{SW} = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_o}{t_0(1 - P_f)}}{\frac{1 - \frac{n_o}{n_f}}{1 + \frac{n_o}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}}(1 - P_f)}$$

Effect of frame loss

Fig. 1. Eficiência do Stop-and-Wait-ARQ.

- Go-Back-N-ARQ

Para o caso de eficiência no Go-Back-N-ARQ foi utilizada a fórmula representada pela Figure 2, sendo o tempo para mandar um frame  $t_f$  caso o primeiro frame da transmissão seja enviado corretamente ou  $t_f + W_s * t_f / (1 - P_f)$  caso contrário.

$$t_{GBN} = t_f(1 - P_f) + P_f \left\{ t_f + \frac{W_s t_f}{1 - P_f} \right\} = t_f + P_f \frac{W_s t_f}{1 - P_f} \text{ and}$$

$$\eta_{GBN} = \frac{n_f - n_o}{R} = \frac{1 - \frac{n_o}{n_f}}{1 + (W_s - 1)P_f}(1 - P_f)$$

Delay-bandwidth product determines  $W_s$

Fig. 2. Eficiência do Go-Back-N-ARQ.

- Selective-Repeat-ARQ

Para a eficiência do Selective-Repeat-ARQ foi utilizada a fórmula representada pela Figure 3.

$$\eta_{SR} = \frac{n_f - n_o}{R} = \frac{t_f / (1 - P_f)}{n_f} = (1 - \frac{n_o}{n_f})(1 - P_f)$$

Fig. 3. Eficiência do Selective-Repeat-ARQ.

Um sumário dos cálculos das eficiências dos três protocolos pode ser visto na Figure 4.

### IV. ANÁLISE

Para velocidades mais baixas, o stop and wait mostrou-se mais eficaz devido à característica de esperar o ack do receptor, em contrapartida, o tempo ocioso para velocidades altas mostrou-se mais alto. O go back n mostrou uma taxa de erros menor, uma vez que a cada erro são reenviados quatro vezes mais quadros que nos outros protocolos. Já o selective

Selective-Repeat:

$$\eta_{SR} = (1 - P_f) \left(1 - \frac{n_a}{n_f}\right) \approx (1 - P_f)$$

Go-Back-N:

$$\eta_{GBN} = \frac{1 - P_f}{1 + (W_s - 1)P_f} = \frac{1 - P_f}{1 + LP_f}$$

Stop-and-Wait:

$$\eta_{SW} = \frac{(1 - P_f)}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}} \approx \frac{1 - P_f}{1 + L}$$

Frame loss probability  
For  $P_f \neq 0$ , SR & GBN same  
For  $P_f \rightarrow 1$ , GBN & SW same

Fig. 4. Comparação entre as eficiências dos protocolos ARQ.

repeat mostrou-se mais eficiente em situações onde a taxa de erros é média ou alta devido ao fato de não reenviar pacotes já recebidos.

## V. CONCLUSÕES

O presente trabalho foi de grande ajuda para entender o funcionamento dos protocolos ARQ da camada de enlace. Houve uma maior dificuldade em representar o caráter paralelo dos protocolos, em que emissor e receptor trabalham de forma simultânea.

[0] Wikipedia

[1] Redes de computadores e a Internet - Kurose - Ross

[2] ARQ Protocols - Prof. Anish Goel (MIT)