

METODOLOGIA PARA IMPLEMENTAÇÃO OTIMIZADA DE PROTOCOLOS DE COMUNICAÇÃO

FRANCO, Álvaro Junio Pereira¹; MIRANDA, Marcio Nunes de²

Palavras-chave: Protocolos de Comunicação, Redes de Petri, Algoritmo Genético.

1. INTRODUÇÃO

As aplicações multimídias demandam cada vez mais rapidez no tráfego de seus dados visto que a qualidade do resultado da execução dessas aplicações depende diretamente do tempo de processamento, da transmissão e da propagação dos pacotes.

Para tentar diminuir o atraso incluído pelos tempos citados, utiliza-se o conhecido protocolo da camada de transporte User Datagram Protocol (UDP). A justificativa para o uso do UDP é que este protocolo não estabelece inicialmente uma conexão, não mantém nenhum tipo de estado da conexão, não se preocupa em controlar o fluxo dos dados e nem o congestionamento da rede. Além disso, utiliza-se de toda a capacidade disponível do enlace para enviar seus pacotes. A grande desvantagem do UDP é que ele não garante a entrega dos pacotes, que podem ser perdidos por várias formas: descartados pelo destinatário (pacote corrompido), descartados por roteadores que estão com buffers cheios, etc. O protocolo Lightweight Streaming Protocol (LSP) [1] tenta resolver estes problemas do UDP e este trabalho aplica a metodologia apresentada por Miranda [2] no LSP.

2. METODOLOGIA

2.1 – Escolha do protocolo: LSP

O LSP foi escolhido por ser um protocolo da camada de aplicação para fluxos multimídia dito semi-confiável que implementa certos conceitos encontrados em protocolos orientados a conexão. Esses conceitos são: retransmissão seletiva usando NAK e controle de congestionamento via *frame dropping* seletivo.

A retransmissão seletiva usando NAK força que os pacotes do protocolo LSP sejam maiores, incluindo dois campos: número de seqüência e tempo de entrega dos dados para a mídia decodificadora. O número de seqüência é utilizado para identificar perdas de pacotes e o tempo de entrega dos dados é utilizado para calcular a estimativa de tempo para a entrega (ETPE) dos pacotes para a mídia. O fluxo normal de pacotes é sempre do remetente para o destinatário. Quando o destinatário identifica uma perda de um pacote, através da recepção de um pacote com um número de seqüência não esperado, este poderá enviar ao remetente um NAK. Isso dependerá da seguinte condição:

$$ETPE > ERTT + 4 * Des_Pad(ERTT) + TAN,$$

onde ERTT é a Estimativa do Round Trip Time feita dinamicamente durante o tráfego de dados, TAN é o atraso incluído na fase de identificação do GAP e processamento do NAK, chamado de Tempo de Atraso do Nak e $4 * Des_Pad(ERTT)$ é uma margem de segurança adotada pelo protocolo como sendo quatro vezes o Desvio Padrão do ERTT. Essa mesma margem é utilizada também no TCP.

O controle de congestionamento via *frame dropping* seletivo é tratado excluindo alguns quadros de dados do fluxo multimídia. No caso da transmissão de vídeo usando o formato MPEG, nem todos os quadros são igualmente importantes. Por exemplo, a exclusão de quadros do tipo-B não irá usualmente produzir uma degradação significativa da qualidade do vídeo [1]. A exclusão destes quadros é feita para diminuir a taxa de envio de dados. É claro que quanto menor essa taxa, maior será o impacto na qualidade do vídeo. Deve-se encontrar um ponto ideal para que ambas as partes (rede e destinatário) fiquem satisfeitas.

Na seção de resultados, o protocolo LSP é especificado por uma Rede de Petri (RP). O primeiro passo para a implementação de um protocolo é justamente esta especificação.

