



Pós-Graduação em Ciência da Computação

ANÁLISE DA QUALIDADE DE TRANSMISSÃO DE VIDEO STREAM SOBRE UDP

Por

EDSON KAZUO HIROSE

Dissertação de Mestrado Profissional



Universidade Federal de Pernambuco
posgraduação@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE, JANEIRO/2012



Universidade Federal de Pernambuco

CENTRO DE INFORMÁTICA

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Edson Kazuo Hirose

Análise da Qualidade de Transmissão de Vídeo Stream sobre UDP

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre Profissional em Ciência da Computação.

ORIENTADOR(A): ***Prof. Djamel Fawzi Hadj Sadok – Ph.D.***

RECIFE, JANEIRO/2012

Catálogo na fonte
Bibliotecário Vimário Carvalho da Silva, CRB 4-1204

Hirose. Edson Kazuo.

Qualidade de transmissão de vídeo stream sobre UDP. / Edson
Kazuo Hirose. - Recife: O Autor, 2012.
xxiv, 152 f.: fig. tab. quadro.

Orientador: Djamel Fawzi Hadj Sadok.

Dissertação (Mestrado) - Universidade Federal de Pernambuco, CIN,
Ciência da Computação, 2012.

Inclui bibliografia, glossário, anexo e apêndice.

1. Processamento de dados. 2. UDP. 3. GOP. 4. *Vídeo Stream*. I.
Sadok, Djamel Fawzi Hadj (orientador). II. Título.

004.2 (22. ed.)

MEI 2012-004

Dissertação de Mestrado Profissional apresentada por **Edson Kazuo Hirose** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título, “**Qualidade de Transmissão de Video Stream sobre UDP**”, orientada pelo **Professor Djamel Fawzi Hadj Sadok** e aprovada pela Banca Examinadora formada pelos professores:

Prof. Stênio Flávio de Lacerda Fernandes
Centro de Informática / UFPE

Prof. Juliana Regueira Basto Diniz
Universidade Federal Rural de Pernambuco

Prof. Djamel Fawzi Hadj Sadok
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 29 de setembro de 2011.

Prof. NELSON SOUTO ROSA
Coordenador da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

Aos meus pais Morihiko Hirose (in memorian)
e Yoko Hirose (in memorian)

Agradecimentos

A minha esposa, Joyce pelo amor, carinho, compreensão, paciência, sabedoria e por todos os momentos que estive ao meu lado me dando apoio.

Ao meu orientador Djamel Fawzi Hadj Sadok, por suas palavras de incentivo, por sua sabedoria, por sua dedicação em orientar.

Aos colegas da minha empresa, que com suas idéias, conselhos e sugestões me impulsionaram nesta longa caminhada.

A todos aqueles que diretamente e indiretamente estiveram do meu lado durante todo esse caminho e que hoje comemoram a sua conclusão.

“Um homem que ousa desperdiçar uma hora do seu tempo ainda não descobriu o valor da vida.”

Charles Darwin

Resumo

A popularização do serviço de Internet e a redução de custo facilitaram o acesso de usuários a diversos provedores de conteúdo, sejam estes baseados em voz, vídeo ou qualquer outro recurso de multimídia. Consequentemente aumentam os benefícios de ter uma rede de comunicações de melhor resiliência, pois quando uma conexão falha os dados podem fluir por conexões alternativas, reduzindo a probabilidade de perdas de pacotes e falhas na transmissão. Dessa forma, abre-se um novo cenário de estudo que tem a intenção de verificar e propor melhoras em formas de uso dos protocolos de comunicação que tenham o objetivo de tornar confiável a escolha dos melhores caminhos de transmissão para cada tipo de conteúdo.

Para a transmissão de *stream* de vídeo, a escolha do protocolo de transporte é tão importante quanto a escolha de algoritmos e estruturas de dados em relação à aplicação rede. O estudo do desempenho de protocolos de rede de transportes é essencial para o desenvolvimento de protocolos tolerantes a falhas. A avaliação de políticas combinado com o caminho de transmissão, e os limiares de detecção de falhas, largura de banda, atrasos e perda de vazão são avaliados através da simulação dos diferentes protocolos de transmissão utilizado. O comportamento dos protocolos são verificados, comparando o tráfego considerando caminhos alternativos simétricas e assimétricas. O protocolo escolhido para o estudo é o UDP, devido à sua popularidade e sua tolerância a perda. Mostramos neste estudo que a configuração do tamanho do GOP (*Group Of Picture*) e a utilização correta do tamanho do *Frame* do UDP podem contribuir para uma transmissão de *Vídeo Stream* com melhor qualidade utilizando-se dos recursos de conexões disponíveis (seleção de caminhos) na rede.

Palavras-chave: UDP, GOP, *Video Streaming*, Seleção de Caminhos.

Abstract

The popularization of the Internet service and cost reduction facilitated user access to various content providers, whether based on voice, video or other multimedia features. Consequently it increased the benefits of having a communications network with the best resiliency, as when a given connection fails the data can flow through alternative routes, reducing the probability of packet loss and the impact of transmission failures. Thus, it opens a new scenario for the study verification, and the proposal of improvements in the use of new forms of existing transport . Our goal is making reliable the end-to-end multimedia session by choosing the best transmission paths for each type of content.

For the transmission of Video Stream the choice of transport protocol is as important as the choice of algorithms and data structures in relation to the application network. The study of the performance of network transport protocols is essential for the development of fault-tolerant protocols. The policy evaluation combined with the transmission path, and fault detection thresholds, buffers and bandwidth, delays and loss of flow rate are evaluated through simulation of the different used transmission protocols. The behavior of the protocols is checked by comparing the traffic considering alternative symmetrical and asymmetrical paths. The protocol chosen for our studies is UDP, due to its popularity, and its loss tolerance. We show in this study that size setting GOP (Group of Picture) and using de correct frame size can contribute to a UDP broadcast stream with better quality video using the resources available connections (Multi-path Selection) in the network.

Keywords: UDP, GOP, Video Streaming, Multi-path Selection.

LISTA DE FIGURAS

Figura 1- Fragmentação de Datagramas - [3]	7
Figura 2 - Pré-processo, simulação NS-2, Pós-processo Evalvid	36
Figura 3 - Topologia de rede utilizado para as simulações	37
Figura 4 - 3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 6..	50
Figura 5 - 3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 8.	51
Figura 6 - 3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 8	51
Figura 7 - 3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 6.	54
Figura 8 - 3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 6...	54
Figura 9 - 3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP=6.....	55
Figura 10 - 3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego	
TCP(230,4kb;115,2kb;38,4kb); UDP- 512; GOP = 6.....	57
Figura 11 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e	
tráfego UDP (230,4kb;115,2kb;38,4kb); UDP 256; GOP = 6.....	58
Figura 12 - 3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP=6.....	58
Figura 13 - 1 via sem perdas - UDP 1024, GOP = 4	70
Figura 14 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) - UDP 1024; GOP = 4	70
Figura 15 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms) - UDP 1024; GOP = 4	71
Figura 16 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms) - UDP 1024; GOP =	71
4.....	
Figura 17 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) -	
UDP 1024; GOP = 4	72
Figura 18 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms) -	
UDP 1024; GOP = 4	72
Figura 19 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms) -	
UDP 1024; GOP = 4	73

Figura 20 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 4	73
Figura 21 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 4	74
Figura 22 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb;) - UDP 1024; GOP = 4	74
Figura 23 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb) - UDP 1024; GOP = 4	75
Figura 24 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms) - UDP 1024; GOP = 4	75
Figura 25 - 1 via sem perdas - UDP 1024; GOP = 6	76
Figura 26 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) - UDP 1024; GOP=6	76
Figura 27 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms) - UDP 1024; GOP = 6.....	77
Figura 28 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms) - UDP 1024; GOP = 6.....	77
Figura 29 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) - UDP 1024; GOP = 6	78
Figura 30 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms) - UDP 1024; GOP = 6	78
Figura 31 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms) - UDP 1024; GOP = 6	79
Figura 32 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb) UDP 1024; GOP = 6	79
Figura 33 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 6.....	80
Figura 34 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 6	80
Figura 35 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 6.....	81
Figura 36 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 1024; GOP = 6	81

Figura 37 - 1 via sem perdas; UDP 1024; GOP = 8.....	82
Figura 38 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP=8	82
Figura 39 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 1024; GOP = 8	83
Figura 40 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 1024; GOP = 8	83
Figura 41 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 8	84
Figura 42 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 8	84
Figura 43 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 8	85
Figura 44 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP(230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 8.....	85
Figura 45 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 8.....	86
Figura 46 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 8	86
Figura 47 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 8.....	87
Figura 48 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 1024; GOP = 8	87
Figura 49 - 1 via sem perdas; UDP 1024; GOP = 16.....	88
Figura 50 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 16.....	88
Figura 51 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 1024; GOP = 16.....	89
Figura 52 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 1024; GOP = 16.....	89
Figura 53 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 16	90

Figura 54 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 16	90
Figura 55 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 1024; GOP = 16	91
Figura 56 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 16	91
Figura 57 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 16.....	92
Figura 58 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 16	92
Figura 59 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 16.....	93
Figura 60 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 1024; GOP = 16	93
Figura 61 - 1 via sem perdas; UDP 512; GOP = 4.....	99
Figura 62 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 4	100
Figura 63 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 512; GOP = 4	100
Figura 64 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 512; GOP = 4	101
Figura 65 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 4	101
Figura 66 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 4	102
Figura 67 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 4	102
Figura 68 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP(230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 4.....	103
Figura 69 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 4.....	103

Figura 70 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 512; GOP = 4	104
Figura 71 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 512; GOP = 4.....	104
Figura 72 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 512; GOP = 4.....	105
Figura 73 - 1 via sem perdas; UDP 512; GOP = 6.....	105
Figura 74 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP=6.	106
Figura 75 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 512; GOP = 6	106
Figura 76 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 512; GOP = 6	107
Figura 77 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP =6	107
Figura 78 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP =6	108
Figura 79 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 6	108
Figura 80 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 6	109
Figura 81 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 6.....	109
Figura 82 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb;) UDP 512; GOP = 6	110
Figura 83 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb) UDP 512; GOP = 6.....	110
Figura 84 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 512; GOP = 6.....	111
Figura 85 - 1 via sem perdas; UDP 512; GOP = 8.....	111
Figura 86 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP=8.	112
Figura 87 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 512; GOP = 8	112

Figura 88 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 512; GOP = 8	113
Figura 89 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP= 8	113
Figura 90 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP= 8	114
Figura 91 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 8	114
Figura 92 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 8	115
Figura 93 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 8	115
Figura 94 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb) UDP 512; GOP = 8	116
Figura 95 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb) UDP 512; GOP = 8	116
Figura 96 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 512; GOP = 8	117
Figura 97 - 1 via sem perdas; UDP 512; GOP = 16	117
Figura 98 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 16	118
Figura 99 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 512; GOP = 16	118
Figura 100 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 512; GOP = 16	119
Figura 101 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP= 16	119
Figura 102 - vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 16	120
Figura 103 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 512; GOP = 16	120

Figura 104 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 16	121
Figura 105 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 16	121
Figura 106 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb) UDP 512; GOP = 16	122
Figura 107 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 512; GOP = 16	122
Figura 108 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 512; GOP = 16	123
Figura 109 - 1 via sem perdas; UDP 256; GOP = 4	129
Figura 110 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 4	129
Figura 111 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 256; GOP = 4	130
Figura 112 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 256; GOP = 4	130
Figura 113 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 4	131
Figura 114 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 4	131
Figura 115 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 4	132
Figura 116 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 4	132
Figura 117 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 4.....	133
Figura 118 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 4	133
Figura 119 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 4	134
Figura 120 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 256; GOP = 4	134

Figura 121 - 1 via sem perdas; UDP 256; GOP = 6	135
Figura 122 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP=6	135
Figura 123 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 256; GOP = 6	136
Figura 124 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 256; GOP = 6	136
Figura 125 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 6	137
Figura 126 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 6	137
Figura 127 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 6	138
Figura 128 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 6	138
Figura 129 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 6.....	139
Figura 130 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 6	139
Figura 131 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 256; GOP = 6	140
Figura 132 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 256; GOP = 6	140
Figura 133 - 1 via sem perdas; UDP 256; GOP = 8	141
Figura 134 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP=8	141
Figura 135 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 256; GOP = 8	142
Figura 136 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 256; GOP = 8	142
Figura 137 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP= 8	143

Figura 138 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP= 8	143
Figura 139 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 8	144
Figura 140 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 8	144
Figura 141 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 8.....	145
Figura 142 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 8	145
Figura 143 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 8	146
Figura 144 - 3 vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 256; GOP = 8	146
Figura 145 - 1 via sem perdas; UDP 256; GOP = 16	147
Figura 146 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 16.....	147
Figura 147 - 3 vias sem perdas com <i>delay</i> (240ms; 500ms; 700ms); UDP 256; GOP = 16.....	148
Figura 148 - 3 vias sem perdas com <i>delay</i> (360ms; 750ms; 1050ms); UDP 256; GOP = 16.....	148
Figura 149 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 16	149
Figura 150 - 3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 16	149
Figura 151 - 3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms; 250ms; 350ms); UDP 256; GOP = 16	150
Figura 152 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 16	150
Figura 153 - 3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 16.....	151

Figura 154 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 16	151
Figura 155 - 3 vias sem perdas com <i>delay</i> (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 16	152
Figura 156 - vias sem perdas com <i>delay</i> (1ms; 1ms; 1ms); UDP 256; GOP = 16	152

LISTA DE TABELAS

Tabela 1 -Estrutura do Protocolo IP [3].....	5
Tabela 2 - Exemplos de Redes - [3].....	7
Tabela 3 - Estrutura do Protocolo UDP [5]	9
Tabela 4 - RTP - Cabeçalho do Pacote [6]	11
Tabela 5 - Estrutura do Pacote de Dados TCP [7]	15
Tabela 6-Comparação entre SCTP, TCP e UDP [9] - (N/A = não aplicável).....	18
Tabela 7- Exemplos de sequências GOP (em ordem natural) e seus usos.[10]	21
Tabela 8- Exemplo de Resolução e Relação de Aspecto [12].	27
Tabela 9 - Escala de Qualidade - ITU - Escala de Imparidade [13].	27
Tabela 10 - Ferramentas Evalvid; Pré-processo, pós-processo[14]......	29
Tabela 11 - Resultados de simulações pós-processados com NS-2 - UDP 1024	64
Tabela 12 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-1)	65
Tabela 13 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-2)	66
Tabela 14 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-3)	67
Tabela 15 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-4)	68
Tabela 16 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-5)	69
Tabela 17 - Resultados de simulações pós-processados com NS-2 - UDP 512	94
Tabela 18 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-1)	95
Tabela 19 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-2)	96
Tabela 20 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-3)	97

Tabela 21 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-4)	98
Tabela 22 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-5)	99
Tabela 23 - Resultados de simulações pós-processados com NS-2 - UDP 256	124
Tabela 24 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-1)	125
Tabela 25 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-2)	126
Tabela 26 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-3)	127
Tabela 27 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-4)	128

LISTA DE EQUAÇÕES

Equação 1-[12] Definição do PSNR entre componente de luminância da fonte de imagem S e imagem de destino D	28
--	----

LISTA DE ACRÔNIMOS

3G -	Terceira geração de padrões e tecnologias de telefonia móvel possuem uma capacidade de rede maior por causa de uma melhora na eficiência espectral.
ACK -	Acknowledgement.
AllRtxAlt -	Todas as retransmissões pelo caminho alternativo.
AllRtxSame -	Todas as retransmissões para o mesmo caminho.
AV -	Áudio e Vídeo.
AVC -	Advanced Video Codec.
Bit -	Contração de Dígitos Binários.
Bytes -	É uma unidade de informação digital em computação e telecomunicações que mais comumente consiste de oito bits.
CC -	CSRC Count.
CD -	Disco Compacto.
CGA -	Color Graphics Adapter.
CIF -	Common Interface Format.
CSRC -	Contributing source.
DCT -	Discrete Cosine Transform.
DTMF -	Dual-tone multi-frequency signaling.
DVD -	Digital Versatile Disc, em português, Disco Digital Versátil.
ECN -	Explicit Congestion Notification.
FrSameRtoAlt -	Retransmissões rápidas pelo mesmo caminho, e retransmissões por limite de tempo pelo caminho alternativo.
G.711 -	Descreve a técnica de codificação para voz em PCM 64 Kbps

G.723 -	Opera a taxas de 6,3 e 5,3 kbit/s produzindo níveis de compressão digital da voz de, respectivamente, 10:1 e 12:1.
G.726 -	Fornece a compressão de 64 Kbps à taxas de 16, 24, 32 e 40 Kbps com base em técnica de Diferencial Adaptável de Modulação de código de pulso (Adaptive Differential Pulse Code Modulation) (ADPCM).
G.729 -	É um algoritmo de compressão de áudio para voz que compacta digital de voz em pacotes de 10 milissegundos duração; padrão G.729 opera com uma taxa de bits de 8 kbit / s, mas existem extensões que fornecem taxas de 6,4 kbit / e 11,8 kbit / s para a qualidade de voz pior e melhor.
GOF -	Grupo de Quadros.
GOI -	Grupo de Imagens.
GOP –	Group Of Picture.
GOV -	Grupos de Objetos de Video .
GSM -	Global System for Mobile Communications
H.261 -	É uma recomendação publicada pela ITU (International Telecommunication Union) que descreve métodos de codificação e decodificação para imagens móveis a ser utilizada em transmissões por rede a taxas de $p \times 64$ kbit/s, onde p tem valor entre 1 e 30.
H.263 -	Um padrão de compressão de vídeo originalmente concebido como um formato compactado bitrate baixo para videoconferência.
H.264 -	Um formato digital avançado para a codificação eficiente de vídeos de alta definição.
HD -	High Definition.
IETF -	Internet Engineering Task Force.
IHL -	Internet Header Length.
IP -	Internet protocol.
IPTV -	Internet Protocol Television.

ITU-T -	International Telecommunication Union.
JPEG -	Joint Photographic Experts Group - é um método comum usado para comprimir imagens fotográficas.
JVC -	Joint Video Team .
LTE -	Lon Term Evolution – Quarta geração de padrões e tecnologias de telefonia móvel.
MJPEG -	Motion JPEG.
MOS -	Mean Opinion Score.
MP3 -	(MPEG-1/2 Audio Layer 3) foi um dos primeiros tipos de compressão de áudio com perdas quase imperceptíveis ao ouvido humano.
MPEG -	Moving Picture Experts Group.
MPEG4 -	Moving Picture Experts Group , padrão número 4.
MSCTP -	Mobile Stream Control Transmission Protocol.
MTU -	Maximum Transfer Unit.
NS-2 -	Network Simulator 2.
NTSC -	National Television System Committee.
PAL -	Phase Alternating Line.
PDU -	Protocol Data Unit.
PMR -	Path.Max.Retrans.
PR-SCTP	Partial-Reliable SCTP.
PSNR -	Peak Signal-to-Noise.
PSTN -	Public switched telephone network.
PT -	Payload Type.
QCELP -	Qualcomm Code Excited Linear Prediction.
QCIF -	Quarter Interface Format .

Qos -	Quality of Service.
QVGA -	Quarter Video Graphics Array.
RFC -	Request For Comment.
RTCP -	Real-time Transport Control Protocol
RTP -	Real-time Transport Protocol
RTT -	Round-Trip Time. Tempo decorrido entre a transmissão do pacote e o recebimento da respectiva confirmação. É essencialmente a soma das latências de ida e de volta da rede.
SCTP -	Stream Control Transmission Protocol.
SSRC -	Synchronization Source.
SVCD -	Super Vídeo Compact Disc
SVGA -	Super Video Graphics Array
SXGA -	Super Extended Graphics Array
SYN -	Sincronização
TCP -	Transmission Control Protocol
TTL -	Time To Live
UDP -	User Datagram Protocol
UWB -	Ultra Wide Band
VGA -	Video Graphics Array
VRML -	Virtual Reality Modeling Language
WiFi -	Wireless Fidelity, ou Fidelidade sem fio
WiMAX -	Worldwide Interoperability for Microwave Access
WSXGA+ -	Widescreen Super eXtended Graphics Array Plus
WXGA -	Wide Screen Monitor

YUV – Y = luminância – o brilho (o preto e branco); U - cromaância (azul);
V = cromaância (vermelho)

SUMÁRIO

2	INTRODUÇÃO.....	1
2.1	Objetivos Gerais.....	1
2.2	Objetivos Específicos.....	2
2.3	Estrutura da Dissertação.....	2
2.4	Considerações Finais do Capítulo	3
3	REFERENCIAL TEÓRICO.....	4
3.1	Protocolo IP	4
3.2	A Fragmentação dos Datagramas IP	7
3.3	Protocolo UDP	8
3.4	Protocolo RTP	10
3.5	Protocolo TCP	13
3.6	Vídeo Digital	18
3.7	Compressão	19
3.8	Compressão Intra-Frame	19
3.9	Compressão Inter-Frame	20
3.10	Tipos de Frame MPEG	20
3.11	Frame Interleaving.....	21
3.12	Interleaving de GOPs.....	22
3.13	MPEG4	23
3.14	Erro Resiliente e Cancelamento	24
3.15	H.264	25
3.16	YUV.....	25
3.17	Resolução, Relação de Aspecto e Taxa de Quadros.....	26
3.18	PSNR – Avaliação de Qualidade.....	27
3.19	A Ferramenta Evalvid.....	28
3.20	Considerações Finais do Capítulo	31
4	METODOLOGIA.....	32
4.1	Considerações Gerais	32
4.2	Trabalhos Relacionados	33

4.3	Classificação da Pesquisa.....	34
4.4	Etapas da Pesquisa	35
5	AVALIAÇÃO DOS RESULTADOS E DISCUSSÕES.....	48
5.1	Resultados Obtidos.....	48
5.2	Discussão dos Resultados.....	59
6	CONCLUSÃO.....	60
7	REFERÊNCIAS.....	61
8	ANEXOS.....	64
8.1	Anexo I: Resultado dos experimentos dos cenários simulados.....	64

2 INTRODUÇÃO

A crescente popularização e utilização de equipamentos como o *tablet*, celulares, *notebook's* e *netbooks* para acessos a multimídias como Youtube e vídeos na internet, em diferente e variados locais, como em aeroportos, a bordo de aviões, escolas, bibliotecas, ambientes de trabalhos nas empresas, sob diferentes provedores de acessos e tecnologias, como *WiFi*, *WiMAX*, *2G*, *3G*, e *LTE* (*Long Term Evolution* - que já está em fase inicial de implantação) motiva o estudo de sistemas multiabrigados (*multihomed*) e o estudo de desempenho dos protocolos de transporte de rede, essencial para o desenvolvimento de protocolos tolerantes a falhas.

Com o crescimento de aplicações em tempo real nas redes, tanto nas redes fixas convencionais como nas redes móveis *2G* e *3G*, a prevenção do atraso fim a fim e a perda de pacotes, são vitais para quesitos de *QoS* destas aplicações.

Portanto, para investigar a qualidade de tais aplicações sob carga pesada de rede, é necessário criar padrões de tráfego real, tanto na rede como na camada de transporte e aplicação. Sendo assim, foi criada uma rede de com três possíveis caminhos, todos independentes, tendo uma origem e um destino, com características de atrasos, *GOP* e quadros de *UDP* que possibilita a medição do desempenho.

Os testes utilizando uma rede real para a transmissão de multimídia ficariam caros e com pouca flexibilidade para a sua configuração e escalabilidade limitada. Para a construção de rede eficazes e personalizados com baixo custo, utilizamos o simulador de domínio público *NS-2*[1]. A escolha deste simulador será explicado no terceiro capítulo.

2.1 Objetivos Gerais

A pesquisa tem propõe verificar e comprovar junto ao corpo científico, que mesmo utilizando um protocolo simples como o *UDP*, podemos otimizar e viabilizar a transmissão de vídeo, utilizando redes similares aos multiabrigados (*multihomed*), tornando o sistema mais tolerantes a falhas.

A política de transmissão, combinada com o processo de escolha de caminhos e limiares de detecção de falhas, buffers e banda disponível, atrasos e perda de taxa de fluxo são avaliados através de simulação do protocolo de transmissão. O comportamento dos protocolos é verificado comparando o tráfego considerando os caminhos alternativos simétricos e assimétricos.

2.2 Objetivos Específicos

Definimos como objetivo desta pesquisa as seguintes características:

1. A qualidade da rede : atrasos fim a fim e perda de pacotes – QoS
2. Simular redes com características diferentes – perdas, atrasos, e ainda ajustando o tamanho do quadro do UDP.
3. Verificar o desempenho da rede na transmissão de vídeo H.264 com codificação de cores YUV e sequência 4:2:0.

2.3 Estrutura da Dissertação

Este trabalho está estruturado da seguinte forma: o capítulo 2 aborda conceitos teóricos envolvidos neste trabalho; o capítulo 3 consiste da Motivação e uma breve revisão sobre trabalhos relacionados, e a descrição dos cenários simulados; os resultados aparecem no capítulo 4; e as conclusões são apresentadas na capítulo 5.

2.4 Considerações Finais do Capítulo

Mesmo utilizando um protocolo simples é possível otimizar e viabilizar a transmissão de vídeo, utilizando redes similares a multiabrigados, tornando o sistema mais tolerantes a falhas.

Comprovamos esta hipótese através de:

1. Escolha do simulador NS-2, devido a valor de investimento, flexibilidade, escalabilidade e compatibilidade com as Evalvid;
2. Simulação de transmissão de *vídeo stream* em redes com características diferentes ajustando o UDP Frame e GOP;
3. Investigação a qualidade de na transmissão de vídeo na topologia escolhida;
4. Verificação de desempenho da rede na transmissão de vídeo H264;

3 REFERENCIAL TEÓRICO

Os protocolos de comunicação são padrões que controlam e possibilitam conexões, comunicação ou transferência de dados entre dois ou mais sistemas computacionais. São as regras controlando a comunicação, que podem ser implementadas pelo *hardware*, *software* ou ambos. Os protocolos variam muito em propósito e sofisticação, onde são especificados uma ou mais das seguintes propriedades [2]:

- detecção da conexão física subjacente ou a existência de um nó;
- *handshaking* (estabelecimento de conexão);
- negociação de várias características de uma conexão;
- como iniciar e finalizar uma mensagem;
- como formatar uma mensagem;
- o que fazer com mensagens corrompidas ou mal formatadas;
- como detectar perda inesperada de conexão e o que fazer em seguida;
- término de sessão ou conexão;

3.1 Protocolo IP

O protocolo IP é uma dos protocolos mais importantes da Internet, pois permite a elaboração e o transporte dos datagramas IP (pacote de dados), sem, contudo, assegurar a entrega. O protocolo IP determina o destinatário da mensagem graças a 3 campos:

- O campo dirige IP : endereço da máquina;
- O campo máscara de sub-rede: uma máscara de sub-rede permite ao protocolo IP determinar a parte do endereço IP que se refere à rede ;
- O campo ponte estreita por defeito: Permite ao protocolo Internet saber a que máquina entregar o datagrama, se por acaso a máquina de destino não está na rede local;

Veja a estrutura do Protocolo IP – Tabela 1[3]:

Tabela 1 -Estrutura do Protocolo IP [3]

<--		32 bits		-->	
Versão (4 bits)	Comprimento de cabeçalho (4 bits)	Tipo de serviço (8 bits)	Comprimento total (16 bits)		
Identificação (16 bits)			Bandeira (3 bits)	Desfasamento fragmento (13 bits)	
Duração de vida (8 bits)		Protocolo (8 bits)	<i>header checksum</i> (16 bits)		
Endereço IP fonte (32 bits)					
Endereço IP destino (32 bits)					
Dados					

Eis o significado dos diferentes campos:

- Versão (4 bits): trata-se da versão do protocolo IP que se utiliza para verificar a validade do datagrama. É codificada em 4 bits.
- Comprimento de cabeçalho, ou IHL (4 bits): trata-se do número de palavras de 32 bits que constituem o cabeçalho (nota: o valor mínimo é 5). Este campo é *Internet Header Length* codificado em 4 bits.
- Tipo de serviço (8 bits): indica a maneira segundo a qual o datagrama deve ser tratado.
- Comprimento total (16 bits) : indica a dimensão total do datagrama em bytes. A dimensão deste campo de 2 bytes, a dimensão total do datagrama não pode exceder 65536 bytes. Utilizado conjuntamente com a dimensão do cabeçalho, este campo permite determinar onde estão situados os dados.

- Identificação, bandeiras (*flags*) e deslocação de fragmento: são campos que permitem a fragmentação dos datagramas, e que serão explicados abaixo.
- Duração de vida chamada também TTL, para *Time To Live* (8 bits): este campo indica o número máximo de *switchs* através dos quais o datagrama pode passar. Assim este campo é reduzido a cada passagem em switch, quando este atinge o valor crítico de 0, o switch destrói o datagrama. Isto evita o congestionamento da rede pelos datagramas perdidos.
- Protocolo (8 bits): este campo, em notação decimal, permite saber de que protocolo procede o datagrama :
TCP : 6
UDP : 17
- Soma de controle do cabeçalho ou, em inglês, *header checksum* (16 bits) : este campo contém um valor codificado de 16 *bits*, que permite controlar a integridade do cabeçalho a fim de determinar se este não foi alterado durante a transmissão. A soma de controle é o complemento de todas as palavras de 16 *bits* do cabeçalho (campo soma de controle excluído). Isto faz-se para que, quando se faz a soma dos campos do cabeçalho (soma de controle incluída), se obtenha um número com todos os *bits* posicionados a 1.
- Endereço IP fonte (32 *bits*) : Este campo representa o endereço IP da máquina emissora, permite ao destinatário responder.
- Endereço IP destino (32 *bits*) : endereço IP do destinatário da mensagem

3.2 A Fragmentação dos Datagramas IP

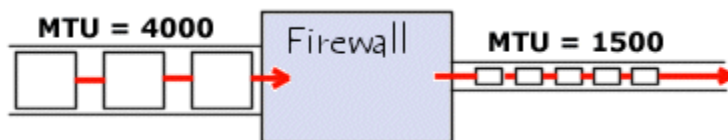
A dimensão máxima de um datagrama é de 65536 *bytes*, apesar de nunca chegarmos a este valor devido a capacidade da rede. Na Internet, utilizamos diferentes tecnologias, de modo que a dimensão de um datagrama varia de acordo com o tipo de rede. A dimensão máxima de uma trama chama-se MTU (*Maximum Transfer Unit*), que provocará a fragmentação do datagrama se este tiver uma dimensão maior do que a MTU da rede. Exemplo na Tabela 2[3]:

Tabela 2 - Exemplos de Redes - [3]

Tipo de rede	MTU (em bytes)
Arpanet	1000
Ethernet	1500
FDDI(<i>Fiber Distributed Data Interface</i>)	4470

A fragmentação ocorre durante a transição de uma rede (*switchs*) cuja MTU é maior para uma rede cuja MTU é mais fraca. Se o datagrama for demasiadamente grande para passar na rede, o *switch* vai fragmentá-lo, de modo que o fragmento seja múltiplo de 8 *bytes* – Figura 1.

Figura 1- Fragmentação de Datagramas - [3]



O switch vai enviar os fragmentos de maneira independentes e re-encapsulá-los (acrescentar cabeçalho a cada fragmento) de maneira a ter a nova dimensão do fragmento. Para ter em conta a fragmentação, cada datagrama possui vários campos que permitem a sua remontagem [3]:

- campo deslocação de fragmento (de 13 *bits*): campo permitindo conhecer a posição do início do fragmento no datagrama inicial. A unidade de medida

deste campo é de 8 *bytes* (o primeiro fragmento que tem um valor de zero).

- campo identificação (16 *bits*) : número atribuído a cada fragmento a fim de permitir a sua remontagem.
- campo comprimento total (16 *bits*): é calculado novamente para cada fragmento.
- campo bandeira (3 *bits*): é composto de três *bits* :
- O primeiro não é utilizado.
- O segundo (chamado DF: *Don' t Fragment*) indica se o datagrama pode ser fragmentado ou não. Se por acaso um datagrama tem este bit posicionado em 1 e o *switch* não pode encaminhá-lo sem o fragmentar, então o datagrama é rejeitado com uma mensagem de erro.
- O último (designado MF : *More Fragments*) indica se o datagrama for um fragmento de dado (1). Se o indicador estiver a zero, isso indica que o fragmento é o último (por conseguinte , que o *switch* deveria estar na posse dos fragmentos precedentes) ou que o datagrama não foi alvo de uma fragmentação.

3.3 Protocolo UDP

O protocolo de transporte UDP (*User Datagram Protocol* – RFC 768) é um dos protocolos originais da arquitetura da Internet [4]. Este protocolo adiciona somente a aplicação de multiplexação (usando números de porta) e *checksum* para os dados antes de ser enviado por IP. Como tal, é um protocolo leve, inserindo somente 8 bytes de sobrecarga para os dados. Em comparação com o protocolo de transporte TCP (*Transmission Control Protocol*) , o UDP não fornece ordenação de dados, controle de fluxo ou confiabilidade (apesar de que todo o pacote sem erro serem passados para a

camadas superiores), e fornece dados enquadrados em datagramas, não como um fluxo contínuo como é o caso do TCP.

UDP é geralmente usado em aplicações de videoconferência ou jogos de computador especialmente afinado para desempenho em tempo real. Para alcançar um maior desempenho, o protocolo permite que os pacotes individuais possam ser descartados (sem retransmissões).

A rúbrica do segmento UDP é por conseguinte muito simples – Tabela 3[5] :

Tabela 3 - Estrutura do Protocolo UDP [5]

Porta Fonte (16 bits)	Porta Destino (16 bits)
Tamanho do datagrama (16 bits)	Checksum (16 bits)
Dados (comprimento variável)	

Significado dos diferentes campos

- Porta Fonte : trata-se do número de porta que corresponde à aplicação emissora do segmento UDP. Este campo representa um endereço de resposta para o destinatário. Assim, este campo é opcional, isto significa que se não se precisar da porta fonte, os 16 bits deste campo serão postas a zero, neste caso o destinatário não poderá responder (isto não é necessário, notadamente para mensagens unidirecionais).
- Porta Destino : Este campo contém a porta que corresponde à aplicação da máquina destinatário à qual nos dirigimos.

- Tamanho do datagrama: Este campo precisa o comprimento total do datagrama, incluindo o cabeçalho. Se cabeçalho tem um comprimento de 4 x 16 *bits* (são 8 x 8 *bits*), então o campo comprimento é necessariamente superior ou igual a 8 *bytes*.
- *Checksum* : UDP *checksums* protegem os dados de adulteração de mensagem. O valor representa uma soma de verificação de codificação dos dados, datagrama calculado primeiro pelo remetente e, posteriormente, pelo receptor. Datagrama individuais adulterados ou corrompidos durante a transmissão, o protocolo UDP detecta uma incompatibilidade de cálculo de *checksum*. Em UDP, *checksumming* é opcional em vez de TCP onde *checksums* são obrigatórios. [5]

3.4 Protocolo RTP

O *Real-time Transport Protocol* (RTP) define o formato de pacote padronizado para entrega de áudio e vídeo em redes IP. Foi desenvolvido pelo Grupo de Transporte de Audio-Vídeo do IETF (*Internet Engineering Task Force*) em 1996 como RFC 1889, substituído pelo RFC 3550 em 2003. É usado extensivamente em sistemas de comunicação e entretenimento que envolvem streaming mídia, tais como vídeo e teleconferência. RTP é usado em conjunto com a RTCP (*Real-time Transport Control Protocol*). RTCP é usado para monitorar as estatísticas de transmissão e qualidade de serviço (QoS) e sincronização de vários fluxos de *stream* – suporta *multicast* – e ocupa aproximadamente 5% do tráfego do RTP. O RTP foi projetado para transferência de fluxo de dados fim a fim, em tempo real, fornece a facilidade para compensação de *jitter* e detecção de chegada dos dados fora de sequência, que são comuns durante as transmissões em uma rede IP. Informações fornecidas por este protocolo incluem *timestamps* (para sincronização), números de sequência (para perda de pacotes e detecção de re-ordenamento) e o formato da carga útil que indica o formato de codificação dos dados. Aplicativos de Multimídia em tempo real exigem a entrega oportuna no tempo da informação e podem tolerar alguma perda para atingir este

objetivo. Por exemplo, a perda de pacotes de aplicativo de áudio pode resultar em perda de uma fração de segundo de dados de áudio, que pode ser imperceptível com a utilização de algoritmos adequados de ocultação de erro. Embora o TCP esteja padronizado para o uso do RTP, não é normalmente utilizado, pois favorece a confiabilidade. A maioria das implementações do RTP são construídas sobre o UDP.

Uma das considerações de projeto do RTP foi apoiar uma variedade de formatos multimídia (como H.264, MPEG-4, MJPEG, etc) e permitir novos formatos para serem adicionados sem a revisão do padrão RTP. As informações precisas exigidas de uma aplicação específica não são incluídas no cabeçalho genérico do RTP, mas através dos perfis RTP e os formatos de carga útil. A especificação completa da RTP para um uso determinado aplicativo vai exigir um perfil de carga e especificação de formato. O perfil define os codec usados para codificar os dados de carga útil e seu mapeamento para os códigos de carga útil no campo Tipo de *Payload* do cabeçalho RTP, apresentada na Tabela 4. Cada perfil é acompanhado por diversas especificações de formato de carga útil, cada um dos quais descreve o transporte de dados em particular de um dado codificado. Alguns dos formatos de *payload* de áudio incluem: G.711, G.723, G.726, G.729, GSM, QCELP, MP3, DTMF, etc, e alguns dos formatos de *payload* de vídeo incluem: H.261, H.263, H.264, MPEG-4 e etc [6].

Tabela 4 - RTP - Cabeçalho do Pacote [6]

RTP packet header							
bit offset	0-1	2	3	4 - 7	8	9 - 15	16-31
0	Version	P	X	CC	M	PT	Sequence Number
32	Timestamp						
64	SSRC identifier(synchronous source)						
96	CSRC identifiers(contributing source)						
	...						
96+32×CC	Profile-specific extension header ID						Extension header length
128+32×CC	Extension header						
	...						

O cabeçalho do RTP tem um tamanho mínimo de 12 bytes. Após o cabeçalho, extensões opcionais podem estar presentes. Isto é seguido pela carga útil RTP, cujo formato é determinado pela classe particular de aplicação.

Os campos do cabeçalho são os seguintes [6]:

- Versão: (2 bits) – indica a versão do protocolo.
- P (*Padding*): (1 bit) – usado para indicar se há bytes estofamentos extra no final do pacote RTP. Um estofamento pode ser usado para preencher um bloco de certo tamanho, por exemplo, como exigido por algoritmo de criptografia.
- X (Extensão): (1 bit) – indica a presença de um cabeçalho de extensão entre o cabeçalho padrão e os dados de carga útil. Esta é a ampliação ou perfil específico.
- CC (*CSRC Count*): (4 bits) – contém o número de identificadores CSRC que segue o cabeçalho fixo.
- M (marcador): (1 bit) usado no nível do aplicativo e definida por um perfil. Se ele estiver definido, isso significa que os dados atuais tem alguma relevância especial para a aplicação.
- PT (*Payload Type*): (7 bits) – indica o formato da carga útil e determina a sua interpretação pela aplicação. Isto é especificado por um perfil de RTP. Por exemplo perfil RTP para conferências de áudio e vídeo com controle mínimo (RFC 3551).
- *Sequence Number*: (16 bits) – o número de sequência é incrementado para cada pacote de dados RTP enviado e será utilizado pelo receptor para detectar perda de pacotes e para restaurar a sequência de pacotes. A RTP não especifica qualquer ação sobre a perda de pacotes, que é deixada para o aplicativo para tomar as medidas adequadas. Por exemplo, aplicações de

vídeo podem jogar o último quadro conhecido no lugar do quadro de falta. De acordo com a RFC 3550, o valor inicial do número de sequência deve ser aleatório para tornar os ataques a criptografia mais difíceis. RTP não fornece nenhuma garantia de entrega, mas a presença de números de sequência torna possível detectar pacotes perdidos.

- *Timestamp*: (32 *bits*) - usado para ativar o receptor a reproduzir as amostras recebidas em intervalos apropriados. Quando vários fluxos de dados estão presentes, os *timestamps* são independentes em cada *stream*, e não pode ser invocado para a sincronização de mídia. A granularidade do tempo é uma aplicação específica. A granularidade do relógio é um dos detalhes que está especificado no perfil de RTP para uma aplicação.
- SSRC: (32 *bits*) – identificador de fonte de sincronização identifica a fonte de um *stream*. As fontes de sincronização dentro da sessão RTP serão únicas.
- CSRC: ID's que enumeraram as fontes que contribuem para um fluxo que tem sido gerados a partir de múltiplas fontes.
- *Extension header*: (opcional) – a palavra de 32 *bits* contém um identificador específico do perfil (16 *bits*) e um comprimento especificador (16 *bits*) que indica o comprimento da extensão (EHL = comprimento do cabeçalho de extensão) em 32 *bits* unidades, excluindo os 32 *bits* do cabeçalho de extensão.

3.5 Protocolo TCP

O protocolo TCP (RFC793) é amplamente utilizado para diferentes tipos de serviços que necessitam de transporte confiável dos dados em rede IP. Um número crescente de aplicações mostrou que o TCP é bastante limitado como mostrado em seguida, por isto, determinadas aplicações optaram por incorporar novos algoritmos para

obter confiabilidade de forma mais flexível, utilizando o protocolo UDP (RFC768). As limitações mais comuns do TCP são:

- O protocolo TCP fornece transporte confiável com uma ordem estrita de transferência de dados. Algumas aplicações precisam de confiabilidade, sem manutenção do número da sequência, ou ainda podem ser satisfeitas com uma ordem parcial. O bloqueio causado por esta política do TCP causa atrasos desnecessários. Um exemplo típico desta categoria de aplicações é o transporte da sinalização em redes de telecomunicações.
- A natureza *stream-oriented* do TCP pode ser considerada um inconveniente na maioria das vezes. As aplicações devem adicionar o seu próprio registro de marcas para delinear suas próprias mensagens, e devem fazer explicitamente o uso desta facilidade para assegurar que a mensagem seja transferida em um tempo razoável.
- O escopo limitado dos *sockets* do TCP complica a tarefa de fornecer alta disponibilidade na transferência de dados para aplicações que utilizam hosts “*multihomed*”.
- O TCP é relativamente vulnerável a ataques, tal como o ataque do SYN para degeneração dos serviços DoS entre outros.[7]

A tabela 5 mostra a estrutura do pacote de dados do TCP:

Tabela 5 - Estrutura do Pacote de Dados TCP [7]

+	Bits 0 - 3	4 - 9	10 - 15	16 - 31		
0	Porta na origem			Porta no destino		
32	Número de sequência					
64	Número de confirmação (ACK)					
96	Offset	Reservados	Flags	Janela Window		
128	Checksum			Ponteiro de urgência		
160	Opções (opcional)					
Padding (até 32)						
224	Dados					
Detalhe do campo <i>Flags</i>						
+	10	11	12	13	14	15
96	<i>UrqPtr</i>	ACK	<i>Push</i>	RST	SYN	FIN

As características fundamentais do TCP são [8]:

- Orientado à conexão - A aplicação envia um pedido de conexão para o destino e usa a "conexão" para transferir dados.
- Ponto a ponto - uma conexão TCP é estabelecida entre dois pontos.
- Confiabilidade - O TCP usa várias técnicas para proporcionar uma entrega confiável dos pacotes de dados, que é a grande vantagem que tem em relação ao UDP, e motivo do seu uso extensivo nas redes de

computadores. O TCP permite a recuperação de pacotes perdidos, a eliminação de pacotes duplicados, a recuperação de dados corrompidos, e pode recuperar a ligação em caso de problemas no sistema e na rede.

- *Full duplex* - É possível a transferência simultânea em ambas as direções (cliente-servidor) durante toda a sessão.
- *Handshake* - Mecanismo de estabelecimento e finalização de conexão a três e quatro tempos, respectivamente, o que permite a autenticação e encerramento de uma sessão completa. O TCP garante que, no final da conexão, todos os pacotes foram bem recebidos.
- Entrega ordenada - A aplicação faz a entrega ao TCP de blocos de dados com um tamanho arbitrário num fluxo (ou *stream*) de dados, tipicamente em octetos. O TCP segmenta estes dados em tamanho especificado pelo valor MTU. Porém, a circulação dos pacotes ao longo da rede (utilizando um protocolo de encaminhamento, na camada inferior, como o IP) pode fazer com que os pacotes não cheguem ordenados. O TCP garante a reconstrução do *stream* no destinatário mediante os números de sequência.
- Controle de fluxo - O TCP usa o campo janela ou *window* para controlar o fluxo. O receptor, à medida que recebe os dados, envia mensagens ACK (=Acknowledgement), confirmando a recepção de um segmento; como funcionalidade extra, estas mensagens podem especificar o tamanho máximo do *buffer* no campo (janela) do segmento TCP, determinando a quantidade máxima de *bytes* aceita pelo receptor. O transmissor pode transmitir segmentos com um número de *bytes* que deverá estar confinado ao tamanho da janela permitido: o menor valor entre sua capacidade de envio e a capacidade informada pelo receptor.

Em contraste, o SCTP (*Stream Control Transmission Protocol*) evoluiu a partir de um protocolo de sinalização telefônico desenvolvido para redes convergentes IP. Em Outubro de 2000 ele foi padronizado pela IETF (RFC 2960). O SCTP foi projetado como protocolo de uso geral para aplicações orientadas a mensagens com um controle de congestionamento compatível com o TCP.

O desenvolvimento do SCTP foi motivado pelas limitações do TCP em transportar sinalização PSTN sobre redes IP, no entanto o SCTP pode ser utilizado em outras aplicações fazendo uso de suas características.

O projeto do protocolo inclui a prevenção de congestionamento e a resistência a ataques do tipo *flooding* (inundação de mensagens) e mascaramento (quando um sistema assume a identidade de outro) .

O SCTP apresenta outra característica marcante, que é o suporte a múltiplos endereços IP em uma mesma “conexão” permitindo a redundância de caminhos em caso de falhas na rede – *multihoming*.

O protocolo SCTP é orientado a conexão, estabelecendo uma associação entre Tx e Rx com um número arbitrário de fluxos simplex de transmissão e recepção.

Diferentemente do TCP que é orientado a octeto, o SCTP é orientado a mensagens, onde as informações transmitidas em uma comunicação que é transportada em blocos de dados (*chunks*), que podem ser de usuários – dados de aplicação - ou de controle do protocolo. O estabelecimento de associações (conexões entre entidades) é realizado pela troca de quatro mensagens de controle entre cliente e servidor conhecido como *four-way handshake* ao invés do *three-way handshake* do TCP para evitar ataques como o do SYN. Esta troca de mensagens garante ao SCTP uma maior flexibilidade para complementações de sua estrutura operacional. Já o TCP possui suas funções de controle embutidas em uma estrutura de segmento, o que traz eficiência computacional de processamento, mas inibe muitas tentativas de adaptação e evolução do protocolo.

O protocolo SCTP tem taxa adaptativa (*rate adaptive*), o que garante a sua adaptação dinâmica às variações e problemas que ocorrem na rede. Os algoritmos de controle de congestionamento do TCP foram aproveitados no SCTP com pequenas variações devido a eficiência comprovada em anos de operação.

O MSCTP (*Mobile Stream Control Transmission Protocol*), é uma extensão que define o uso do SCTP em conjunto com mecanismos de configuração dinâmica de endereços numa associação. O MSCTP explora as características do *multihoming* do SCTP em conjunto com a configuração dinâmica de endereços, o que permite operações de *handover* (mudança de servidores sem desconexão) em ambientes que necessitam de mobilidade, garantindo assim a “mobilidade terminal” no nível de transporte [7].

A Tabela 6 mostra um comparativo de funções entre os principais protocolos de transporte mencionados.

Tabela 6-Comparação entre SCTP, TCP e UDP [9] - (N/A = não aplicável).

Função/Serviço	SCTP	TCP	UDP
Permite conexões <i>half-closed</i>	Não	Sim	N/A
Agrupamento de PDU por aplicativo	Sim	Sim	Não
Fragmentação de PDU por aplicativo	Sim	Sim	Não
Controle de congestionamento	Sim	Sim	Não
Orientado à conexão	Sim	Sim	Não
Capaz de usar ECN	Sim	Sim	Não
Controle de fluxo	Sim	Sim	Não
<i>Full duplex</i>	Sim	Sim	Sim
<i>Multi-homing</i>	Sim	Não	Não
<i>Multistreaming</i>	Sim	Não	Não
Distribuição ordenada de dados	Sim	Sim	Não
Transferência parcial confiável de dados	Opcional	Não	Não
Descoberta do MTU do caminho	Sim	Sim	Não
Preserva limites da mensagem	Sim	Não	Sim
Protege contra ataques de inundação SYN	Sim	Não	N/A
Pseudocabeçalho para o <i>checksum</i>	Usa vtags	Sim	Sim
Verificação de acessibilidade	Sim	Sim	Não
Transferência confiável de dados	Sim	Sim	Não
Acks seletivos	Sim	Opcional	Não
Estado de tempo de espera	para vtags	para 4-tuple	N/A
Distribuição de dados não-ordenada	Sim	Não	Sim

3.6 Vídeo Digital

Streaming de mídia inclui a codificação e intercambio de mídia que é consumida (vista ou ouvida principalmente na forma de *clips*) enquanto ele está sendo entregue. *Streaming* é mais que uma propriedade da entrega do sistema do que a própria mídia. A distinção é normalmente aplicada aos meios de comunicação que são distribuídas em redes de computadores, e outros meios de comunicação (rádio, televisão), ou inerentemente não *streaming* (livros, CD's) . “*Stream*” também é usado como verbo, ou seja para entregar media *streaming*.

O vídeo digital é uma sequência de imagens codificadas mostradas em rápida sucessão. Para o sistema de recepção visual humano é um movimento contínuo, devido a persistência da visão. A diferença de tempo entre as imagens pode ser de 0,1s para movimento muito lento, e menos de 0,05 (mais de 20 frames por segundo), não apresenta uma apreciável diferença de qualidade. A maioria dos sistemas de vídeo usa 15, 24, 25 ou 30 quadros por segundo.

Vídeo é um meio intenso de dados. Especialmente dentro das redes. Os vídeos avançam para uma distribuição mais ampla de dispositivos (tele-moveis, leitores de MP3) e alcançam maiores resoluções, gerando novas tecnologias de vídeos. A tecnologia IPTV emprega um vídeo em formato de distribuição de demanda em camadas sobre IP, com avançada funcionalidade de consumo, tais como publicidade e gerenciamento personalizado de conteúdo. Outra evolução da tecnologia que está ganhando força é a entrega de vídeo de alta resolução (HD).

