REDES DIGITAIS I - parte I do projecto

Américo Tomás (54149), Ianick Insaly(60585), Eduardo Gonçalves(60596)

Março 2014

Contents

Chapter 1

Introdução

Sem deteção e correção de erros teríamos sérios problemas de integração e utilização das redes. A quem nunca aconteceu ficar meia dúzia de segundos à espera que uma página na web abra e Nada. Sem abordarmos estes assuntos seriamente, isso aconteceria a todo o instante.

Neste projeto vamos abordar deteção e correção de erros. Sendo importante distinguir claramente a diferença entre ambos. Existem algoritmos que detetam, mas não corrigem. Outros detetam e corrigem, mas poucos erros e assim sucessivamente. Cada um tem vantagens e desvantagens. Sendo a sua utilização muito variante do tipo de utilização que está a ser dado à rede.

Também vamos abordar ritmo binário, bem como valores médios de repetição de envio de PDU's (no nível 2: tramas).

Naturalmente o ritmo a que conseguimos aceder a informação, enviando e recebendo a mesma é vital. Saber os atuais limites para tirar o máximo proveito das redes já existentes. Ao mesmo tempo que é possível procurar e planear de forma a criar infra-estruturas, que juntamente com a investigação que está a decorrer a nível mundial, irá permitir dar aos atuais clientes, novas velocidades, maior estabilidades, sempre tendo em consideração o consumo energético.

Neste projeto iremos procurar perceber o que é uma rede, bem como lidar com alguns dos desafios inerentes ao seu funcionamento. Desde a criação de expressões gerais para abordarmos estes problemas, testando vários valores. Como posteriormente fazendo a implementação de da mesma.

Este parte do projeto foi feita no editor de texto LateX. Muito bom para escrita de linguagem matemática, de forma a provar os assuntos que vamos abordando. Existem anexos (A, B e C), feitos tanto em Excel, como cópias de ecrã para poder analisar os resultados das experiências feitas.

Com este Trabalho pretende-se estudar algumas das funcionalidades que estão presentes nos níveis de ligações de dados no modelo de OSI do ISO, nomeadamente: O ritmo binário máximo que um cliente pode colocar na interface com o nível ligação de dados, O valor médio para o atraso sofrido por uma mensagem que é colocada na interface do nível ligação de dados e a probabilidade de ser entregue ao cliente destino uma mensagem com erros. Nesta parte do trabalho iremos apresentar resultados analíticos, recorrendo a um conjunto de expressões e aproximações teóricas.

`redes_introducao.jpg`

Figure 1.1: Legenda

Chapter 2

Detecção de erros

2.1 Criar o CRC_8my do nosso grupo

Após utilizar o CRC-8my.java, obtivemos para o nosso grupo o seguinte número Binário:

- Padrão binário gerado: 111100011
- Polinómio: $1 * x^8 + 1 * x^7 + 1 * x^6 + 1 * x^5 + 0 * x^4 + 0 * x^3 + 0 * x^2 + 1 * x^1 + 1 * x^0$
- Eliminando os elementos a zero: $(x) = x^8 + x^7 + x^6 + x^5 + x^1 + 1$

2.2 Expressão da taxa de redundância

A taxa de redundância do código para os seguintes algoritmos está apresentado em anexo (A), numa folha de Excel.

2.2.1 cálculo do n:

Para obtermos a Taxa de redundância, precisamos primeiro de calcular o r (bit de redundância) para cada um dois algoritmos.

- Bit de paridade: r=1;
- Hamming: r = $\log_2 m + 1$;
- CRC: r = depende (polinómio de maior grau)

$\rightarrow Sendo : \boldsymbol{n = m + r}$

2.2.2 cálculo da taxa:

$$TaxaDeRedundância = \frac{r}{m + r} \tag{2.1}$$

2.3 Probabilidade de estar ok

Para obter a probalilidade de uma trama ser enviada sem erros para os 3 tipos de algoritmos, colocamos em anexo (B) a respectiva informação, numa folha de excel.