2.2 – Fases do processo de otimização

A primeira fase da metodologia é destinada à especificação do protocolo. Esta especificação será feita em RP. A validade desta rede será testada pela ferramenta: Analisador / Simulador de Redes de Petri (ARP) [3]. A visualização desta especificação será feita pela ferramenta: JARP [5]. A ferramenta TANGRAM-II [9] trabalha juntamente com o algoritmo genético fazendo a otimização do protocolo. Uma explicação mais completa desta metodologia poderá ser encontrada em Miranda [2].

3. RESULTADOS E DISCUSSÃO

3.1 – Especificação do protocolo LSP em redes de Petri

Para especificar um protocolo em RP, deve-se inicialmente identificar os lugares, as transições, as relações entre estes e a marcação inicial [4]. A especificação visual do protocolo LSP em RP é mostrada a seguir.

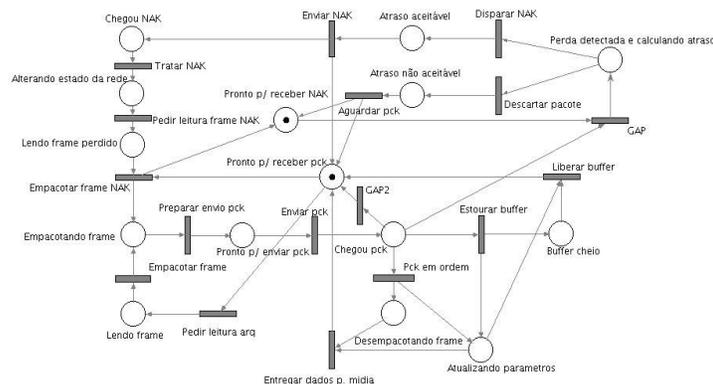


Figura 1: Especificação do protocolo LSP em redes de Petri

Os lugares identificados no LSP estão descritos na figura através dos círculos e as transições através das barras. Os relacionamentos são os arcos e a marcação inicial é dada pelas fichas (tokens) existentes nos lugares.

A especificação anterior foi submetida à ferramenta ARP [3] que têm, como um de seus objetivos, analisar a RP e obter suas propriedades. Algumas dessas propriedades são desejáveis, tais como: a rede é livre de bloqueios, ou seja, não existe uma marcação (ou estado) que bloqueie sua execução; a rede é livre de *live-locks*, ou seja, a execução do protocolo não entra em *loop* em um local fixo da RP; a rede é reiniciável (pode sempre voltar à marcação inicial); e a rede é limitada, ou seja, o número de fichas não cresce indefinidamente. A marcação da rede, ou seja, a posição das fichas dentro dos lugares, define o **estado do protocolo**.

3.2 – Modelagem no TANGRAM-II

A RP utilizada para modelar o protocolo LSP é estocástica, ou seja, o tempo de disparo de cada transição, uma vez que a mesma esteja habilitada a disparar, tem uma distribuição exponencial. Molloy [7] mostra que uma RP estocástica é isomorfa com uma cadeia de Markov a tempo contínuo (CMTC) [8]. Nesta fase é feito uma modelagem do protocolo em RP para o TANGRAM-II. Esta modelagem é feita da seguinte forma: para cada lugar da RP é definida uma variável de estado na ferramenta e para cada transição da RP é definido um evento na ferramenta. As condições para os eventos serem disparados são dadas pela posição das fichas na RP.

3.3 – Desempenho desejado

O parâmetro de desempenho escolhido para se otimizar o protocolo LSP foi a probabilidade de perda de pacotes (PLP) em função do número de reconhecimentos negativos enviados pelo destinatário (NNACK). A curva de desempenho desejada será dada pela PLP em função do NNACK. Foi utilizada a expressão $DPLP > PLP^{NNACK}$, onde DPLP significa a probabilidade de perda desejada fixada em 0,01 (1%). Ela foi especificada considerando-se que, para uma

aplicação de áudio, é uma perda aceitável. A expressão citada anteriormente foi extraída de [1] e o gráfico associado a ela é mostrado a seguir.