3.7 Compressão

Vídeo Digital tem várias informações redundantes no mesmo, que podem ser classificados em *inter-frame* (entre quadros sucessivos) ou *intra-frames* (dentro de um único *frame*). Diferentes métodos são utilizados para codificar essas redundâncias de forma eficiente.

3.8 Compressão *Intra-Frame*

Analizando apenas um *frame* de vídeo, a compressão *intra-frame* não difere da compressão de imagem normal. Técnicas de compressão tais como *Walvelet* e *Discrete Cosine Transform* (DCT) (como aquele encontrado em JPEG de compressão de imagem) são utilizados [10], sendo referidos como codificação de textura. A codificação *intra-frame* é usada com perdas de codificação de imagem. A codificação *intra-frame* é muito mais utilizada que a de *inter-frame*.

3.9 Compressão *Inter-Frame*

Os dados de vídeo contemplam várias redundâncias temporais, onde cada quadro está intimamente relacionado com uma área no próximo quadro, na maioria dos casos. Uma previsão pode ser feita quando da codificação de uma pequena área de *pixels* em um quadro para o próximo quadro, pois estes, vão estar intimamente relacionados de uma área para o próximo. Este conceito de compensação de movimento permite comprimir melhor o movimento da câmera e objetos no quadro, que são expressas como vetores de movimento.

Vetores de movimento são “ponteiros” para a área semelhante à atual no anterior, em um quadro anterior ou posterior. Os vetores normalmente são limitados a que medida eles podem apontar para a redução do espaço de busca da área semelhante e tempo de compressão, e manter o tamanho compactado do vetor de movimento menor. Esta forma de compensação de movimento funciona bem para objetos móveis dentro do quadro, onde pode ser usado um método “global” de compensação de movimento para melhorar a compressão apenas sobre os vetores móveis. As áreas que não podem ser codificadas pelos métodos de compensação de movimento são codificadas utilizando compressão *intra-frame*, como são as regiões onde o erro na imagem reconstruída é grande [10].

3.10 Tipos de Frame MPEG

Na maioria dos sistemas de compressão de vídeo, existem três tipos de quadros, aqueles que têm apenas compressão *intra-frames* (*I-Frames*), aqueles que estão previsto a partir de um quadro anterior (*P-Frames*), e os bi-predictos (bidirecionalmente previstos) a partir de um quadro anterior e um futuro quadro (*B-Frames*).

I-Frames são os maiores em termos de conteúdo, são a representação mais precisa, e não dependem de quadros anteriores para a interpretação das suas informações.

P-Frames são menores, mas dependem do anterior I ou P frames para algumas informações. Eles fazem parte do vetor de movimento e oferecem o restante da codificação de textura.

B-Frames são os menores, mas dependem do anterior I-frames ou P-frames, ou do próximo I-frames ou P-frames de informação para prever o conteúdo do quadro. Esses quadros são em sua maioria vetores de movimento a partir de qualquer quadro anterior ou posterior.

3.11 *Frame Interleaving*

O vídeo digital é organizado em uma sequência recorrente de tipos de quadros. Estas sequências são conhecidas como um grupo de imagens (GOP – *Group Of Picture*), Tabela 7. Existem outras padronizações com diferentes padrões de organização, que chamam de Grupo de Quadros (GOF), Grupos de Objetos de Video (GOV) e Grupo de Imagens (GOI), que tem conceitos semelhantes, mas podem ter características diferentes.

Tabela 7- Exemplos de sequências GOP (em ordem natural) e seus usos.[10]

Sequência	Comprimento	# de Quadros B	Exemplo/Usos
I	1	0	vídeo de stúdio, Movimento-jpeg, alta taxa de bits e qualidade.
IP	2	0	vídeo de stúdio, melhor compressão.
IBPB	4	1	Vídeo com movimento muito alto, com rápida mutação
IBBPBB	6	2	DivX codec de alto movimento , bom para cenas de ação.
IBBBPBBB	8	3	
IBBPBBPBB	9	2	
IBBPBBPBBPBB	12	2	DivX codec de baixo movimento, bom para o vídeo típico.
IPPPPPPPPPPPPPPP	15	0	Vídeo conferência

O GOP sempre inicia com o *Frame I* e tem somente um destes em cada sequência. O *Frame P* ocorre a intervalos fixos, com a exceção onde o *Frame I* é o início. O *Frame B* preenche o resto da sequência.

Por exemplo, para vídeo conferência temos um longo período de *Frames P*, pois temos pouco movimento, a taxa de *bits* deve ser minimizada e o atraso deve ser mantido baixo. Os *Frame B* não podem ser usados, pois aumentariam o atraso até a chegada do próximo *Frame P*. Quando uma sequência de 15 quadros por segundos é usada, o *Frame I* é enviado a cada segundo para corrigir eventuais erros que se acumulam através dos *Frames P*.

Para codificar (ou decodificar) um GOP, os quadros devem se codificadas na ordem do menos dependente para os mais dependentes, ou seja *I*, *P*, e *Frame B*. Para a sequência do GOP IBBPBB (escrito como $\text{GOP } I_1 B_1^0 B_1^1 P_1 B_1^2 B_1^3$ quando o número da sequência é 1), primeiro o *Frame I* é codificado. Os dois últimos *Frames B* dependem do *Frame P* (ou *I*) precedente e do *Frame I*. Na sequência temos o *Frame P*, precedido de dois *Frames B* formando o GOP. Assim temos o $\text{GOP } I_1 B_0^2 B_0^3 P_1 B_1^0 B_1^1$ que é referenciado como a sequência de codificação. A decodificação ocorre na mesma ordem, assim como o armazenamento e transmissão.

3.12 Interleaving de GOPs

Em redes de transporte confiável ou parcialmente confiável, quando um pacote é perdido por motivo de congestionamento, erro, ou roteamento, este pacote é retransmitido. Estas retransmissões ocorrem cerca de $2 \times \text{RTT}$ (*round Trip Time*) mais tarde do que a transmissão original, ou mais tarde, se o controle de fluxo ou congestionamento está envolvido como no caso do TCP. Quando um vídeo é recebido após o tempo que era obrigado a ser decodificado antes de ser exibido ele é descartado. Um *frame* tem um limite de tempo (*deadline*) em que este deve chegar ao receptor e ser decodificado. Quando um *frame* é descartado a qualidade do vídeo apresentado diminui, e os quadros que dependiam do quadro perdido também diminuem a qualidade já que sua decodificação é incompleta. *Frames* que chegam antes do seu prazo de codificação, são armazenados no *buffer* até que seja necessário. O número de *frames* no *buffer* deve ser o

menor possível, para minimizar o uso de memória, e grande o bastante que não preencha totalmente o *buffer*, pois quando totalmente preenchido os próximos quadros podem ser descartados. Este *buffer* pode ser utilizado também para reordenar a sequência de codificação.

3.13 MPEG4

MPEG4 (*Moving Picture Experts Group* , padrão número 4) é um padrão para áudio e vídeo (AV) de acordo com a ISO (*International Standards Organization*).

Criado em 1988 e prorrogado em 2003, MPEG4 absorve muito das funcionalidades do MPEG1, MPEG2 e outros padrões relacionados, adicionando novas funcionalidades como suporte ao VRML (*Virtual Reality Modeling Language*), melhora na compressão e qualidade. A norma definiu o *bitstream*, e não como o vídeo é codificado. Isto permite muitas implementações e abordagens diferentes para a codificação, em vista de uma melhor qualidade para uma determinada taxa de *bits*. Para reduzir os custos, são escolhidos subconjuntos de características que serão utilizados com o intuito de reduzir a complexidade na codificação e/ou decodificação. O MPEG4 é ainda um padrão em desenvolvimento e está dividido em várias partes. Inicialmente o MPEG4 era destinado a vídeos de baixo *bit-rate*, entretanto, sua abrangência foi expandida posteriormente para muito mais que um padrão de codificação multimídia. O MPEG4 é eficiente através de uma variedade de *bit-rates* indo desde poucos *kilobits* por segundo até dezenas de *megabits* por segundo. MPEG4 fornece as seguintes funcionalidades:

- Eficiência de codificação melhorada
- Possibilidade de codificação de diferentes media (vídeo, áudio, fala)
- Flexibilidade a erros para possibilitar transmissões robustas
- Possibilidade de interação com a cena áudio-visual gerada através do receptor

3.14 Erro Resiliente e Cancelamento

O padrão MPEG4 especifica uma coleção de métodos para lidar com erros. Estas foram testadas com taxas de erros de até 10^{-3} e erros de rajadas com durações acima de 20 ms.[11]

São usados dois filtros para cancelar os erros de transmissão e codificação. O filtro de desbloqueio é usado para remover a disparidade visual entre blocos vizinhos de imagens, causados pelos Vetores de movimento que não referenciam a área de correspondência e falta de bloco de textura (erro ou não recebido). O filtro *Deringing* contra age o “toque” visual de arestas na imagem pela *Discreta Cosine Transform* (DCT) usada para codificação de textura. Símbolos de sincronização são inseridos em intervalos fixos no fluxo e são usados para recuperar-se de falta ou parte corrompida do fluxo. Padrões mais antigos como H.263 e MPEG2 somente permitem a sincronização no *Frame I*, já que contem dados para cada *pixel* independente do *frame* anterior. A codificação utilizada no MPEG-4 permite a previsão dos dados desaparecidos, ou localizar o início dos próximos blocos de dados que pode ser decodificado, tais como parte de um *frame* ou o próximo *frame*.

MPEG-2 acrescentou o conceito de camadas de melhoria, utilizados para melhorar a qualidade de vídeo, adicionando informações ao fluxo. Quando necessário, um servidor de vídeo complexo poderia removê-las para reduzir a taxa de *bits* para um cliente. MPEG-4, introduz camadas de fino grão acessório, camadas estas de reforço, que podem ser truncadas em qualquer ponto da camada e ainda ser camadas válidas. Isto dá maior flexibilidade para o servidor de vídeo ou serviço de processamento para fornecer uma gama de taxa de dados com resoluções diferentes, e não apenas uma ou duas. O MPEG tem produzido vários padrões de vídeo internacional, integrando os dois meios de compressão, espacial e temporal. Compressão espacial (ou redundância espacial) leva vantagem de *pixels* vizinhos que são semelhantes. Compressão temporal (ou redundância) aproveita a semelhança de quadros sucessivos. Os padrões populares para a entrega de vídeo são:

- MPEG-2 – DVD e SVCD (super vídeo *Compact Disc* – predecessor de DVD) e

- MPEG-4 – fornece *bit-rate* baixo para alcançar maior compressão, o que minimiza o espaço de armazenamento.

3.15 H.264

O padrão H.264, é também conhecido como MPEG-4 parte 10 da ISO, e *Advanced Video Codec (AVC)* do *Joint Video Team (JVC)* de ambas as comissões ITU-T e ISO. A norma criada em 2003 é baseada no padrão MPEG4 com algumas melhorias e é implementada como um perfil de MPEG4. H.264 especifica uma previsão de movimento para diferentes vetores de movimento que produz melhor compressão e qualidade na mesma taxa de transmissão. Quadros B generalizados foram adicionados para referenciar qualquer quadro I ou P recente (dentro de limites do frame) para a previsões a serem feitas a partir de, em vez de apenas os últimos (e seguinte) P ou I *frame*. O padrão foi ajustado para dar melhor qualidade a baixas taxas de *bits*.

3.16 YUV

YUV é o formato RAW sem compressão intermediário para vídeo, que utiliza uma cor equivalente a um esquema de representação para percepção humana. YUV pode representar ambas as variantes, digitais e analógicas. É comumente representado em duas interfaces em diferentes formatos utilizados para padronizar resoluções horizontais e verticais: *Interface Comum format (CIF)* e *Quarter Interface Format (QCIF)*. As características de YUV são:

- Y Representa a luminância – o brilho (o preto e branco) do *frame*
- U Representa um dos valores de cromaticidade (cor) usado para fornecer o espectro de cores do vídeo. Na maioria dos casos refere-se ao espectro azul.
- V Representa o valor de cromaticidade que na maioria dos casos refere-se ao espectro vermelho.

3.17 Resolução, Relação de Aspecto e Taxa de Quadros

Com vídeo digital, além da qualidade de compressão de um fator, outros parâmetros importantes incluem a resolução (dimensões do *pixel* do vídeo), a taxa de quadros (número de quadros por segundo) e a relação de aspecto (as dimensões da tela de vídeo/espectador). Um *pixel* representa um elemento de imagem (*pixel element*) definido por uma cor. Uma imagem é um conjunto de *pixels* organizados em uma matriz visual. Quanto maior o número de *pixels* numa área, maior será a qualidade de imagem. Os vídeos são reproduzidos sequencialmente com imagens sucessivas que seguem a mesma estrutura. Outro aspecto de resolução de vídeo é a atualização do vídeo que são normalmente aplicáveis a tecnologias de televisão com estreita largura de banda. Isto pode ser entrelaçado (atualização ocasional de linhas separadamente) ou progressivo (atualizar todas as linhas na tela) de uma vez somente.

A resolução é definida como um conjunto horizontal de *pixels* versus o conjunto vertical de *pixels*. Quanto maior a resolução, maior o espaço de armazenamento necessário porque é necessário armazenar cada *pixel* e a sua informação. Entretanto, a resolução do monitor também pode determinar a proporção ideal para visualização do vídeo. A proporção é dada pelas as dimensões da tela / espectador e é uma medida do tamanho da tela de visualização. Esta é uma medida de uma relação, por exemplo 4:3 implica na relação entre o comprimento horizontal e comprimento vertical é de 4 a 3.

Existe uma relação entre a largura de banda utilizada e a densidade de movimento dentro de uma cena de vídeo (cena em movimento).

A tabela 8 mostra algumas das resoluções populares acompanhado por razões de seu aspecto, e exemplos de aplicações.

Tabela 8- Exemplo de Resolução e Relação de Aspecto [12].

Resolução	Relação de Aspecto	Termo	Exemplo de Aplicação
320x200	8:5	CGA	mp3 players, cellphones
320x240	4:3	QVGA	PDA's
640x460	4:3	VGA	Portable Game Console
720x480	3:2	NTSC	American TV Monitor
768x576	4:3	PAL	European TV Monitor
800x600	4:3	SVGA	Old 15Monitor
1280x720	16:9	HD 720	HD TV Screen
1280x800	8:5	WXGA	Wide Screen Monitor
1280x1024	5:4	SXGA	Standard 19LCD Monitor
1680x1050	8:5	WSXGA+	HI-RES Monitor
1920x1080	16:9	HD 1080	Full HD TV

3.18 PSNR – Avaliação de Qualidade

A qualidade de transmissão de vídeo depende da impressão de como um observador humano recebe o vídeo entregue. Os resultados dos testes subjetivos de vídeo de qualidade são expressos, por exemplo, por meio de pontuação média de opinião (MOS) como definido pela ITU e mostrado na Tabela 9.[13]

Tabela 9 - Escala de Qualidade - ITU - Escala de Imparidade [13].

Escala	MOS	Imparidade	PSNR [dB]
5	Excelente	Imperceptível	> 37
4	Bom	Perceptível, mas não irritante	31 - 37
3	Justo	ligeiramente irritante	25 - 31
2	Pobre	Irritante	20 - 25
1	Ruim	Muito Irritante	< 20

Basicamente, as métricas de qualidade objetiva de vídeo são calculadas pelo pixel métrica baseadas em SNR ou PSNR e métrica psico-visual. A Equação 1 é a definição do PSNR entre a componente de luminância da fonte de imagem S e imagem de destino D.

Equação 1-[12] Definição do PSNR entre componente de luminância da fonte de imagem S e imagem de destino D

$$PSNR(n)_{dB} = 20 \log_{10} \left(\frac{V_{pico}}{\sqrt{\left(\frac{1}{N_{col}N_{col}}\right) \sum_{i=0}^{N_{col}} \sum_{j=0}^{N_{linha}} [Y_s(n, i, j) - Y_D(n, i, j)]^2}} \right)$$

onde:

$V_{pico} = 2^k - 1$
 N = Vídeo Frame
 V = Máximo possível valor de pixel
 N_{col} = Número de colunas de pixels sem frame
 N_{linha} = Número de linhas de pixels sem frame
 k = Número de bits por pixels (componente de luminância)
 i, j = combinação de pixel por frame

A parte sob a fração nada mais é do que o erro quadrado médio (MSE). Assim a fórmula para PSNR pode ser abreviado como $PSNR = 20 \log (V_{pico} / MSE)$.

Quando o PSNR é calculado quadro a quadro, para vídeos consistindo de milhares de quadros esta fórmula pode ser inconveniente. Assim, podemos obter o PSNR, comparando o vídeo recebido (possivelmente distorcida) com o vídeo sem distorções e comparando as médias dos desvios padrão. Outra opção é calcular primeiro o MOS (*Mean Opinion Score*) e calcular o percentual de quadros com MOS pior do que o vídeo enviado (sem distorções). Este método tem a vantagem de mostrar claramente a distorção causada pela rede em um relance.

3.19 A Ferramenta Evalvid

A ferramenta Evalvid [14] é um projeto *open source* que suporta a geração de arquivos de rastreamento de vídeo MPEG4, H.263, bem como vídeo H.264. Evalvid é adotado juntamente com o NS-2 através da interface sugerida por Ke – C.H. [15]. A qualidade é percebida e medida objetivamente incluindo o PSNR – *Peak Signal-to-Noise Ratio*. O cálculo do índice (PSNR) pode ser obtido após a simulação da rede.

A ferramenta Evalvid, além de permitir o rastreamento de vídeos, permite avaliar a qualidade de transmissão, através de medidas como MOS, PSNR, e *jitter*.

Na tabela 10, mostramos as ferramentas Evalvid[14] utilizadas no processo de simulações.

Tabela 10 - Ferramentas Evalvid; Pré-processo, pós-processo[14].

Ferramenta	Propósito
<i>xvid_encraw.exe</i>	Pré-processo: codificação de vídeo formato YUV em vídeo MPEG4. Cria vídeos com <i>x frames</i> por segundo, GOP de <i>x</i> quadros e nenhum <i>frame B</i> .
<i>MP4Box.exe</i>	Pré-processo: cria arquivo de vídeo ISO MP4 contendo as amostras (<i>frames</i>) e descreve como empacotar os quadros para o transporte com a RTP (<i>Real-time Transport Protocol</i>).
<i>etmp4.exe</i>	Pós-processo: reconstrução do vídeo transmitido como é visto pelo receptor. Gera arquivo de vídeo (possivelmente corrompido), onde todos os quadros corrompidos são excluídos da faixa de vídeos originais. Inserções do último quadro decodificado com sucesso são inseridos devido a queda ou atrasos na chegada dos <i>frames</i> .
<i>ffmpeg4</i>	Pós-processo: decodifica o vídeo recebido em formato YUV. O arquivo resultante YUV contém exatamente o mesmo número de <i>frames</i> do arquivo original YUV.
<i>psnr.exe</i>	Pós-processo: Calcula o PSNR
<i>mos.exe</i>	Pós-processo: mapeia valores de MOS de PSNR

Na tabela 11, mostramos as ferramentas de integração entre NS-2 e Evalvid desenvolvidas por Ke, C.H. [15] utilizadas no processo de simulações.

Tabela 11 - simulação NS-2, [15].

Ferramenta	Propósito
<i>MyTrafficTrace</i>	<i>MyTraffic Trace</i> é empregada para extrair o tipo de frame e o tamanho do frame do arquivo de rastreamento de vídeo gerado a partir da saída do componente anterior. Além disto este agente fragmenta os <i>frames</i> de vídeo em segmentos menores que são enviados para a camada inferior do UDP no momento apropriado de acordo com as configurações do usuário especificado no arquivo de <i>script</i> de simulação.
<i>MyUDP</i>	<i>MyUDP</i> é uma extensão do agente UDP, que permite aos usuários especificar o nome do arquivo de saída do arquivo de rastreamento do remetente, e registra o <i>timestamp</i> de cada pacote transmitido, o ID do pacote, tamanho e carga do pacote. Corresponde à tarefa que ferramentas como <i>TCPdump</i> ou <i>win-dump</i> fazem em ambiente de rede real.
<i>MyUDPSink</i>	<i>MyUDPSink</i> é o agente de recepção para os pacotes de vídeo fragmentado quadro enviado por <i>MyUDP</i> . Este agente também registra o <i>timestamp</i> , ID de pacote, <i>payload</i> e tamanho de cada pacote recebido no arquivo de usuário especificado

3.20 Considerações Finais do Capítulo

Neste capítulo, relembra-se os protocolos básicos utilizados na internet. Ressalta-se as características do protocolo RTP, que fornece facilidade para compensação do *jitter* e detecção de chegada fora de sequência. O RTP incluem o *timestamp* (para sincronização) e número de sequência (para perda de pacote e detecção de re-ordenamento), e o formato da carga útil, que indica o formato de codificação dos dados.

Na maioria dos sistemas de compressão de vídeo, existem três tipos de quadros, aqueles que têm apenas compressão intra-frames (I-Frames), aqueles que estão previsto a partir de um quadro anterior (P-Frames), e os bi-predictos (bidirecionalmente previstos) a partir de um quadro anterior e um futuro quadro (B-Frames).

O padrão MPEG4 especifica uma coleção de métodos para lidar com erros. Estas foram testadas com taxas de erros de até 10^{-3} e erros de rajadas com durações acima de 20 ms. Símbolos de sincronização são inseridos em intervalos fixos no fluxo e são usados para recuperar-se de falta ou parte corrompida do fluxo. A codificação utilizada no MPEG-4 permite a previsão dos dados desaparecidos, ou localizar o início dos próximos blocos de dados que pode ser decodificado, tais como parte de um *frame* ou o próximo *frame*.

É importante ressaltar a importância das ferramentas Evalvid no processo de simulação e as adaptações no NS-2 sugeridas por Ke, C.H. Por exemplo, o *etmp4.exe*, utilizado para reconstrução do vídeo transmitido como é visto pelo receptor. Gera arquivo de vídeo (possivelmente corrompido), onde todos os quadros corrompidos são excluídos da faixa de vídeos originais. Inserções do último quadro decodificado com sucesso são inseridos devido a queda ou atrasos na chegada dos frames. Devido a esta etapa, nos resultados de simulações observaremos vídeos com altas taxas de erros, onde recuperamos o vídeo.

4 METODOLOGIA

4.1 Considerações Gerais

Atualmente se verifica um aumento exponencial da convergência de redes de computadores, dispositivos eletroeletrônicos e aparelhos celulares nos meios de comunicações. Desta forma surge um novo ambiente de entretenimento digital que permite o uso múltiplo de stream de vídeo, áudio e outras mídias que integram o referido ambiente. A correta parametrização lógica das redes que fazem parte deste novo cenário é fator fundamental para obter o máximo de desempenho com a menor quantidade de recursos (hardware/software).

O *streaming* de mídia pode ser definido como uma forma de distribuir informações multimídia em uma rede de dados através de pacotes. É composto de vários componentes que funcionam de forma lógica para proporcionar um fluxo contínuo de dados de uma fonte (ou fontes) para um ou mais pontos finais (destinos).

Este trabalho é baseado em estudos e interpretação de resultados de simulações de parametrizações do protocolo UDP realizadas em uma topologia de rede já consagrada no meio científico. As simulações foram feitas com o objetivo geral investigar o comportamento da qualidade e desempenho de uma transmissão de um *stream* de vídeo variando a parametrização da rede UDP. Em conjunto, tenta-se verificar o melhor conjunto de parâmetros para tal transmissão.

A ferramenta de simulação utilizada é um software de domínio público, o NS-2. Em conjunto com o NS-2, a avaliação da transmissão do vídeo é feita utilizando a ferramenta Evalvid [14]. Com base neste ambiente integrado, realiza-se uma série de experimentos de diferentes tipos de características e cenários observando métricas como: banda, perda, tráfego de *background* e variação do tamanho de *frames* do UDP na topologia de teste escolhida.

4.2 Trabalhos Relacionados

Em Khater [13] verifica-se a avaliação e simulação da transmissão de vídeo MPEG em redes sem fio de malha com equipamentos de banda ultra-larga. O objetivo foi realizar uma investigação abrangente fim a fim do QoS de transmissão de vídeo através de uma rede UWB (Ultra Wide Band) , sem fio e com topologia fixa. Como resultado da pesquisa, estabeleceram-se limites empíricos superiores para o número de fluxo de dados e mistura de fluxo de dados que podem render uma qualidade satisfatória para uma determinada densidade de nós de rede. O ambiente de avaliação é formado pelo simulador o NS-2 e o novo módulo Evalvid desenvolvido para modelar redes multi-percurso sem fio.

Em Gavriloff [16] implementa-se uma técnica de selecionar o caminho de menor latência para proporcionar uma melhora na qualidade de transmissão de dados. Nesta pesquisa, também foi utilizado o *software* de simulação NS-2 (*Network Simulator 2*) para implementação e simulação da técnica proposta.

Podemos citar ainda o “*Comparative Study of Real-Time Multimedia Transmission over Multi-homing Transport Protocols*” de Changqiao Xu [16], que mostra o desempenho do sistema Evalvid-SCTP – análise do desempenho de transmissão em tempo real de multimídia através de protocolos *mult-homing* de transporte de rede tolerante a falhas. Este estudo concentra-se nas três políticas adotadas pelo protocolo PR-SCTP (*Partial-Reliable SCTP*) de retransmissão [17]:

- 1) *AllRtxSame*: todas as retransmissões para o mesmo caminho
- 2) *AllRtxAlt* : todas as retransmissões pelo caminho alternativo e
- 3) *FrSameRtoAlt*: retransmissões rápidas pelo mesmo caminho, e retransmissões por limite de tempo pelo caminho alternativo.

Os resultados deste experimento mostraram que para a configuração com PMR=0 (sem retransmissões ou *Path.Max.Retrans*=0) para ambientes com falhas agressivas, o caminho simétrico tem um melhor desempenho, no entanto, em ambientes menos agressivos, a configuração PMR=1 apresentou melhor desempenho para caminhos

assimétricos. Retransmissões de todos os vídeos com caminhos alternativos com PMR=0 apresentou melhor desempenho em caminhos simétricos. Ou seja, a escolha adequada dos parâmetros do SCTP pode melhorar sua performance em cenários com perdas, sugerindo valores baixos para valores de PMR (0 ou 1) para transferência de dados. Caro[18], em 2006, também chega à mesma conclusão e ainda propõe uma política alternativa de retransmissão. Apesar dos falsos-positivos para falhas causadas por valores mais baixos de PMR, o aumento na velocidade de reação a falhas de caminho compensa e garante um ganho de vazão ou *throughput*.

O trabalho aqui desenvolvido se diferencia dos trabalhos citados, na obtenção de uma melhor qualidade de vídeo utilizando um protocolo simples, bastante conhecido e muito utilizado como o UDP. Além disso, é sugerida a utilização de multipercurso através da configuração do sistema no nível de aplicação e não no nível de protocolo como fazem os demais trabalhos. Ou seja, propõe-se obter resultados melhores, apenas melhorando a eficácia da solução de um problema conhecido.

4.3 Classificação da Pesquisa

A abordagem apresentada a cerca da pesquisa, é a forma clássica descrita a seguir:

- Do ponto de vista da natureza, trata-se de uma pesquisa aplicada, pois gera conhecimento para sua aplicação prática, dirigido a solução de problemas específicos.
- Do ponto de vista da forma de abordagem do problema, trata-se de uma pesquisa quantitativa com avaliação formal dos resultados;
- Do ponto de vista dos objetivos, trata-se de uma pesquisa descritiva, pois envolve coleta de dados com uso de técnicas padronizadas pelo IETF, assumindo uma forma de levantamento.

4.4 Etapas da Pesquisa

4.4.1 Escolha do Software de Simulação

Para configurar uma rede real para testes de rede multimídia requer um esforço e investimentos consideráveis além de oferecer pouca flexibilidade e escalabilidade. Em vez de utilizar emulações ou medição em rede, foi utilizado a ferramenta NS-2 [1], que possibilita a construção de redes eficazes personalizados, flexíveis, a um custo baixo, em tempo hábil em seus testes em detalhes.

Existem no mercado vários simuladores disponíveis para uso. A seguir, segue uma visão geral de alguns simuladores populares atualmente disponíveis:

- CSIM-19 - orientado simulador de eventos discretos implementados em C / C ++, e desenvolvida principalmente para a modelagem de sistemas completos. Produto comercial.
- OPNET - é usado principalmente no desenvolvimento de redes de comunicação, tecnologias redes e protocolos por engenheiros do mundo de redes. Produto comercial.
- QualNet - oferece tanto de simulação como emulação de rede. Oferece meio ambiente portátil e escalável para a implantação de redes virtuais. Representa com precisão um clone digital com condições da rede física.
- NS-2 - Amplamente utilizado em ambientes de pesquisa acadêmica. Documentação disponível na Internet. Extensível acesso a fonte Kernel, permite alterar protocolos existentes e criar novos.

Motivo da escolha do software NS-2:

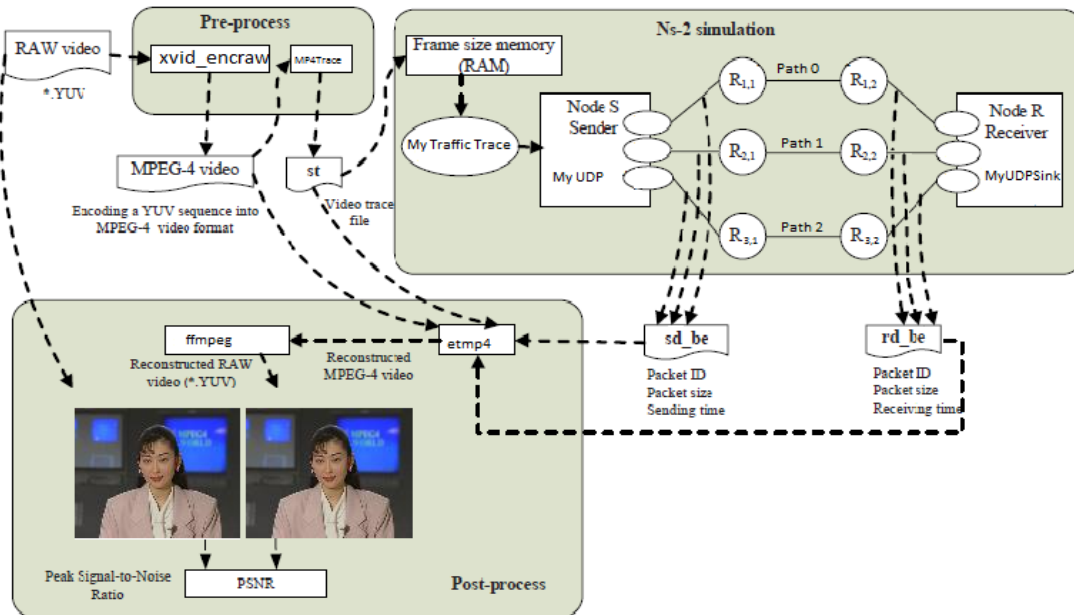
1. Valor do investimento

2. Flexibilidade
3. Escalabilidade
4. Tempo

4.4.2 Sistema de Simulação Utilizando a Ferramenta Evalvid

A Figura 2 mostra o processo de simulação utilizando a ferramenta Evalvid e NS-2 utilizado no processo de simulação. Descrição em detalhes de cada etapa será explicado no item 3.4.4.

Figura 2 - Pré-processo, simulação NS-2, Pós-processo Evalvid



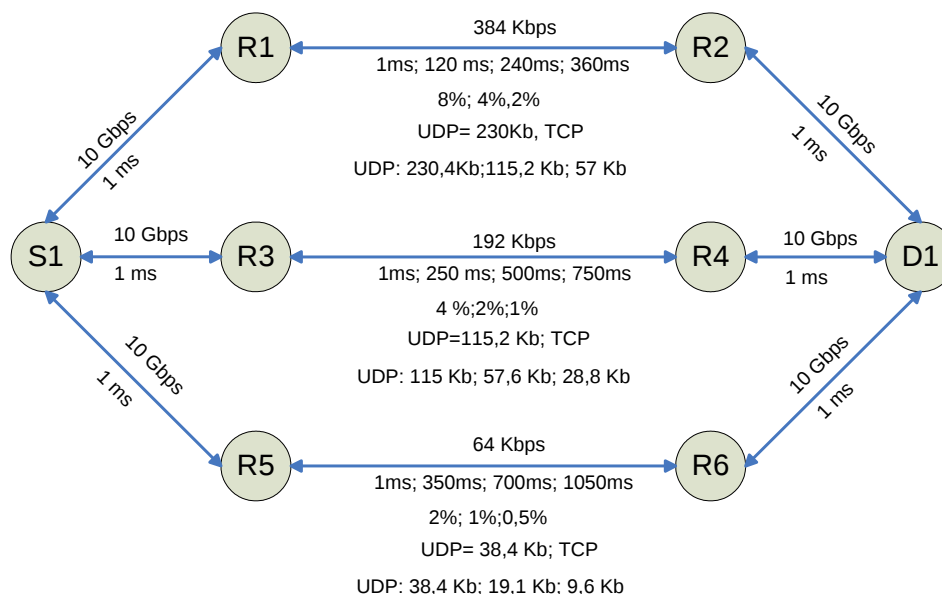
4.4.3 Ambiente de Laboratório

Para a realização do experimento, projetamos uma solução de vídeo MPEG4 para investigação do desempenho da transmissão multi-percurso UDP em NS-2 [19], baseado nas modificações Evalvid [15], onde transmitimos não só em uma via, mas em vários,

onde simulamos as variações de disponibilidade de banda, taxa de erro, congestionamento e até mesmo a variação do tamanho do *frame*, para verificar a melhor configuração nesta situação.

Apresentamos abaixo a figura 3, onde mostramos a topologia escolhida para o processo de simulação. Esta topologia foi escolhida, pois sintetiza uma rede real, pois apresenta as principais características encontradas numa rede, por exemplo a internet, onde temos possibilidade de conexões do ponto de origem ao destino utilizando mais de um percurso, sendo em cada percurso uma banda disponível devido a tráfegos concorrentes existentes nas redes e atrasos e taxas erros diferentes. Para verificarmos o desempenho nas condições adversas, configuramos a rede com diversos atrasos, bandas e tráfego concorrente, além de variação no tamanho do *Frame* do UDP.

Figura 3 - Topologia de rede utilizado para as simulações



4.4.4 Procedimento de Experimentação

Os experimentos foram executados em ambientes:

- Linux: simulação da topologia da rede, com transmissão de vídeo, obtendo uma tabela com tempo do envio, tamanho do pacote, tipo do pacote e tempo de recepção – figura 3;
- Windows: embaralhamento do vídeo, para melhor distribuição das perdas;
- e ambiente Linux no Windows, cygwin [20]: avaliação da qualidade no receptor, mapeia os valores de MOS, PSNR e decodifica o vídeo recebido em formato YUV.

Para a preparação do vídeo, foi escolhido o vídeo *akiyo_cif.yuv* (352x288) [21].

Foram em seguida executados os seguintes passos:

1. Codificou-se o arquivo *yuv* no formato MPEG. Criou-se o vídeo com uma taxa de 25 quadros por segundo, um comprimento do GOP com quadros de 4, 6, 8 e 16 quadros sem nenhum frame B, pois o controle da taxa de bits do XviD não funciona, portanto foi omitido. [22]
2. Criou-se arquivo ISO MP4 de vídeo com amostras de *frames* e descrição de como empacotar os quadros para transporte com RTP (*Real-time Transport Protocol*).
3. Com a ferramenta *Evalvid mp4trace* especificou-se o destino (*host*) do arquivo mp4 sugerido por RTP/UDP.
4. O vídeo foi dividido em 3 partes nas proporções 3; 2;1 de GOP intercaladamente, para transmissão em três vias paralelamente.
5. A topologia foi criada e simulada para a transmissão do vídeo MP4.
6. O vídeo foi remontado, de acordo com a sequência de chegada do *frame*.
7. O PSNR e MOS de cada *frame* de vídeo recebido foi calculado e comparado para cada *frame* recebido com o vídeo original.

4.4.5 Descrição dos Cenários

Os experimentos são compostos de cenários onde variamos os seguintes fatores:

- Tamanho do Frame do UDP: 256, 512 e 1024
- Tamanho do GOP: 4, 6, 8 e 16

Estas variações de parâmetros são aplicadas em 12 situações de redes distintas, onde são simulados fatores de banda, delay de transmissão e quantidade de vias físicas. Estas situações foram escolhidas para varrer uma gama suficiente de situações de redes permitindo uma análise mais aprofundada do sistema com um todo. No total foram simulados 144 (cento e quarenta e quatro) cenários, como detalhado a seguir:

- **Cenário: UDP –Frame 1024 Bytes**

Foi escolhido um tamanho do quadro do UDP de 1024 *bytes*, acrescido do cabeçalho UDP – de 8 *bytes* - e acrescido do cabeçalho IP – com 20 *bytes*.

Os experimentos foram conduzidos com uma taxa de transmissão de 25 *frames* por segundo e:

➤ **GOP = 4**

1. 1 (uma) via de transmissão sem perda, banda 384 Kbps e *delay* 1ms.
2. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 120 ms; a segunda via com 198Kbps, *delay* 250 ms e a terceira via com 64 Kbps, *delay* 350 ms.
3. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 240 ms; a segunda via com 198Kbps, *delay* 500 ms e a terceira via com 64 Kbps e *delay* 700 ms.

4. 3 (três) vias de transmissão com *delay's* diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 360 ms; a segunda via com 198 Kbps, *delay* 750 ms e a terceira via com 64 Kbps, *delay* 1050 ms.
5. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 8%; a segunda via com 198 Kbps, *delay* 250 ms, perda 4% e a terceira via com 64 Kbps, *delay* 350 ms, perda 2%.
6. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 4%; 2) via com 198Kbps, *delay* 250 ms, perda 2%; 3) via com 64 Kbps, *delay* 350 ms, perda 1%.
7. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 2%; a segunda via com 198Kbps, *delay* 250 ms, perda 1% e a terceira via com 64 Kbps, *delay* 350 ms, perda 0,5%.
8. 3 (três) vias com tráfego background , sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego TCP e UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego TCP e UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego TCP e UDP com *rate* 38,4 Kbps,
9. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 115,2 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 57,6 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 19,2 Kbps.
10. 3 (três) vias com tráfego background sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 57,6 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 28,8 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 9,6 Kbps;

11. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 38,4 Kbps;
12. 3 (três) vias de transmissão sem perdas, a primeira via com banda de 384 Kbps, *delay* 1 ms; a segunda via com 198 Kbps, *delay* 1 ms e a terceira via com 64 Kbps, *delay* 1 ms;

➤ **GOP = 6**

1. 1 (uma) via de transmissão sem perda, banda 384 Kbps e *delay* 1ms.
2. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 120 ms; a segunda via com 198Kbps, *delay* 250 ms e a terceira via com 64 Kbps, *delay* 350 ms.
3. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 240 ms; a segunda via com 198Kbps, *delay* 500 ms e a terceira via com 64 Kbps e *delay* 700 ms.
4. 3 (três) vias de transmissão com *delay's* diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 360 ms; a segunda via com 198 Kbps, *delay* 750 ms e a terceira via com 64 Kbps, *delay* 1050 ms.
5. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 8%; a segunda via com 198 Kbps, *delay* 250 ms, perda 4% e a terceira via com 64 Kbps, *delay* 350 ms, perda 2%.

6. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 4%; 2) via com 198Kbps, *delay* 250 ms, perda 2%; 3) via com 64 Kbps, *delay* 350 ms, perda 1%.
7. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 2%; a segunda via com 198Kbps, *delay* 250 ms, perda 1% e a terceira via com 64 Kbps, *delay* 350 ms, perda 0,5%.
8. 3 (três) vias com tráfego background , sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego TCP e UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego TCP e UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego TCP e UDP com *rate* 38,4 Kbps,
9. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 115,2 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 57,6 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 19,2 Kbps.
10. 3 (três) vias com tráfego background sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 57,6 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 28,8 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 9,6 Kbps;
11. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 38,4 Kbps;

12. 3 (três) vias de transmissão sem perdas, a primeira via com banda de 384 Kbps, *delay* 1 ms; a segunda via com 198 Kbps, *delay* 1 ms e a terceira via com 64 Kbps, *delay* 1 ms;

➤ **GOP = 8**

1. 1 (uma) via de transmissão sem perda, banda 384 Kbps e *delay* 1ms.
2. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 120 ms; a segunda via com 198Kbps, *delay* 250 ms e a terceira via com 64 Kbps, *delay* 350 ms.
3. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 240 ms; a segunda via com 198Kbps, *delay* 500 ms e a terceira via com 64 Kbps e *delay* 700 ms.
4. 3 (três) vias de transmissão com *delay's* diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 360 ms; a segunda via com 198 Kbps, *delay* 750 ms e a terceira via com 64 Kbps, *delay* 1050 ms.
5. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 8%; a segunda via com 198 Kbps, *delay* 250 ms, perda 4% e a terceira via com 64 Kbps, *delay* 350 ms, perda 2%.
6. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 4%; 2) via com 198Kbps, *delay* 250 ms, perda 2%; 3) via com 64 Kbps, *delay* 350 ms, perda 1%.
7. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 2%; a segunda via com 198Kbps, *delay* 250 ms, perda 1% e a terceira via com 64 Kbps, *delay* 350 ms, perda 0,5%.

8. 3 (três) vias com tráfego background , sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego TCP e UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego TCP e UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego TCP e UDP com *rate* 38,4 Kbps,
9. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 115,2 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 57,6 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 19,2 Kbps.
10. 3 (três) vias com tráfego background sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 57,6 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 28,8 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 9,6 Kbps;
11. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 38,4 Kbps;
12. 3 (três) vias de transmissão sem perdas, a primeira via com banda de 384 Kbps, *delay* 1 ms; a segunda via com 198 Kbps, *delay* 1 ms e a terceira via com 64 Kbps, *delay* 1 ms;

➤ **GOP = 16**

1. 1 (uma) via de transmissão sem perda, banda 384 Kbps e *delay* 1ms.

2. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 120 ms; a segunda via com 198Kbps, *delay* 250 ms e a terceira via com 64 Kbps, *delay* 350 ms.
3. 3 (três) vias de transmissão com *delay's* diferentes sendo a primeira via com banda de 384 Kbps, *delay* 240 ms; a segunda via com 198Kbps, *delay* 500 ms e a terceira via com 64 Kbps e *delay* 700 ms.
4. 3 (três) vias de transmissão com *delay's* diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 360 ms; a segunda via com 198 Kbps, *delay* 750 ms e a terceira via com 64 Kbps, *delay* 1050 ms.
5. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 8%; a segunda via com 198 Kbps, *delay* 250 ms, perda 4% e a terceira via com 64 Kbps, *delay* 350 ms, perda 2%.
6. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 4%; 2) via com 198Kbps, *delay* 250 ms, perda 2%; 3) via com 64 Kbps, *delay* 350 ms, perda 1%.
7. 3 (três) vias de transmissão com perdas diferentes, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, perda 2%; a segunda via com 198Kbps, *delay* 250 ms, perda 1% e a terceira via com 64 Kbps, *delay* 350 ms, perda 0,5%.
8. 3 (três) vias com tráfego background , sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego TCP e UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego TCP e UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego TCP e UDP com *rate* 38,4 Kbps,
9. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 115,2 Kbps; a segunda via

com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 57,6 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 19,2 Kbps.

10. 3 (três) vias com tráfego background sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 57,6 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 28,8 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 9,6 Kbps;

11. 3 (três) vias com tráfego background, sendo a primeira via com banda de 384 Kbps, *delay* 120 ms, tráfego UDP com *rate* 230,4 Kbps; a segunda via com 198 Kbps, *delay* 250 ms, tráfego UDP com *rate* 115,2 Kbps e a terceira via com 64 Kbps, *delay* 350 ms, tráfego UDP com *rate* 38,4 Kbps;

12. 3 (três) vias de transmissão sem perdas, a primeira via com banda de 384 Kbps, *delay* 1 ms; a segunda via com 198 Kbps, *delay* 1 ms e a terceira via com 64 Kbps, *delay* 1 ms;

- **Cenário: UDP Frame 512 Bytes**

Foi escolhido um tamanho do quadro do UDP de 512 *bytes*, acrescido do cabeçalho UDP – de 8 *bytes*- e acrescido do cabeçalho IP – com 20 *bytes*.

Utilizou-se as mesmas condições de rede acima, alterando somente o tamanho do quadro do UDP de 512 *bytes*.

- **Cenário: UDP Frame 256 Bytes**

Foi escolhido tamanho do quadro do UDP de 256 *bytes*, acrescido do cabeçalho UDP – de 8 *bytes*- e acrescido do cabeçalho IP – com 20 *bytes*.

Utilizou-se as mesmas topologias do cenário anterior, alterando somente o tamanho do quadro do UDP 256 *bytes*.

4.4.6 Considerações Finais do Capítulo

Neste capítulo foi descrito a topologia utilizada, o sistema de simulação utilizando Evalvid, as adaptações sugeridas por Ke, C.H.. Descrito o ambiente de laboratório e as configurações de sistemas utilizados para simulação.

Com relação aos trabalhos relacionados, a pesquisa tem relevância, pois, esta pesquisa mostra que a configuração combinadas do UDP e GOP em sistemas multipercurso (a nível de aplicação) alteram o desempenho do sistema de transmissão de *Video Stream*.

Já o trabalho de Khater, pesquisa em uma rede UWB os limites empíricos superiores para o número de fluxo de dados. Gavriloff pesquisou a seleção do caminho de menor latência e Changqiao Xu pesquisou retransmissões rápidas pelo mesmo caminho, e retransmissões por limite de tempo pelo caminho alternativo, todas as retransmissões pelo caminho alternativo e todas as retransmissões pelo mesmo caminho alterando a configuração do protocolo SCTP.

5 AVALIAÇÃO DOS RESULTADOS E DISCUSSÕES

Neste capítulo são apresentados e discutidos os resultados de nosso estudo de transmissão de vídeo usando múltiplos percursos. Após a execução dos experimentos, os dados foram formatados e analisados, procedimento este realizado para cada um dos 144 cenários apresentados: UDP frame size: 256, 512, 1024 e GOP: 4, 6, 8, 16 sendo 12 configurações diferentes por GOP.

5.1 Resultados Obtidos

5.1.1 Cenário: UDP – Frame 1024

Os dez melhores resultados em um total de 48 simulações para UDP configurado para quadros com 1024 bytes; o motivo das perdas em % são elevados elevados, e mesmo assim, ser possível assistir a *vídeo stream* é devido a utilização de *etmp4.exe* e a utilização de RTP. Os resultados foram classificados baseados no MOS, pois apesar da perda ser maior no item 6 do que no item 7, o MOS do item 6 é maior. O motivo da escolha do MOS deve-se ao fato de que a qualidade depende do frame % perdido – I, P ou B.

Tabela 12 - Resultados de simulações pós-processados com NS-2 - UDP 1024 – dez melhores resultados

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
1	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 6	45,70963333	39,10488	14 / 16	15,7	4,61
2	sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 8	45,79896667	34,25445	68.4 / 62.2	63	3,95

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
3	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 8	45,79896667	34,21365	68.4 / 61.8	62,7	3,94
4	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb) UDP 1024; GOP = 8	45,79896667	34,17371	68.4 / 61.8	62,7	3,94
5	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 1024; GOP = 8	45,79896667	34,17371	68.4 / 61.8	62,7	3,94
6	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 8	45,79896667	33,84852	71.1 / 64.9	65,7	3,92
7	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP=8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87
9	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87
10	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 8	45,79896667	33,13428	76.3 / 66	67,3	3,82

Observa-se pelos dez melhores resultados, tendo MOS como referência, que para UDP configurado com quadros de 1024 *bytes*, o melhor desempenho é com GOP = 6, com atraso médio. O mesmo ocorre com o segundo melhor desempenho – GOP = 8, onde foi observado um atraso médio predominando na rede. Além disto, o sexto melhor desempenho é uma rede com perdas, pequenas, mas com perdas, tendo melhor

desempenho do que uma rede sem perdas – oitavo melhor desempenho. Entre o nono e décimo melhor desempenho temos uma rede com maior perda tendo um desempenho ligeiramente melhor. Observa-se que nestes resultados o que predominam os tempos de atrasos menores e $GOP = 8$. Todas as dez topologias, são com três vias de comunicação.