2.3.1 cálculo de probabilidade de estar ok

$$P_{ok} = (1 - P_{eb})^n \tag{2.2}$$

• tendo o n sido calculado anteriormente, variando para cada um dos algoritmos consoante o valores de r dos mesmos.

2.4 Bit de paridade: expressão exata

• Bit de paridade (d=2), (**expressão exata**):
 - numerador: Os somatórios dos pares.
 - denominador: Probabilidade condicionada

$$P_{nd|e}(n, Peb) = \frac{\sum_{(i=par)}^{n} C_i^n * P_{eb}^i * (1 - P_{eb})^{n-i}}{1 - (1 - P_{eb})^n} \tag{2.3}$$

• Na expressão exata, o raciocínio usado foi o de calcular todos os valores possíveis, que o algoritmo Par assume que estão certos, mas que podem não estar. Sendo o operador somatório utilizado para representar todos os valores possíveis até $'n'$.

2.5 Bit de paridade: expressão aproximada

• Bit de paridade (d=2), (**expressão aproximada**):

$P_{nd|e}^{dh}(n, Peb) \approx \frac{1-(P_0+P_1)}{1-(1-P_{eb})^n}$

$$P_{nd|e}^{dh}(n, Peb) \approx \frac{1 - [(1 - P_{eb})^n] - [C_1^n * P_{eb}^1 * (1 - P_{eb})^{n-1}]}{1 - (1 - P_{eb})^n} \tag{2.4}$$

• Na expressão aproximada, o raciocínio utilizado foi uma espécie de 'negação'. Ou seja, queremos todos os valores (daí termos o valor $'1'$), subtraindo aqueles que estão certos de certeza (que são os valores da P_0 e da P_1). Estamos a usar uma aproximação por Excesso (são praticamente sempre por excesso), pois consideramos todos os casos $'1'$, quando no algoritmo Par apenas nos interessaria os pares. Nesta situação, o operador somatório desapareceu, sendo simplesmente substituído pelo valor $'1'$

2.6 Código de hamming: expressão aproximada

• Código Hamming (d=3), (**expressão aproximada**):

$P_{nd|e}^{dh}(n, Peb) \approx \frac{1-(P_0+P_1+P_2)}{1-(1-P_{eb})^n}$

$$P_{nd|e}^{dh}(n, Peb) \approx \frac{1 - [(1 - P_{eb})^n] - [C_1^n * P_{eb}^1 * (1 - P_{eb})^{n-1}] - [C_2^n * P_{eb}^2 * (1 - P_{eb})^{n-2}]}{1 - (1 - P_{eb})^n} \tag{2.5}$$

• Na expressão aproximada mas para o Código Hamming, o raciocínio usado é semelhante ao do Bit paridade, com a exceção de o $d = 3$, sendo d a distância de Hamming. Por esse motivo quero todos os valores ($'1'$), exceto o P_0, P_1 e P_2.

2.7 Código de hamming

- $G_{4_{ITU}}(x) = G_{19}(x) = x^4 + x + 1;$
- $G_{8_{CCITT}}(x) = G_{263}(x) = x^8 + x^2 + x + 1;$
- $G_{8_{my}}(x) = G_{483}(x) = x^8 + x^7 + x^6 + x^5 + x^1 + 1;$

(para valores exatos): $P_{nd} = \frac{\sum_{(i)}^{n} W_i * P_{eb}^i * (1-P_{eb})^{n-i}}{1-(1-P_{eb})^n}$ (2.6)

(para valores aprox.): $P_{nd} \approx \frac{\sum_{(i)}^{10} W_i * P_{eb}^i * (1-P_{eb})^{n-i}}{1-(1-P_{eb})^n}$ (2.7)

- A diferença entre os valores exatos e aproximados está num pedido feito pelo enunciado, para restringirmos a pesquisa a 10 elementos; $W = weight = pesos$;

- $\text{Ham}_{7,4} \rightarrow \text{W} = 3;$
- $\text{Ham}_{31,26} \rightarrow \text{W} = 3;$

Logo, a expressão geral aproximada (Hamming: d é sempre igual a 3). Assim, usamos a equação 2.5 para o cálculo de Hamming.