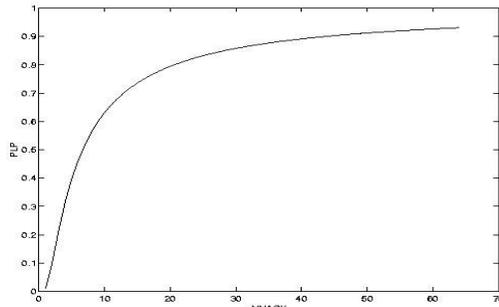


Figura 2: PLP em função do NNACK

Deseja-se encontrar as taxas descritas anteriormente que mais se aproximam desta curva. As transições que tiverem maior vazão, ou seja, maior $\lambda * \pi$ [2], serão aquelas escolhidas para serem implementadas em hardware enquanto que as de menor vazão serão implementadas em software.

4. CONCLUSÃO

Para a aplicação da metodologia descrita neste trabalho (criada por Miranda [2]), é necessário que várias ferramentas trabalhem em conjunto. A ferramenta ARP [3] foi utilizada para identificar as propriedades das RP modeladas, a ferramenta JARP [5] é um auxiliar para a ARP utilizada principalmente para especificar visualmente as RP modeladas. A ferramenta TANGRAM-II [9] foi utilizada para se modelar a especificação em RP. A partir desta modelagem pode ser realizada a otimização do protocolo através de um algoritmo genético. O resultado será uma forte indicativa de quais operações do protocolo deverão ser implementadas em hardware e quais operações deverão ser implementadas em software.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] MULABEGOVIC, E., SCHONFELD, D., ANSARI, A., *Lightweight Streaming Protocol (LSP)*, in *MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia*. New York, NY, USA: ACM Press, 2002, pp. 227-230.
 - [2] MIRANDA, M. N., *Uma metodologia hw/sw codesign de protocolos de comunicação baseada na otimização de desempenho por algoritmos genéticos*, Ph.D. dissertation, Universidade Federal do Rio de Janeiro – UFRJ, Rio de Janeiro, RJ, Brasil, 2002.
 - [3] ARP – *Analizador/Simulador de Redes de Petri* – LCMI – Departamento de Engenharia Elétrica da Universidade Federal de Santa Catarina, Internet Draft – <http://www.ppgia.pucpr.br/~maziero/diversos/petri/manual-pt.html>.
 - [4] PETERSON, J. L., *Petri Nets*, Computing Surveys, Vol. 9 No. 3, 1977.
 - [5] JARP – *Auxiliar para a ferramenta ARP*, Internet Draft – <http://jarp.sourceforge.net/br/index.html>.
 - [6] SILVA, E. S., LEÃO, R. M. M., *The Tangram-II environment*, in *TOOLS '00: Proceedings of the 11th International Conference on Computer Performance Evolution: Modelling Techniques and Tools*. London, UK: Springer-Verlag, 2000, pp. 366-369.
 - [7] MOLLOY, M. K., *Performance analysis using stochastic petri nets*. *IEEE Trans. Computers*, vol. 31, no. 9, pp. 913-917, 1982.
 - [8] ROSS, S. M., *Stochastic Processes*. Wiley, 1983.
 - [9] SILVA, E. S., LEÃO, R. M. M., *The Tangram-II environment*, in *TOOLS '00: Proceedings of the 11th International Conference on Computer Performance Evolution: Modelling Techniques and Tools*. London, UK: Springer-Verlag, 2000, pp. 366-369.
- FONTE DE FINANCIAMENTO – CNPq/PIBIC - UFG

¹ Bolsista de iniciação científica. Instituto de Informática – GEAPIS – Grupo de Estudos Aplicados à Internet e a Sistemas Distribuídos, alvaro@inf.ufg.br

² Orientador/Instituto de Informática/UFG, miranda@inf.ufg.br