Os gráficos apresentados nas figuras 4, 5 e 6 ilustram os três melhores resultados de desempenho com $UDP = 1024$:

Figura 4 - 3 vias sem perdas com *delay* (240ms;500ms;700ms); $UDP\ 1024$; $GOP = 6$

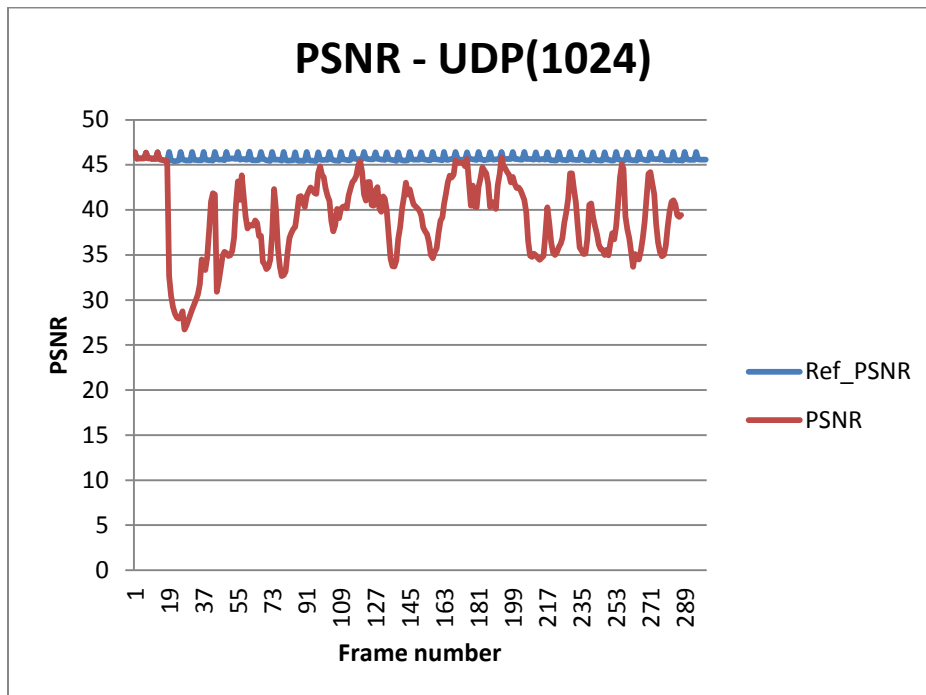


Figura 5 - 3 vias sem perdas com *delay* (240ms;500ms;700ms); UDP 1024; GOP = 8

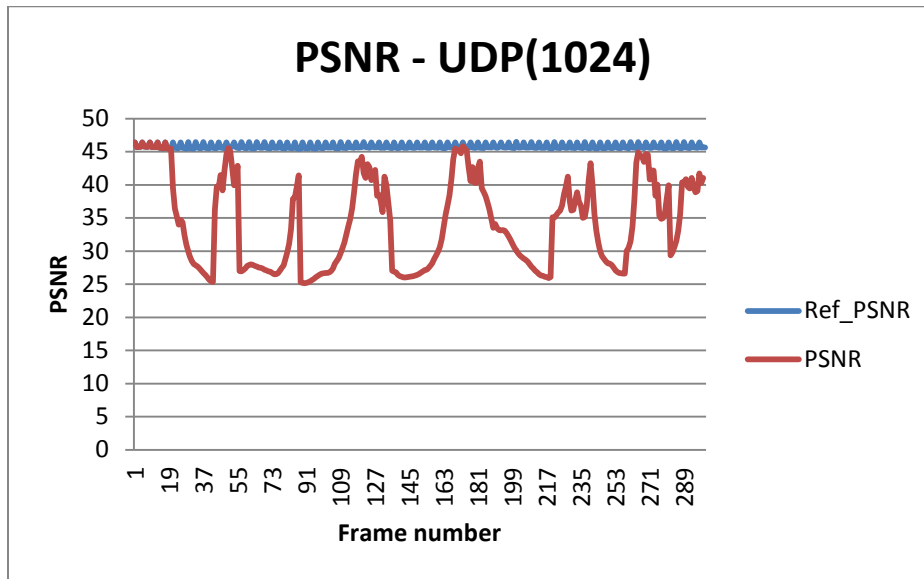
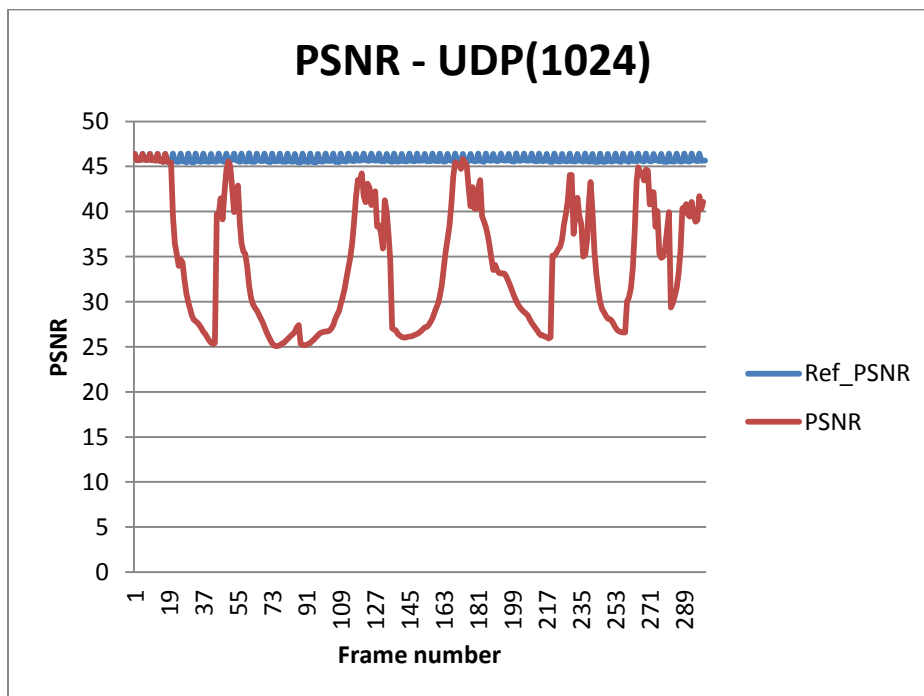


Figura 6 - 3 vias sem perdas com *delay* (360ms;750ms;1050ms); UDP 1024; GOP = 8



Já observando o gráfico de PSNR, observa-se que a configuração GOP=6 apresenta variações muito menores que o GOP =8, tendo uma variação predominante de PSNR de 35 a 45 – de bom a excelente, enquanto o GOP =8 varia o PSNR de 25 a 45 - de pobre a excelente.

5.1.2 Cenário: UDP – *Frame 512*

Os dez melhores resultados no universo de 48 simulações para UDP configurado para quadros com 512 *bytes*; o motivo das perdas em % são elevados elevados, e mesmo assim, ser possível assistir a *vídeo stream* é devido a utilização de *etmp4.exe* e a utilização de RTP. Os resultados foram classificados baseados no MOS. O motivo da escolha do MOS deve-se ao fato de que a qualidade depende do frame % perdido – I, P ou B.

Tabela 13- Resultados de simulações pós-processados com NS-2 - UDP 512 – dez melhores resultados

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
1	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 6	45,70963333	31,11372	86 / 79.6	80,7	3,45
2	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 6	45,70963333	31,08986	86 / 80.8	81,7	3,45
3	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP=6	45,70963333	29,88737	92 / 85.2	86,3	3,34
4	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
5	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb); UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
6	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb); UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
7	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
9	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb); UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
10	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb); UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29

Nesta configuração com UDP configurado com quadros de 512 *bytes* e mantendo MOS como referência, foram obtidos os sete primeiros melhores desempenho para GOP =6 e sendo a melhor rede com maior atrasos e sem perdas. Comparando a rede com tráfego de fundo com TCP, observa-se que houve melhora da oitava (UDP *Frame* = 1024 e GOP =8) posição para a quarta posição (UDP *Frame* 512 e GOP=6) na escala de desempenho.

Os gráficos apresentados nas figuras 7, 8 e 9 ilustram os três melhores resultados de desempenho com UDP = 512:

Figura 7 - 3 vias sem perdas com *delay* (360ms;750ms;1050ms); UDP 512; GOP = 6

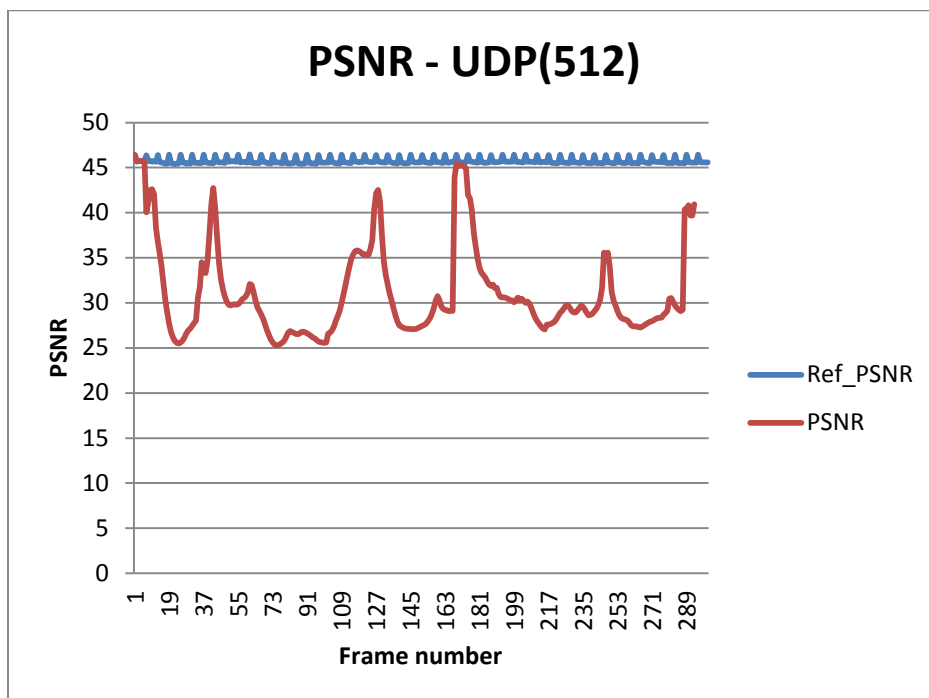


Figura 8 - 3 vias sem perdas com *delay* (240ms;500ms;700ms); UDP 512; GOP = 6

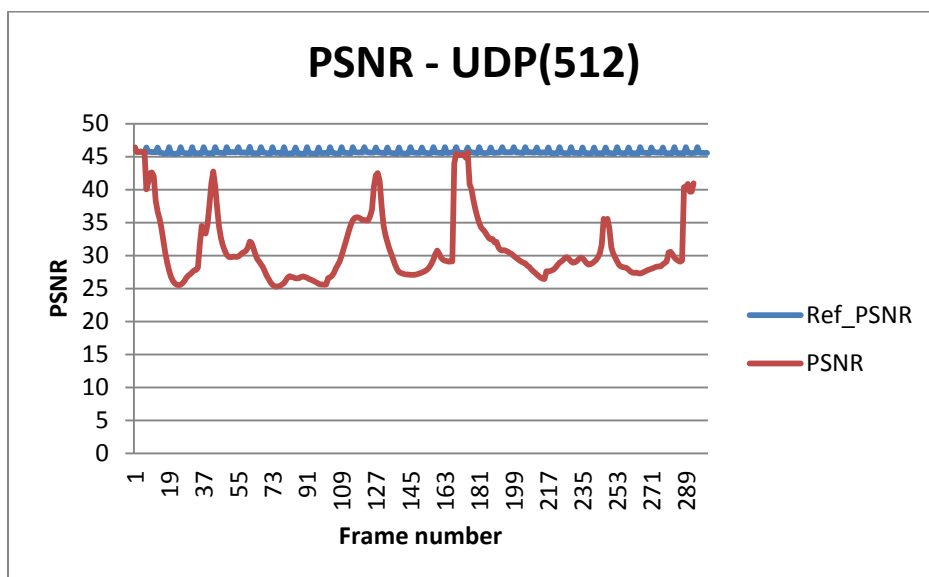
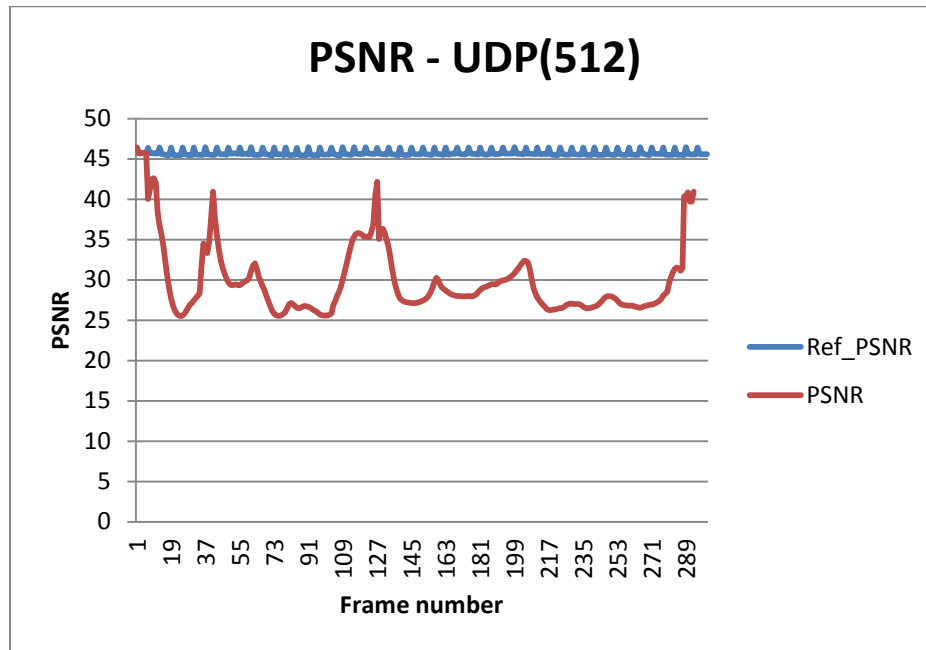


Figura 9 - 3 vias sem perdas com *delay* (120ms;250ms;350ms); UDP 512; GOP=6



Na análise gráfica para UDP configurado com quadros de 512 *bytes*, os comportamentos do gráfico PSNR são muito similares.

5.1.3 Cenário: UDP – *Frame* 256

Os dez melhores resultados no universo de 48 simulações para UDP configurado para quadros com 256 *bytes*; o motivo das perdas em % são elevados elevados, e mesmo assim, ser possível assistir a *vídeo stream* é devido a utilização de *etmp4.exe* e a utilização de RTP. Os resultados foram classificados baseados no MOS. O motivo da escolha do MOS deve-se ao fato de que a qualidade depende do frame % perdido – I, P ou B.

Tabela 14 - Resultados de simulações pós-processados com NS-2 - UDP 256 – dez melhores resultados

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
1	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) udp 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
2	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb); udp 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
3	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb); udp 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
4	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) udp 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
5	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); udp 256; GOP=6	45,70963333	25,03486	92 / 90.4	90,7	3,2
6	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); udp 256; GOP = 4	45,79896667	28,36344	94.7 / 88	89,7	3,09
7	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); udp 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) udp 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
9	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb); udp 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
10	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) udp 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08

Nesta rede com UDP configurado com quadros de 256 bytes tivemos novamente a melhor rede com $GOP = 6$ com tráfego TCP de fundo (UDP Frame 256 e $GOP=6$) aparece agora como o melhor desempenho. Novamente observam-se a predominância de rede com $GOP = 6$ e o $GOP = 4$ aparecendo do sexto ao décimo melhor desempenho.

Os gráficos apresentados nas figuras 10, 11 e 12 ilustram os três melhores resultados de desempenho com UDP = 256:

Figura 10 - 3 vias sem perdas com *delay* (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb); UDP- 512; $GOP = 6$.

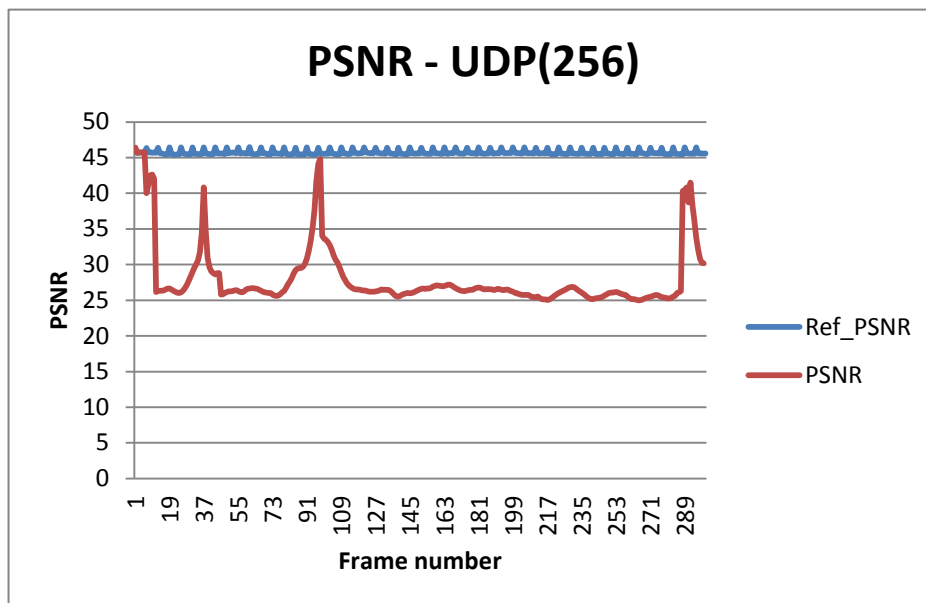


Figura 11 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb); UDP 256; GOP = 6

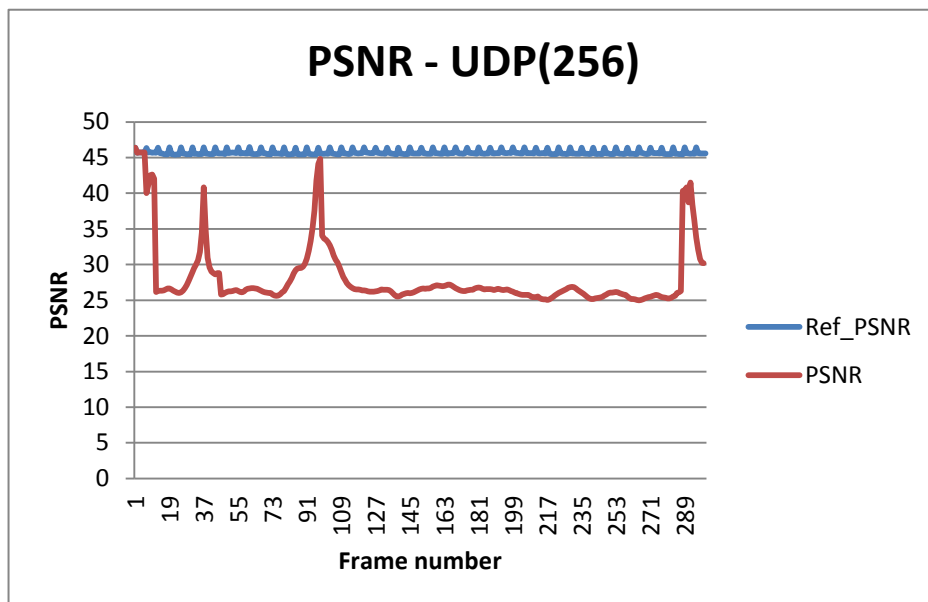
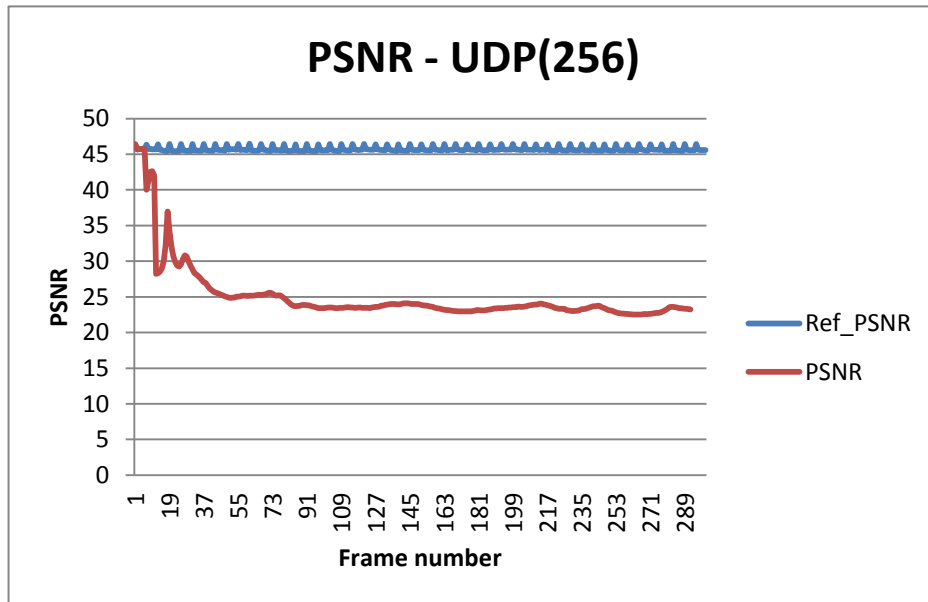


Figura 12 - 3 vias sem perdas com *delay* (120ms;250ms;350ms); UDP 512; GOP=6



Graficamente para redes com UDP configurado com quadros de 256 bytes, o desempenho é muito inferior às demais com média de PSNR de 25.

5.2 Discussão dos Resultados

Diante dos resultados obtidos nas simulações, observou-se que entre os dez melhores desempenhos em cada UDP *Frame size* não aparecem redes simples, o que demonstra que UDP em multipercurso tem um desempenho muito mais elevado. Para UDP de 1024 *bytes*, temos entre os dez melhores desempenho redes com perdas na sexta, nona e décima posição, sendo a nona posição, com perdas elevadas de 8%, 4% e 2%, e tráfego UDP concorrente elevado, o que mostra que redes com UDP multipercurso melhora e altera o desempenho do sistema para transmissão de vídeo.

Para UDP de 512 *bytes* temos na quinta posição, rede com perdas elevadas e tráfego UDP concorrente elevado, o que mostra a melhora no desempenho para as redes com perdas e comprovando que redes UDP Multipercurso melhoram o desempenho.

Também para redes com UDP de 256 *bytes* temos a rede com perdas elevadas e *delay* menor e tráfego elevado UDP concorrente, apresenta o segundo melhor desempenho, ou seja, cada vez que diminuimos o tamanho do Frame UDP, temos uma melhora significativa de desempenho para redes com perdas elevadas e atrasos médios.

Em nenhuma das configurações, redes com apenas um percurso aparecem entre os dez melhores desempenhos. Também não aparecem entre os dez melhores desempenho o GOP = 16.

Para as três configurações de tamanho de UDP temos o melhor desempenho para UDP com rede Multipercurso, para GOP = 6 e com variação no *delay*. Também observou-se que para UDP de 1024 *bytes* o resultado geral é melhor, diminuindo a medida que diminuimos o tamanho do UDP.

Este trabalho foi elaborado em uma rede onde procuramos recriar um ambiente de redes com banda limitada apresentando perdas e atrasos variados, conforme pode ser visto na figura 2 . Em cima deste cenário variamos o GOP de 4, 6, 8 e 16, para verificar a melhor configuração, para uma rede genérica, com características diferentes. Para simular tráfego concorrente, inserimos tráfego UDP, e TCP.

Tendo estes resultados, estende-se a interpretação do resultado para a rede de Internet, onde podemos prever a melhor configuração de vídeo, já na sua criação, para transmissão em redes, utilizando-se dos benefícios de ter uma rede de comunicações de melhor resiliência.

6 CONCLUSÃO

Concluí-se que o tamanho do GOP e da configuração do UDP, influem diretamente no desempenho de transmissões de vídeos. O melhor desempenho obtido foi com UDP-*Frame* 1024 e GOP=6 tendo um percentual de *frame loss* de 15,7% apenas, comparado a aproximadamente 80% ou pior perda em outras configurações. Observa-se ainda que, pelo fato de utilizar multipercurso, e ter o vídeo transmitido de forma embaralhadas pelas rotas alternativas, o gráfico apresenta na maioria dos casos, picos de PSNR.

Como resultado deste trabalho, temos a contribuição de informações para melhorar as configurações de sistemas para transmissões de vídeo, mostrando a importância da configuração.

Como trabalho futuro pode-se sugerir a avaliação do desempenho de transmissão de vídeo sob diferentes topologias e condições de tráfego de fundo para tráfegos de *vídeo stream* em altas resoluções (1090x1080) e HD-TV e utilizando a internet real no sistema de simulação.

7 REFERÊNCIAS

- [1] NS-2, <http://nile.wpi.edu/NS/>
- [2] Protocolos de Comunicação,
[http://pt.wikipedia.org/wiki/Protocolo_\(ci%C3%A7%C3%A3o_da_computa%C3%A7%C3%A3o\)](http://pt.wikipedia.org/wiki/Protocolo_(ci%C3%A7%C3%A3o_da_computa%C3%A7%C3%A3o))
- [3] Protocolo IP, <http://pt.kioskea.net/contents/internet/protip.php3>
- [4] J. Postel. RFC 768: User datagram protocol. In IETF Internet Standards. IETF, August 1980.
- [5] Protocolo UDP, <http://pt.kioskea.net/contents/internet/udp.php3>
- [6] Protocolo RTP, http://en.wikipedia.org/wiki/Real-time_Transport_Protocol#cite_ref-Colins_56_7-0
- [7] Quintero, Nestor Felipe Maya. Middleware para os Protocolos SCTP e Hip. 2007, PUC RJ, 26 págs. Disponível em <http://www-di.inf.puc-rio.br/~endler/courses/Mobile/Monografias/07/Mid.SCTP.HIP-FelipeMaya-mono.pdf>. Acesso em abril 2010.
- [8] Protocolo TCP,
http://pt.wikipedia.org/wiki/Transmission_Control_Protocol#Caracter.C3.ADsticas_t.C3.A9nicas
- [9] Stalvig, Paul. Introdução ao Protocolo de Controle de Fluxo de Transmissão (SCTP), A próxima geração do Protocolo de Controle de Transmissão (TCP). White Paper, 2007, F5 Network. 7 págs, <http://www.f5networks.com.br/pdf/white-papers/introducao-ao-stream-control-transmission-protocol-sctp-wp.pdf>. Acesso abril 2010.
- [10] Begg, C. Lee. High Quality Video Streaming with SCTP over CDMA2000, Canterbury, 2007.

- [11] Moving Picture Experts Group, International Organisation for Standardisation. , November 2005, <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm>. Website.
- [12] Patrick Mulumba, A Common Analysis Framework for Simulated Streaming-Video Networks, Master Thesis, Grahamstown, South Africa, April 2009.
- [13] Khater, Jad. NS-2 simulation based study of E2E vídeo streaming over ultra-wideband (UWB) wireless mesh networks. Master's thesis, 2005/2006, Athens, Greece.
- [14] Evalvid, http://hpd.ee.ncku.edu.tw/~smallko/ns2/Evalvid_in_NS2.htm
- [15] Ke, C.H.: How to evaluate MPEG video transmission using the NS2 simulator. [Online], [http://hpd.ee.ncku.edu.tw/smallko/ns2/Evalvid in NS2.htm](http://hpd.ee.ncku.edu.tw/smallko/ns2/Evalvid_in_NS2.htm)
- [16] Gavriloff, I. Análise de aspectos envolvidos no mecanismo de seleção de caminho baseado em atraso para sistemas multiabrigados utilizando SCTP. Master's thesis, 2009, UFPR, Curitiba, Brazil.
- [17] Xu, Changqiao; Qiao Yuansong; Muntean, Gabriel-Miro; Jacob Paul; Hanley, Austin; Fallon, Enda. Comparative Study of Real-Time Multimedia Transmission over Multi-homing Transport Protocols, 2008, <http://dl.acm.org/citation.cfm?id=1434267>
- [18] Caro, A. L., Amer, P. D., and Stewart, R. R. (2006). Rethinking end-to-end failover with transport layer multihoming. Annals of Telecommunications
- [19] UC Berkeley, LBL, USC/ISI, and Xerox Parc ns-2 documentation and software, Version 2.29 , 2005, <http://www.isi.edu/nsnam/ns>
- [20] Cygwin, <http://www.cygwin.com/>
- [21] Yuv Video Sequences, <http://trace.eas.asu.edu/yuv/index.html>.
- [22] How to evaluate H.263/H.264/MPEG4 video transmission using the NS2 simulator?, <http://140.116.72.80/~smallko/ns2/myevalvid2.htm>.
- [23] Fall, Kevin (hkfall@ee.lbl.gov, Editor), Varadhan, Kannan (kannan@catarina.usc.edu, Editor), The ns Manual (formerly ns Notes and

Documentation), The VINT Project, A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, March 14, 2008,
<http://www.isi.edu/nsnam/ns/ns-documentation.html>

[24] Caro, A., Iyengar, J.: ns-2 SCTP module, Version 3.5,
<http://www.armandocaro.net/software/ns2sctp>

8 ANEXOS

8.1 Anexo I: Resultado dos experimentos dos cenários simulados

8.1.1 UDP Frame 1024:

Tabela 11 - Resultados de simulações pós-processados com NS-2 - UDP 1024

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
1	1 via sem perdas; UDP 1024; GOP = 4	45,79993289	25,2938	96 / 59.1	79,3	2,37
2	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 4	45,79896667	30,29751	88 / 74.7	78	3,31
3	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 4	45,79896667	31,14852	84 / 72	75	3,41
4	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 4	45,79896667	31,57037	82.7 / 71.1	74	3,49
5	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 4	45,79896667	27,86848	97.3 / 76.9	82	2,91
6	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 4	45,79896667	29,88364	92 / 78.2	81,7	3,25
7	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 4	45,79896667	29,94532	90.7 / 76	79,7	3,26
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 4	45,79896667	30,29751	88 / 74.7	78	3,31

Tabela 12 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-1)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
9	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 4	45,79993289	30,26956	88 / 74.7	78	3,31
10	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 1024; GOP = 4	45,79993289	30,26956	88 / 74.7	78	3,31
11	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 1024; GOP = 4	45,79896667	30,29751	88 / 74.7	78	3,31
12	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 1024; GOP = 4	45,79993289	31,06783	88 / 76.4	79,3	3,43
13	1 via sem perdas; UDP 1024; GOP = 6	45,70963333	25,39586	94 / 69.2	73,3	2,41
14	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP=6	45,70963333	33,12261	72 / 66.8	67,7	3,76
15	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 6	45,70963333	39,10488	14 / 16	15,7	4,61
16	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 6	45,79896667	32,59602	74 / 66.8	68	3,7
17	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 6	45,70963333	30,69585	90 / 70.8	74	3,37

Tabela 13 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-2)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
18	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 6	45,79896667	33,1367	86 / 68.4	71,3	3,8
19	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 6	45,79896667	32,47588	82 / 68.4	70,7	3,68
20	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 6	45,70963333	33,12261	72 / 66.8	67,7	3,76
21	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 6	45,70963333	33,12261	72 / 66.8	67,7	3,76
22	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 1024; GOP = 6	45,79896667	32,66699	72 / 66.8	67,7	3,71
23	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 1024; GOP = 6	45,79896667	29,33676	90 / 86	86,7	3,2
24	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 1024; GOP = 6	45,70963333	33,00054	72 / 66.4	67,3	3,75
25	1 via sem perdas; UDP 1024; GOP = 8	45,66863333	25,48387	92.1 / 69.1	72	2,37
26	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP=8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87

Tabela 14 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-3)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
27	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 8	45,79896667	34,25445	68.4 / 62.2	63	3,95
28	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 8	45,79896667	34,21365	68.4 / 61.8	62,7	3,94
29	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 8	45,66863333	20,26657	89.5 / 69.5	72	2,06
30	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 8	45,79896667	33,13428	76.3 / 66	67,3	3,82
31	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 8	45,79896667	33,84852	71.1 / 64.9	65,7	3,92
32	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87
33	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 8	45,66863333	33,61201	68.4 / 61.8	62,7	3,87
34	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb19,2kb;) UDP 1024; GOP = 8	45,79896667	34,17371	68.4 / 61.8	62,7	3,94

Tabela 15 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-4)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
35	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 1024; GOP = 8	45,79896667	34,17371	68.4 / 61.8	62,7	3,94
36	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 1024; GOP = 8	45,66863333	33,26237	68.4 / 63.4	64	3,81
37	1 via sem perdas; UDP 1024; GOP = 16	45,57866667	27,44168	89.5 / 62.6	64,3	2,6
38	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 16	45,57866667	32,5704	68.4 / 61.2	61,7	3,68
39	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 1024; GOP = 16	45,79896667	33,20829	68.4 / 62.6	63	3,74
40	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 1024; GOP = 16	45,79896667	33,28054	68.4 / 62.6	63	3,75
41	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 16	45,57866667	23,20348	94.7 / 68	69,7	2,46
42	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 16	45,79896667	31,68716	89.5 / 64.8	66,3	3,61
43	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 1024; GOP = 16	45,79896667	32,10906	78.9 / 61.9	63	3,68

Tabela 16 - Resultados de simulações pós-processados com NS-2 - UDP 1024 (continuação-5)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	Overall Frame Loss in %	MOS
44	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 16	45,57866667	32,5704	68.4 / 61.2	61,7	3,68
45	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 1024; GOP = 16	45,57866667	32,5704	68.4 / 61.2	61,7	3,68
46	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 1024; GOP = 16	45,79896667	33,41258	68.4 / 61.2	61,7	3,78
47	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 1024; GOP = 16	45,79896667	33,41258	68.4 / 61.2	61,7	3,78
48	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 1024; GOP = 16	45,57866667	31	73.7 / 61.2	62	3,51

Figura 13 - 1 via sem perdas - UDP 1024, GOP = 4

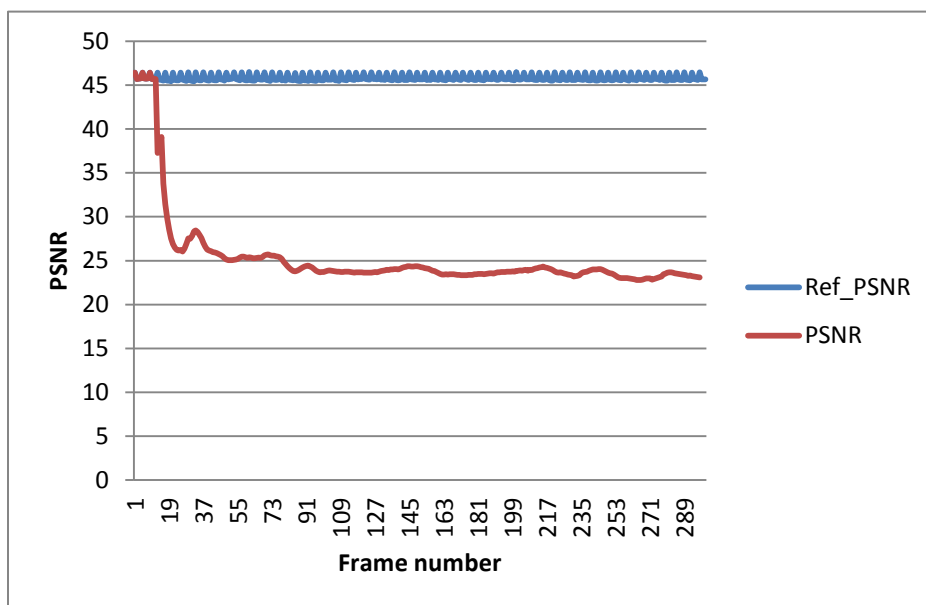


Figura 14 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 4

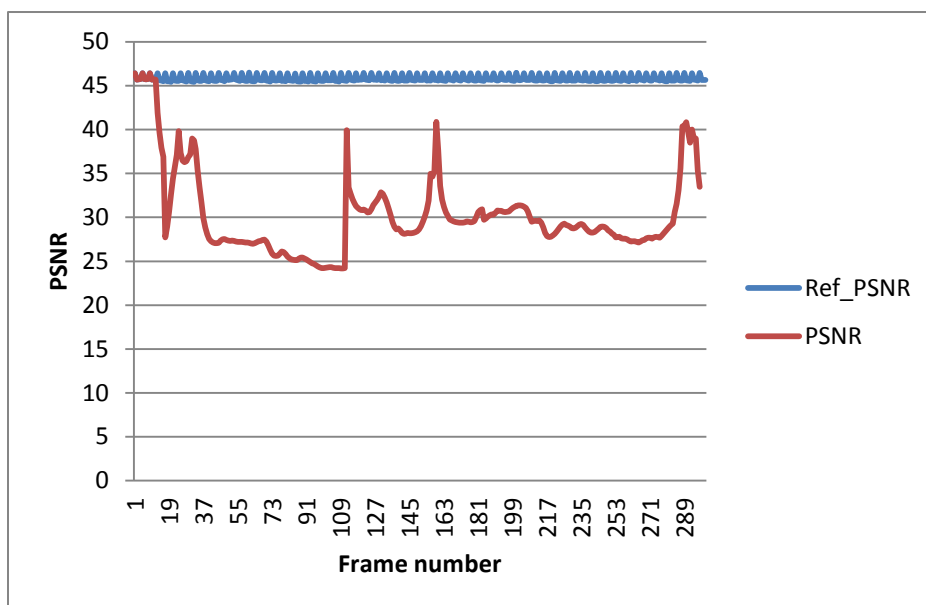


Figura 15 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms) - UDP 1024; GOP = 4

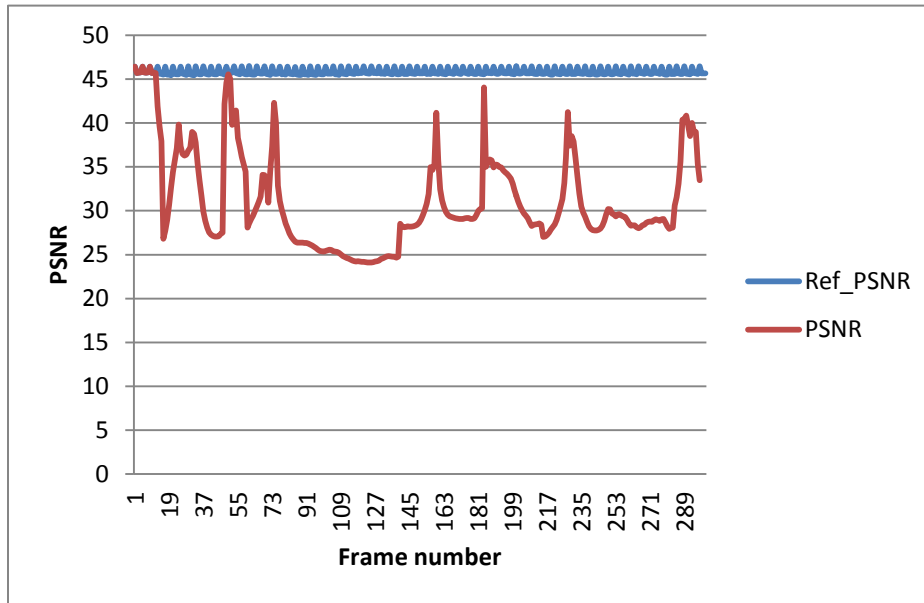


Figura 16 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms) - UDP 1024; GOP = 4

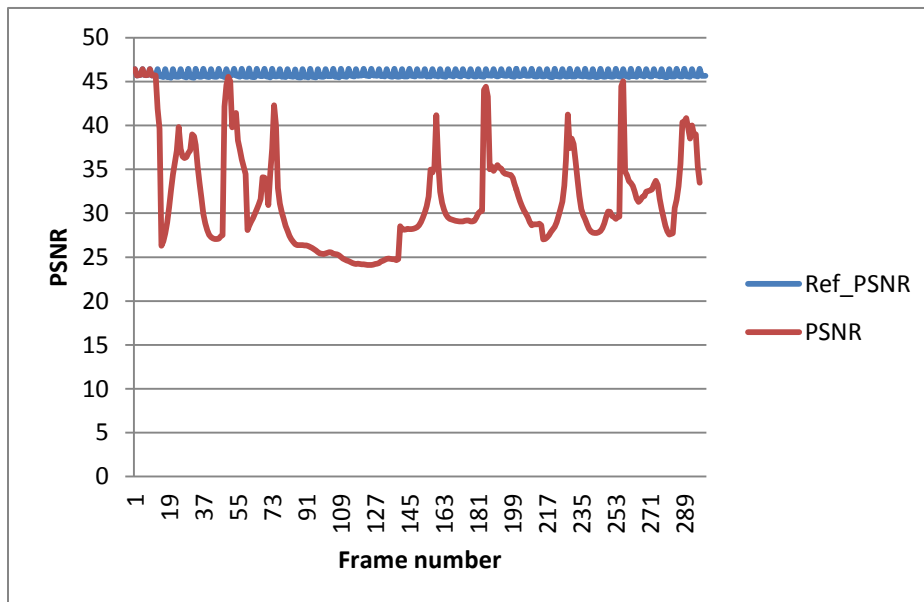


Figura 17 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 4

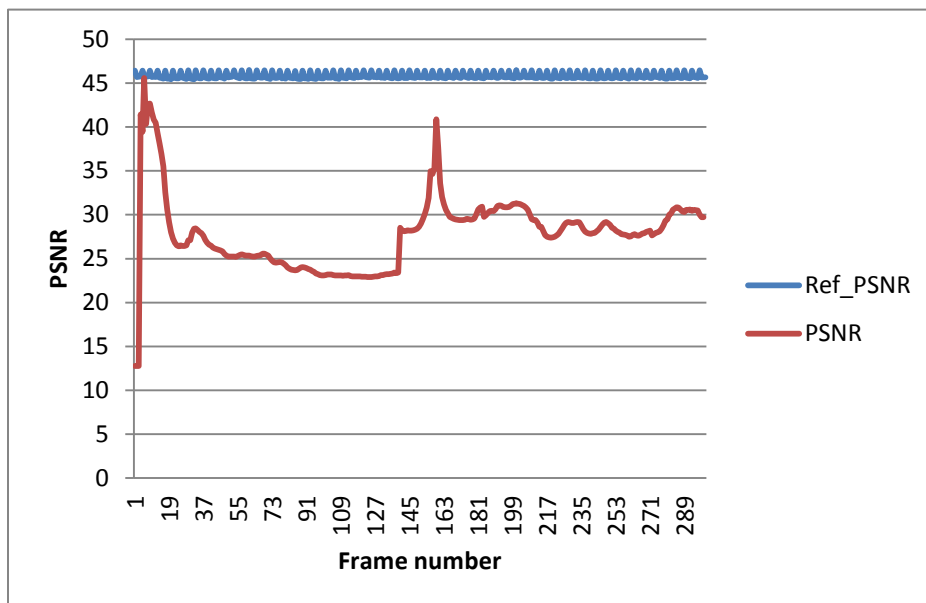


Figura 18 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 4

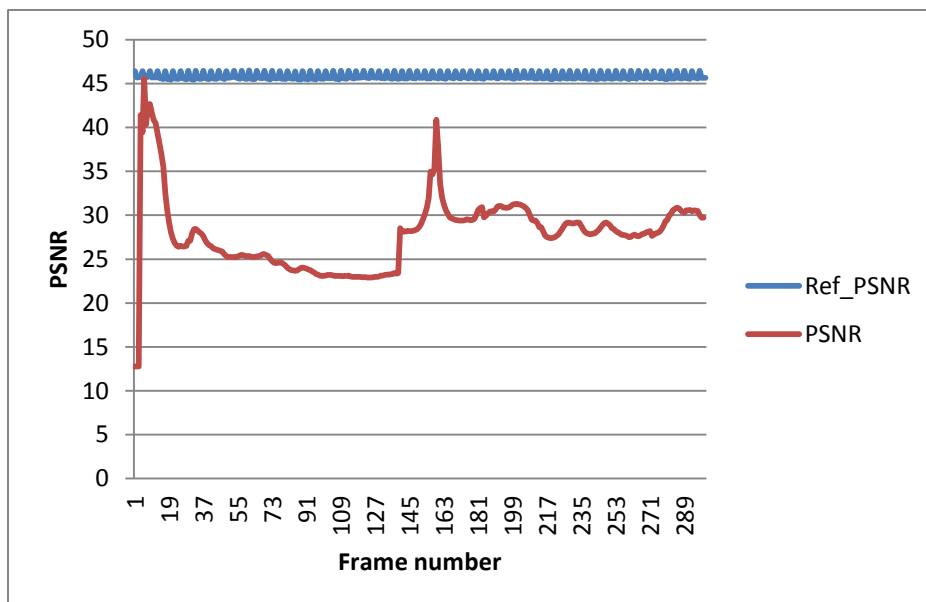


Figura 19 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 4

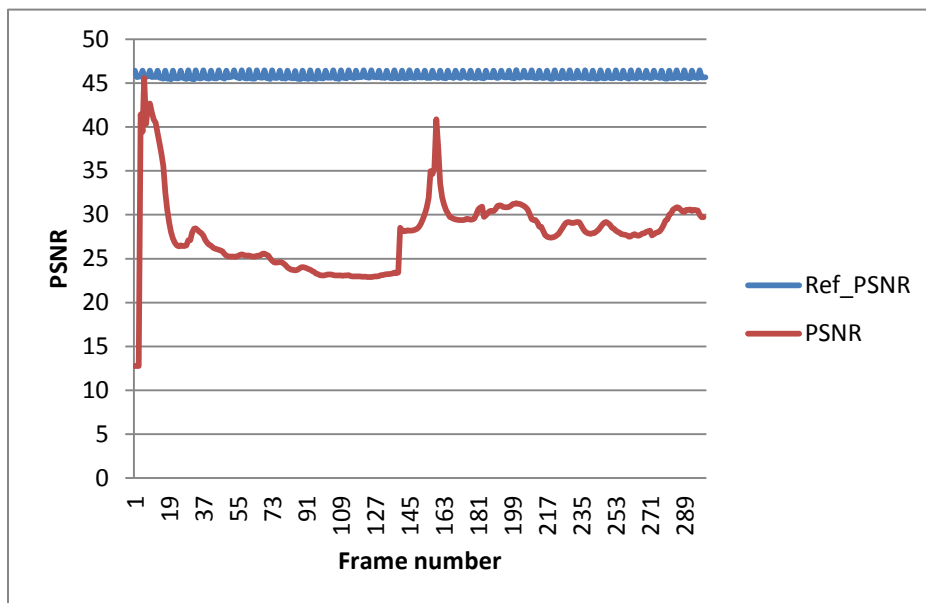


Figura 20 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 4

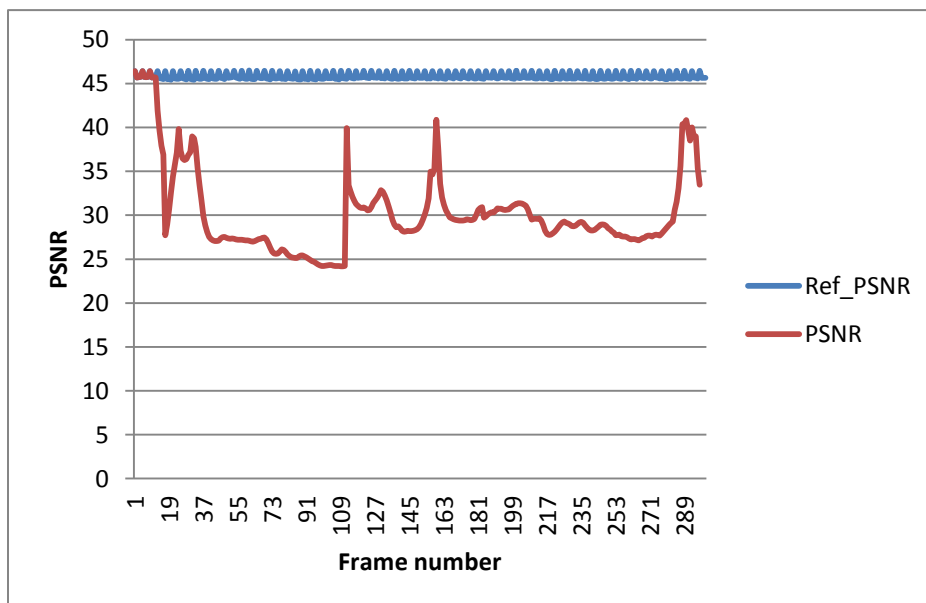


Figura 21 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 4

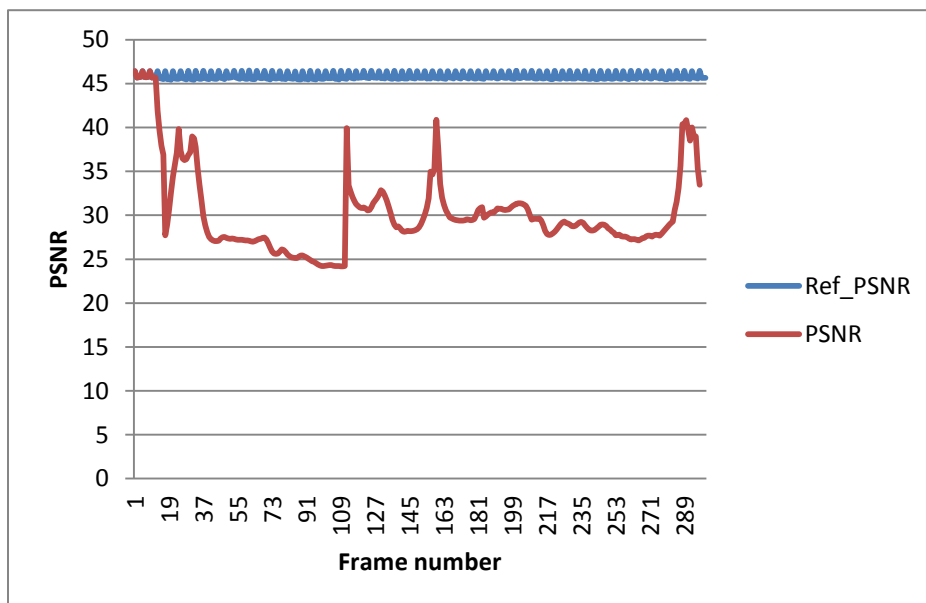


Figura 22 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb;) - UDP 1024; GOP = 4

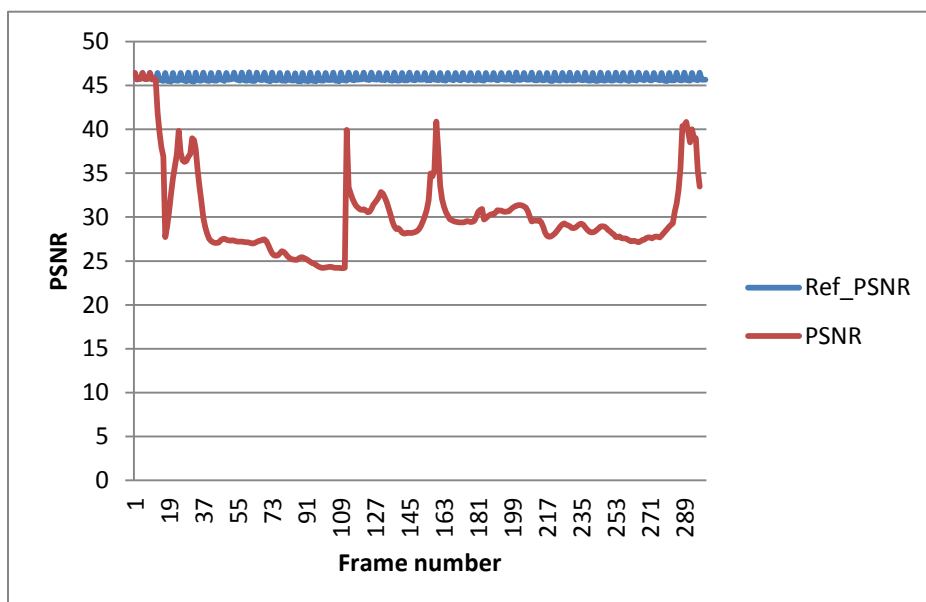


Figura 23 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb) - UDP 1024; GOP = 4