2.8 cógigo CRC

- G_{4-ITU},m=4$\rightarrow$W = erro;
- $\text{G}_{8-CCITT}$,m=4$\rightarrow$W = 4;
- G_{8-my},m=4$\rightarrow$W = 4;

- G_{4-ITU},m=26$\rightarrow$W = 2;
- $\text{G}_{8-CCITT}$,m=26$\rightarrow$W = 4;
- G_{8-my},m=26$\rightarrow$W = 4;

- A expressão geral para os $W = 4$ é:

$$P_{nd}^{dh}(n, Peb) \approx \frac{1-[(1-P_{eb})^n]-[C_1^n * P_{eb}^1 * (1-P_{eb})^{n-1}]-[C_2^n * P_{eb}^2 * (1-P_{eb})^{n-2}]-[C_3^n * P_{eb}^3 * (1-P_{eb})^{n-3}]-[C_4^n * P_{eb}^4 * (1-P_{eb})^{n-4}]}{1-(1-P_{eb})^n}$$

Chapter 3

Correção de Erros

3.1 Método de correção de erros FEC

3.1.1 Ham(7,4)

- $= (1 - P_0) + (1 - P_1)$

$$= (1 - P_{eb})^7 + +C_1^7 * P_{eb} * (1 - P_{eb})^6] \tag{3.1}$$

3.1.2 Ham(31,26)

$$= (1 - P_{eb})^{31} + C_1^{31} * P_{eb} * (1 - P_{eb})^{30}] \tag{3.2}$$

3.2

3.2.1 Expressão de tempo de transferência

- O tempo de atraso vai ser igual à soma do Tempo de propagação mais o Tempo de transferência da trama.

$$T_{atraso} = T_{propagação} + T_{transferência} \tag{3.3}$$

- Assumimos que não existe tempo de atraso entre o emissor e o recetor. Se existir um T_{delay}, só precisamos de somar mais uma parcela.

3.3 Método de correção de erros ARQ: stopAndWait

3.3.1 expressão média do tempo de atraso (código CRC)

- Usando o código CRC, queremos a expressão média do tempo de atraso:

$\sum_{i=1}^{+\infty} n * p^n = \frac{p}{(1-p)^2}$

$E[D] = N_{re-transmissões} + T_{transferência}$

sendo: $N_{re-transmissões} = N_{transmissões}$

então: $N_{transmissões} = \frac{1}{P_{sucesso}} + T_{rtt} = 2 * T_{atraso}$

$$N_{transmissões} = \frac{1}{(1 - P_{ok}) + (1 - P_{nd)})} + (2 * T_{atraso}) \tag{3.4}$$

Chapter 4

Ritmo binário útil

4.1 Método de correção de erros ARQ: stopAndWait

4.1.1 taxa de utilização e ritmo binário útil

$$r_{binário} = \frac{L}{T_{tx}} \tag{4.1}$$

- $U = \frac{r_{binário}}{E[D]}$

$$U = \frac{\frac{L}{T_{tx}}}{N_{re-transmissões} + T_{transferência}} \tag{4.2}$$

Chapter 5

Geração de tráfego e filas de espera

5.1 Expressão geral

Expressão geral para calcular valor médio na fila em função de λ (demonstração):

- É-nos dado no enunciado que: $E[w] = \frac{\lambda*(E[D^2])}{2*(1-\rho)}$

- Também sabemos que: $N_T = \frac{1}{P_{sucesso}} = \frac{1}{(1-P_{eb})}$

- Por sua vez ($D = delay$):