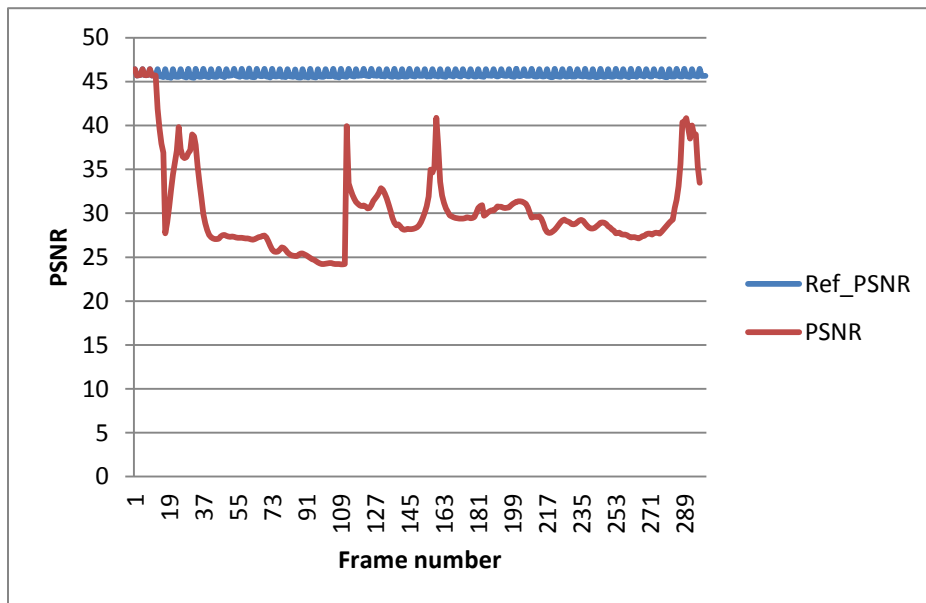


Figura 24 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms) - UDP 1024; GOP = 4

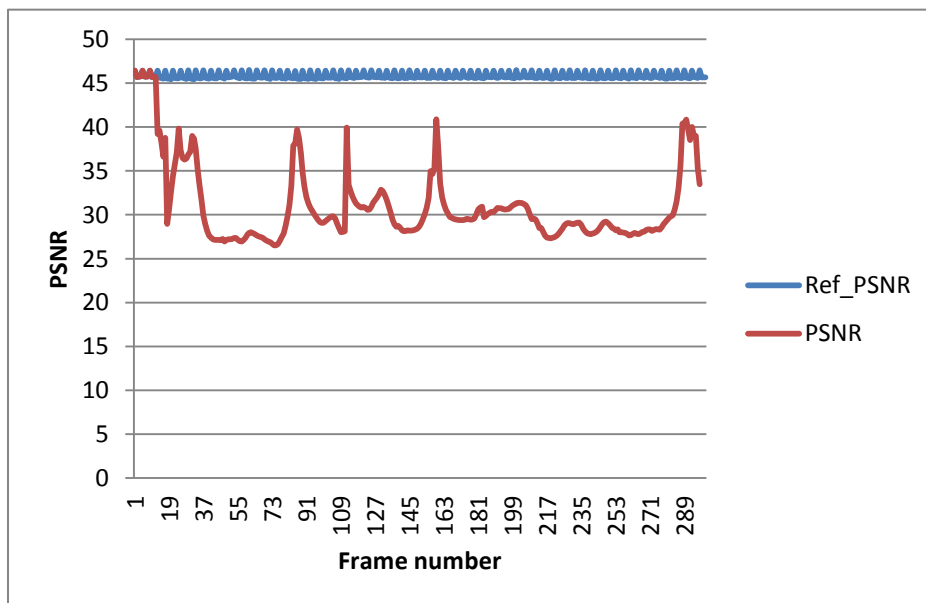


Figura 25 - 1 via sem perdas - UDP 1024; GOP = 6

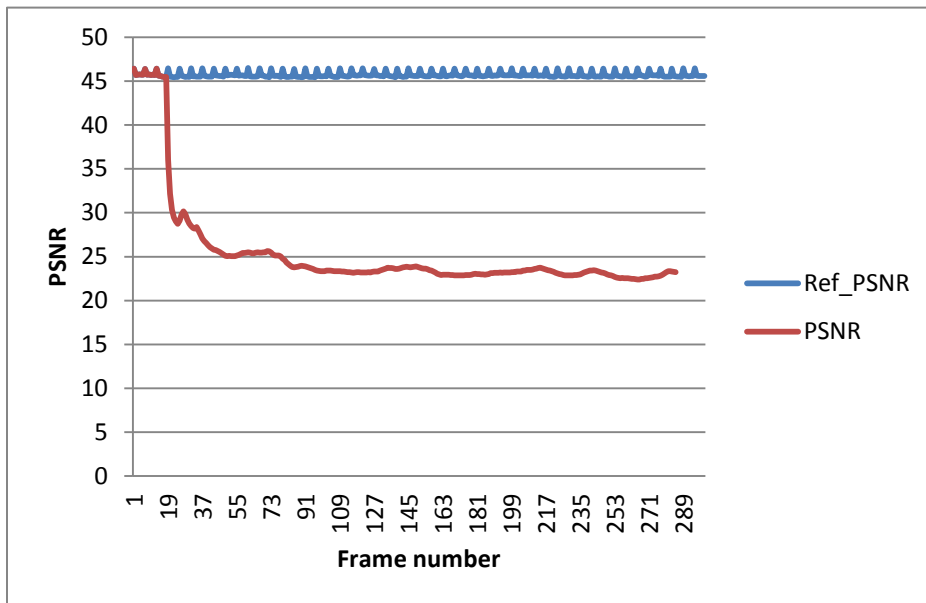


Figura 26 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP=6

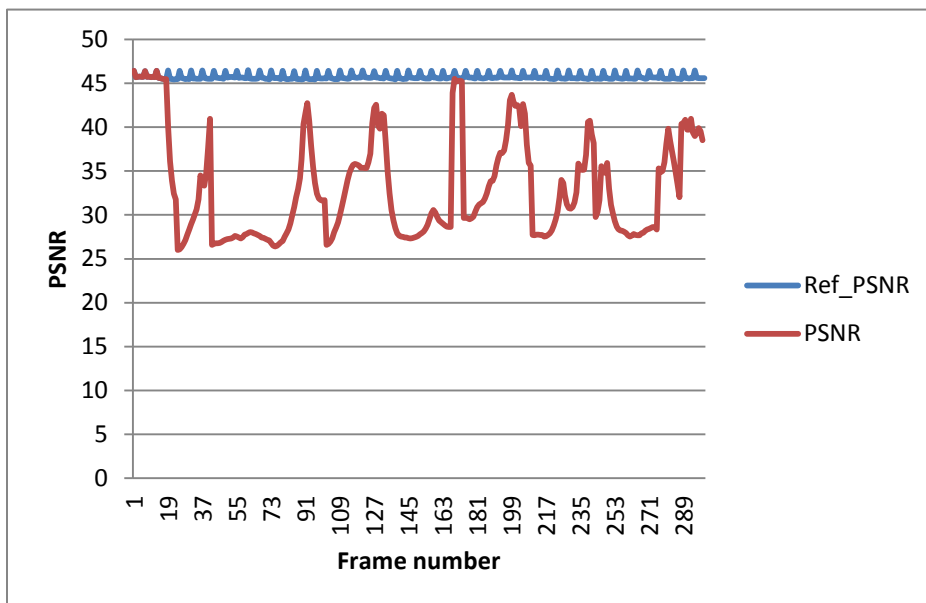


Figura 27 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms) - UDP 1024; GOP = 6

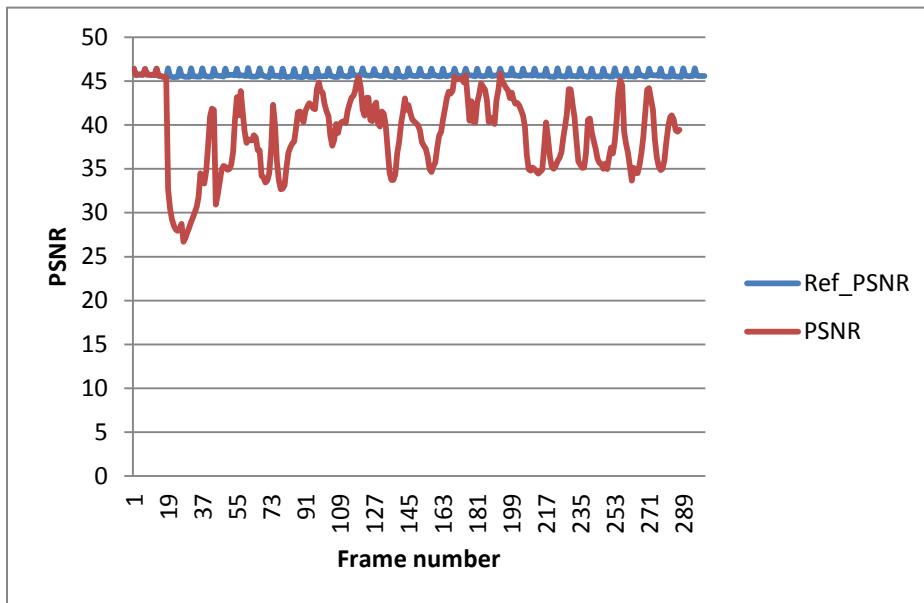


Figura 28 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms) - UDP 1024; GOP = 6

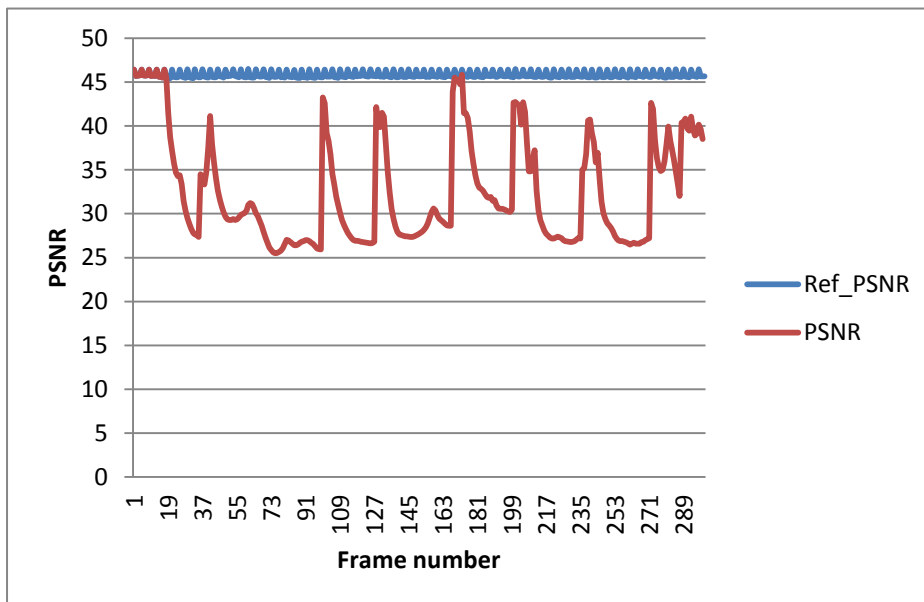


Figura 29 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 6

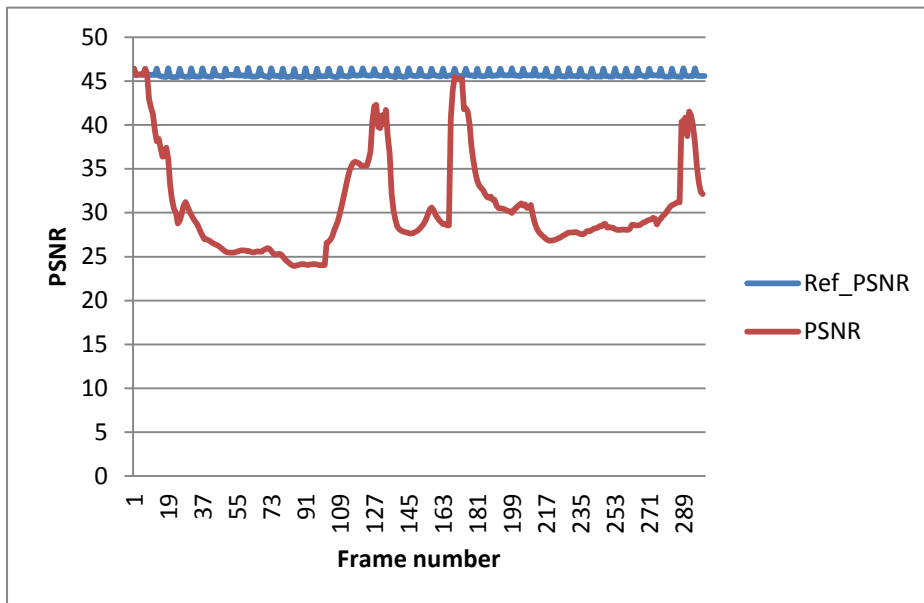


Figura 30 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 6

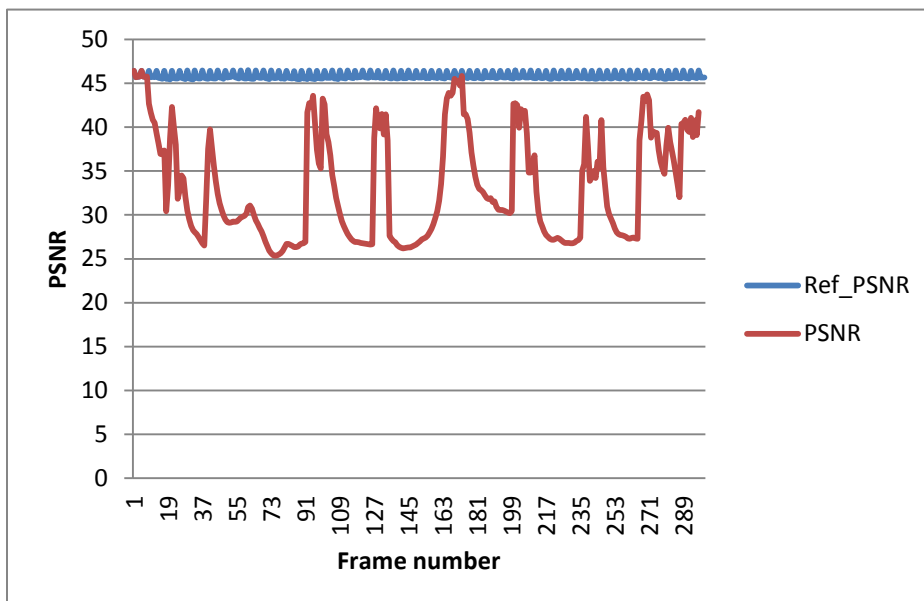


Figura 31 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms) - UDP 1024; GOP = 6

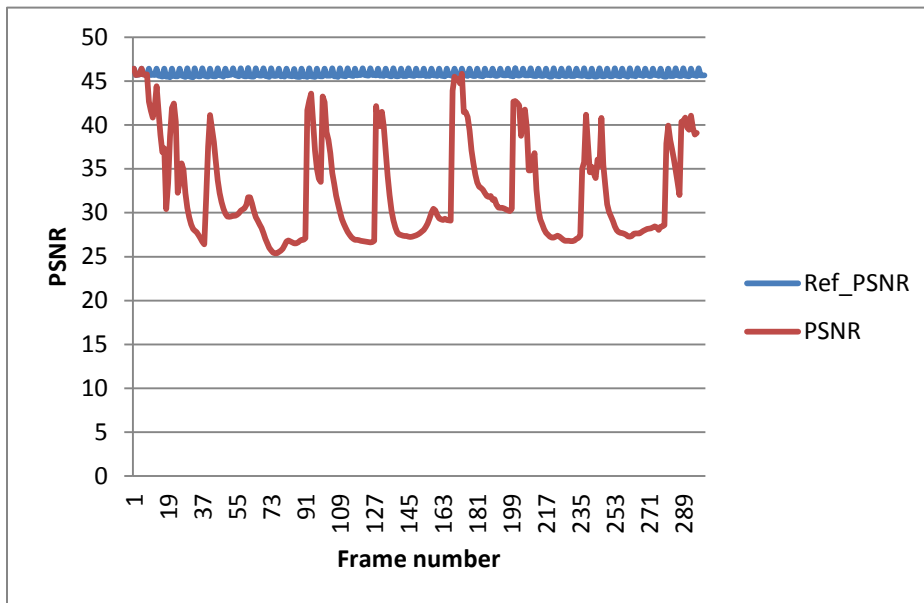


Figura 32 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb) UDP 1024; GOP = 6

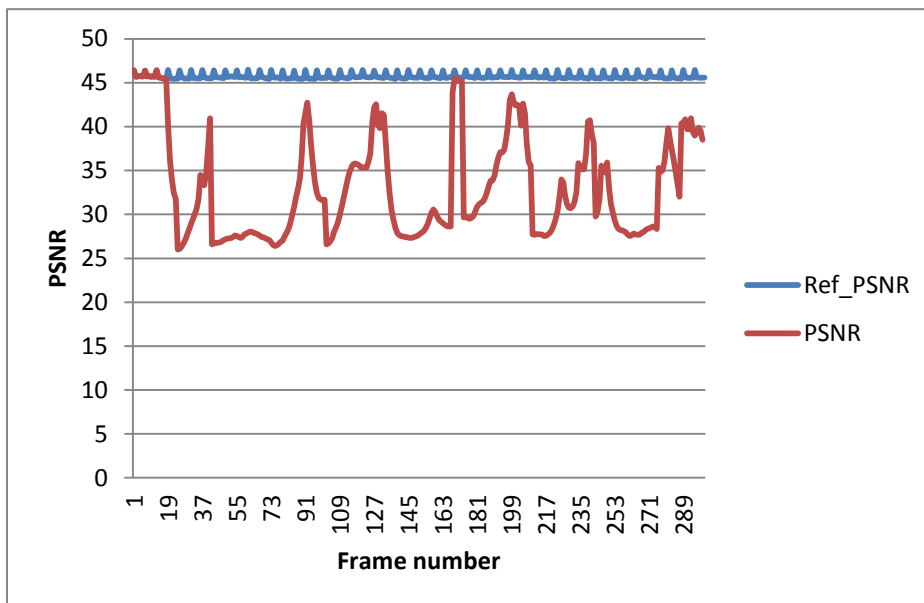


Figura 33 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) - UDP 1024; GOP = 6

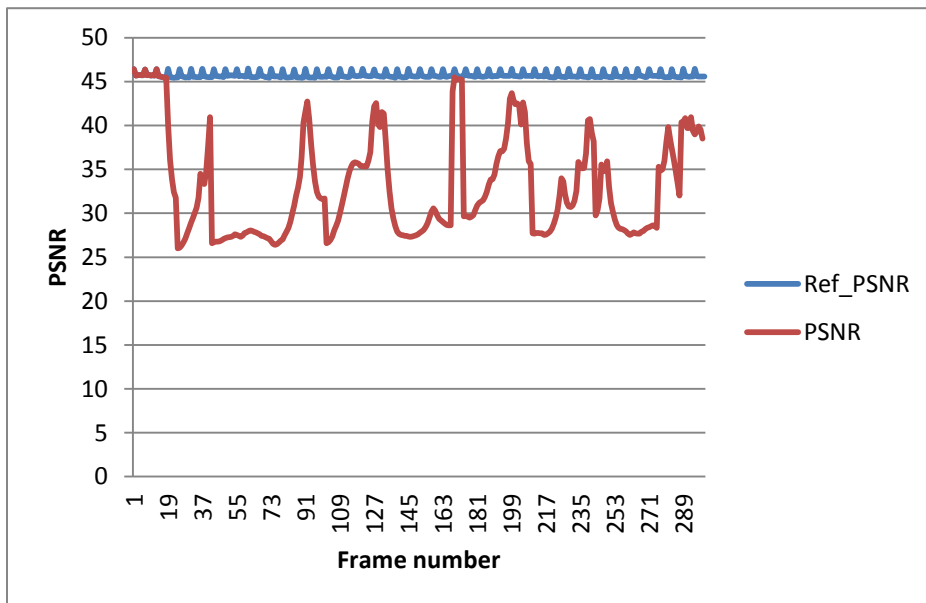


Figura 34 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 6

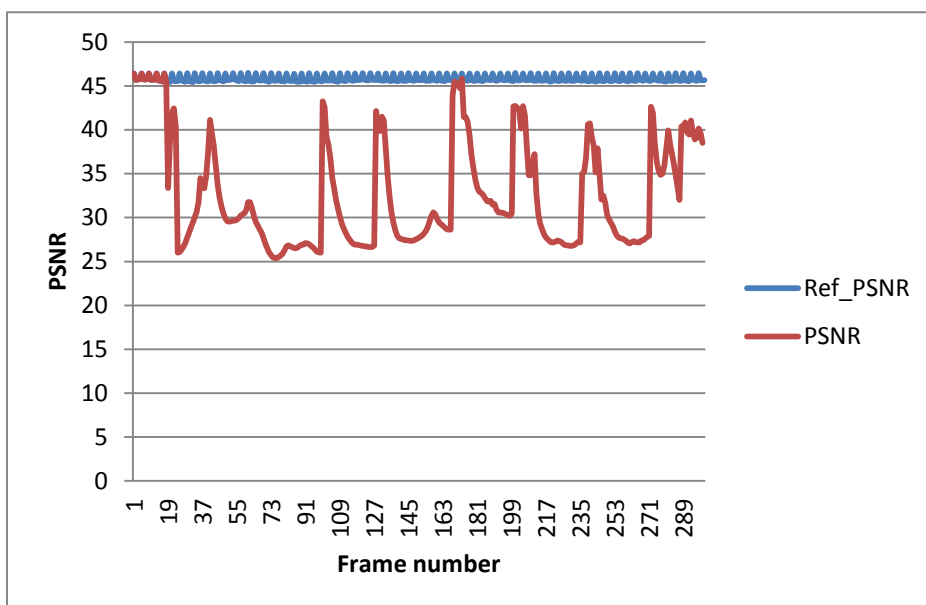


Figura 35 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 6

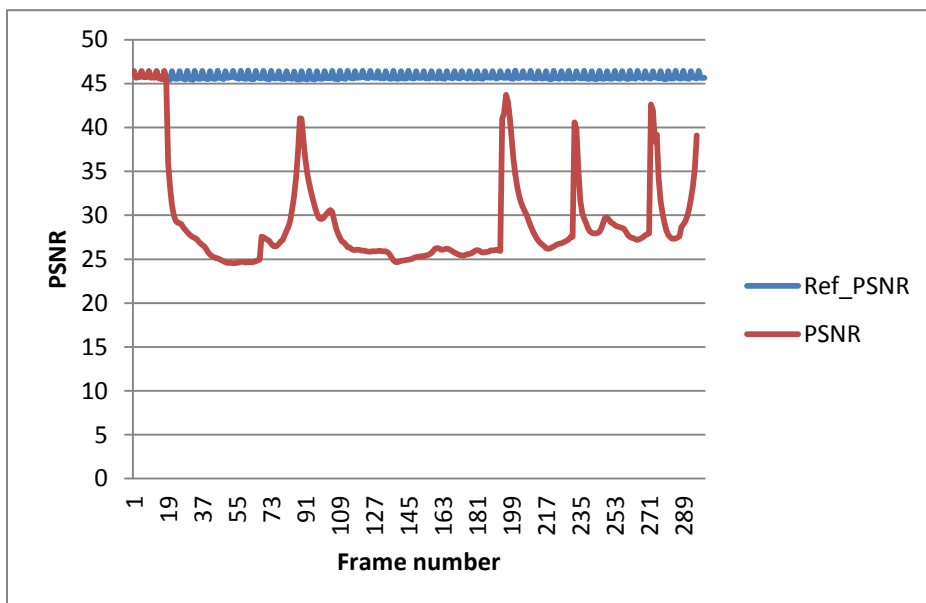


Figura 36 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 1024; GOP = 6

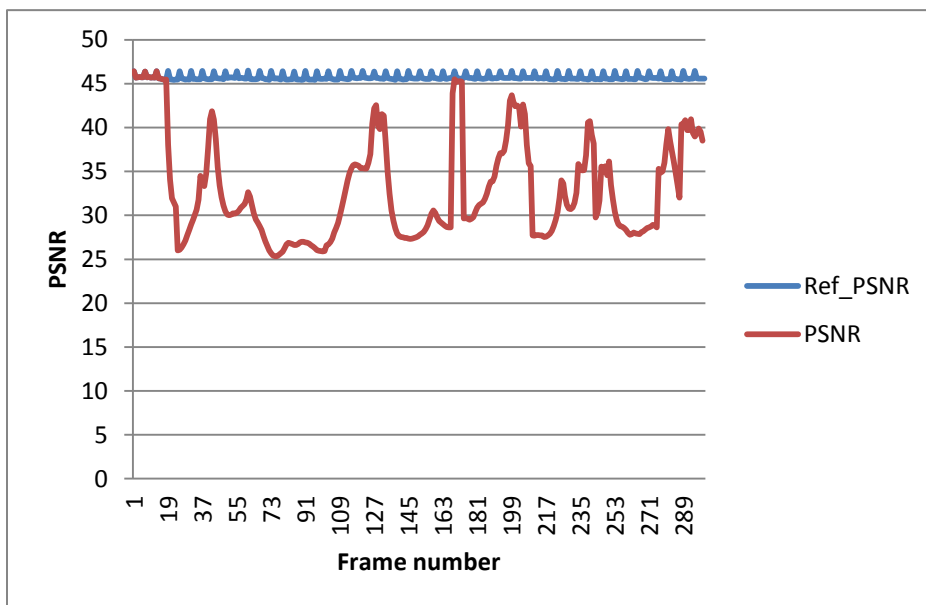


Figura 37 - 1 via sem perdas; UDP 1024; GOP = 8

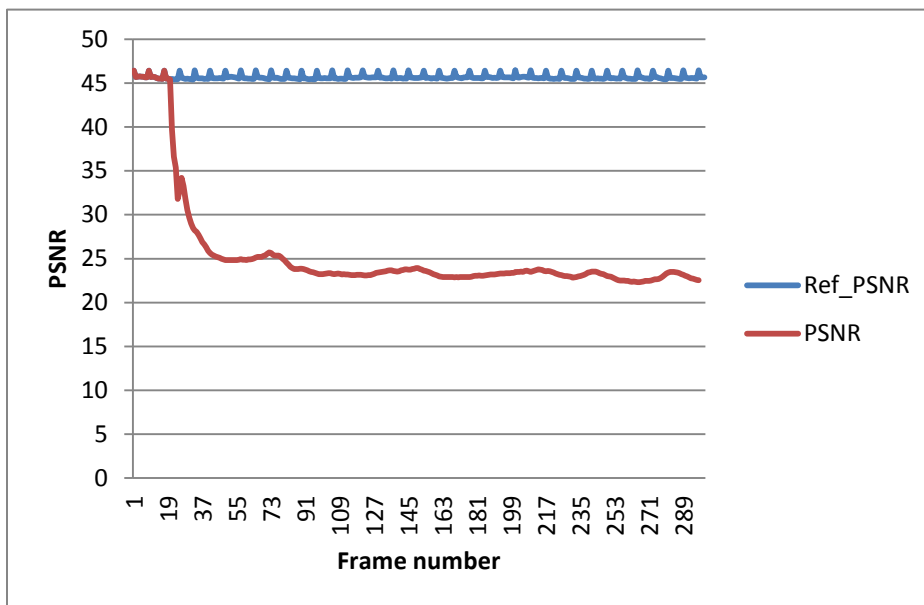


Figura 38 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP=8

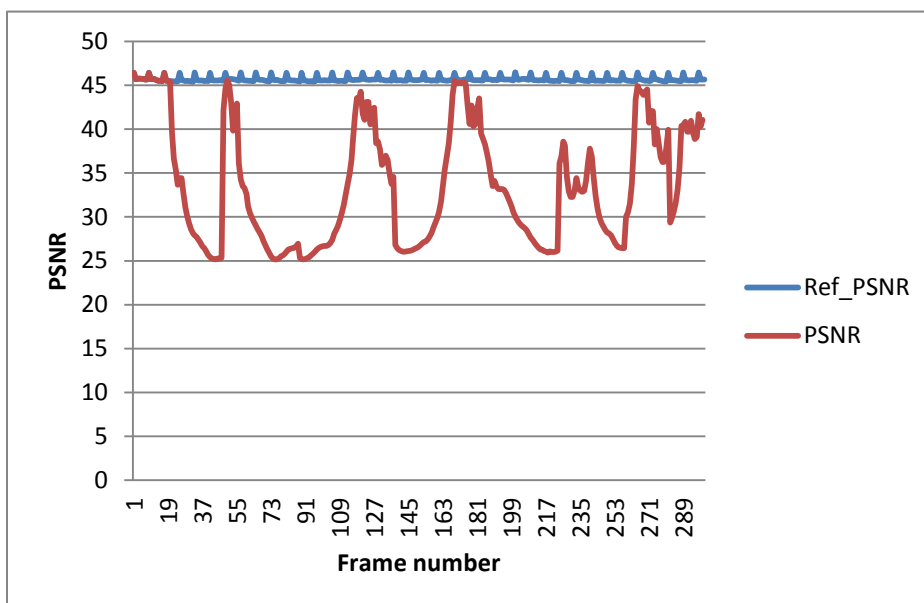


Figura 39 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 1024; GOP = 8

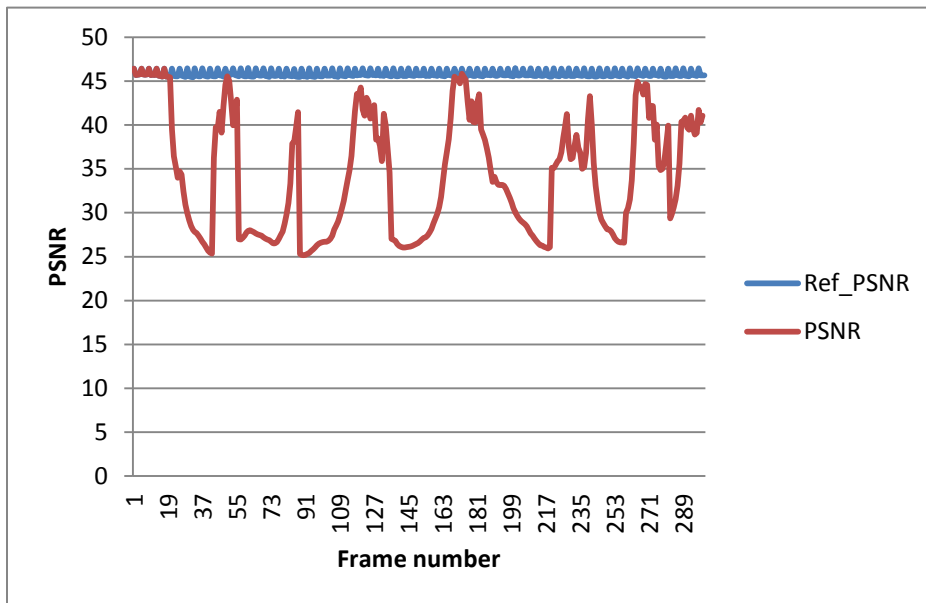


Figura 40 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 1024; GOP = 8

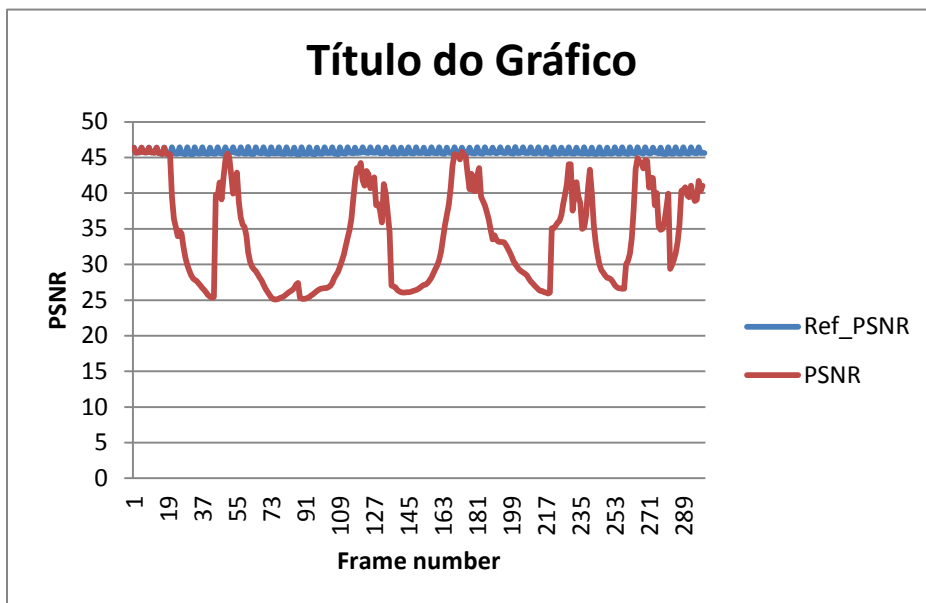


Figura 41 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 8

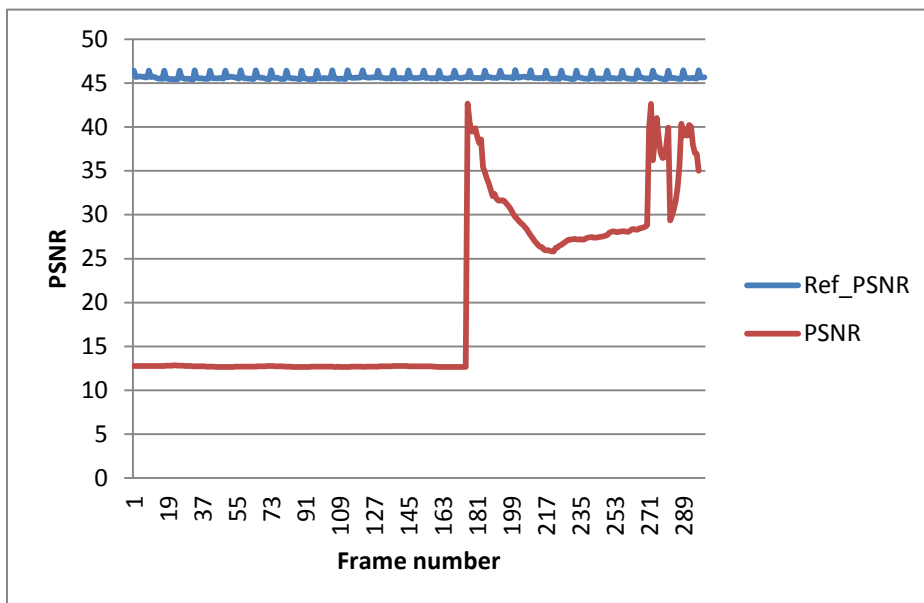


Figura 42 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 8

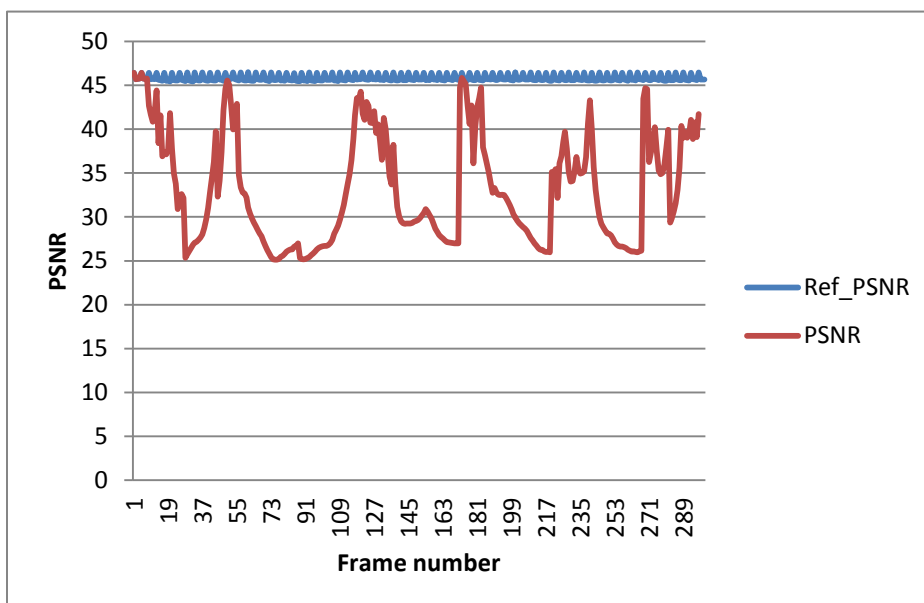


Figura 43 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 8

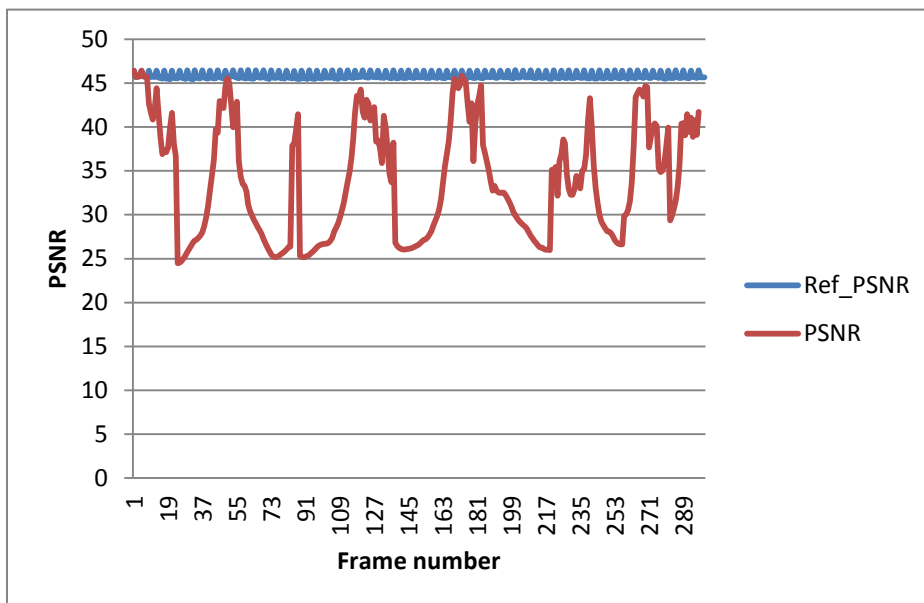


Figura 44 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP(230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 8

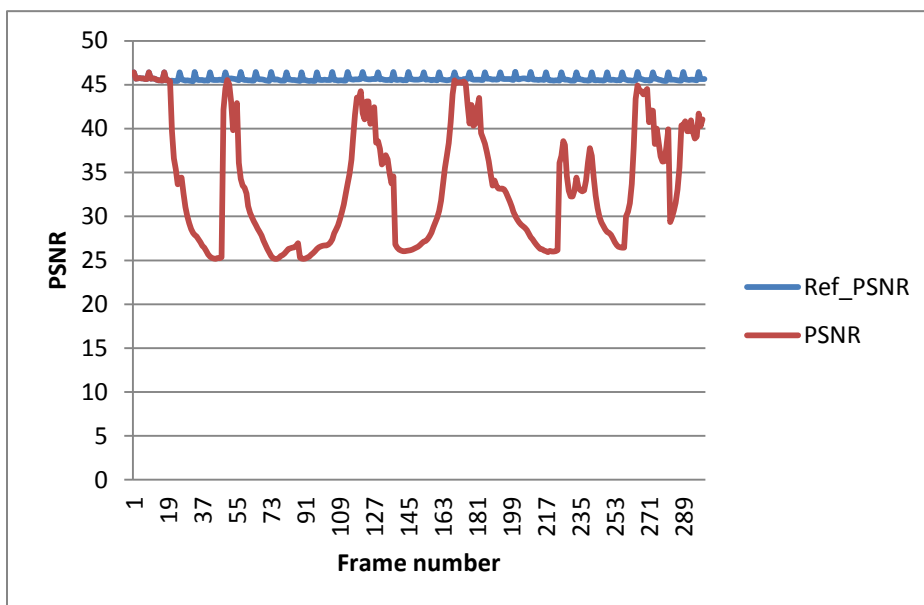


Figura 45 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 8

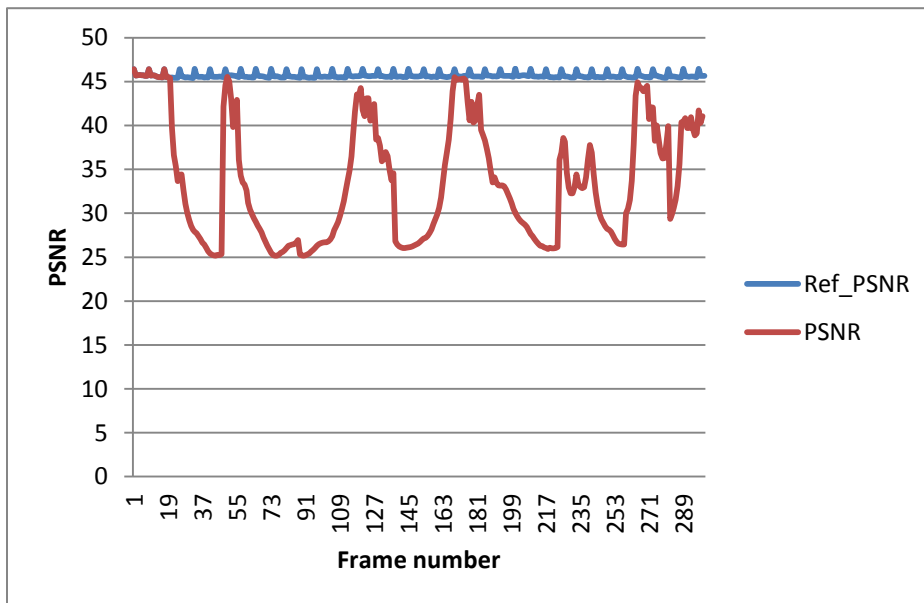


Figura 46 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 8

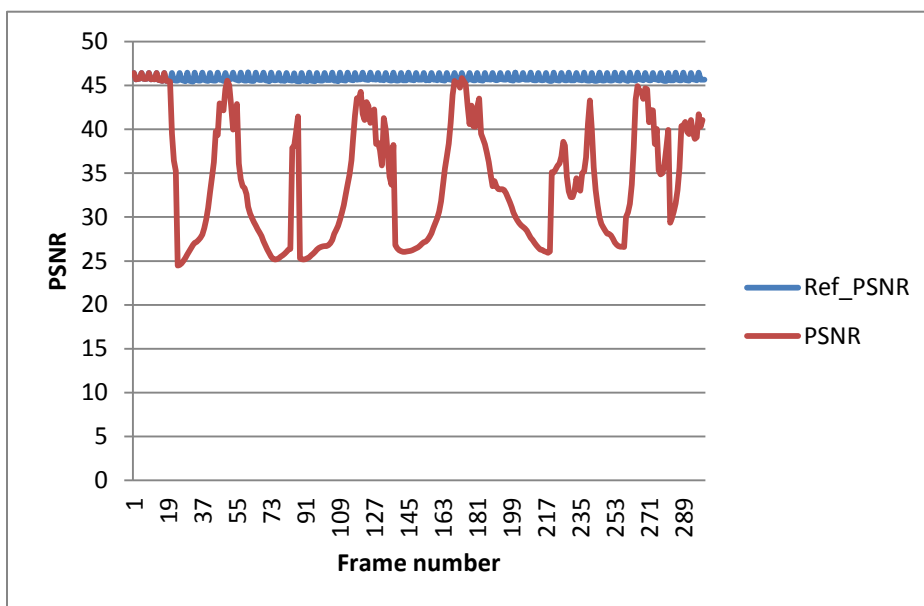


Figura 47 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 8

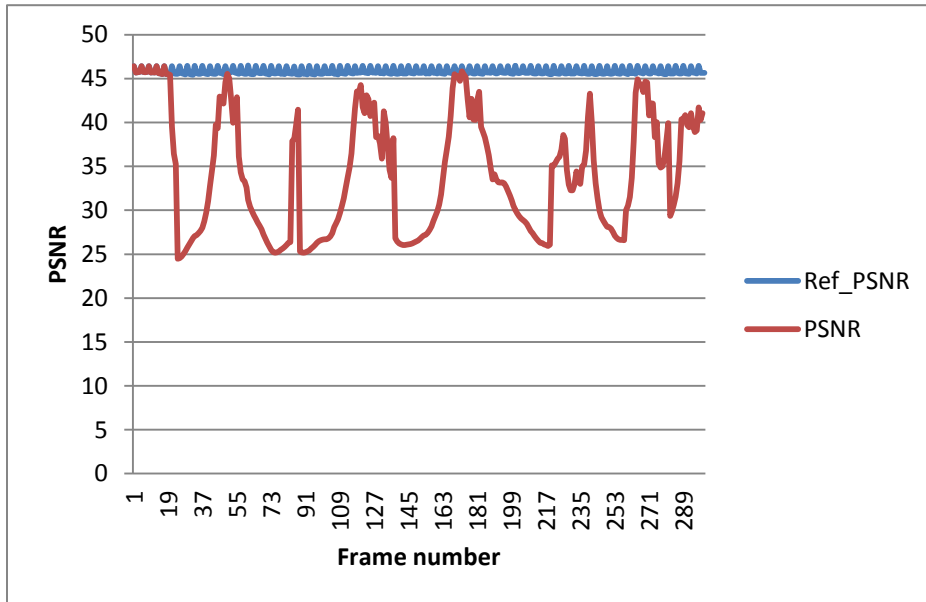


Figura 48 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 1024; GOP = 8

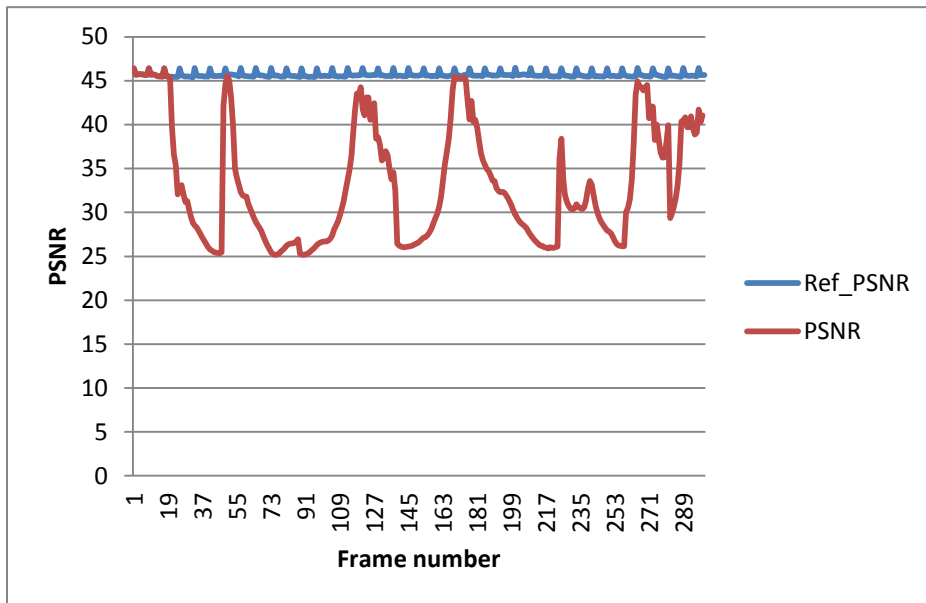


Figura 49 - 1 via sem perdas; UDP 1024; GOP = 16

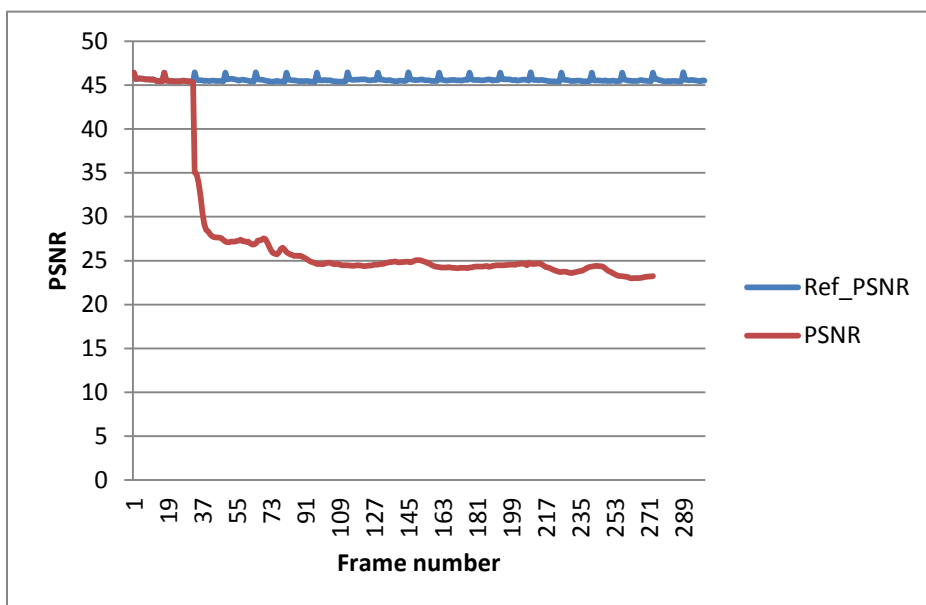


Figura 50 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 16

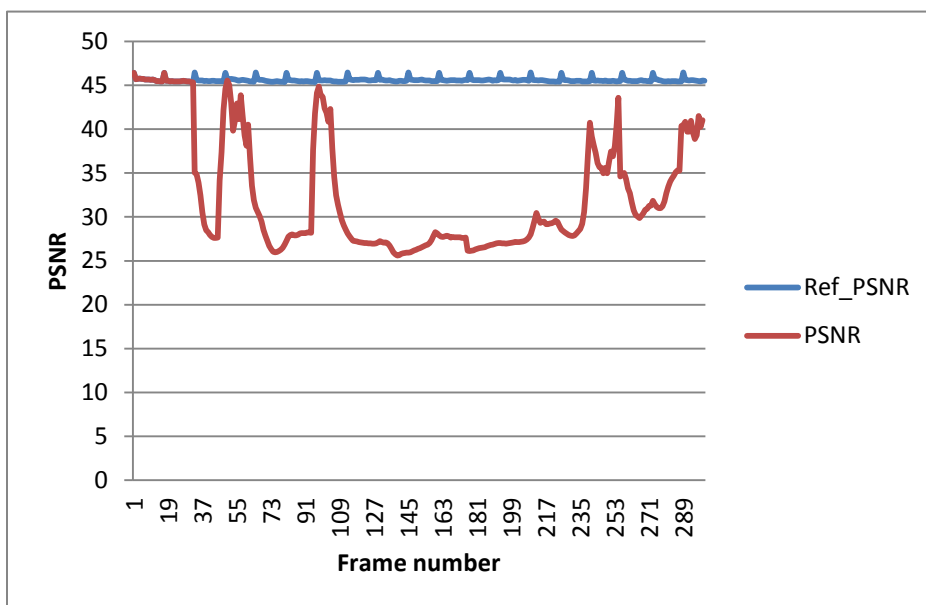


Figura 51 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 1024; GOP = 16

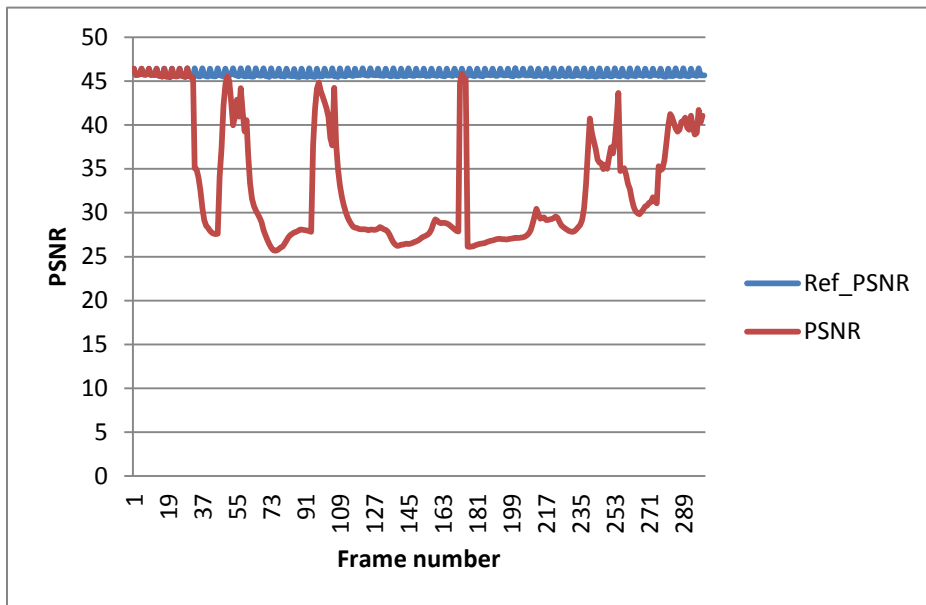


Figura 52 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 1024; GOP = 16

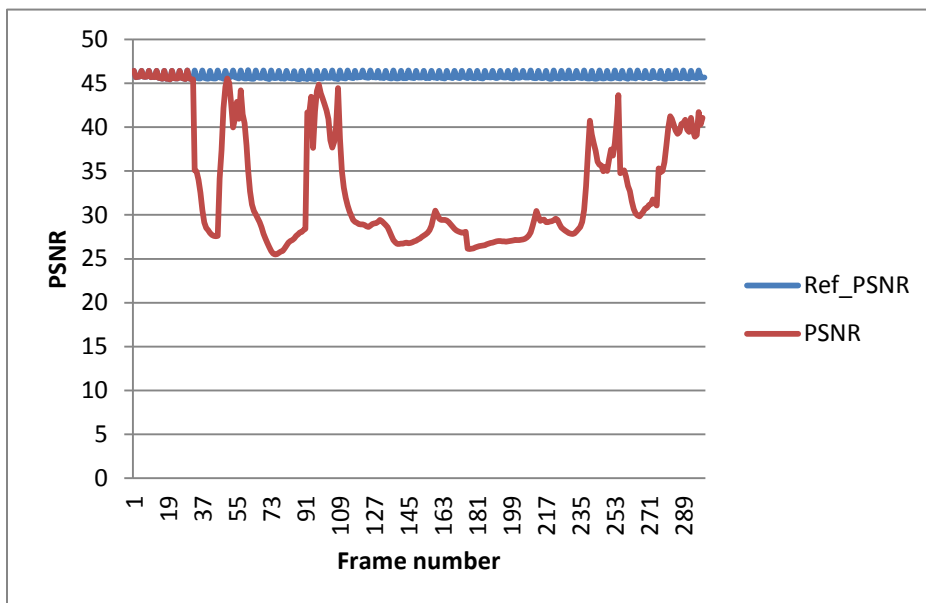


Figura 53 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 16

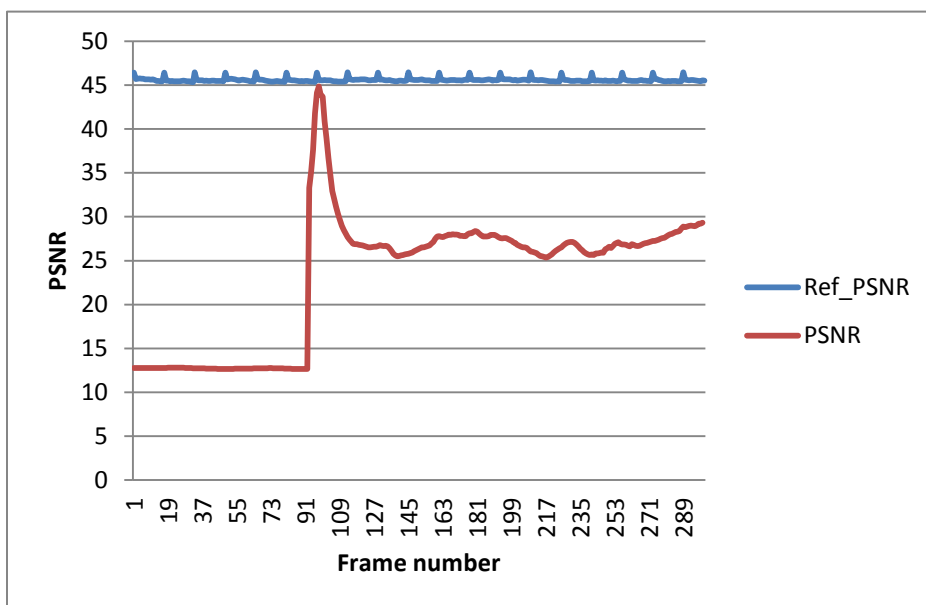


Figura 54 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 16

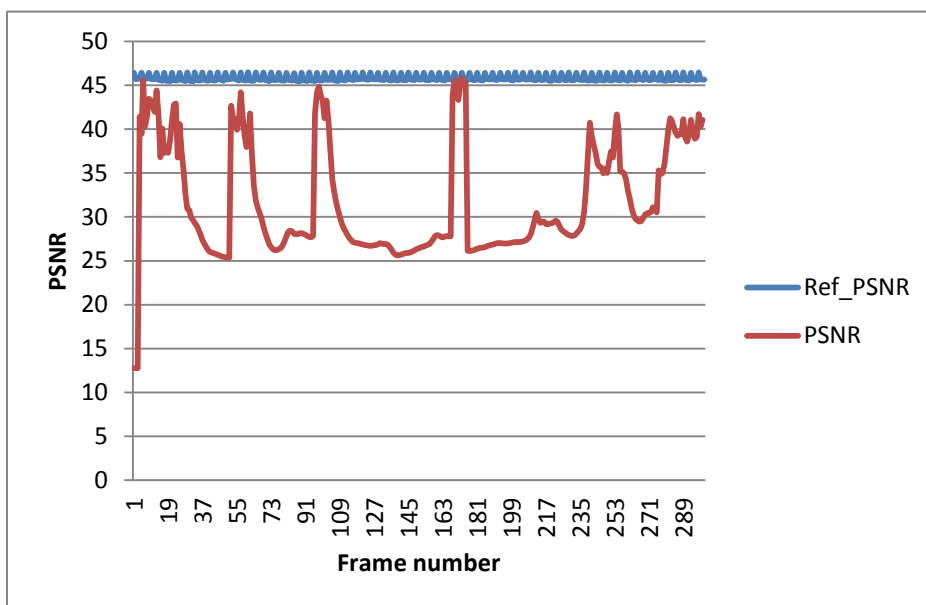


Figura 55 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 1024; GOP = 16

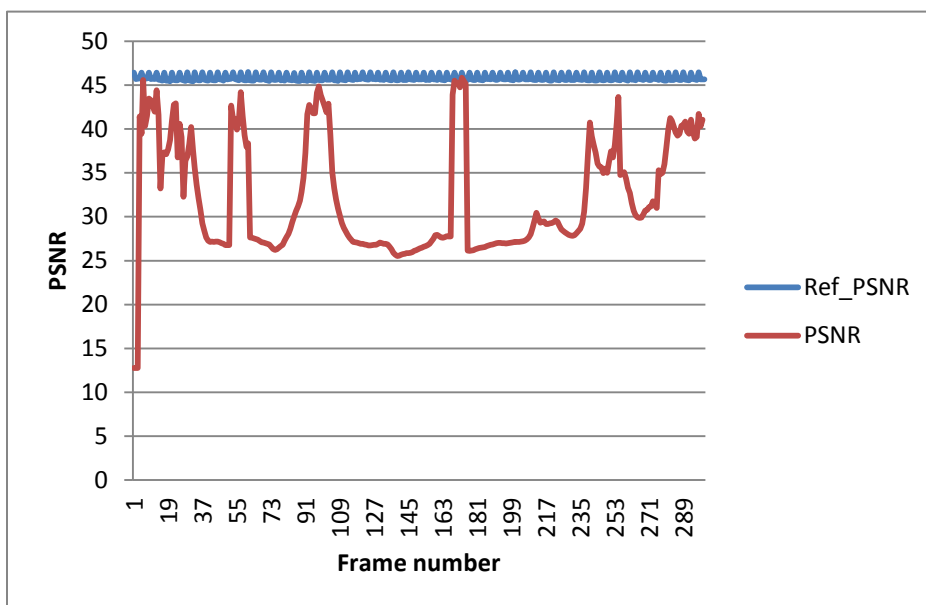


Figura 56 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 16

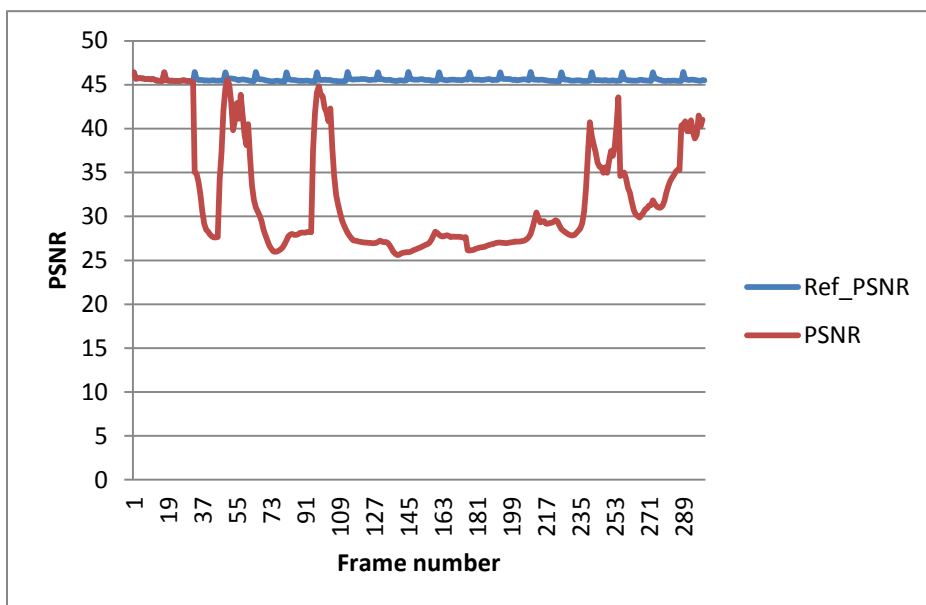


Figura 57 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 1024; GOP = 16

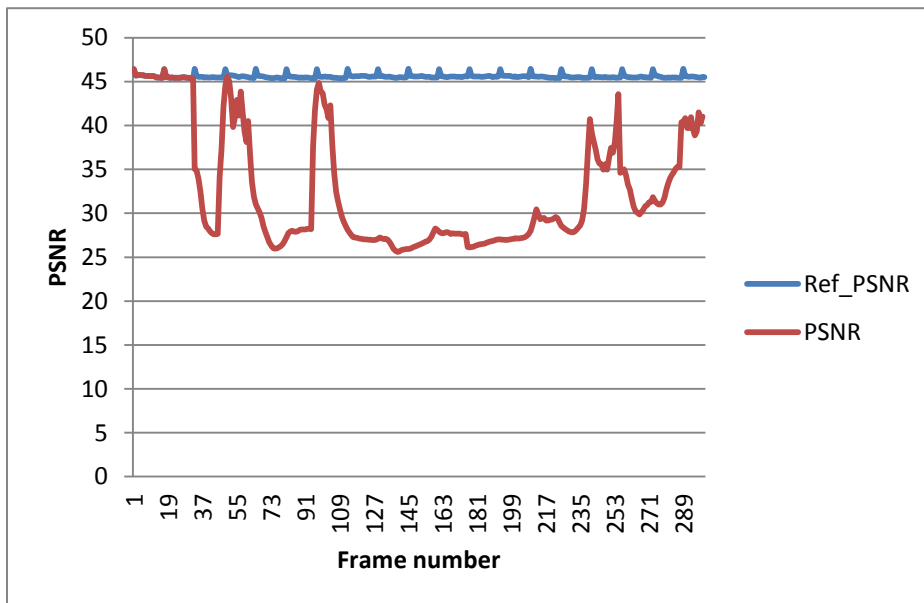


Figura 58 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 1024; GOP = 16

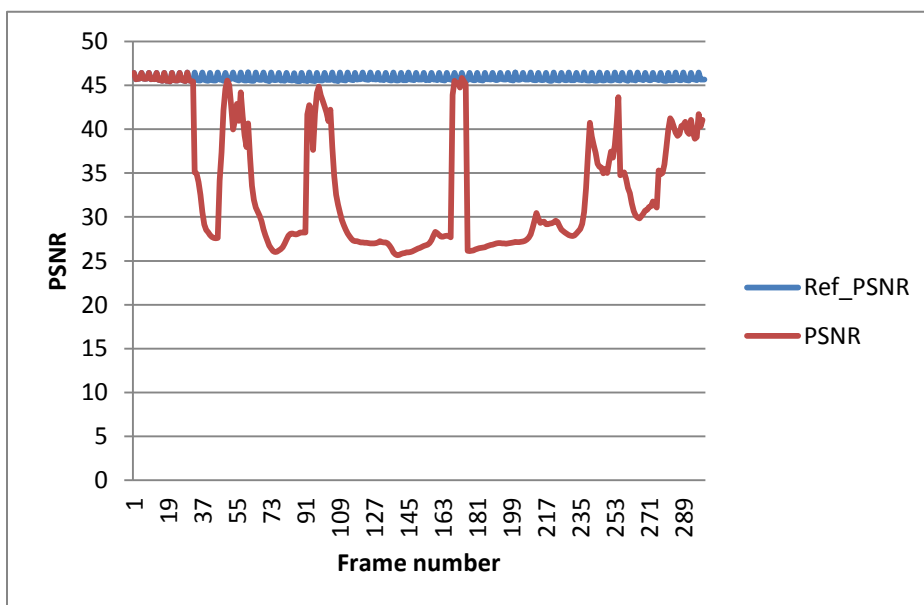


Figura 59 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 1024; GOP = 16

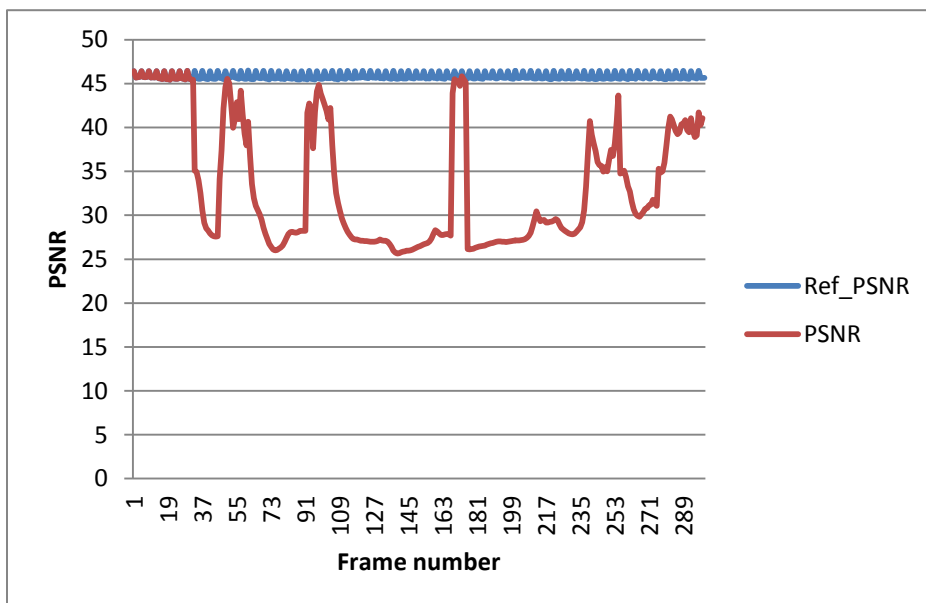
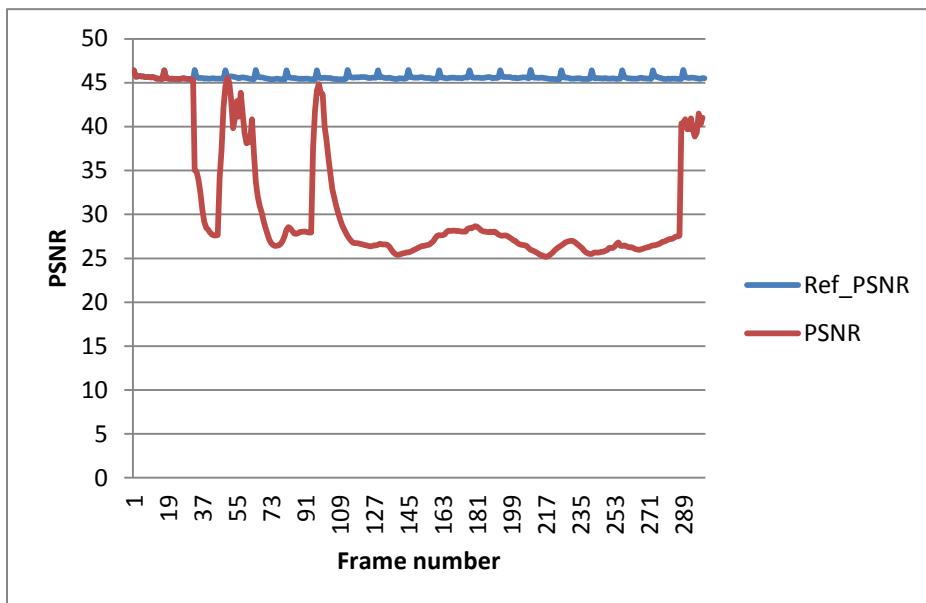


Figura 60 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 1024; GOP = 16



1.1.1 UDP Frame 512:

Tabela 17 - Resultados de simulações pós-processados com NS-2 - UDP 512

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
1	1 via sem perdas; UDP 512; GOP = 4	45,79993289	24,57112	98.7 / 84	87,7	2,25
2	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 4	45,79896667	29,92742	938.3 / 85.8	87,7	3,29
3	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 4	45,79896667	27,80895	93.3 / 91.6	92	3,06
4	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 4	45,79896667	26,43413	94.7 / 86.2	88,3	2,63
5	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 4	45,79896667	12,7309	100 / 88.4	91,3	1
6	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 4	45,79896667	27,54678	96 / 88	90	3,03
7	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 4	45,79896667	27,54678	96 / 88	90	3,03
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
9	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29

Tabela 18 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-1)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
10	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
11	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
12	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 512; GOP = 4	45,79896667	29,92742	93.3 / 85.8	87,7	3,29
13	1 via sem perdas; UDP 512; GOP = 6	45,71043624	24,29688	98 / 77.2	80,7	2,25
14	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP=6	45,70963333	29,88737	92 / 85.2	86,3	3,34
15	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 6	45,70963333	31,08986	86 / 80.8	81,7	3,45
16	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 6	45,70963333	31,11372	86 / 79.6	80,7	3,45
17	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 6	45,70963333	13,30232	98 / 83.6	86	1,08
18	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 6	45,79896667	30,29751	96 / 89.2	90,3	2,4
19	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 6	45,70963333	25,22884	96 / 87.6	89	2,4

Tabela 19 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-2)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
20	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP (230,4kb;115,2kb;38,4kb) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
21	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
22	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
23	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 512; GOP = 6	45,70963333	29,88737	92 / 85.2	86,3	3,34
24	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 512; GOP = 6	45,70963333	26,52863	94 / 87.2	88,3	2,57
25	1 via sem perdas; UDP 512; GOP = 8	45,66863333	24,4812	97.4 / 77.5	80	2,27
26	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP=8	45,66863333	27,7766	89.5 / 85.5	86	2,94
27	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 8	45,66863333	27,62721	92.1 / 84	85	2,93
28	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 8	45,66863333	27,58902	92.1 / 84	85	2,9
29	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 8	45,66863333	12,73445	100 / 88.5	90	1

Tabela 20 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-3)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
30	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 8	45,66863333	23,91619	97.4 / 88.5	89,7	2,4
31	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 8	45,66863333	24,03739	97.4 / 87	88,3	2,42
32	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 512; GOP = 8	45,66863333	27,7766	89.5 / 85.5	86	2,94
33	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 512; GOP = 8	45,66863333	27,7766	89.5 / 85.5	86	2,94
34	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 512; GOP = 8	45,66863333	27,7766	89.5 / 85.5	86	2,94
35	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 512; GOP = 8	45,66863333	27,7766	89.5 / 85.5	86	2,94
36	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 512; GOP = 8	45,66863333	27,74333	89.5 / 85.5	86	2,94
37	1 via sem perdas; UDP 512; GOP = 16	45,57866667	24,64441	94.7 / 76.2	77,3	2,28
38	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88

Tabela 21 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-4)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
39	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 512; GOP = 16	45,57866667	27,90431	84.2 / 86.1	86	2,87
40	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88
41	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 16	45,57866667	12,71625	100 / 88.3	89	1
42	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 16	45,57866667	14,34485	94.7 / 86.1	86,7	1,23
43	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 512; GOP = 16	45,57866667	26,07047	89.5 / 86.1	86,3	2,65
44	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88
45	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88
46	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88

Tabela 22 - Resultados de simulações pós-processados com NS-2 - UDP 512 (continuação-5)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
47	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88
48	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 512; GOP = 16	45,57866667	27,89171	84.2 / 85.8	85,7	2,88

Figura 61 - 1 via sem perdas; UDP 512; GOP = 4

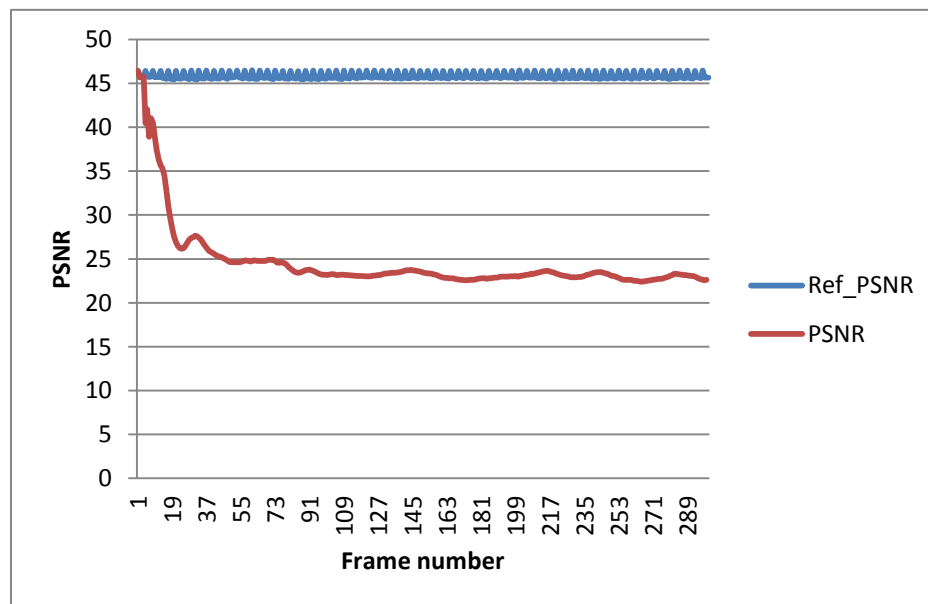


Figura 62 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 4

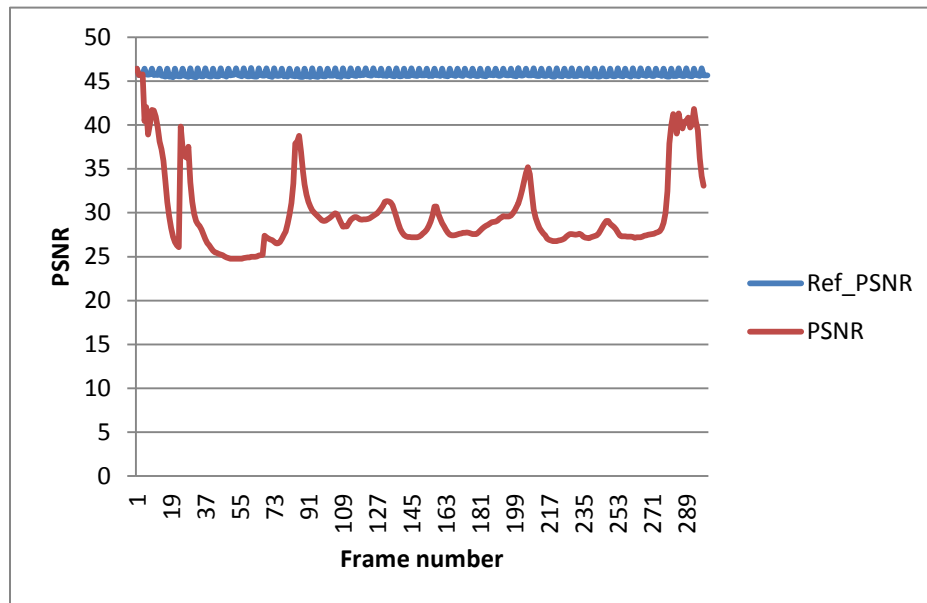


Figura 63 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 512; GOP = 4

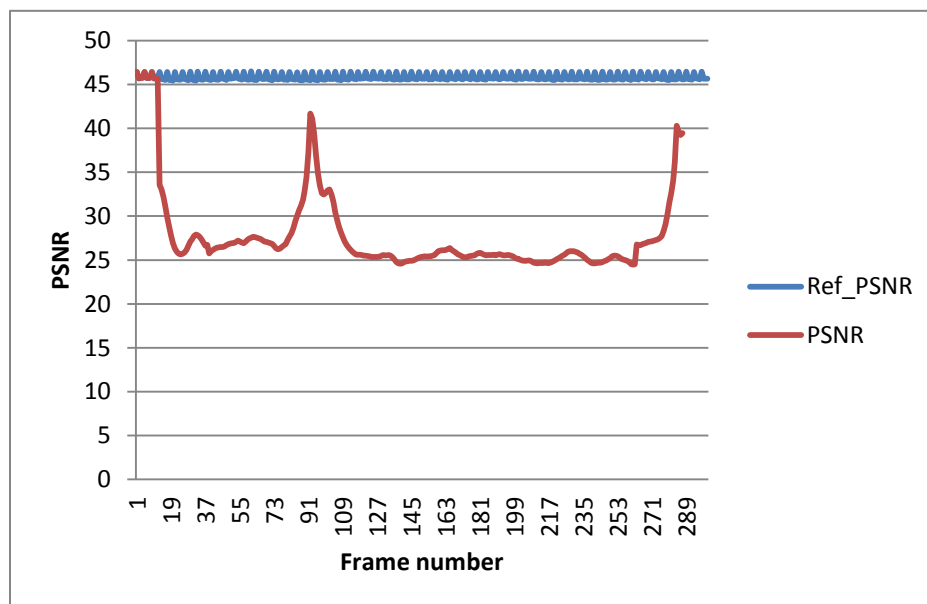


Figura 64 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 512; GOP = 4

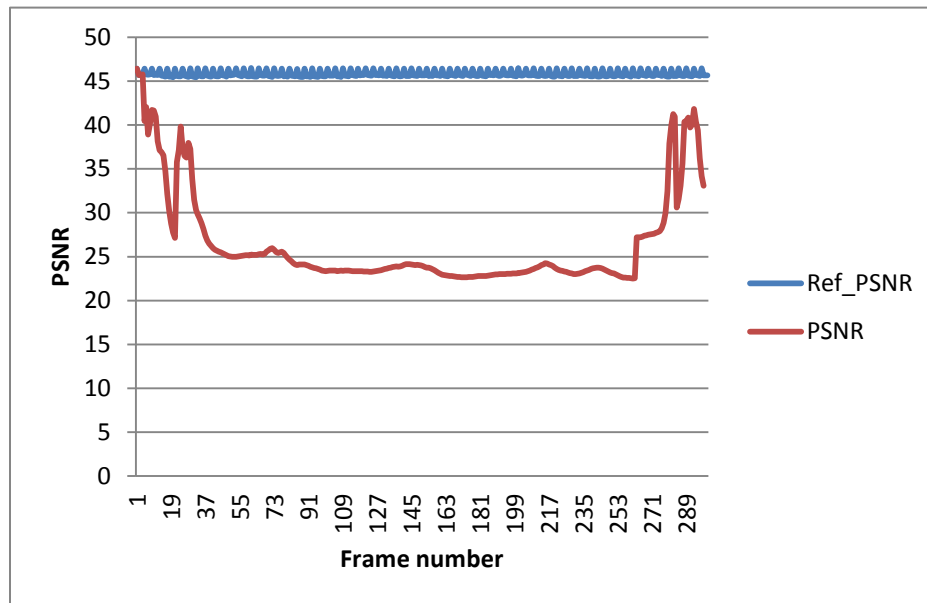


Figura 65 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 4

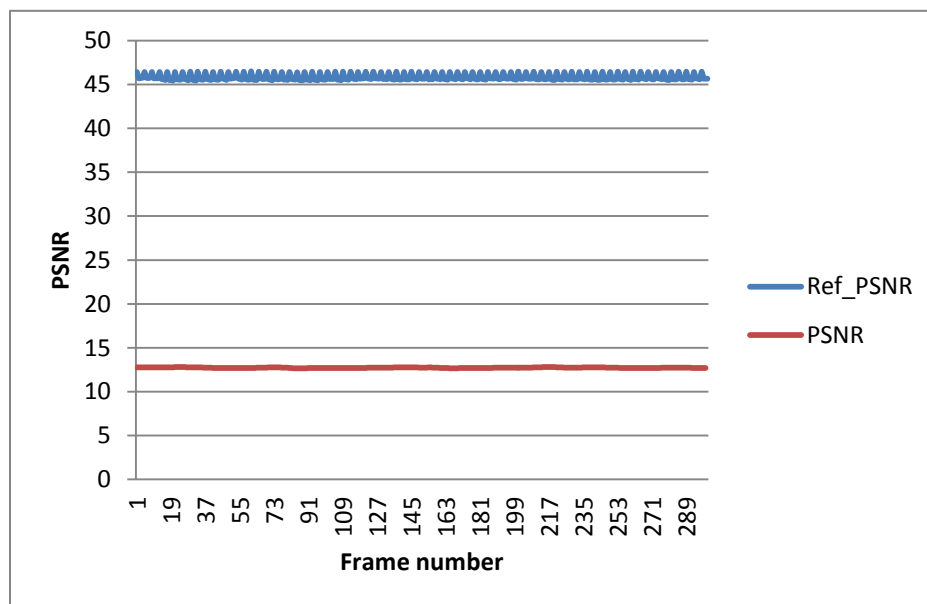


Figura 66 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 4

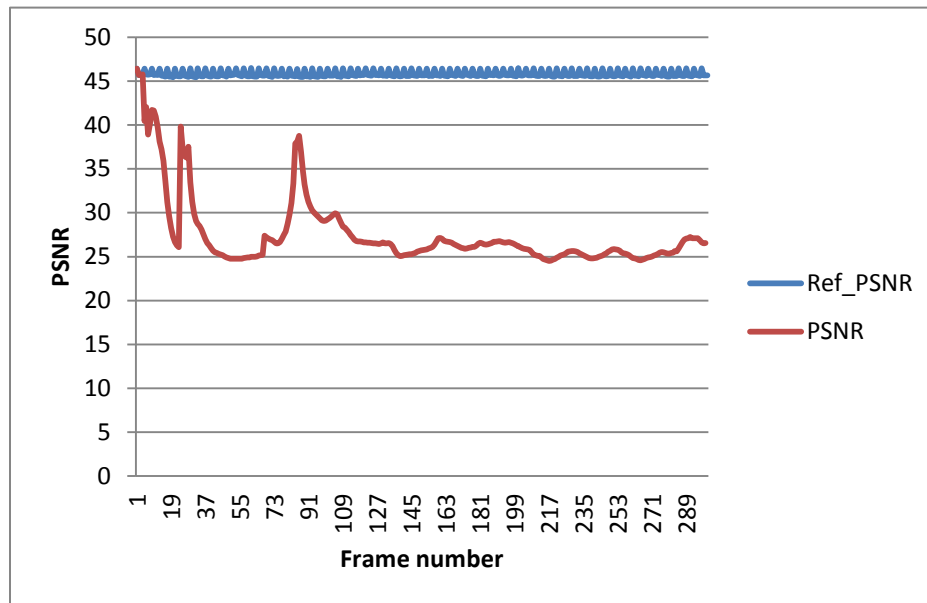


Figura 67 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 4

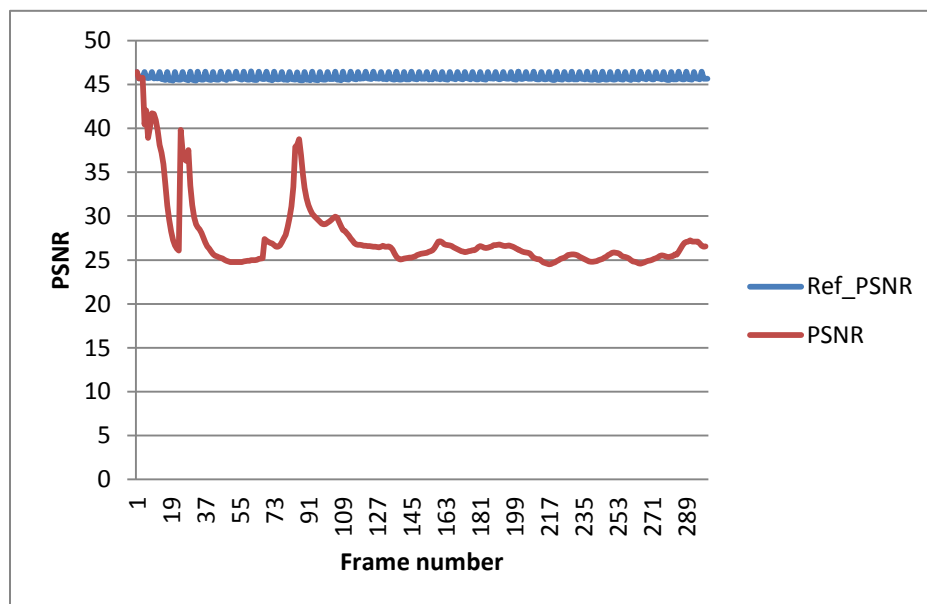


Figura 68 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP(230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 4

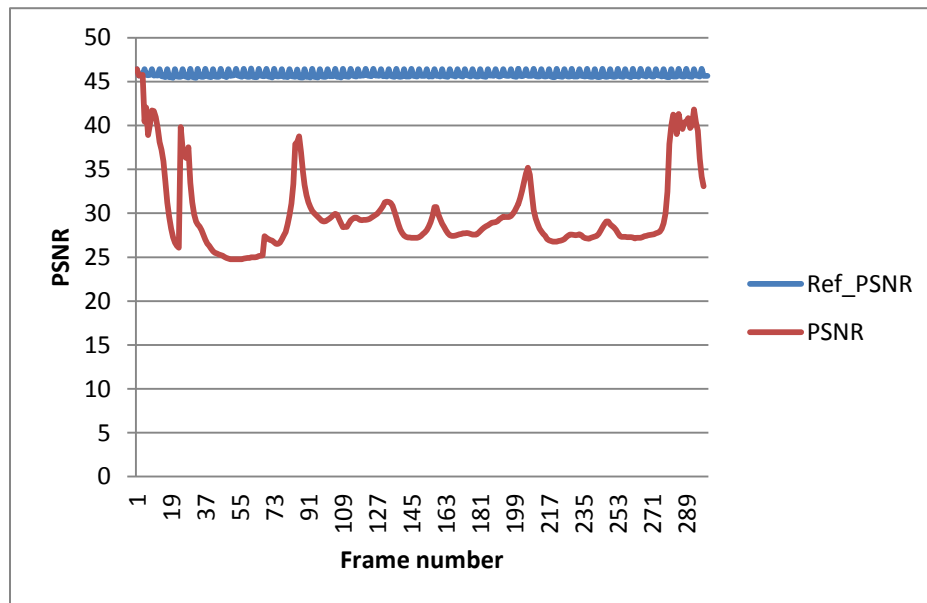


Figura 69 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 4

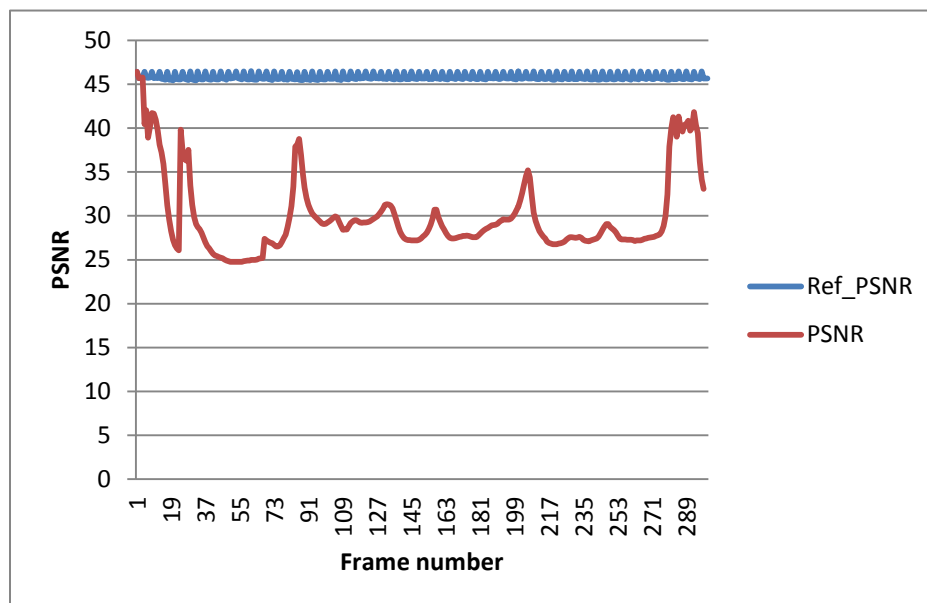


Figura 70 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 512; GOP = 4

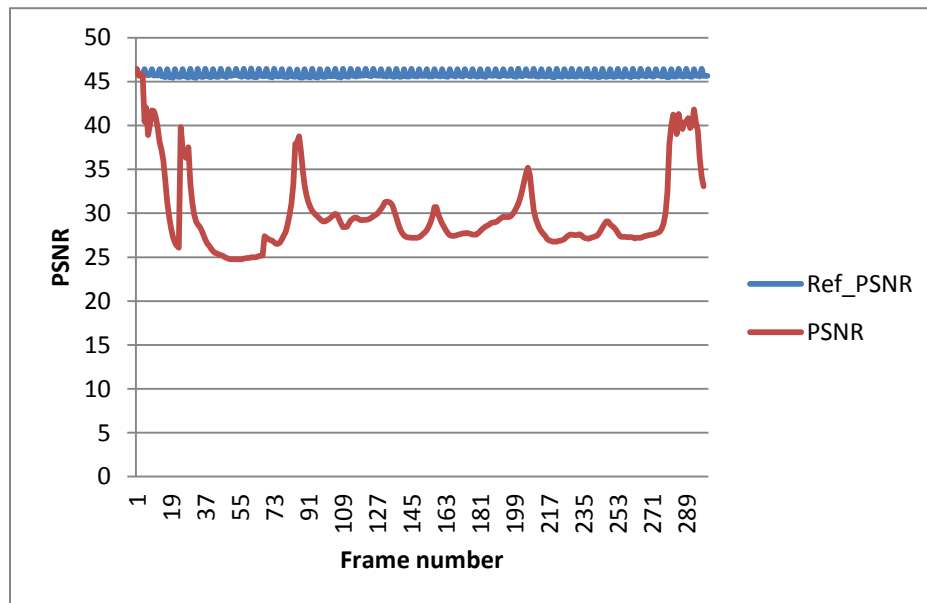


Figura 71 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 512; GOP = 4

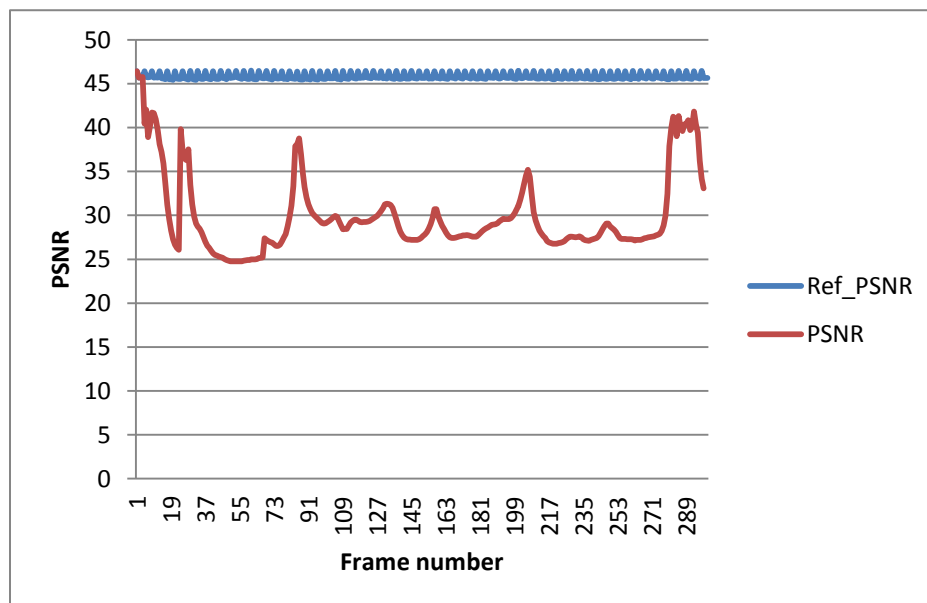


Figura 72 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 512; GOP = 4

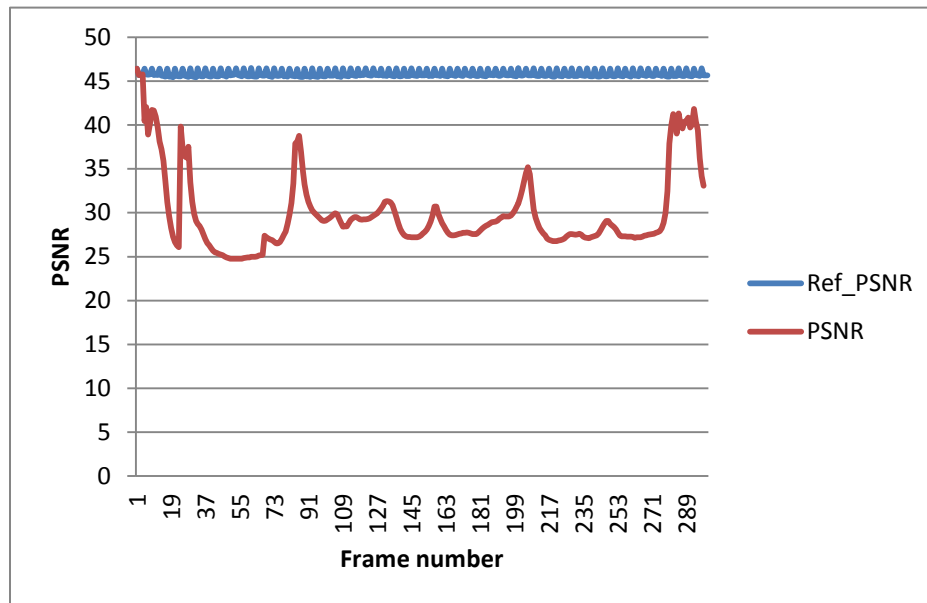


Figura 73 - 1 via sem perdas; UDP 512; GOP = 6

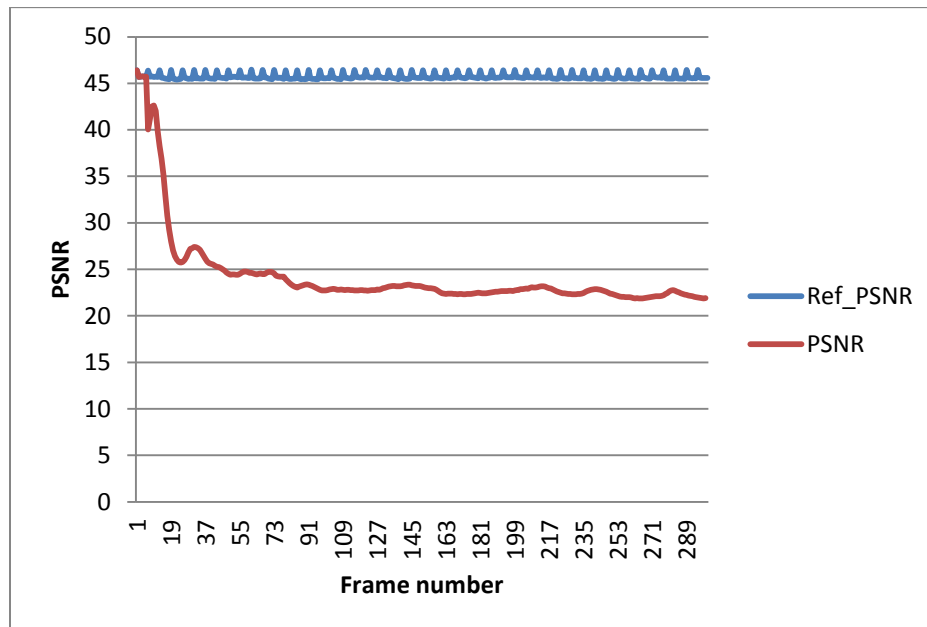


Figura 74 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 512; GOP=6

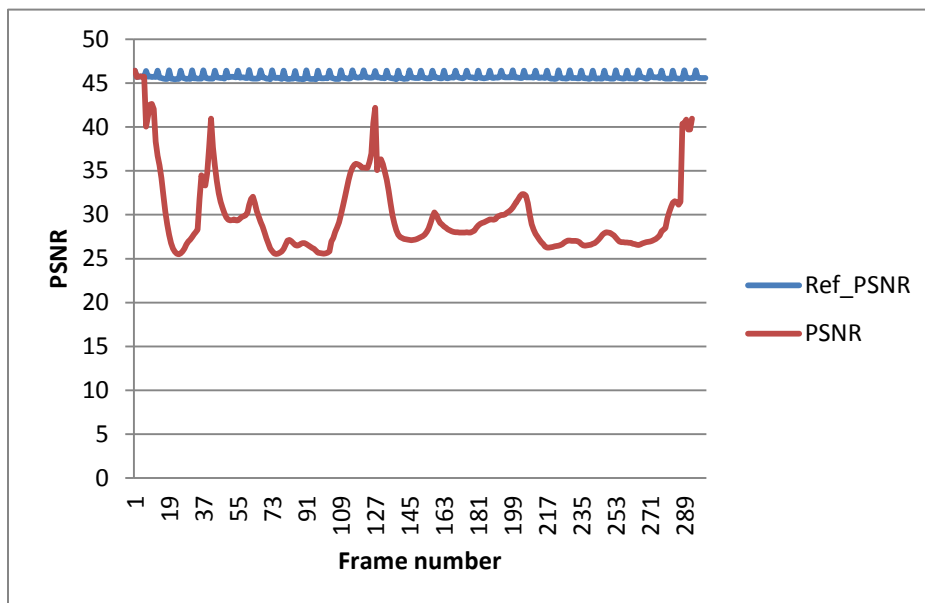


Figura 75 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 512; GOP = 6

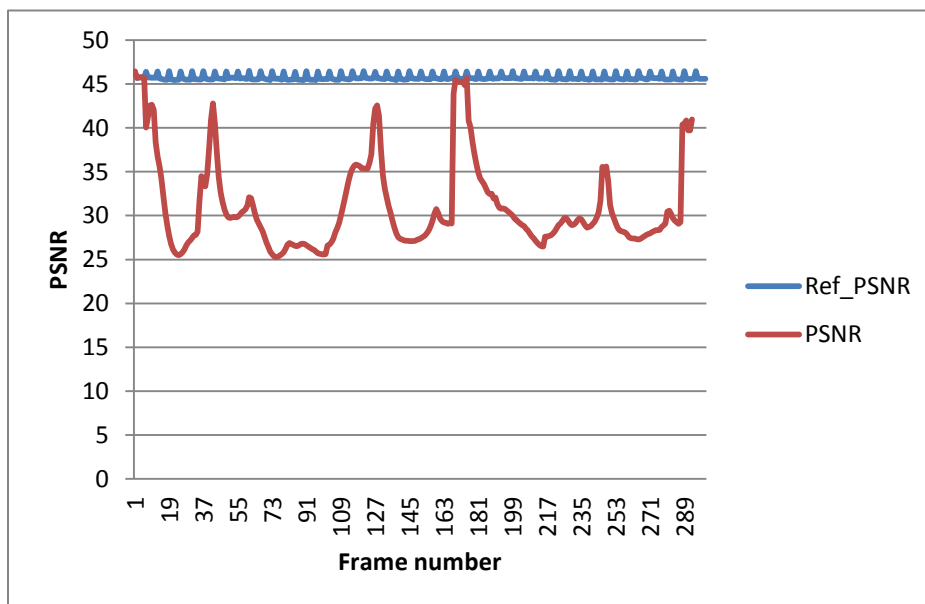


Figura 76 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 512; GOP = 6

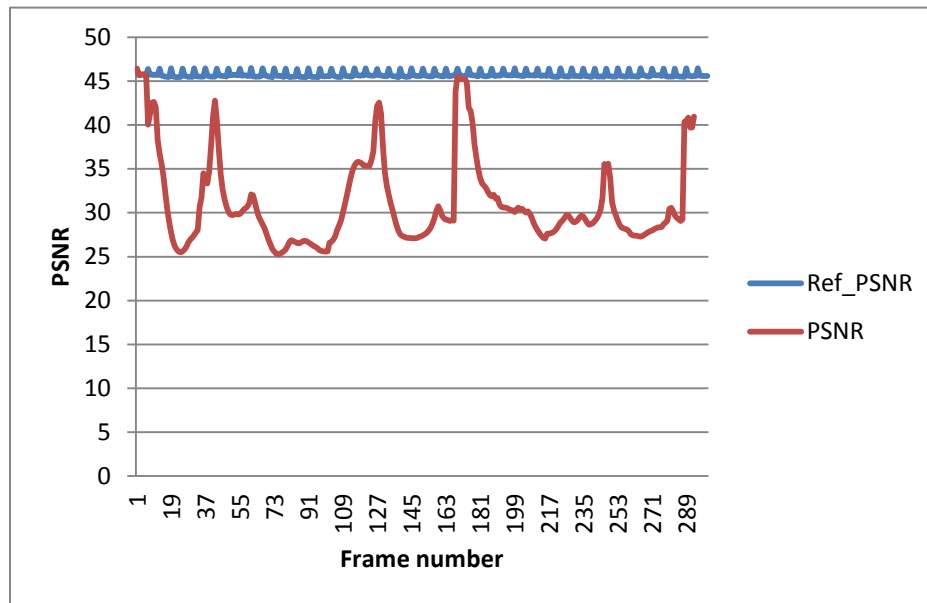


Figura 77 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP =6

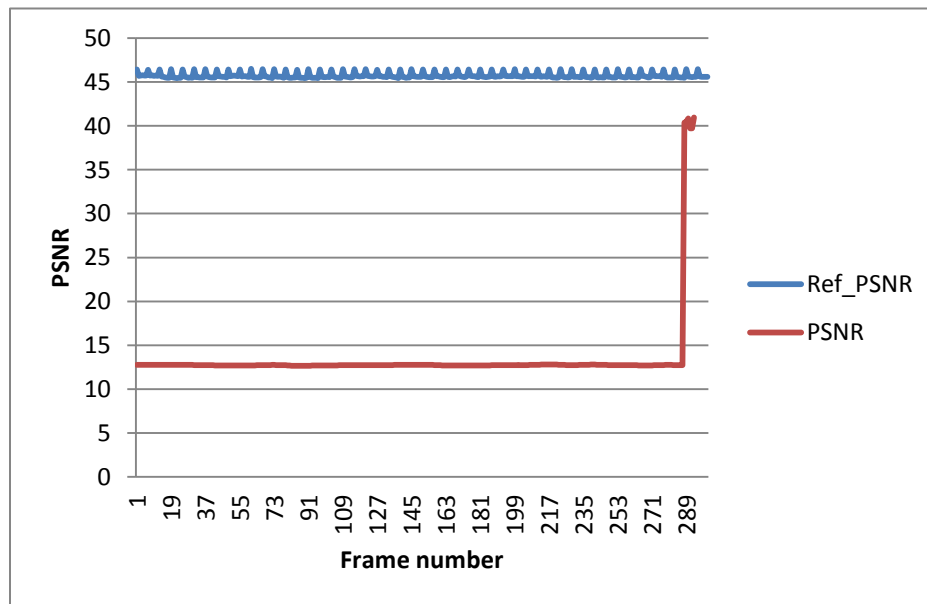


Figura 78 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP =6

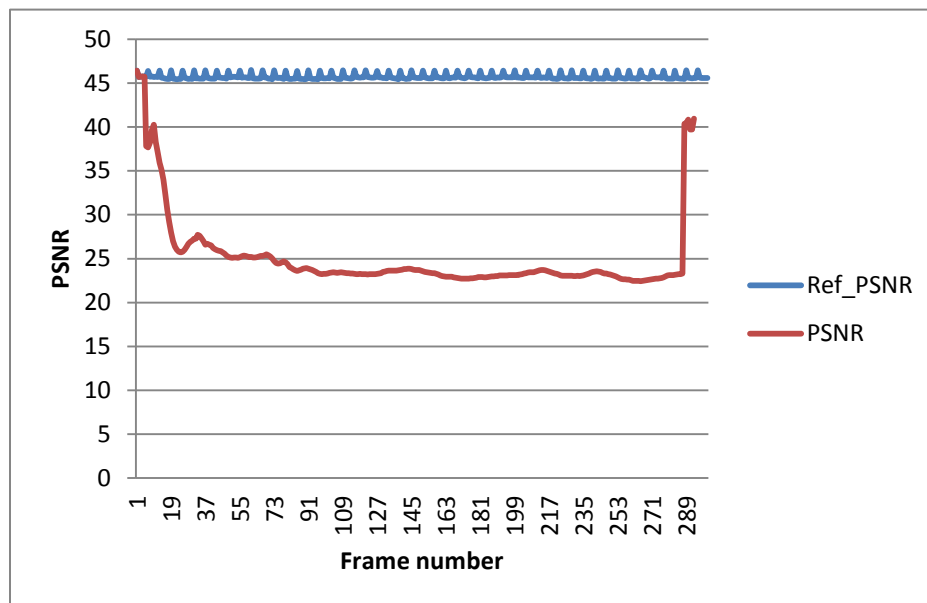


Figura 79 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 6

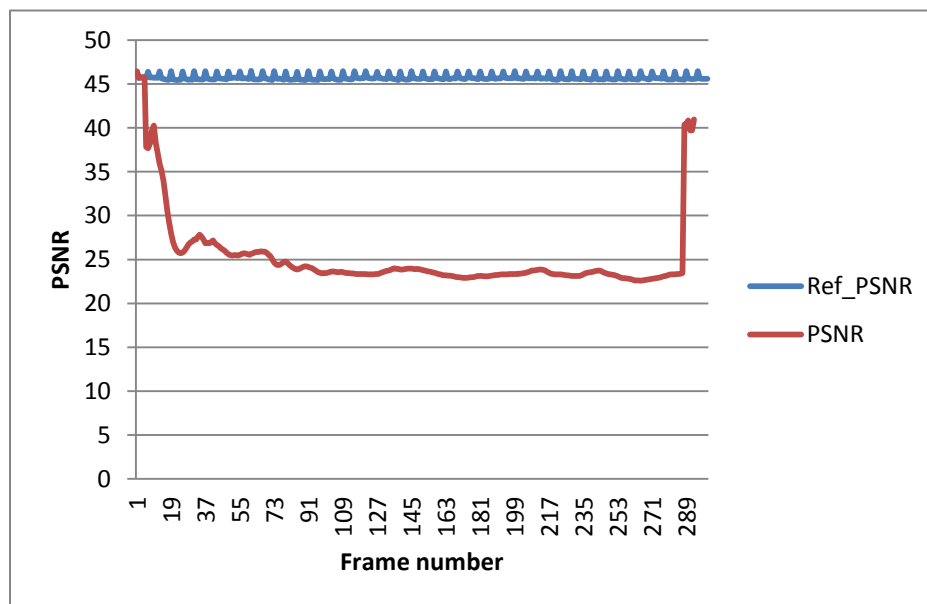


Figura 80 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb)
UDP 512; GOP = 6