$D = T_{rtt} * N_T = T_{rtt} * \frac{1}{(1-P{:}eb)}$

- Calculando primeiro ($E[D]$):

sendo:

$D_i = T_{rtt} * i;$

$P_i = P_{eb}^{i-1} * (1 - P_{eb})$

$\sum_{i=1}^{+\infty} i * P_{eb}^{i} = \frac{P_{eb}}{(1-P_{eb})^2}$

então:

$E[D] = \sum_{i=1}^{+\infty} D_i * P_i = \sum_{i}^{+\infty} (T_{rtt} * i) * [P_{eb}^{i-1} * (1 - P_{eb})]$

$E[D] = T_{rtt} \sum_{i}^{+\infty} (i) * [P_{eb}^{i} * P_{eb}^{-1} * (1 - P_{eb})]$

$E[D] = T_{rtt} * \frac{1}{(1-P_{eb})}$

- Passando para o cálculo de D^2: $(E[D^2])$

sendo:

$\sum_{i=1}^{+\infty} i^2 * P_i = P_{eb} * (\frac{(1+P_{eb})}{(1-P_{eb})^3)}$

$E[D^2] = \sum_{i=1}^{+\infty} (T_{rtt} * i)^2 * [P_{eb}^{i-1} * (1 - P_{eb})]$

$E[D^2] = T_{rtt}^2 * \sum_{i=1}^{+\infty} (i)^2 * [P_{eb}^{i} * P_{eb}^{-1} * (1 - P_{eb})]$

$E[D^2] = T_{rtt}^2 * \sum_{i=1}^{+\infty} (i)^2 * [P_{eb}^{i} * (1 - P_{eb}) * P_{eb}^{-1} * (\frac{P_{eb}}{P_{eb}})]$

$E[D^2] = T_{rtt}^2 * P_{eb} * \frac{(1+P_{eb})}{(1-P_{eb})^2} * (\frac{1}{P_{eb}})$

$E[D^2] = T_{rtt}^2 * \frac{(1+P_{eb})}{(1-P_{eb})^2}$

- Para terminar, é dado no enunciado que:

$E[w] = \frac{\lambda*(E[D^2])}{2*(1-\rho)}$

$$E[w] = \frac{\lambda * (T_{rtt}^2 * \frac{(1+P_{eb})}{(1-P_{eb})^2})}{2 * (1 - \rho)} \qquad (5.1)$$

Chapter 6

Conclusão

Concluímos esta primeira parte do projeto, ficando com as experiência calculadas e algumas experiências já feitas. Ao longo do mesmo, ficou claro a importância do bit redundante (r). A distância de Hamming, conceito transversal a vários algoritmos. A utilidade de utilizar probabilidades ao abordar Redes digitais. Pois é uma forma de termos uma ideia de onde vamos investir tempo a resolver problemas, nos concentrando naqueles em que é mais provável acontecerem.

Qual o processo de envio, re-envio, de emissor para receptor. E quais os diferentes tempos envolvidos no processo. Terminando esta parte do projecto com o cálculo de uma expressão para valores médios.

Aspetos a concluir são a criação de gráficos em Excel, de forma a visualizar e perceber numa função o comportamento de uma Rede. É uma maneira extremamente visual, que é amplamente usada na área da engenharia para mais facilmente ver onde está um problema. É comum olhar para expressões analíticas. E nem sempre ser capaz de retirar informações delas. Mas os gráficos, bem como regressões lineares, ajudam muito os engenheiros a melhor performance, eficiente e produtividade de sistemas, como é o caso das Redes.

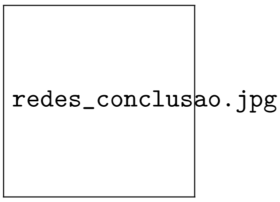

Figure 6.1: Legenda

.