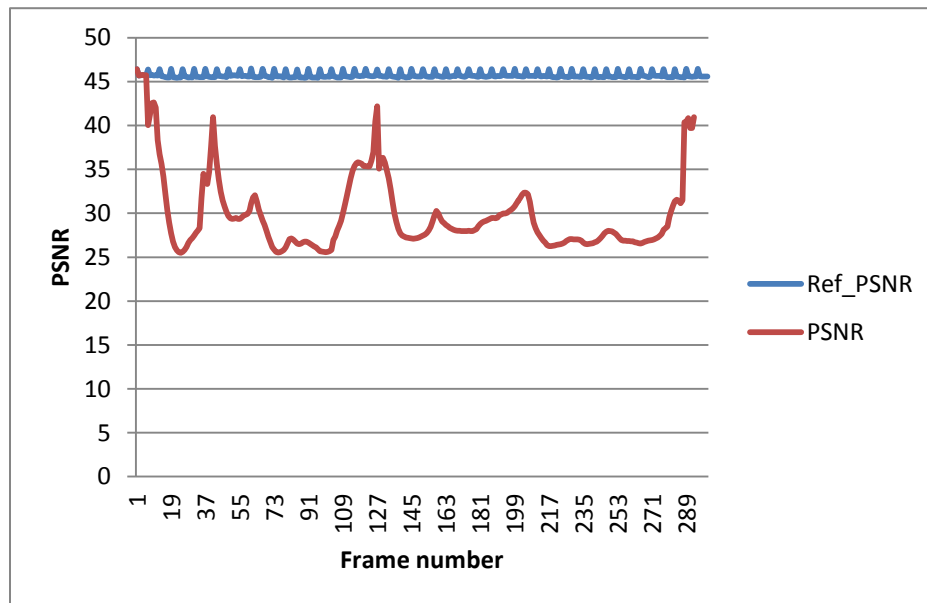


Figura 81 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 6

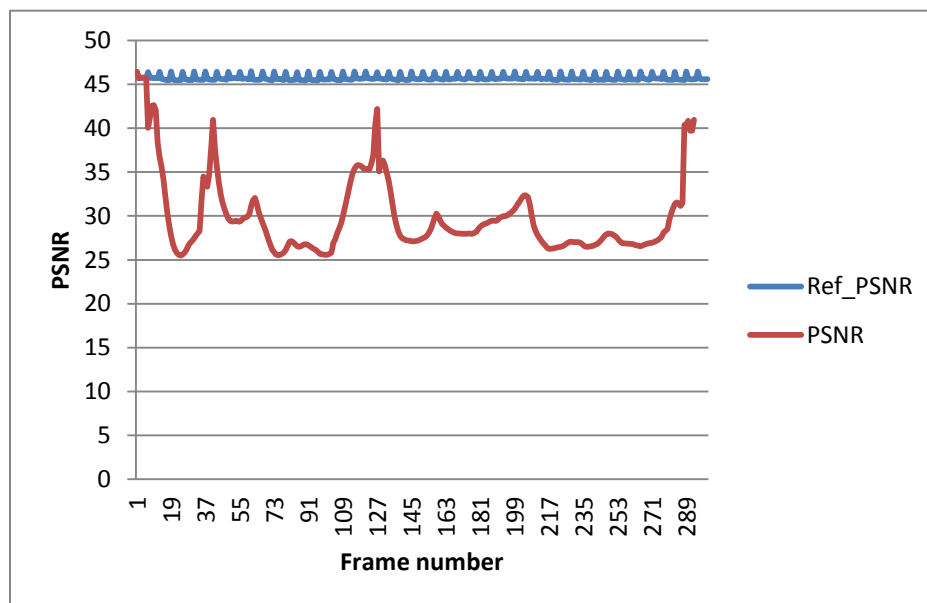


Figura 82 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb;) UDP 512; GOP = 6

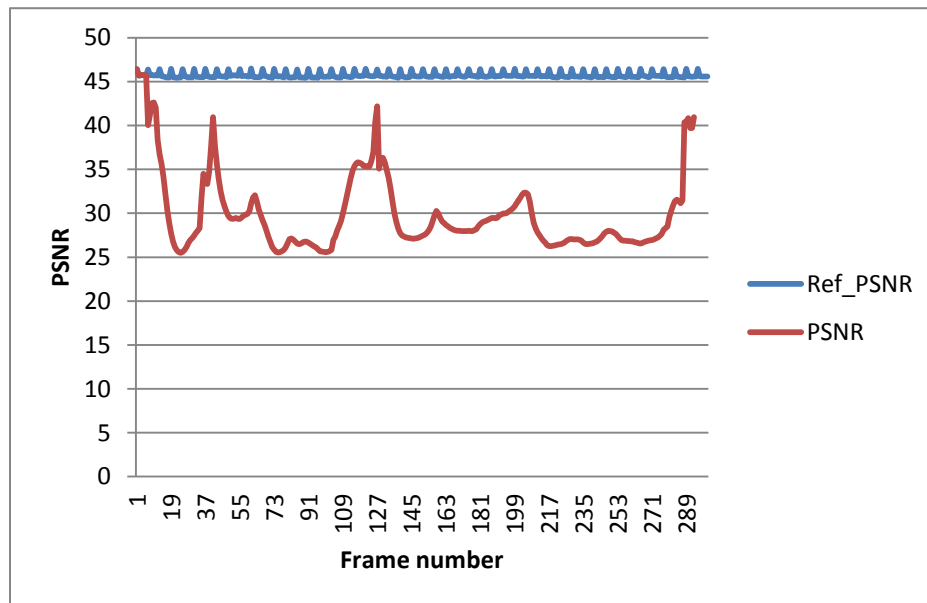


Figura 83 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb) UDP 512; GOP = 6

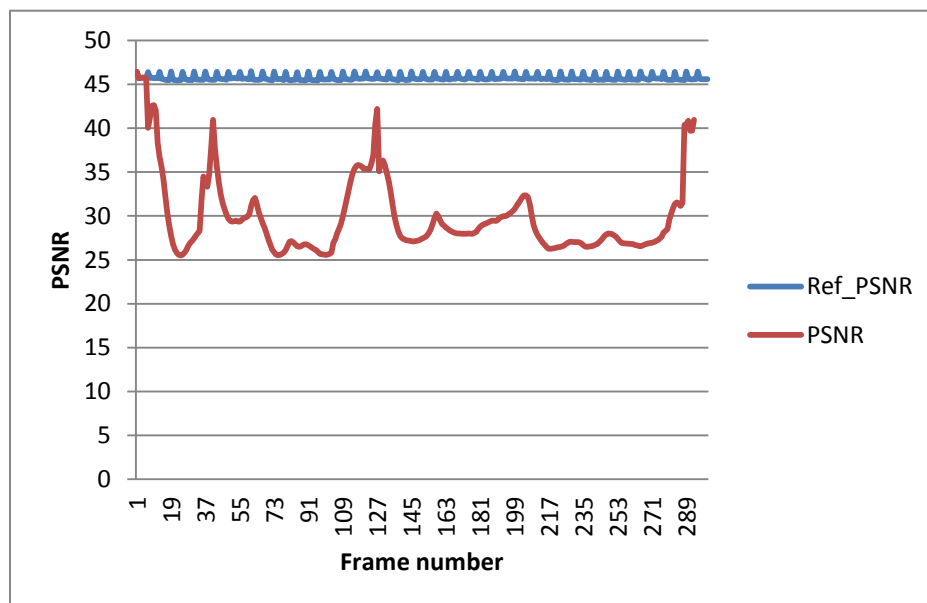


Figura 84 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 512; GOP = 6

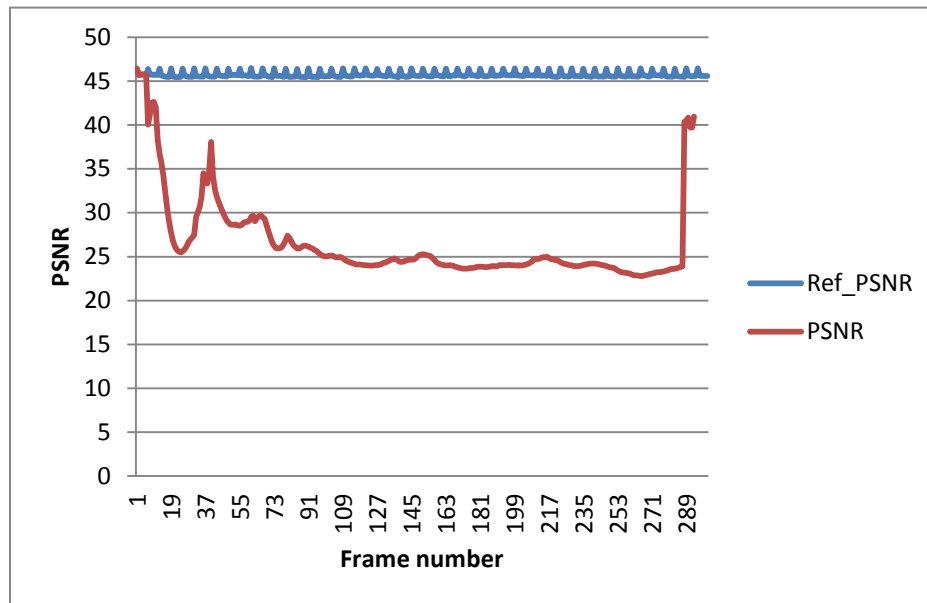


Figura 85 - 1 via sem perdas; UDP 512; GOP = 8

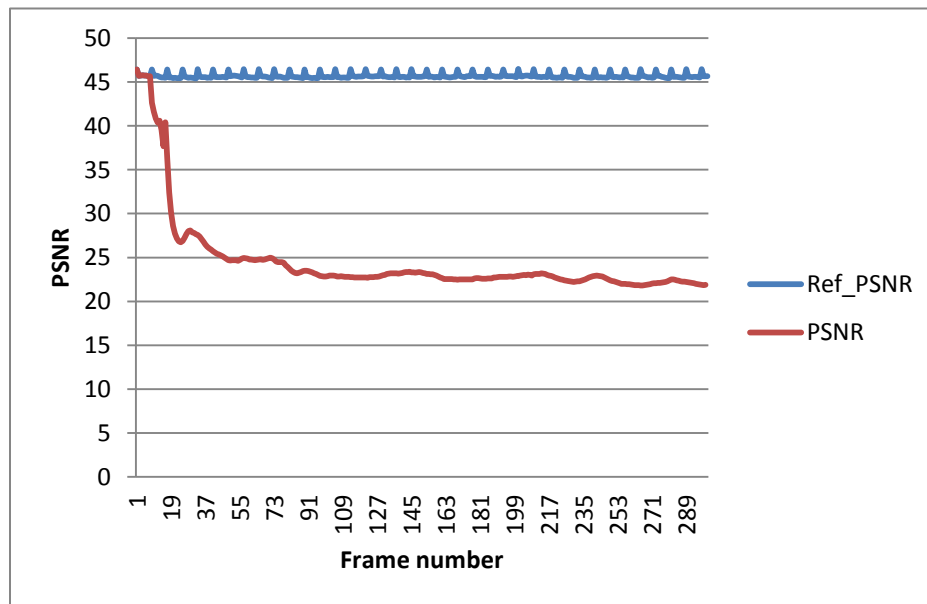


Figura 86 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 512; GOP=8

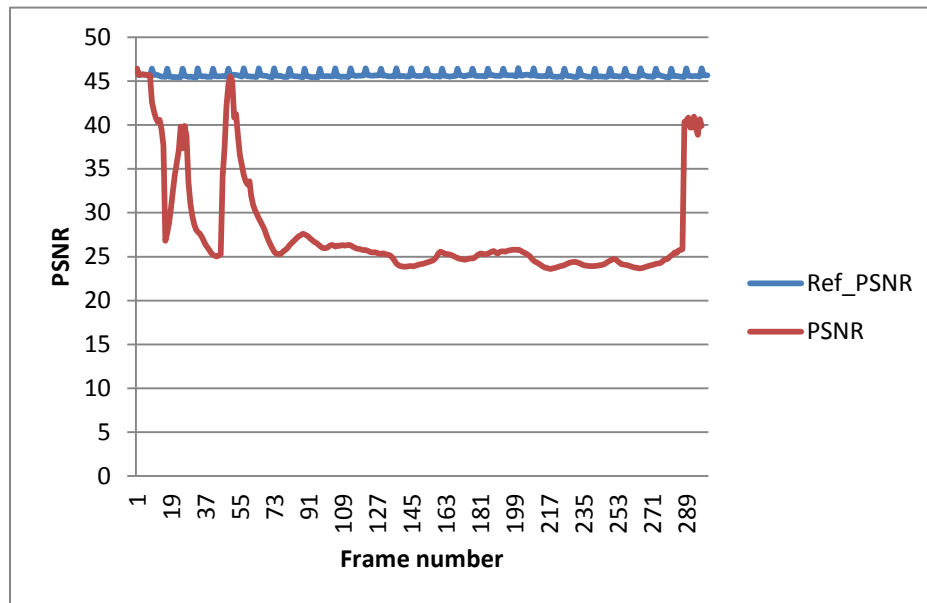


Figura 87 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 512; GOP = 8

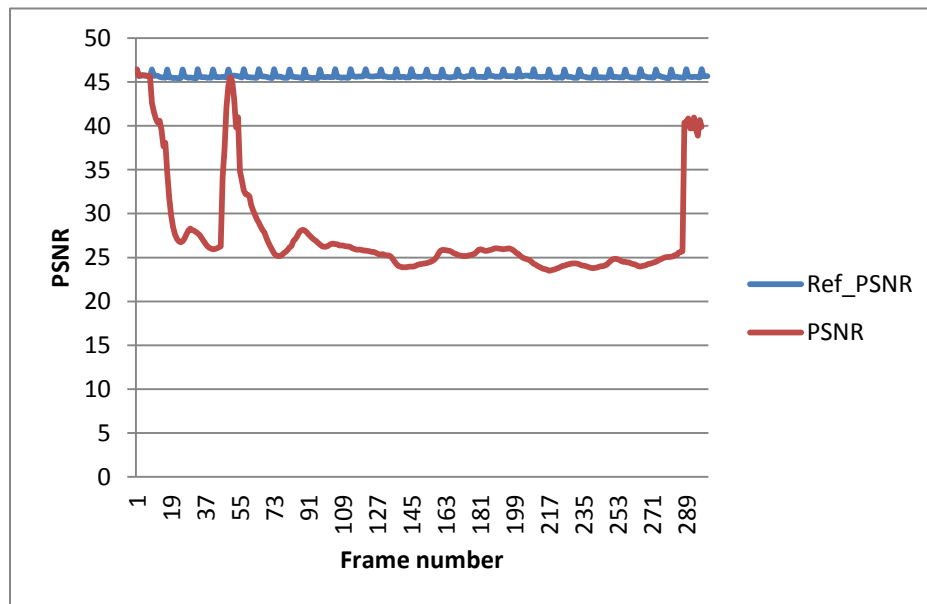


Figura 88 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 512; GOP = 8

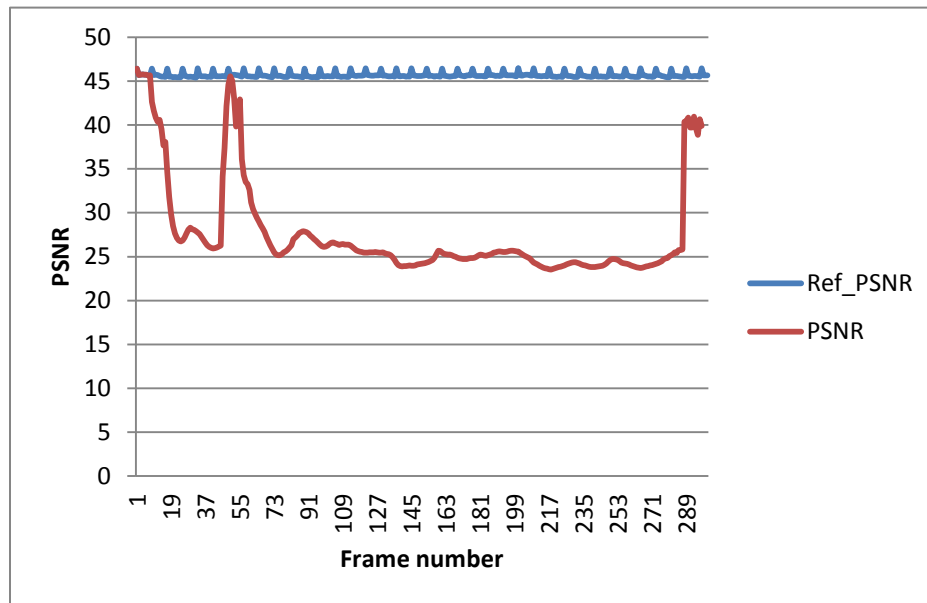


Figura 89 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP= 8

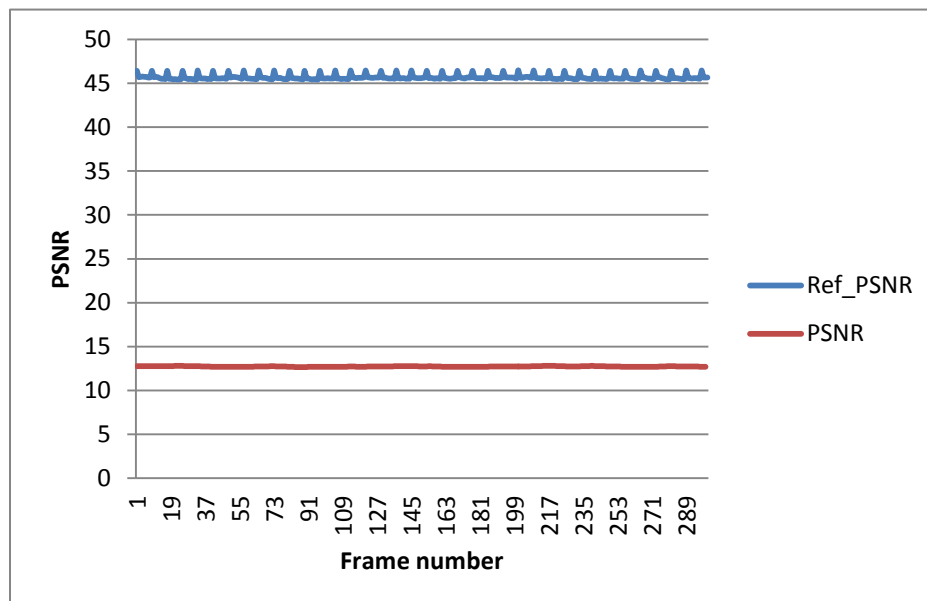


Figura 90 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP= 8

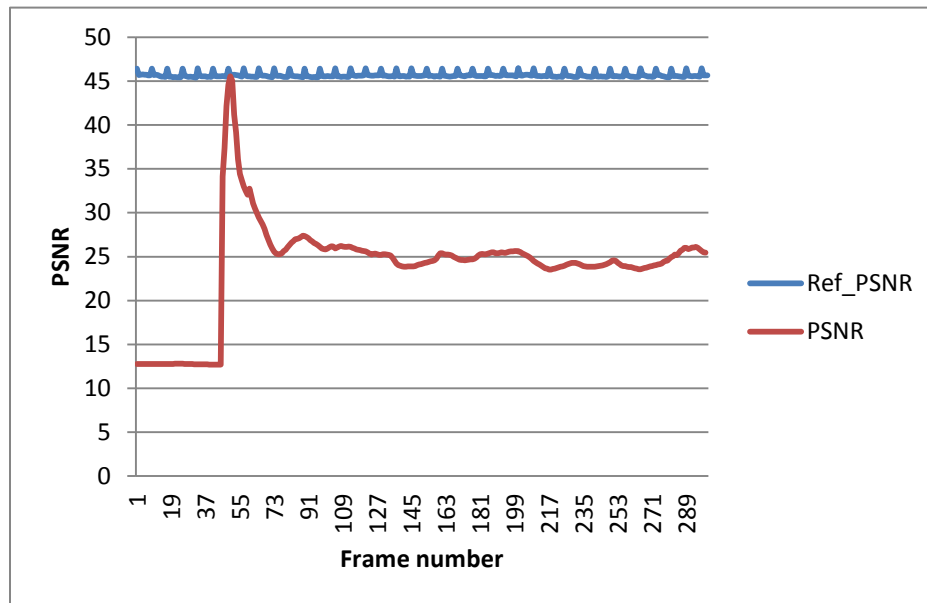


Figura 91 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 8

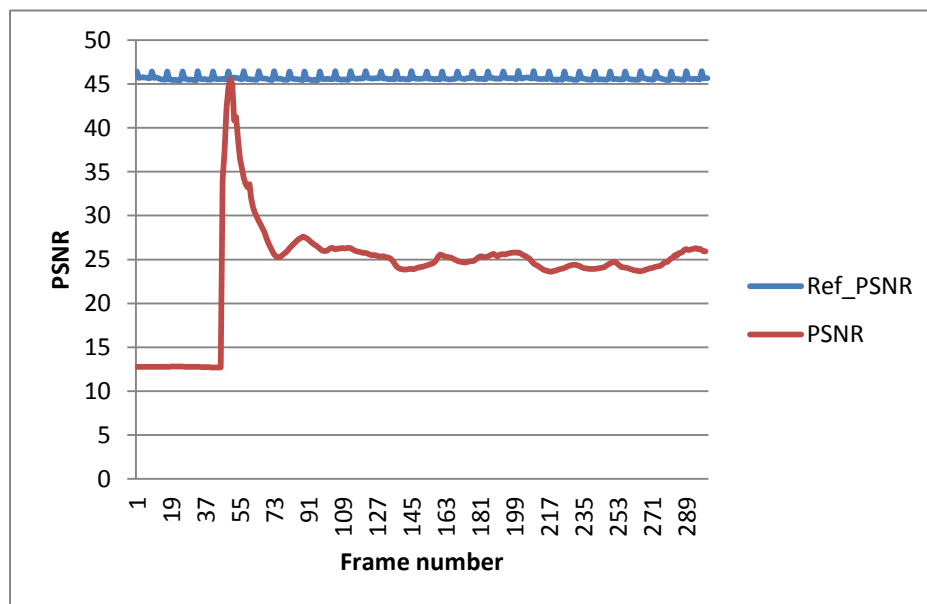


Figura 92 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 8

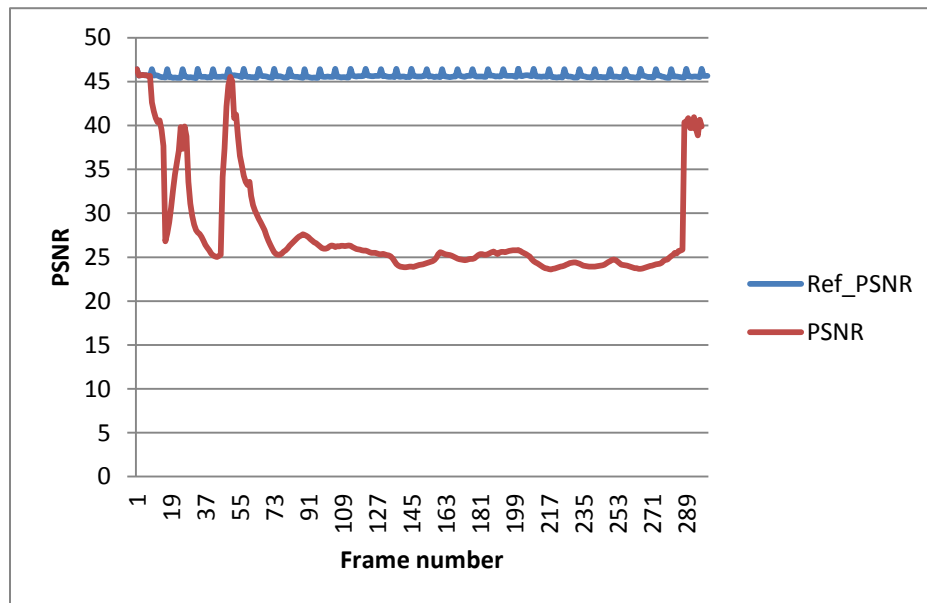


Figura 93 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 8

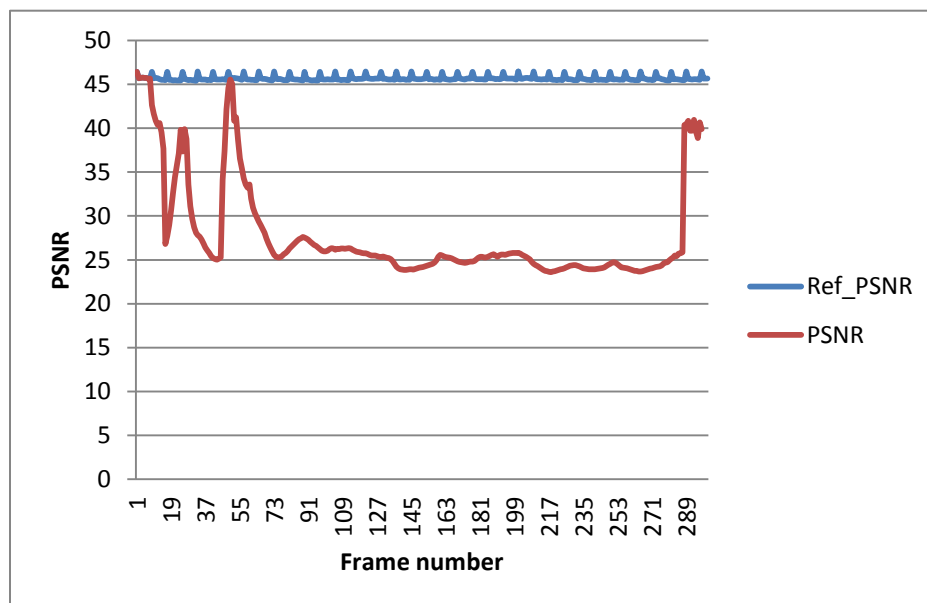


Figura 94 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb)
UDP 512; GOP = 8

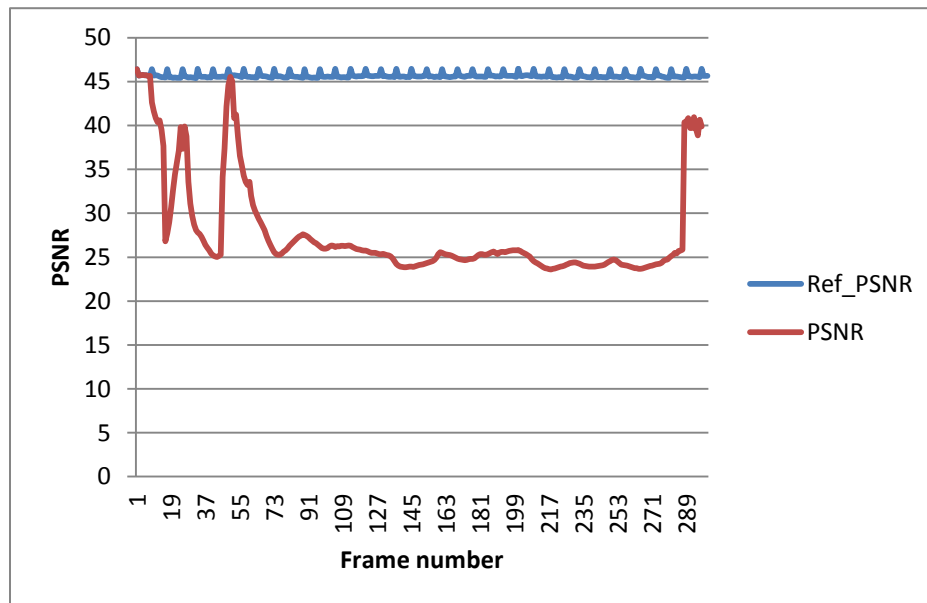


Figura 95 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb)
UDP 512; GOP = 8

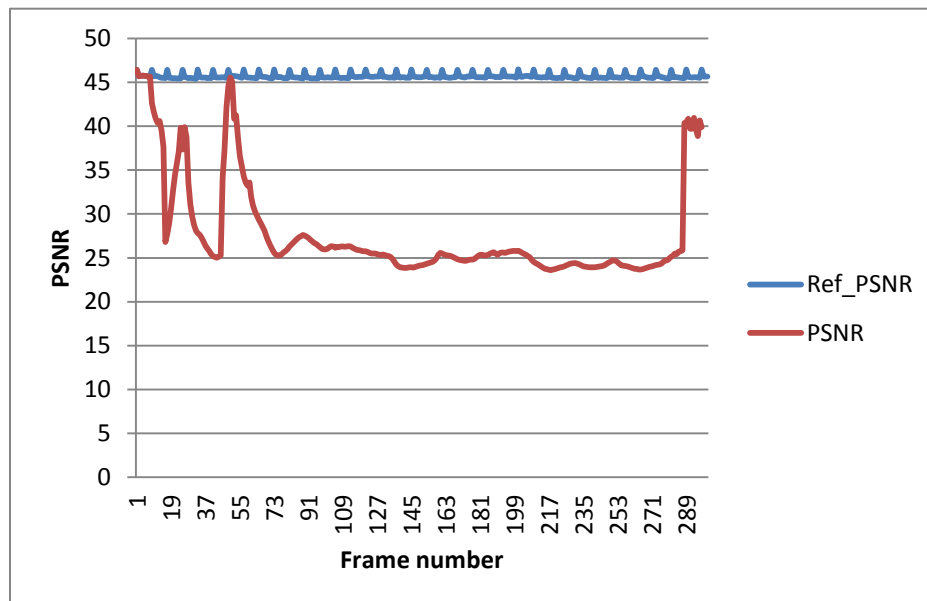


Figura 96 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 512; GOP = 8

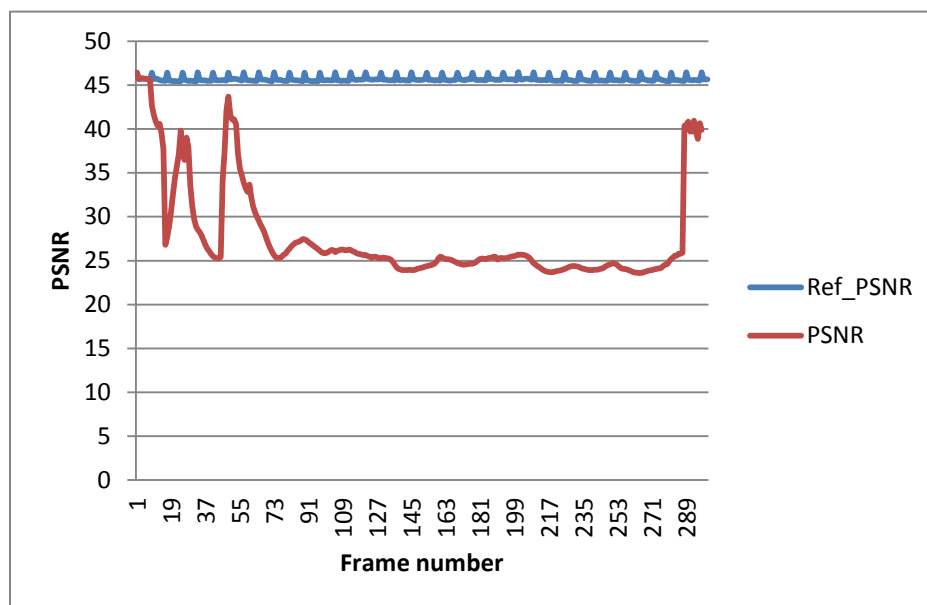


Figura 97 - 1 via sem perdas; UDP 512; GOP = 16

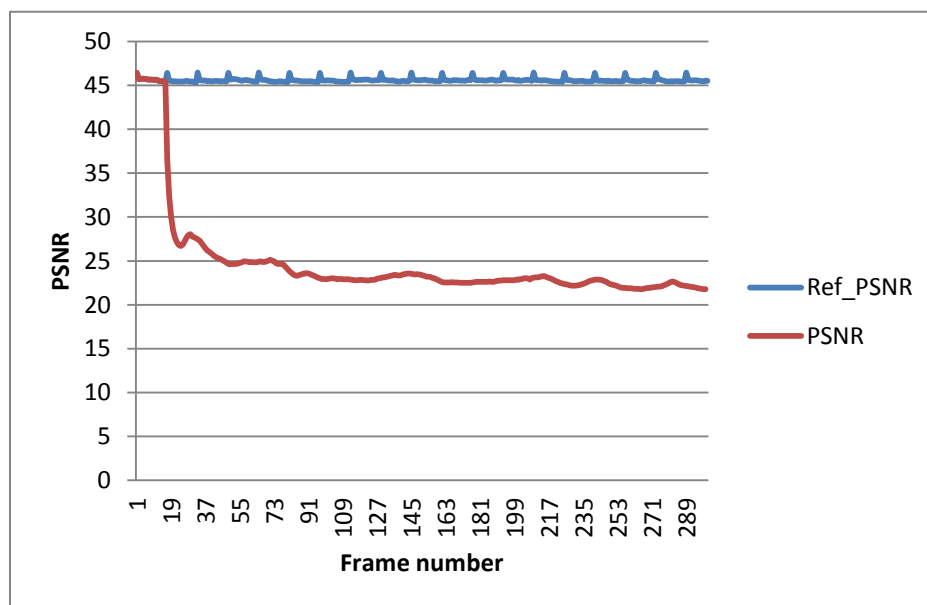


Figura 98 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 16

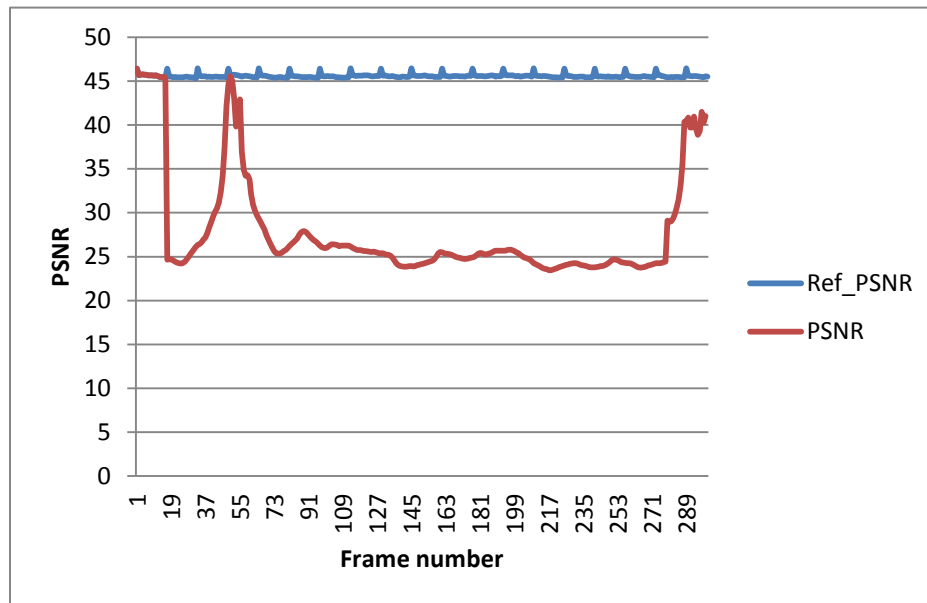


Figura 99 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 512; GOP = 16

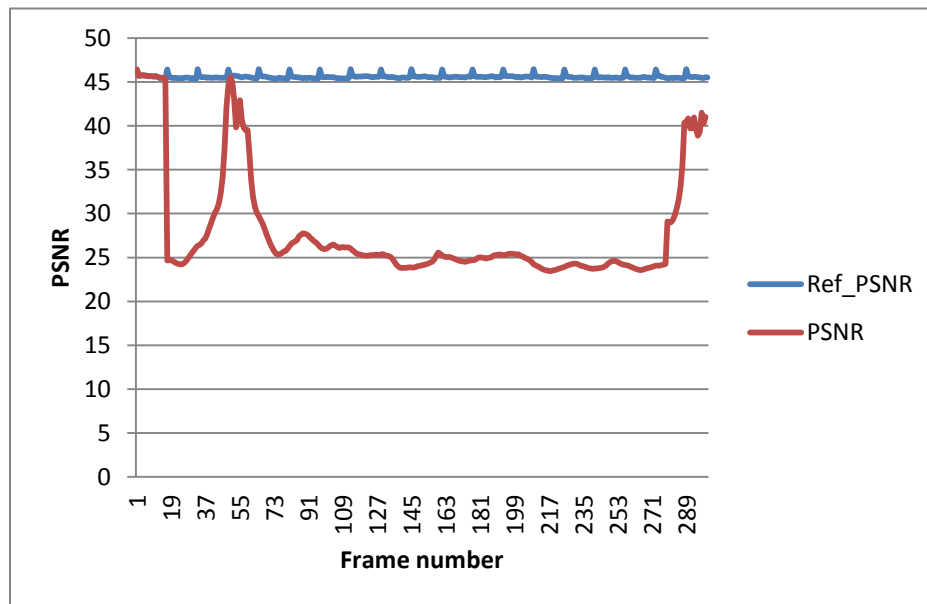


Figura 100 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 512; GOP = 16

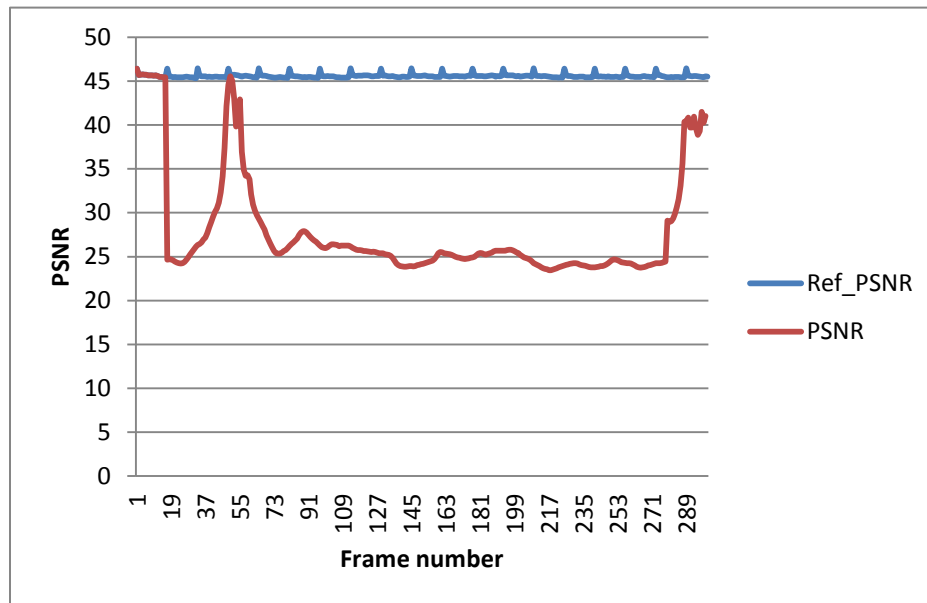


Figura 101 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP= 16

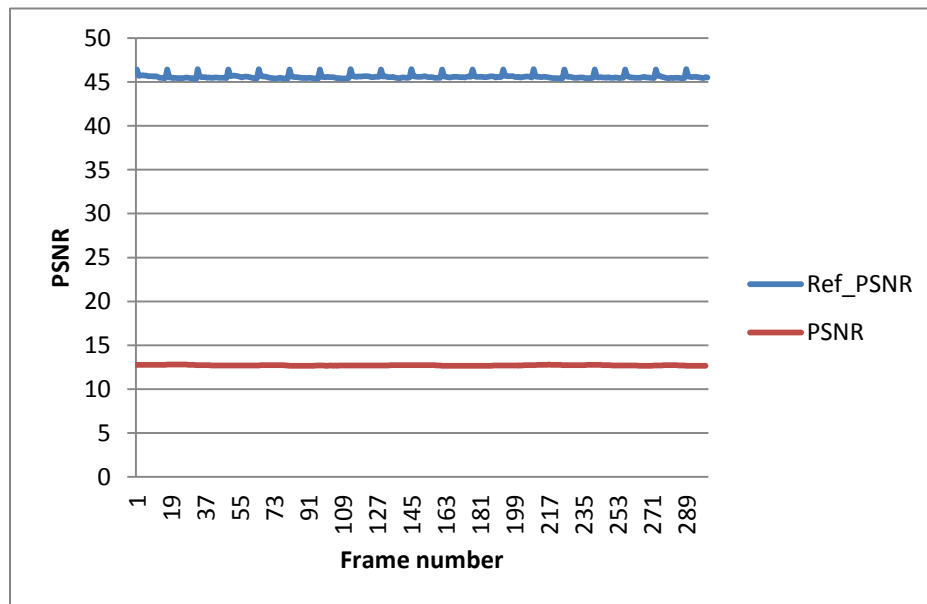


Figura 102 - vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 16

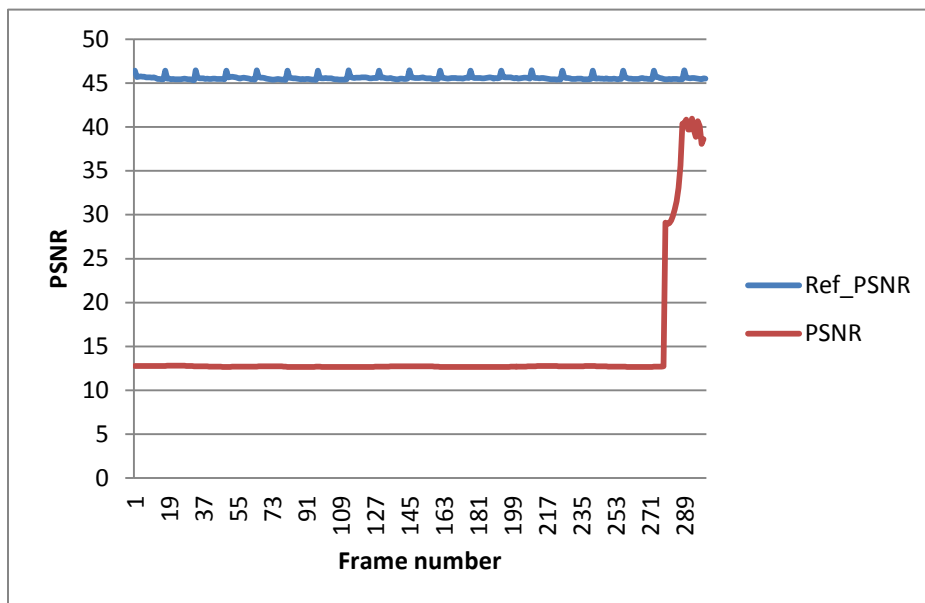


Figura 103 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 512; GOP = 16

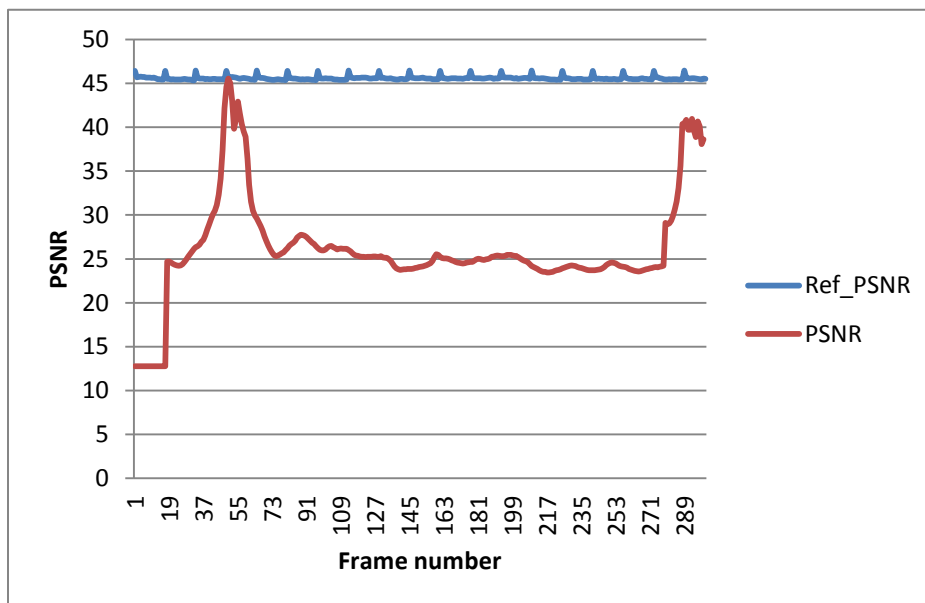


Figura 104 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 512; GOP = 16

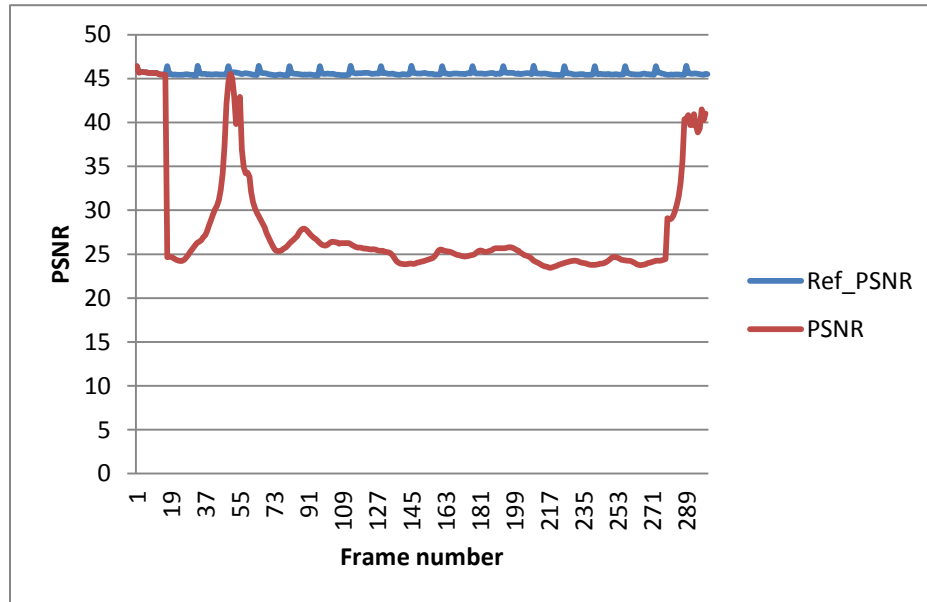


Figura 105 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb) UDP 512; GOP = 16

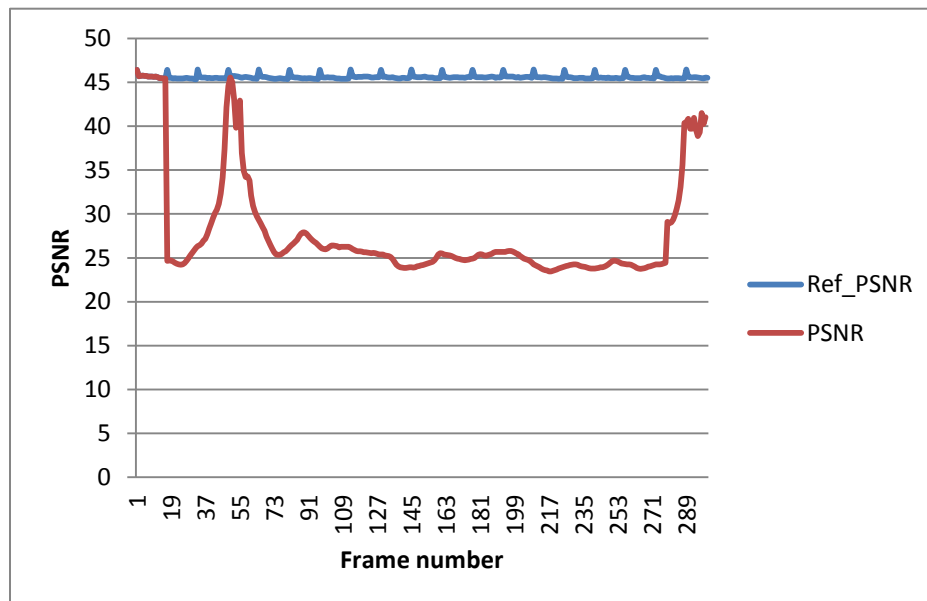


Figura 106 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb)
UDP 512; GOP = 16

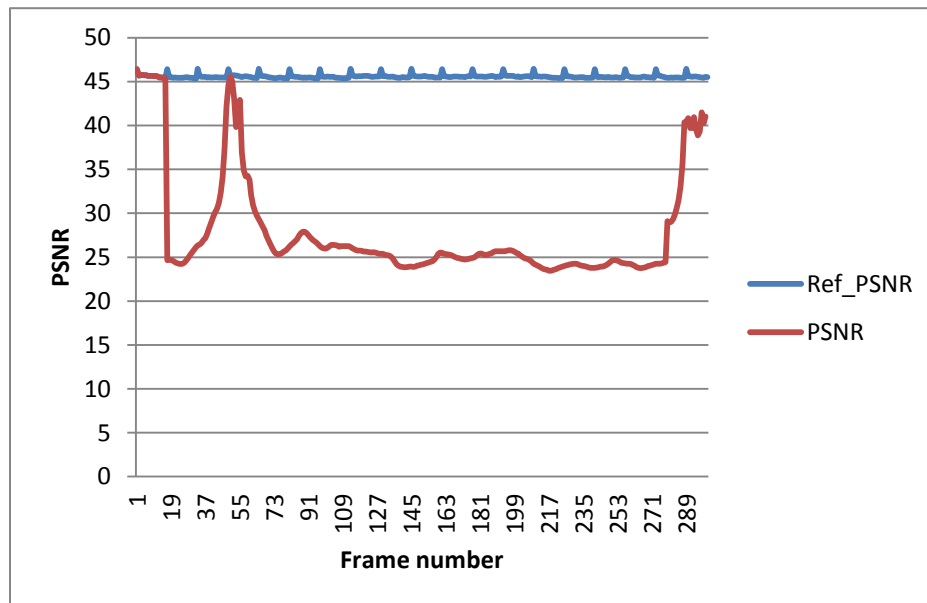


Figura 107 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb);
UDP 512; GOP = 16

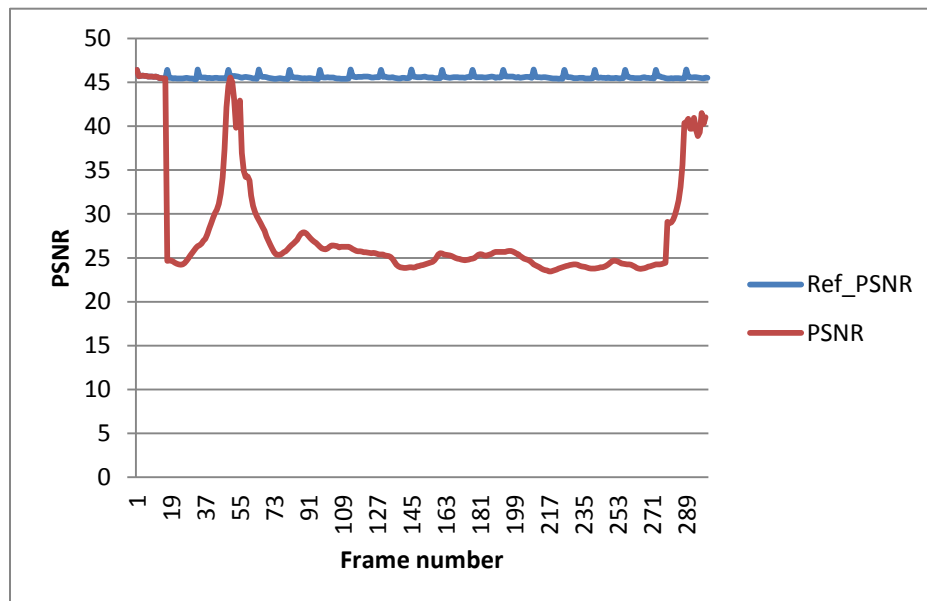
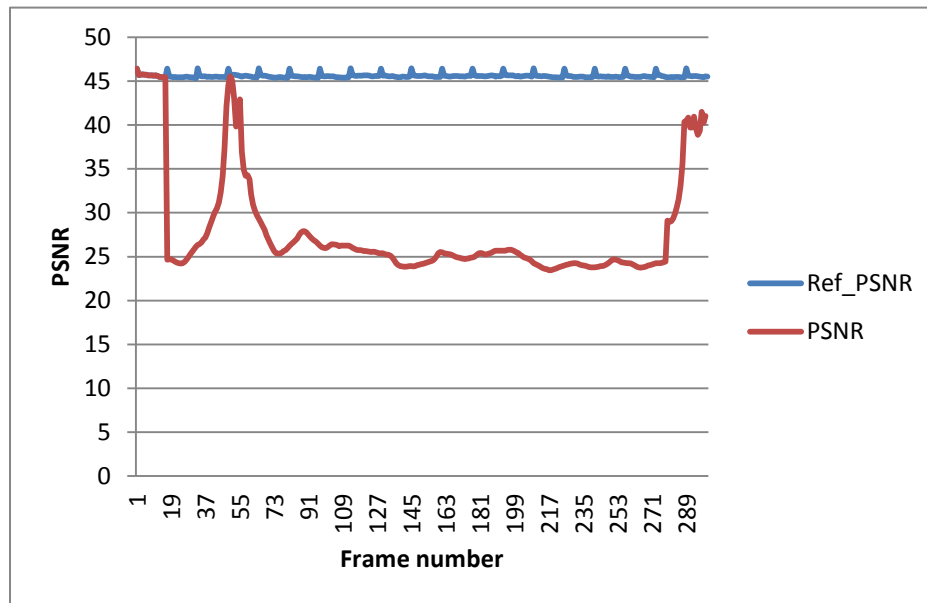


Figura 108 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 512; GOP = 16



1.1.2 UDP Frame 256:

Tabela 23 - Resultados de simulações pós-processados com NS-2 - UDP 256

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
1	1 via sem perdas; UDP 256; GOP = 4	45,79993289	24,57112	98.7 / 84	87,7	2,25
2	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
3	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 256; GOP = 4	45,79896667	28,36344	94.7 / 88	89,7	3,09
4	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 256; GOP = 4	45,79896667	28,2491	94.7 / 86.2	88,3	3,08
5	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 4	45,79896667	13,32081	98.7 / 92.4	94	1,08
6	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 4	45,79896667	21,64802	97.3 / 90.2	92	2,2
7	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 4	45,79896667	21,65423	97.3 / 89.8	91,7	2,21
8	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
9	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
10	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb) UDP 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08

Tabela 24 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-1)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
11	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 256; GOP = 4	45,79896667	28,45281	94.7 / 88.9	90,3	3,08
12	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 256; GOP = 4	45,79896667	25,69043	96 / 90.7	92	2,45
13	1 via sem perdas; UDP 256; GOP = 6	45,71043624	25,03486	96 / 82.4	84,7	2,34
14	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP=6	45,70963333	25,03486	92 / 90.4	90,7	3,2
15	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 256; GOP = 6	45,70963333	27,7699	92 / 84.4	85,7	2,89
16	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 256; GOP = 6	45,70963333	28,06094	92 / 85.2	86,3	2,83
17	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 6	45,70963333	13,30232	98 / 92.8	93,7	1,14
18	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 6	45,79896667	30,29751	98 / 92.8	93,7	1,14
19	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 6	45,70963333	26,9395	94 / 91.6	92	3,04
20	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2

Tabela 25 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-2)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
21	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
22	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb) UDP 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
23	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 256; GOP = 6	45,70963333	28,06094	92 / 90.4	90,7	3,2
24	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 256; GOP = 6	45,70963333	26,14839	94 / 90.4	91	2,51
25	1 via sem perdas; UDP 256; GOP = 8	45,66863333	25,01452	97.4 / 76.7	79,3	2,34
26	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP=8	45,66863333	24,78241	97.4 / 87.4	88,7	2,29
27	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 256; GOP = 8	45,66863333	24,70844	97.4 / 87.4	88,7	2,28
28	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 256; GOP = 8	45,66863333	24,69929	97.4 / 85.9	87,3	2,31
29	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 8	45,66863333	12,73003	100 / 90.5	91,7	1
30	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 8	45,66863333	12,72424	100 / 89.7	91	1

Tabela 26 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-3)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
31	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 8	45,66863333	12,72424	100 / 89.7	91	1
32	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 256; GOP = 8	45,66863333	24,78241	97.4 / 87.4	88,7	2,29
33	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 256; GOP = 8	45,66863333	24,78241	97.4 / 87.4	88,7	2,29
34	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb) UDP 256; GOP = 8	45,66863333	24,78241	97.4 / 87.4	88,7	2,29
35	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 256; GOP = 8	45,66863333	24,78241	97.4 / 87.4	88,7	2,29
36	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 256; GOP = 8	45,66863333	24,78488	97.4 / 86.6	88	2,29
37	1 via sem perdas; UDP 256; GOP = 16	45,57866667	25,56227	89.5 / 75.8	76,7	2,41
38	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
39	3 vias sem perdas com <i>delay</i> (240ms;500ms;700ms); UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
40	3 vias sem perdas com <i>delay</i> (360ms;750ms;1050ms); UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9

Tabela 27 - Resultados de simulações pós-processados com NS-2 – UDP 256 (continuação-4)

Item	Modelos de Rede	Average PSNR-Reference	Average PSNR	Frames Dropped (I/P) %	overall frame loss in %	MOS
41	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 16	45,57866667	12,72963	100 / 91.5	92	1
42	3 vias com perdas (0,04; 0,02; 0,01) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 16	45,57866667	12,72258	100 / 89	89,7	1
43	3 vias com perdas (0,02; 0,01; 0,005) com <i>delay</i> (120ms;250ms;350ms); UDP 256; GOP = 16	45,57866667	12,71264	100 / 87.9	88,7	1
44	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego TCP(230,4kb;115,2kb;38,4kb) UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
45	3 vias com perdas (0,08; 0,04; 0,02) com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (230,4kb;115,2kb;38,4kb) UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
46	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (115,2kb;57,6kb;19,2kb;) UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
47	3 vias sem perdas com <i>delay</i> (120ms;250ms;350ms) e tráfego UDP (57,6kb;28,8kb;9.6kb) UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9
48	3 vias sem perdas com <i>delay</i> (1ms;1ms;1ms); UDP 256; GOP = 16	45,57866667	27,48563	84.2 / 87.5	87,3	2,9

Figura 109 - 1 via sem perdas; UDP 256; GOP = 4

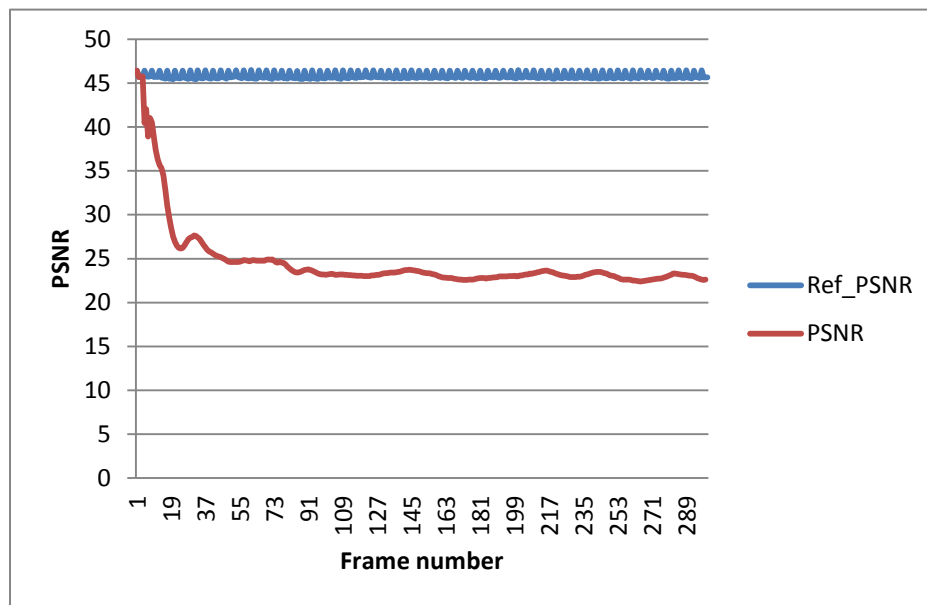


Figura 110 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 4

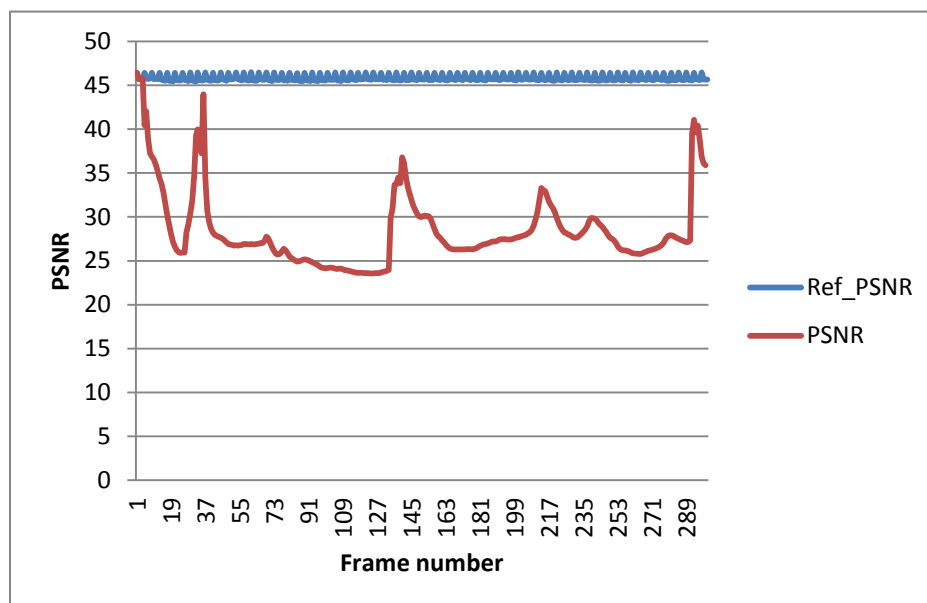


Figura 111 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 256; GOP = 4

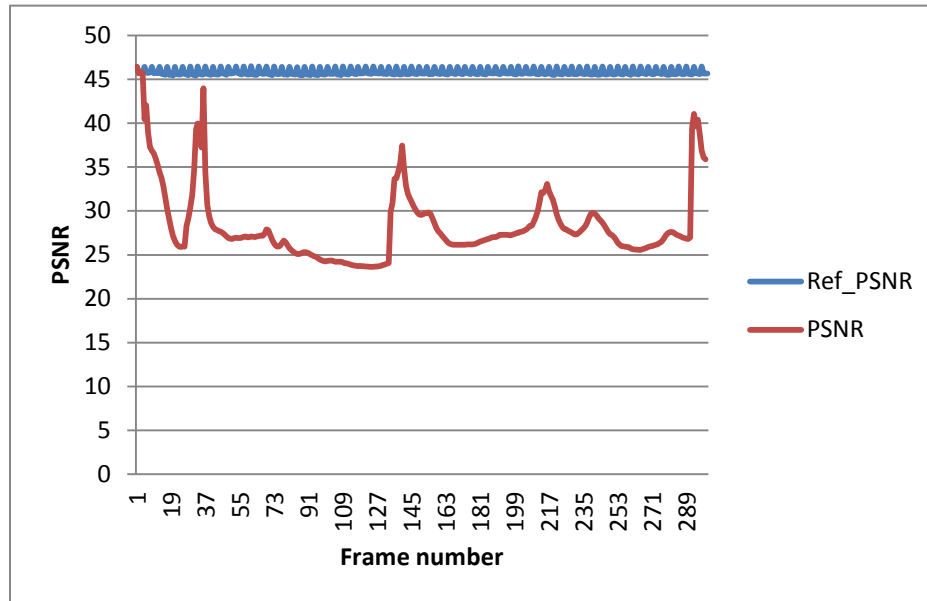


Figura 112 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 256; GOP = 4

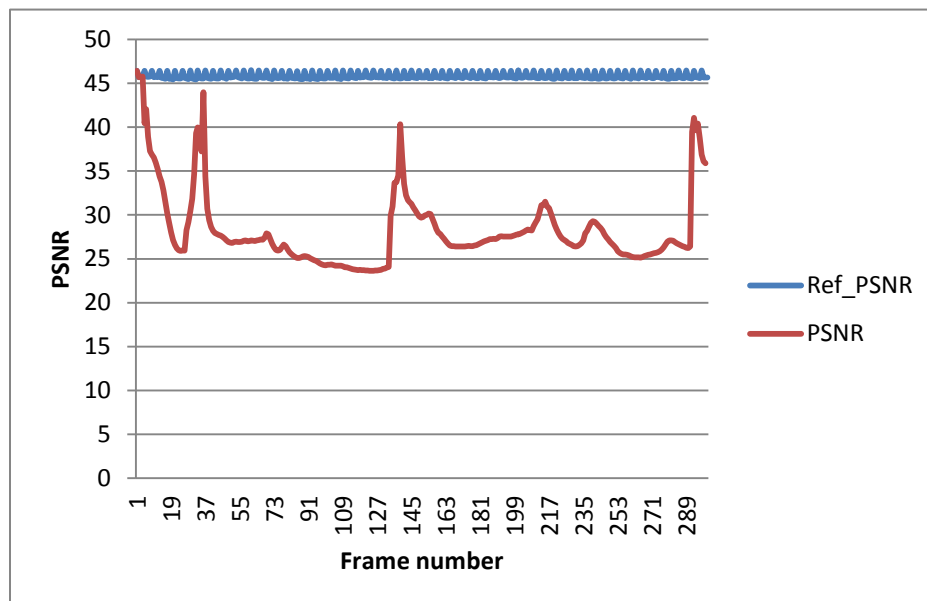


Figura 113 - 3 vias com perdas (0,08; 0,04; 0,02) com delay (120ms; 250ms; 350ms); UDP 256; GOP = 4

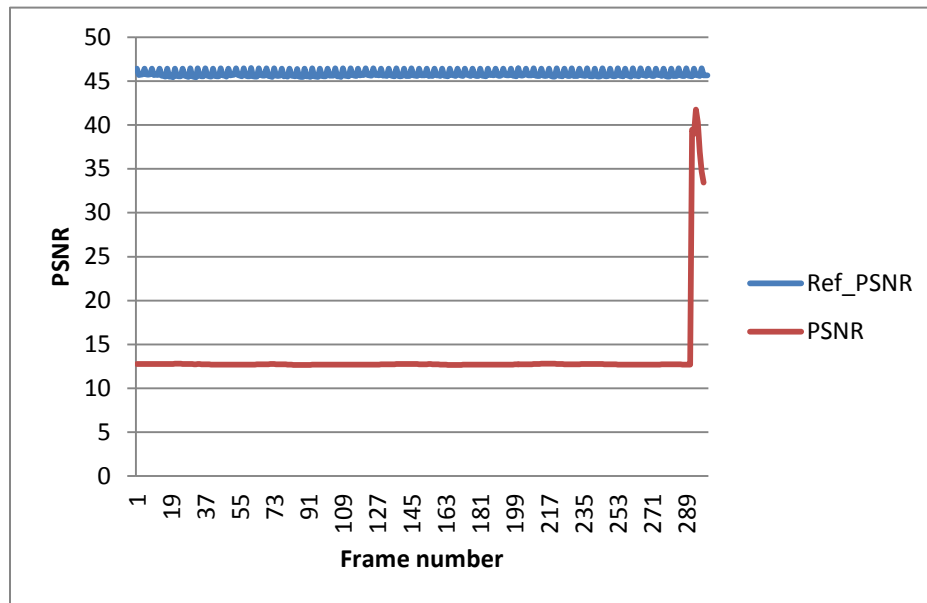


Figura 114 - 3 vias com perdas (0,04; 0,02; 0,01) com delay (120ms; 250ms; 350ms); UDP 256; GOP = 4

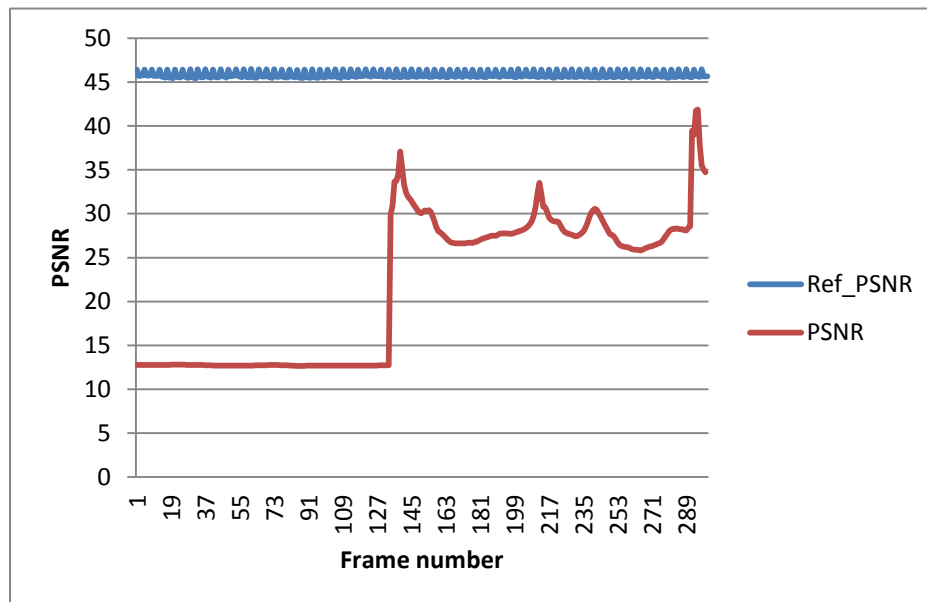


Figura 115 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 4

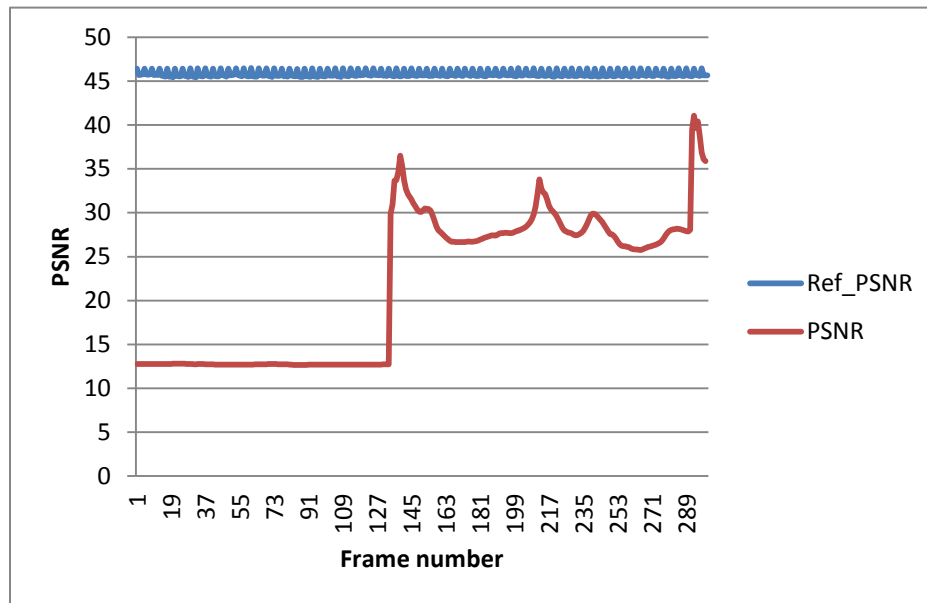


Figura 116 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 4

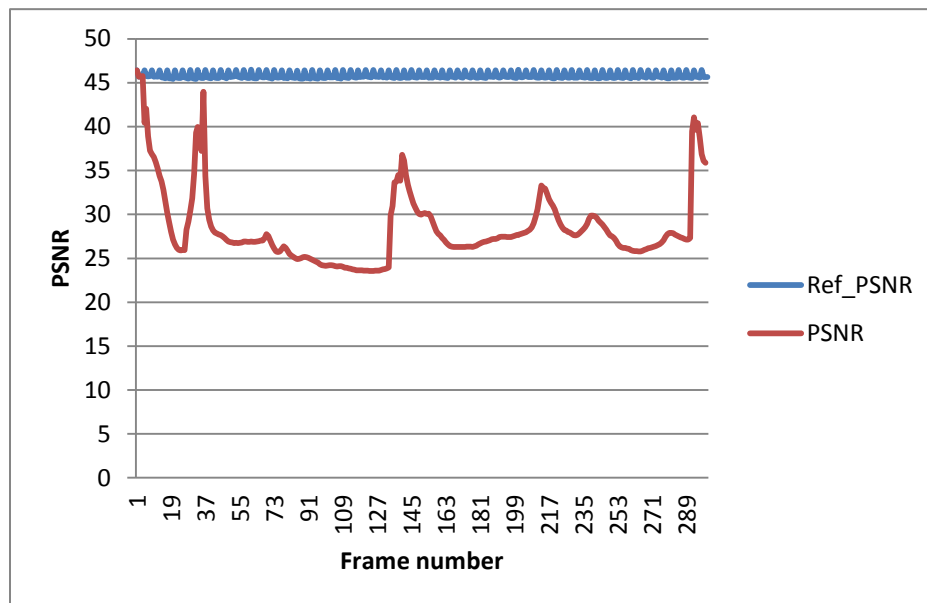


Figura 117 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 4

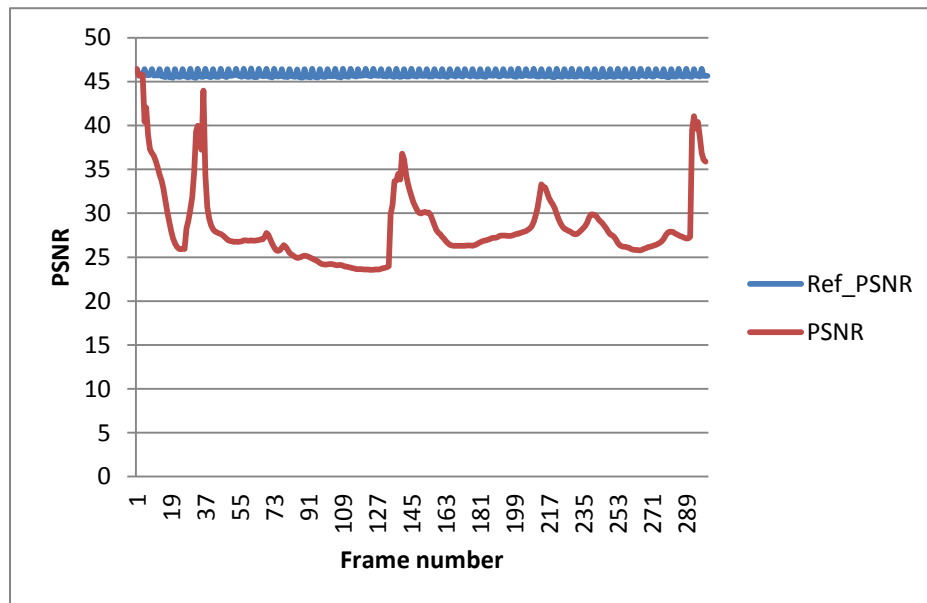


Figura 118 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 4

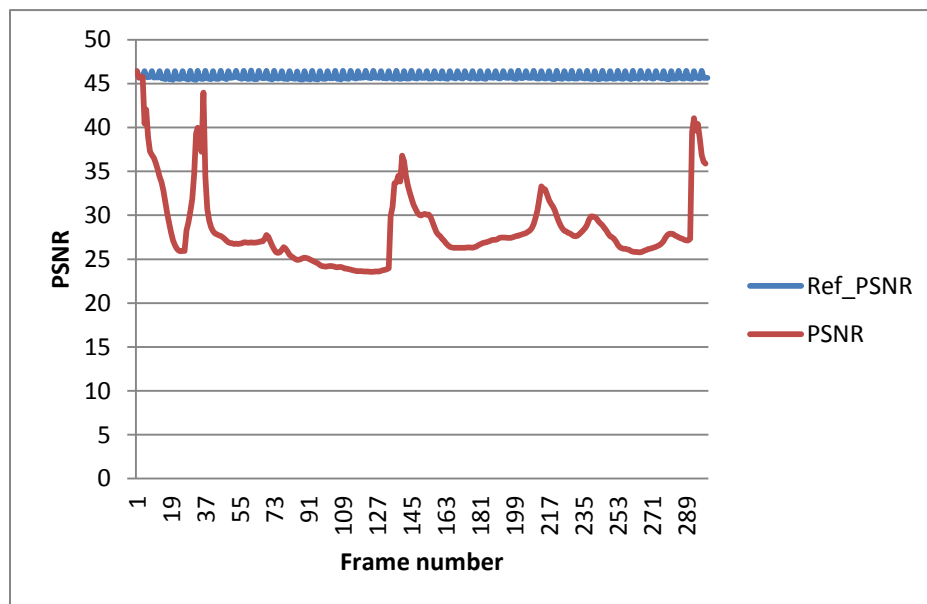


Figura 119 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 4

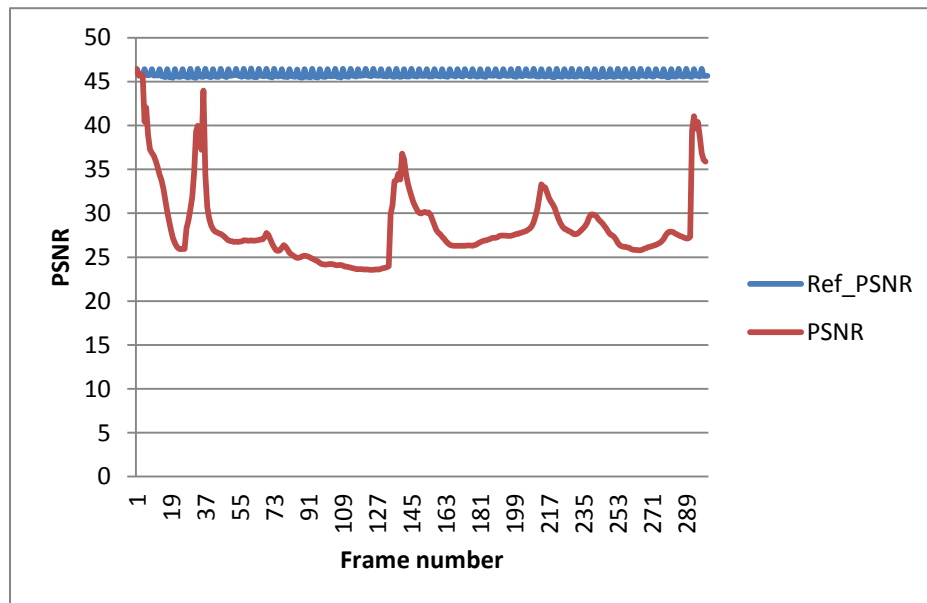


Figura 120 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 256; GOP = 4

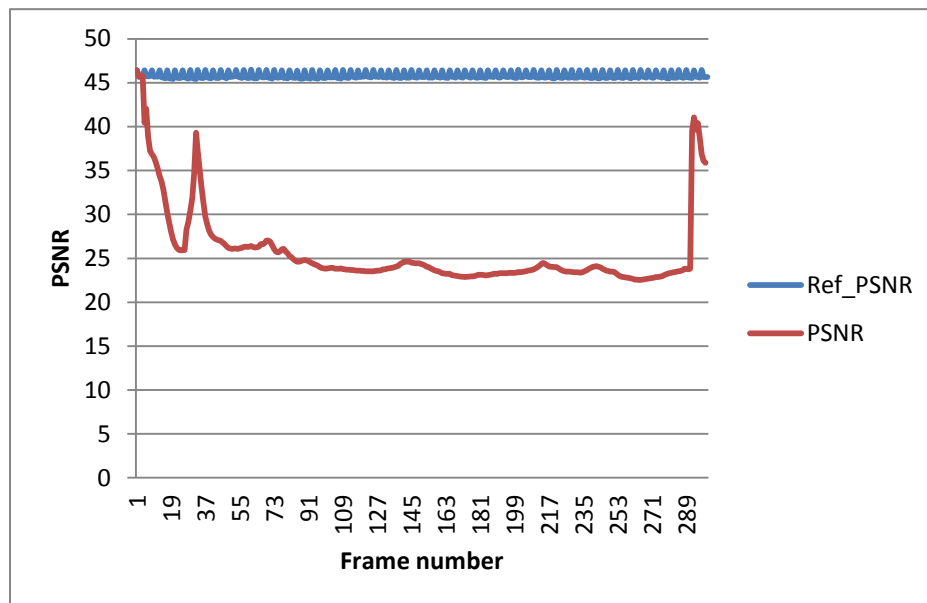


Figura 121 - 1 via sem perdas; UDP 256; GOP = 6

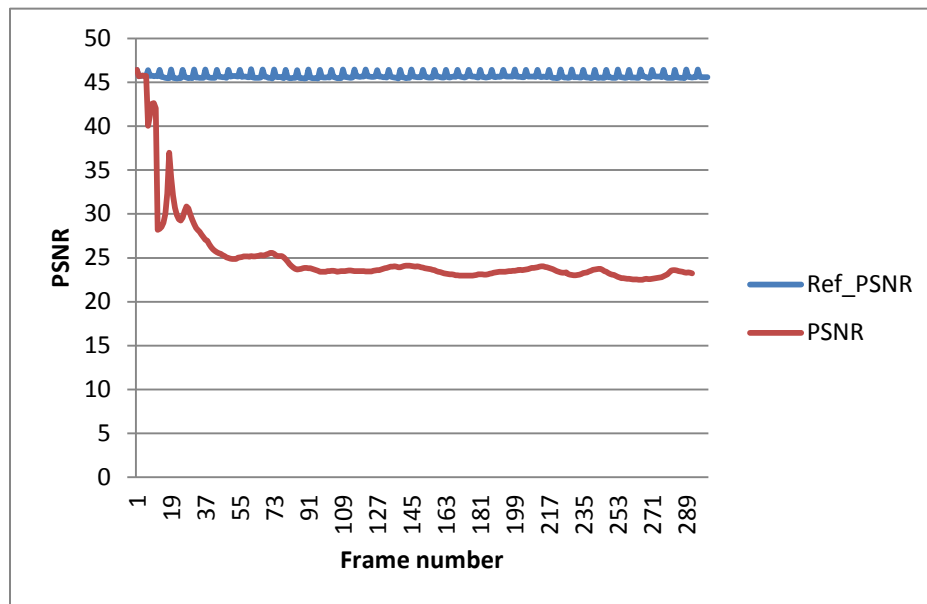


Figura 122 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 256; GOP=6

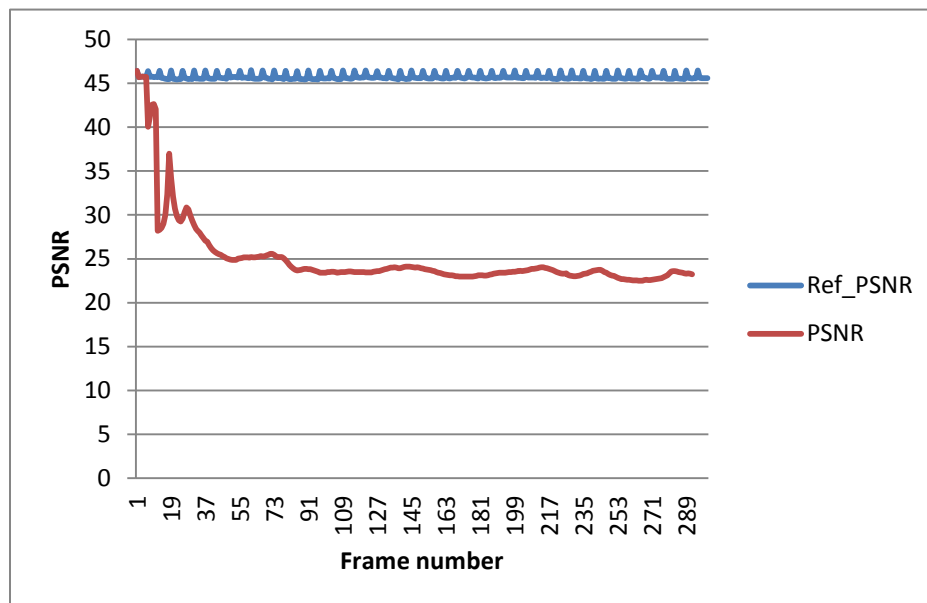


Figura 123 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 256; GOP = 6

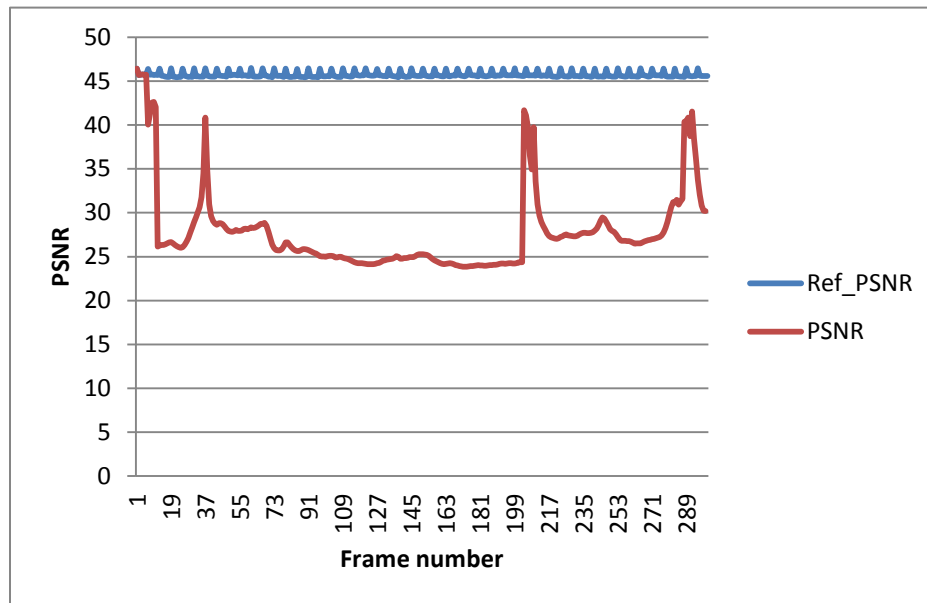


Figura 124 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 256; GOP = 6

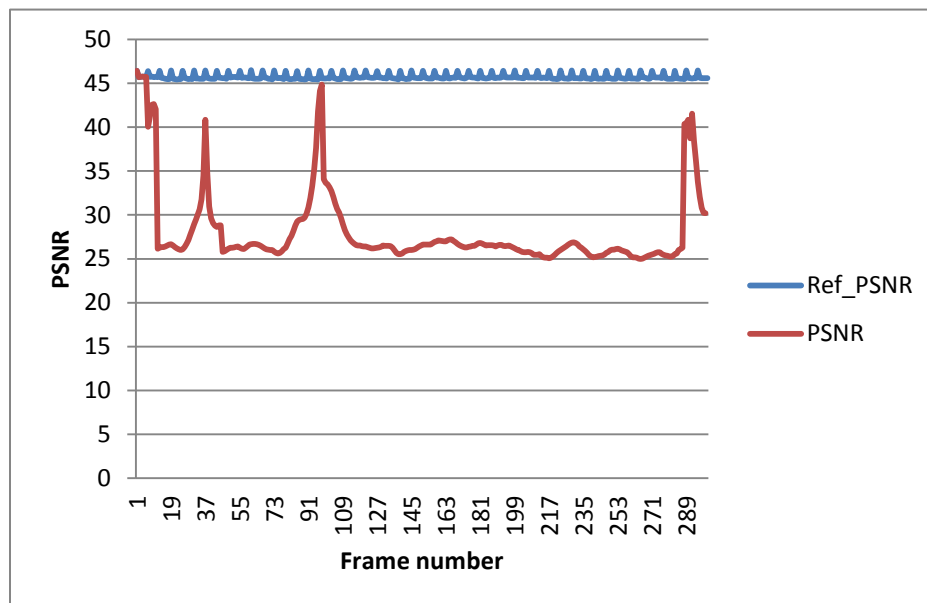


Figura 125 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 6

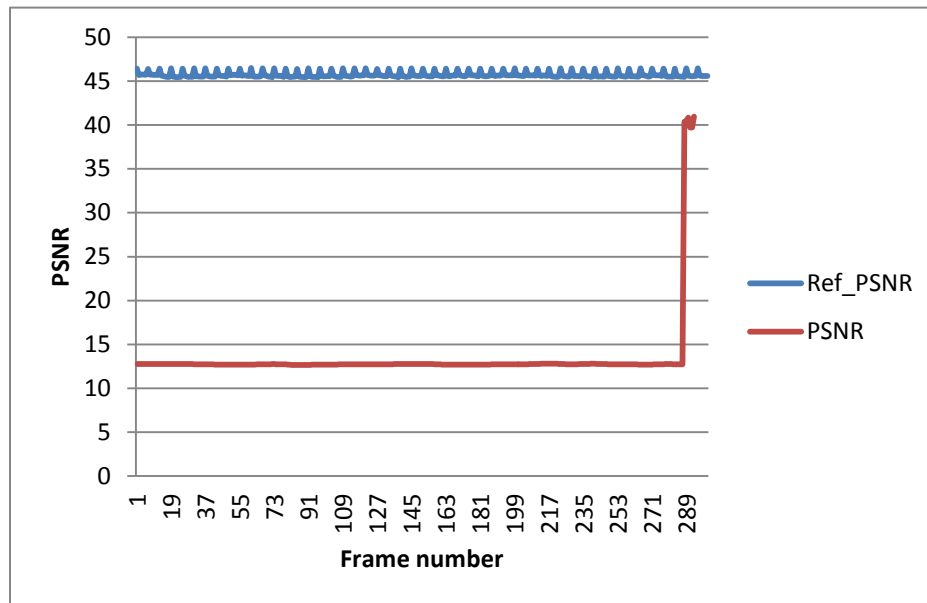


Figura 126 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 6

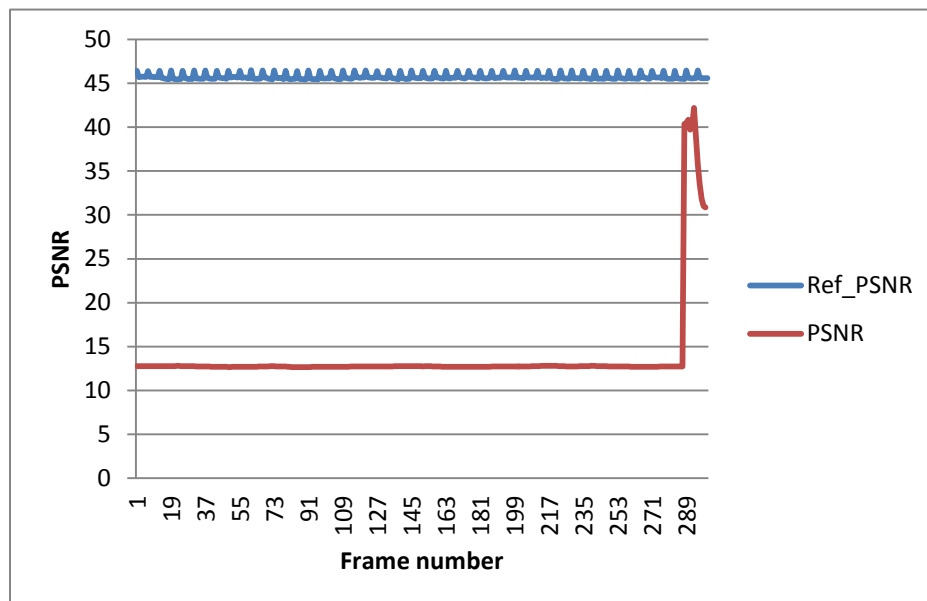


Figura 127 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 6

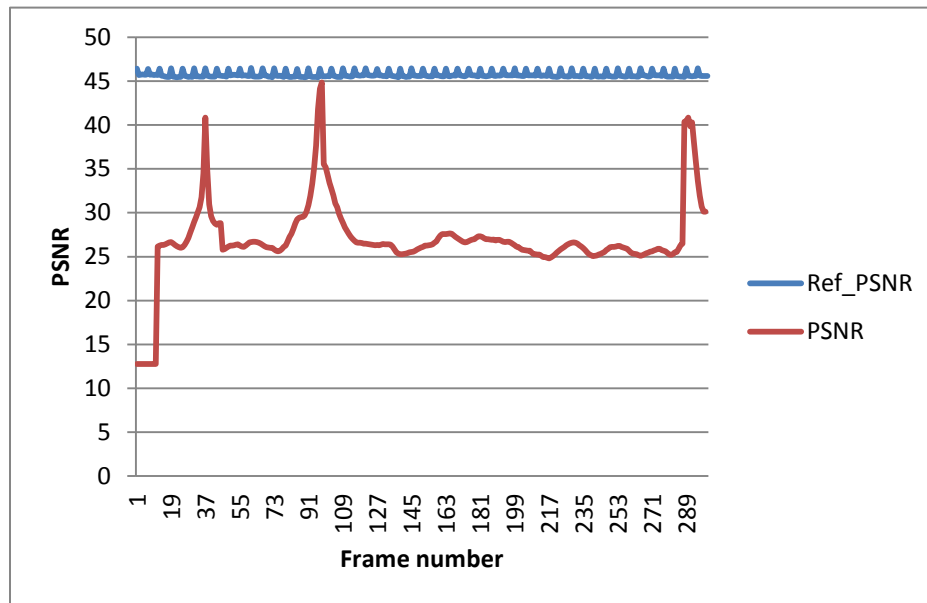


Figura 128 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 6

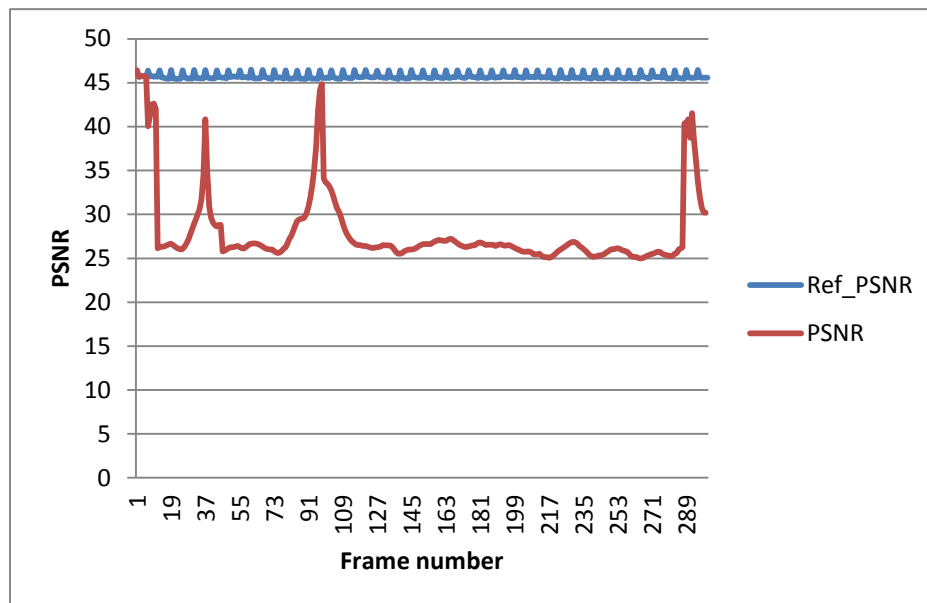


Figura 129 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 6

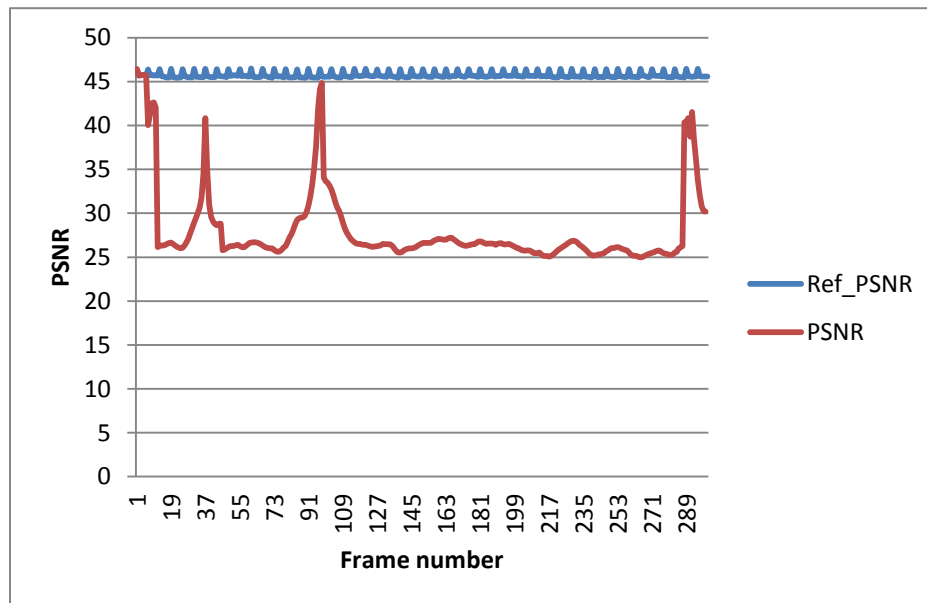


Figura 130 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 6

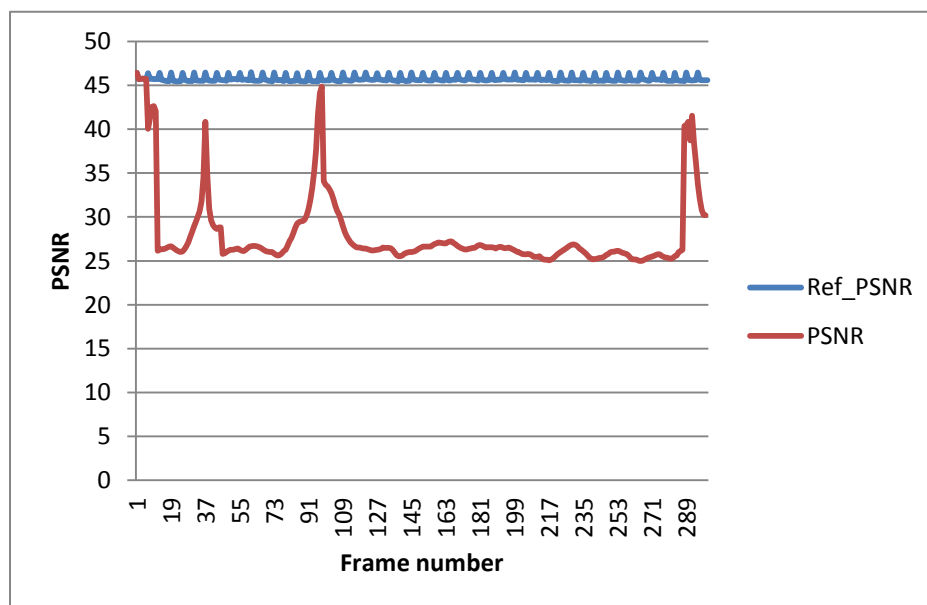


Figura 131 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9.6kb); UDP 256; GOP = 6

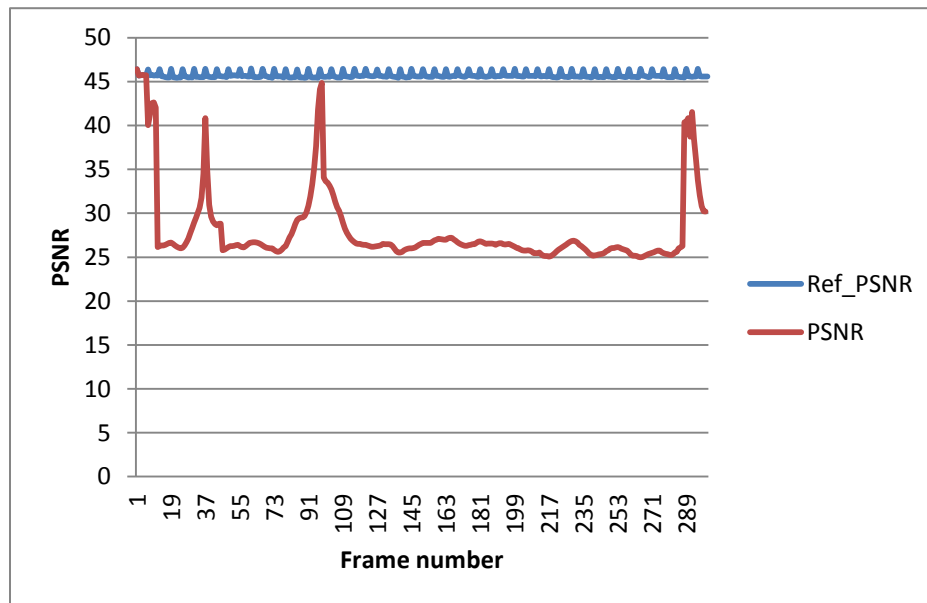


Figura 132 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 256; GOP = 6

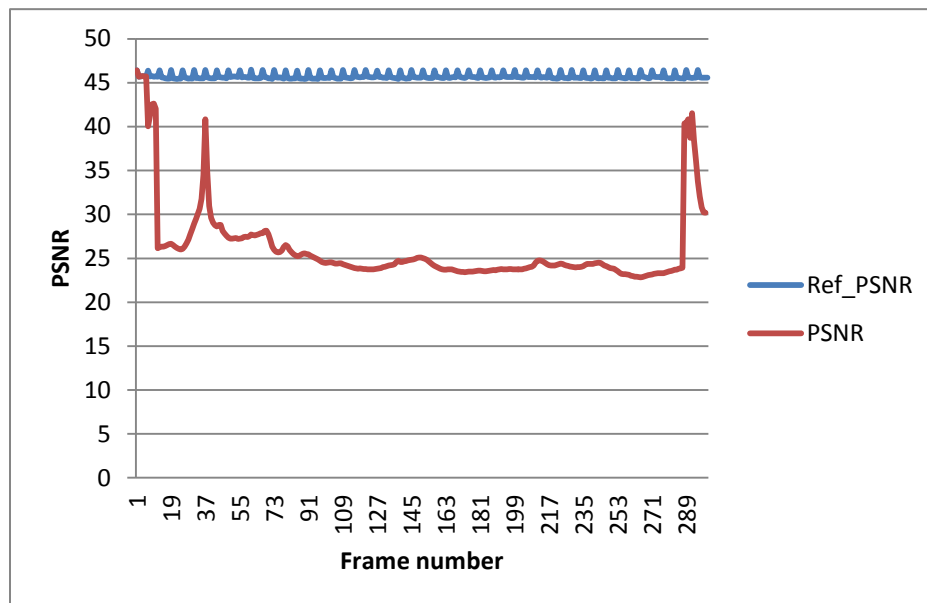


Figura 133 - 1 via sem perdas; UDP 256; GOP = 8

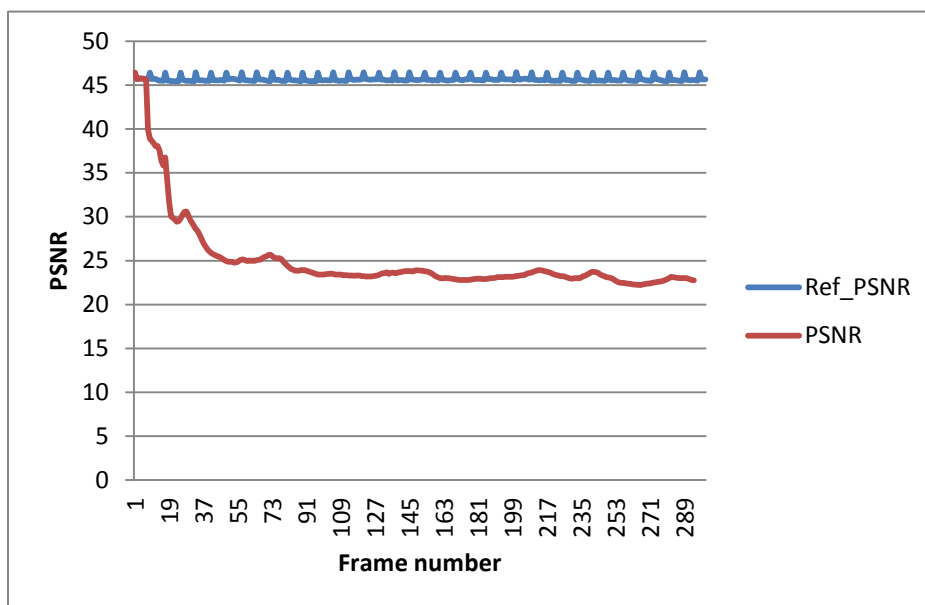


Figura 134 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 256; GOP=8

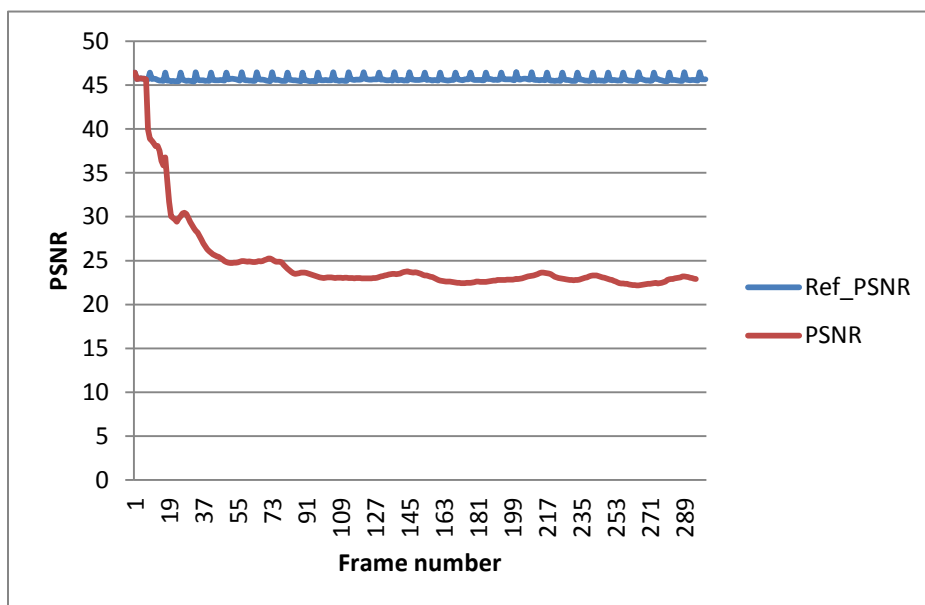


Figura 135 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 256; GOP = 8

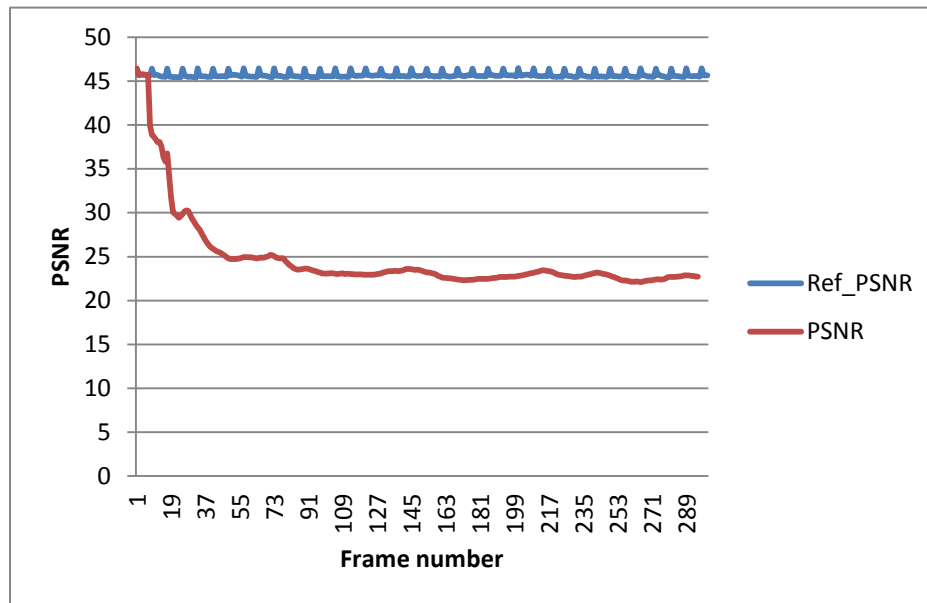


Figura 136 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 256; GOP = 8

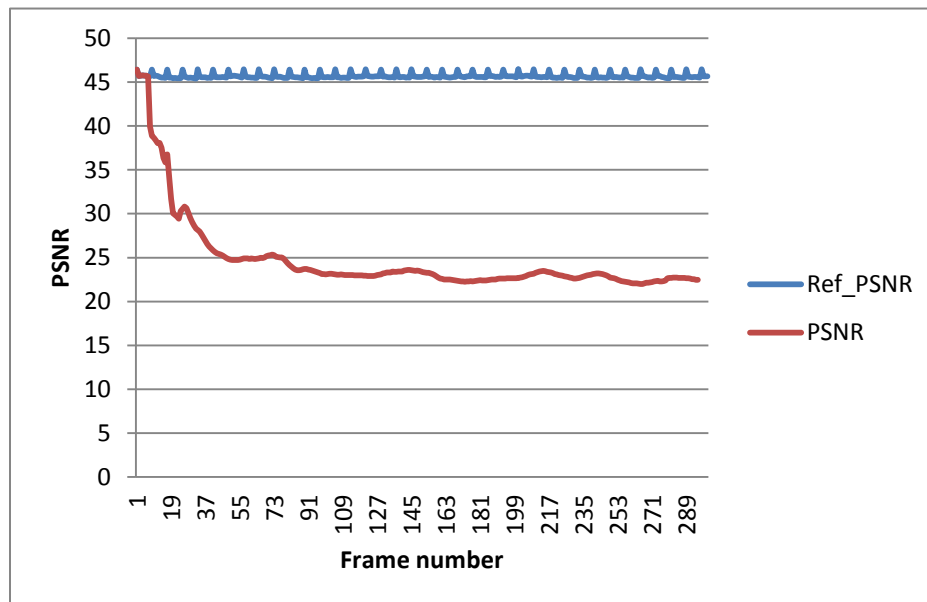


Figura 137 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP= 8

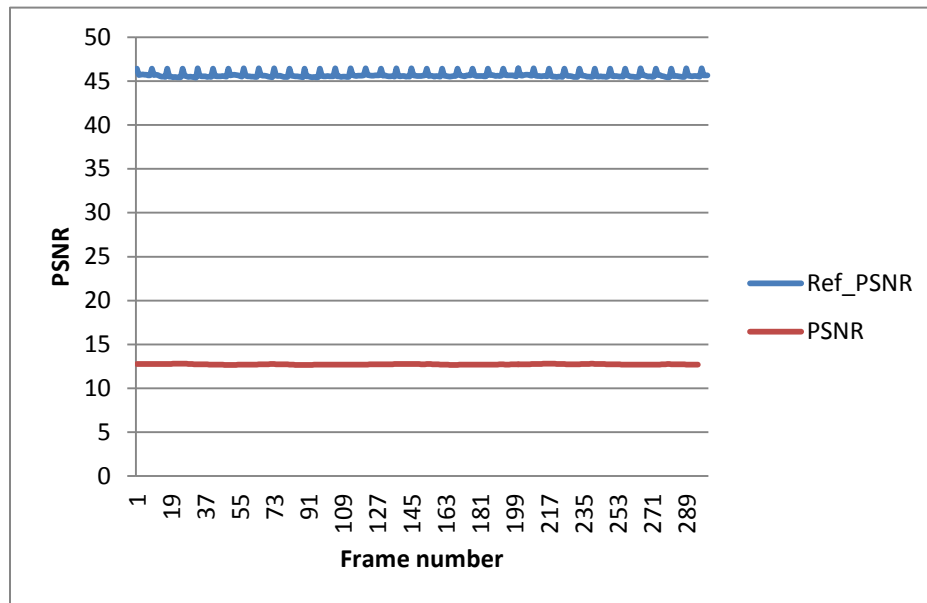


Figura 138 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP= 8

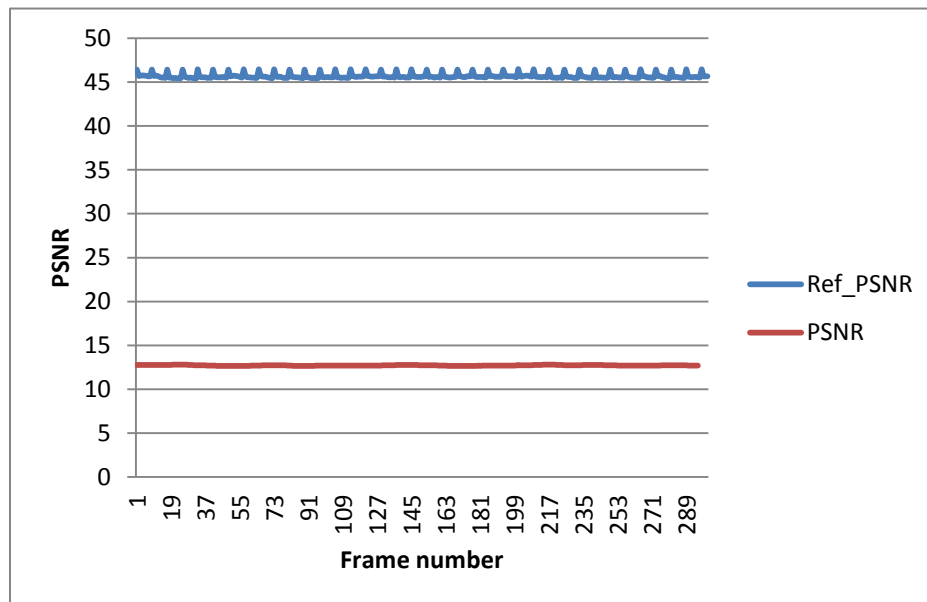


Figura 139 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 8

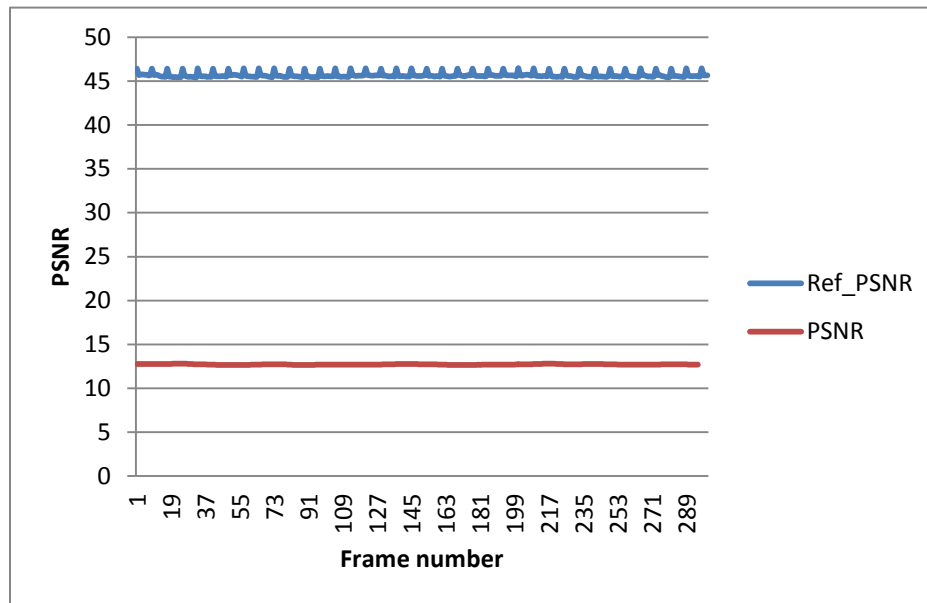


Figura 140 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 8

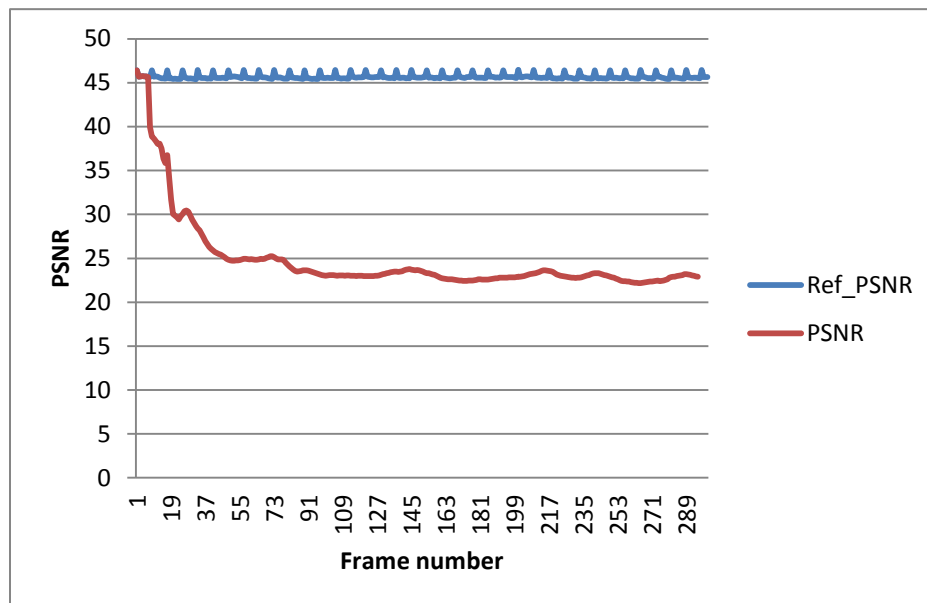


Figura 141 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 8

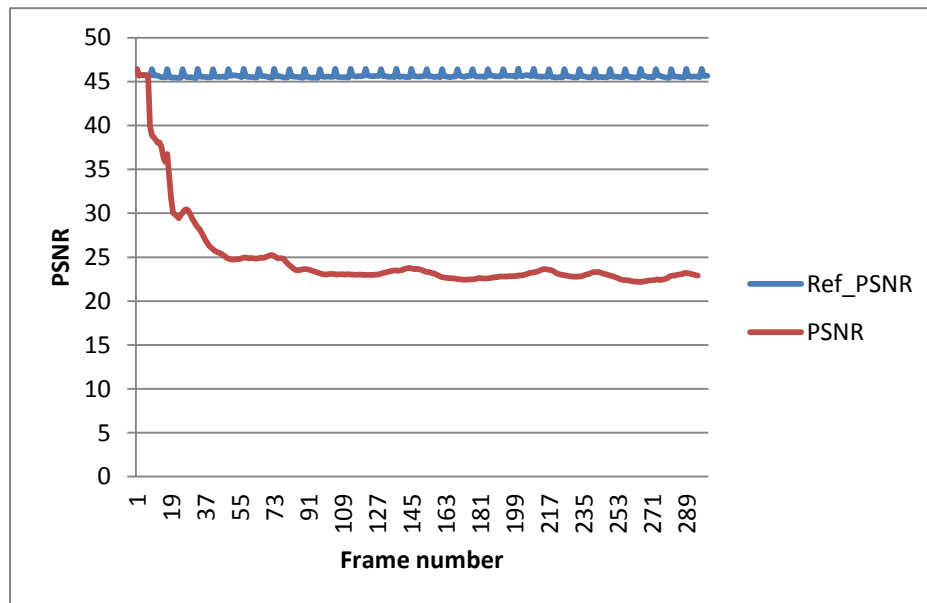


Figura 142 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 8

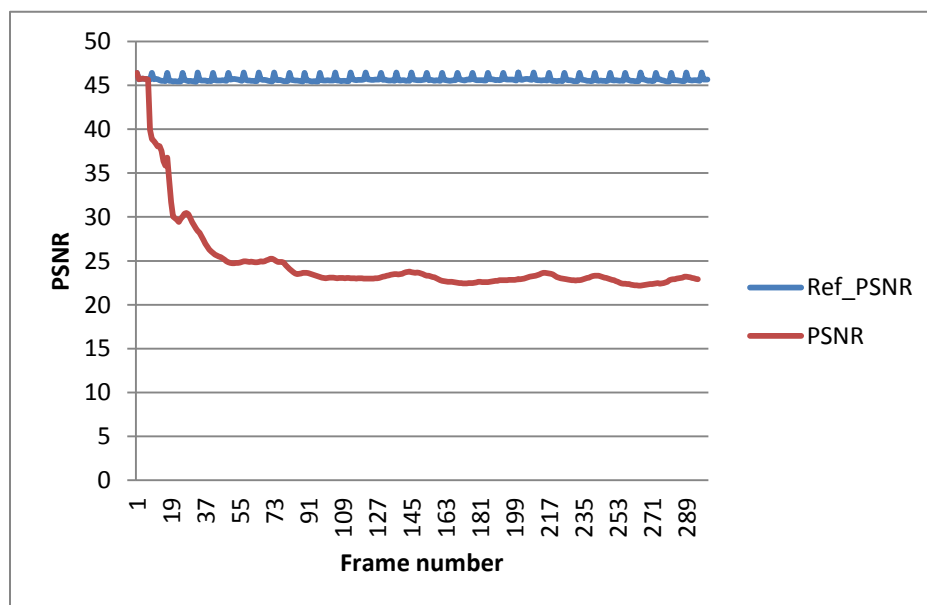


Figura 143 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 8

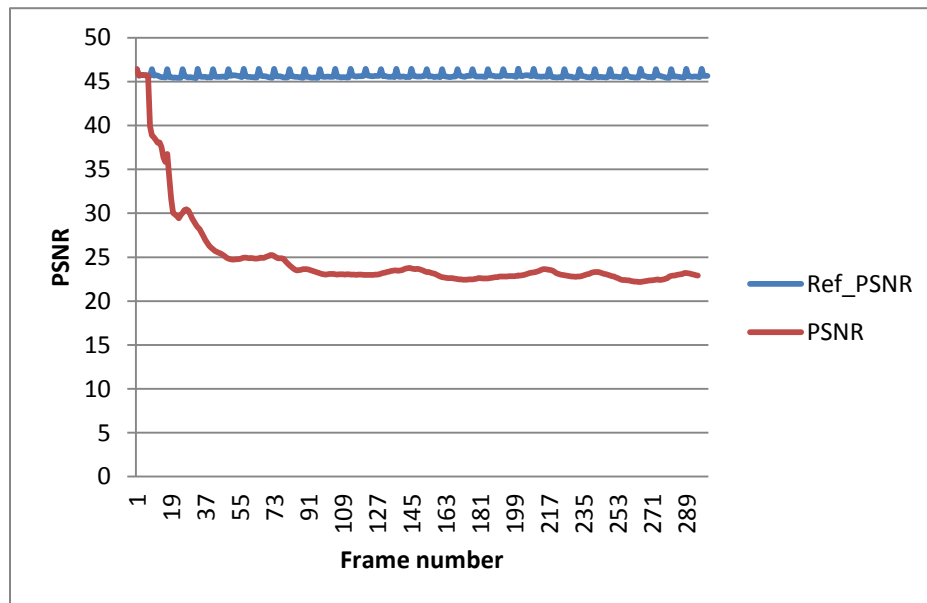


Figura 144 - 3 vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 256; GOP = 8

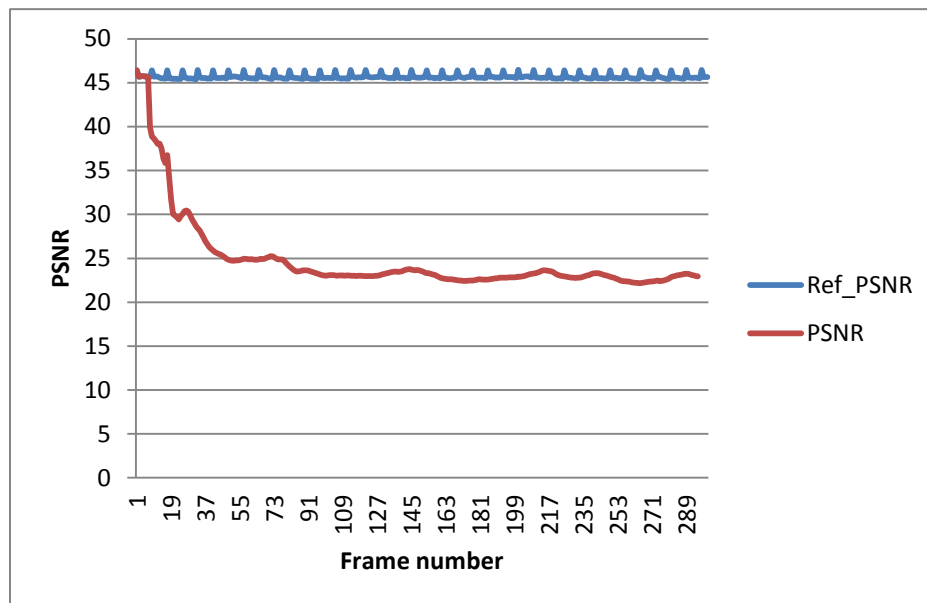


Figura 145 - 1 via sem perdas; UDP 256; GOP = 16

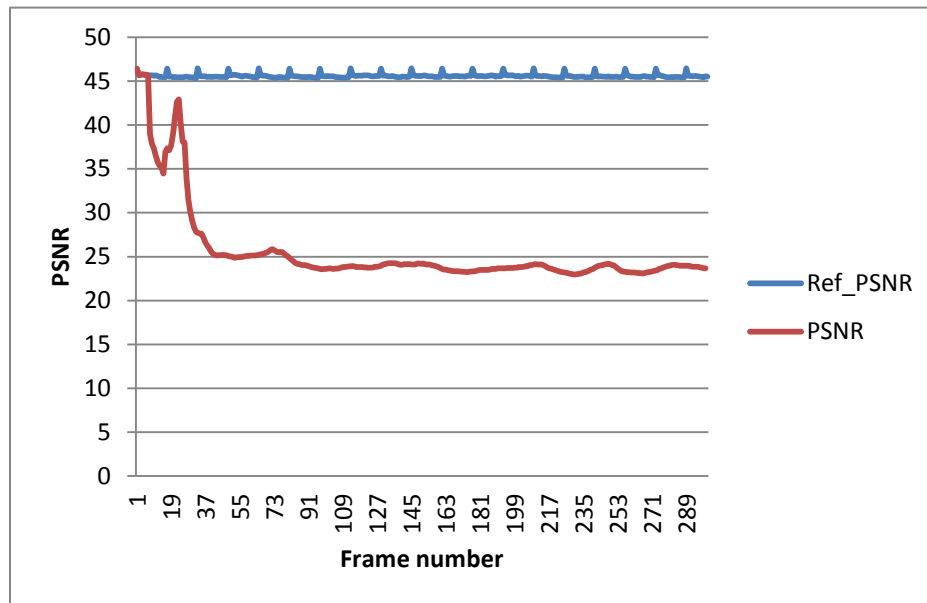


Figura 146 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 16

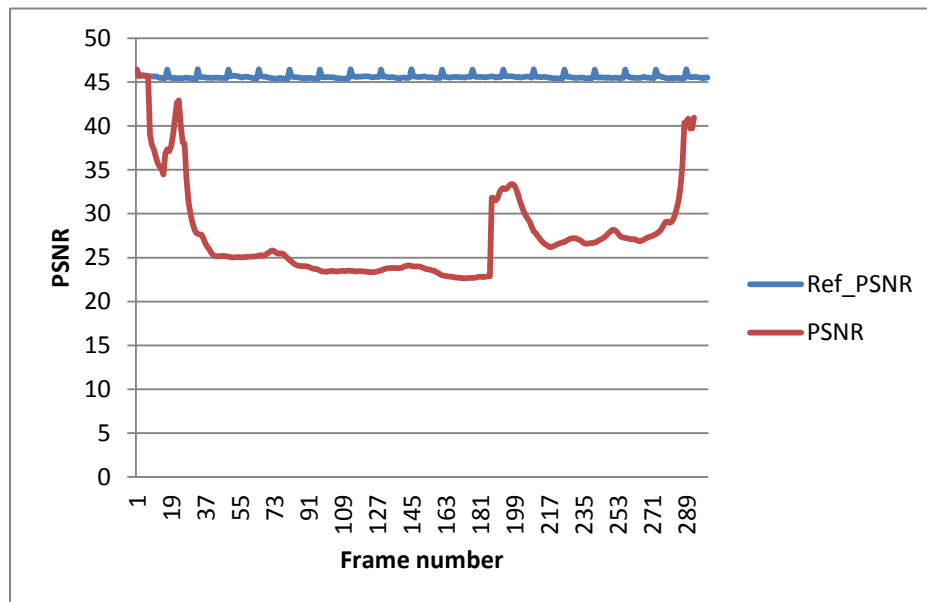


Figura 147 - 3 vias sem perdas com *delay* (240ms; 500ms; 700ms); UDP 256; GOP = 16

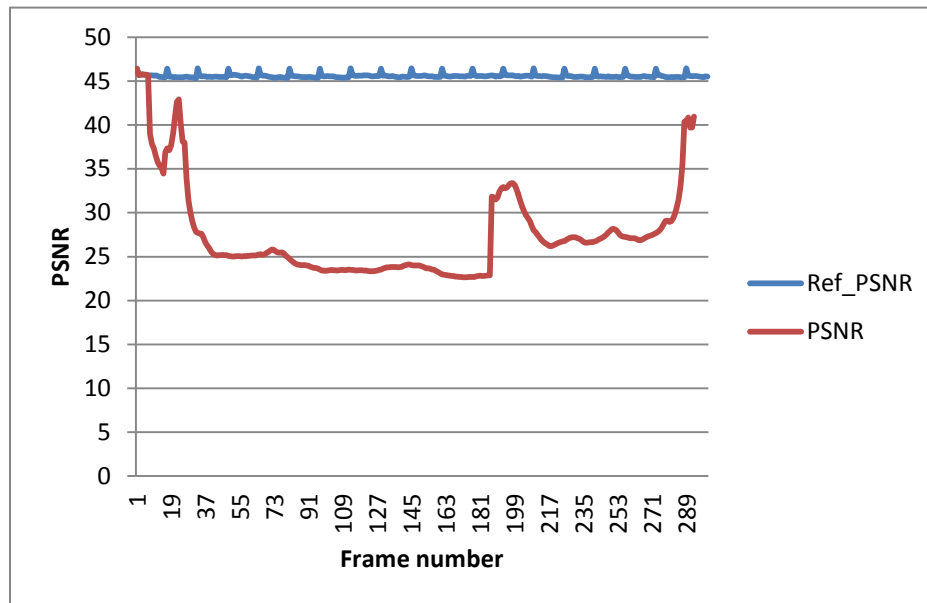


Figura 148 - 3 vias sem perdas com *delay* (360ms; 750ms; 1050ms); UDP 256; GOP = 16

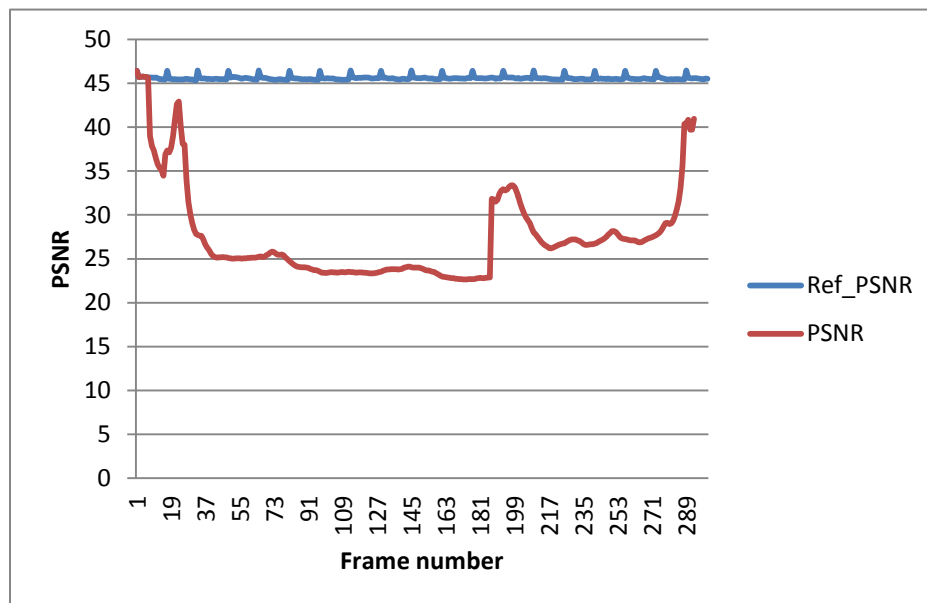


Figura 149 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 16

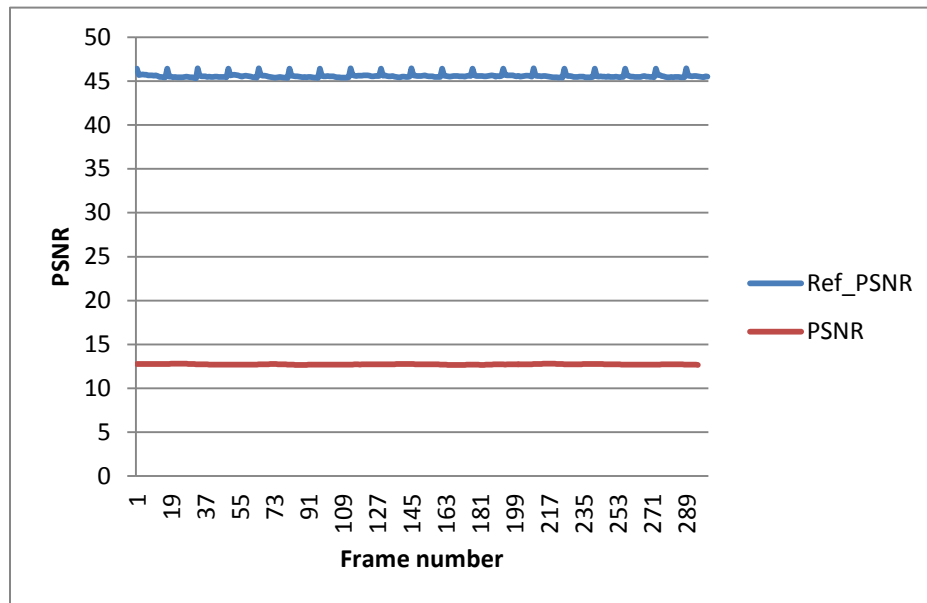


Figura 150 - 3 vias com perdas (0,04; 0,02; 0,01) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 16

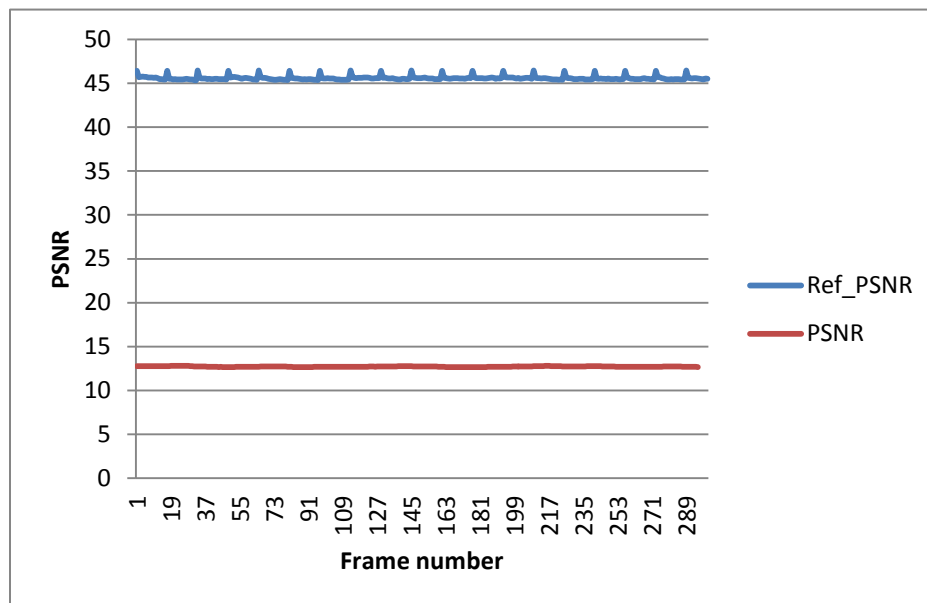


Figura 151 - 3 vias com perdas (0,02; 0,01; 0,005) com *delay* (120ms; 250ms; 350ms); UDP 256; GOP = 16

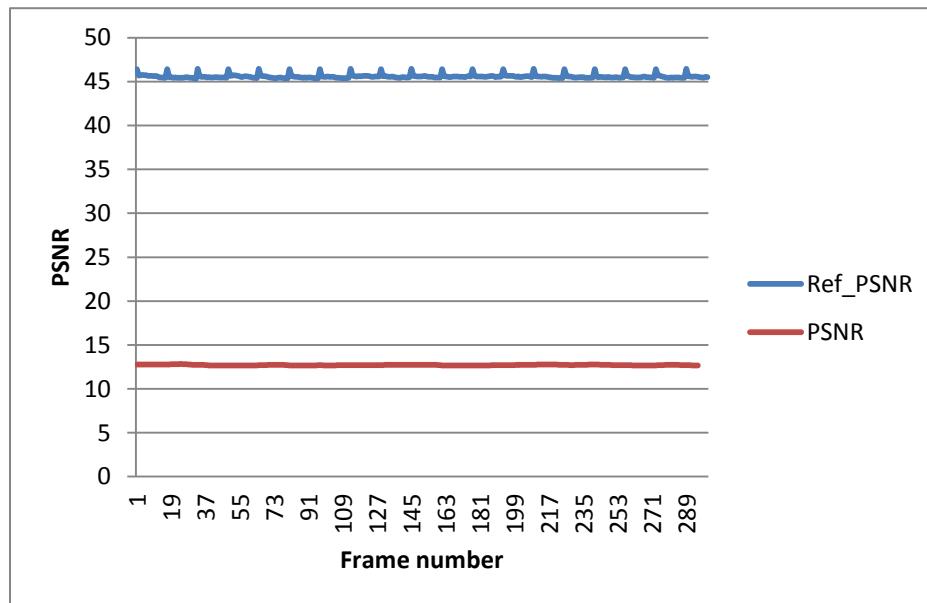


Figura 152 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego TCP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 16

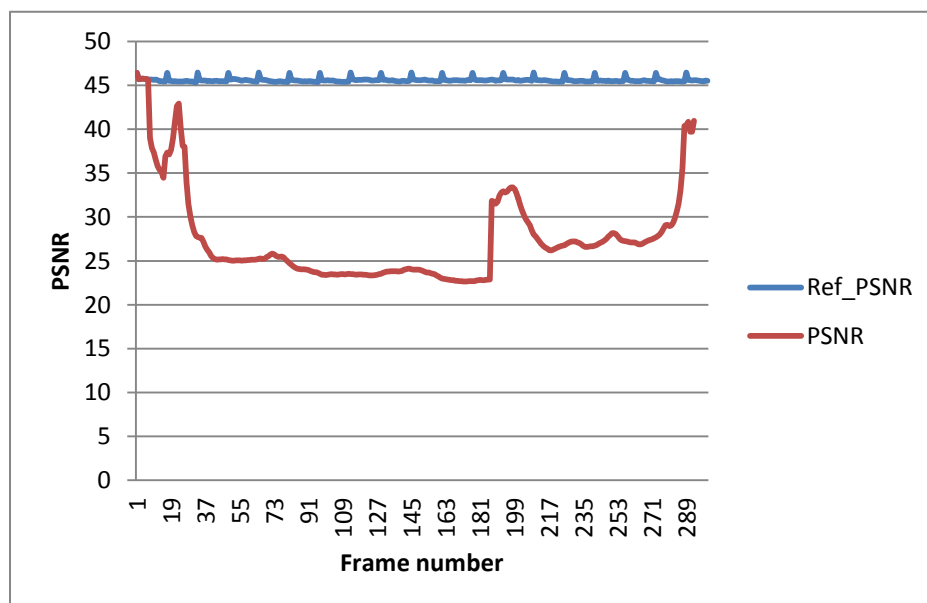


Figura 153 - 3 vias com perdas (0,08; 0,04; 0,02) com *delay* (120ms; 250ms; 350ms) e tráfego UDP (230,4kb; 115,2kb; 38,4kb); UDP 256; GOP = 16

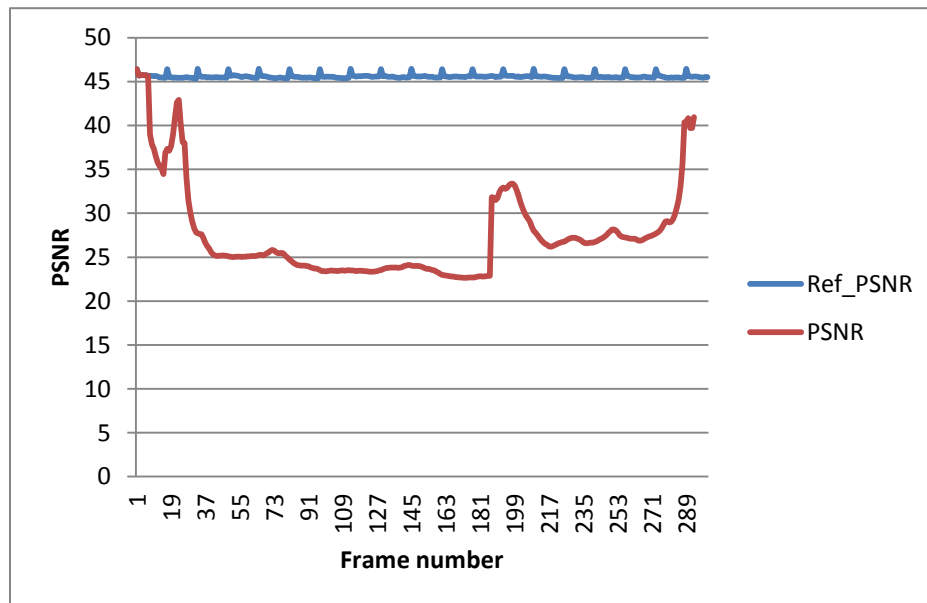


Figura 154 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (115,2kb; 57,6kb; 19,2kb); UDP 256; GOP = 16

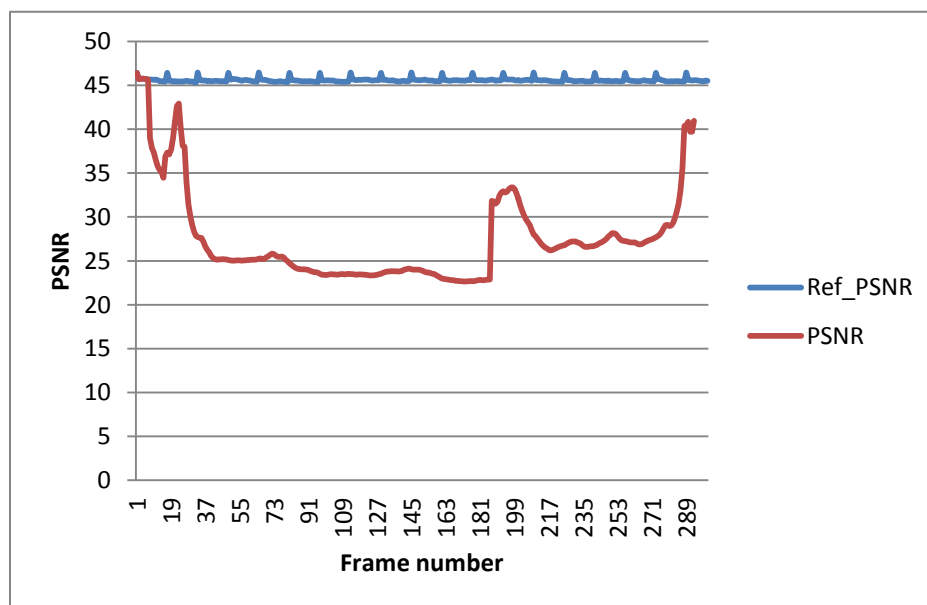


Figura 155 - 3 vias sem perdas com *delay* (120ms; 250ms; 350ms) e tráfego UDP (57,6kb; 28,8kb; 9,6kb); UDP 256; GOP = 16

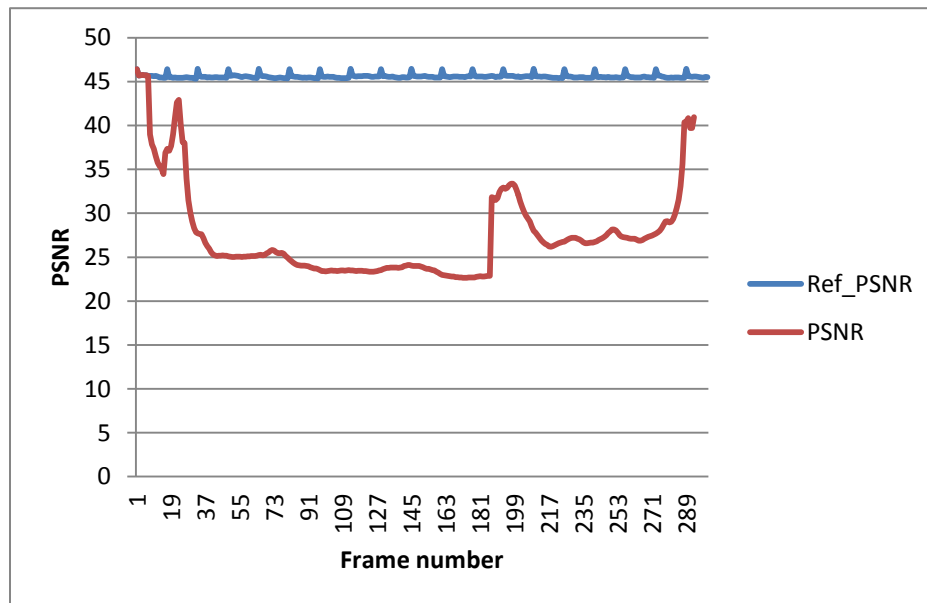


Figura 156 - vias sem perdas com *delay* (1ms; 1ms; 1ms); UDP 256; GOP = 16

