Efficient Hardware Architecture of a Modified 2-D Transform for the HEVC Standard

Fatma Belghith*[1], Hassen Loukil[2], Nouri Masmoudi[3]

University of Sfax, National Engineering School of Sfax, Electronics and Information Technology Laboratory
BP W 3038 Sfax, TUNISIA

*[1] fatmabelghithenis@gmail.com; [2] hassenloukil@gmail.com; [3] nouri.masmoudi@enis.rnu.tn

Abstract

This paper presented an algorithm to compute the 4x4, 8x8 and 16x16 efficient two-dimensional (2-D) transform for the HEVC standard providing less complexity and its hardware design. As HEVC large transforms suffered from the huge number of computations especially multiplications, this paper presented a proposition of a modified algorithm reducing the computational complexity. The goal is to ensure the maximum circuit reuse during the computation while keeping the same quality of encoded videos. The proposed hardware has been implemented in VHDL. The VHDL RTL code works at 240 MHZ in an Altera Stratix III FPGA. Results showed frequency improvement reaching 96% when compared to an architecture described as the direct transcription of the algorithm. The designed architecture has achieved a throughput reaching 4 Gsamples/s. With these results, the proposed is able to process even HD videos in real time.

Key-Words

HEVC; HM5.0; Transformation; FPGA

Introduction

Recently, the joint collaborative team on video coding has developed the new video coding standard HEVC which provides more coding efficiency. This standard uses the same block-based hybrid coding (inter/intra picture) used in the previous standards since H.261. The process of encoding to produce an HEVC compliant bitstream would proceed as Figure 1 shows. Each picture is firstly divided into block-shaped regions with the exact block partitioning using the quadtree structure. The first picture of the video sequence is usually intra coded. For the others, both intra and inter can be used for the prediction. The residual signal which is the difference between the original block and its prediction is transformed, scaled then quantized. Finally the entropy coding is carried out, in order to be transmitted with the prediction information to the decoder (JCT, VCT,2011). Transformation is one of the popular techniques for data compression. In recent years, some transform architectures have been proposed for video applications.

Authors in (Ashfaq A., 2011) have implemented a fast 16-point DCT using a multiplier-less architecture. The VLSI implementation has been carried out for a 90-nm standard cell technology at a clock frequency of 150 MHZ.

Ricardo's work presented in (Ricardo J., 2012) proposed an architecture synthesized using Cyclone II and Stratix III reducing 72% compared to the original architecture.

Martuza's work in (Muhammed M., 2012) proposed unified hybrid architecture to compute the 8x8 integer inverse discrete cosine transform of multiple video codec implemented on FPGA and synthesized in CMOS 0.18 um technology.

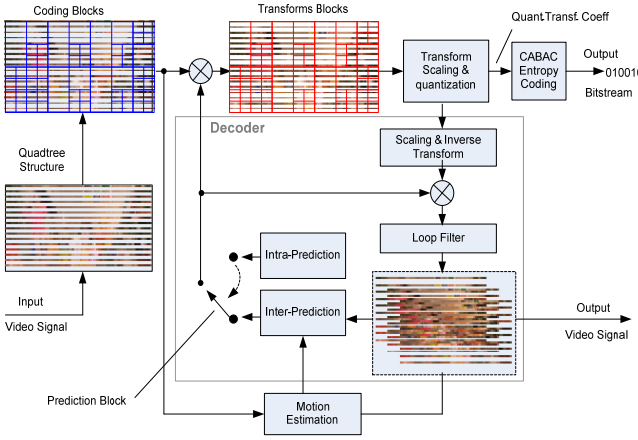


FIG. 1 HEVC ENCODER

For HEVC, the two dimensional transform are computed by applying the one-dimensional transform in both horizontal and vertical directions. Consequently, more multiplications are required. In this paper, we started exploring possible the hardware implementation of this unit discussed in the draft (Benjamin B. , 2011). The reference software is called HM (HEVC test model). The version used in this work is HM5.0 (HM5.0) which is the latest available Test model when realizing this work. This paper will introduce a proposed algorithm with the aim to reduce

the complexity of the transform implementation. The rest of this paper will be organized as follows. In section II, a review of the HEVC transform and the proposed matrix factorizations have been described. Section III presented the comparison of the computational complexity and the quality evaluation of the proposed algorithm compared to the original one. Section IV detailed the hardware implementation of the whole architecture. Finally, Section V presented the experimental results.

HEVC Transform

Review of the HEVC Transform

As HEVC was designed especially for higher resolutions, larger transforms were proposed in order to better the coding efficiency. The new standard uses the Residual Quadtree (RQT) structure which employs a flexible block partitioning using large and multiple block sizes. The quadtree structure uses three different basic units, namely, CU (Coding unit), PU (Prediction Unit) and TU (Transform Unit). The size of the transform unit varies from 4x4 to 32x32.

For this reason, each transform size will be computed using its own matrix. These matrices are used to code with transformation residuals. Matrices are presented as follows.

If the transform unit is 4x4, the matrix used is

$$\mathbf{M_4} = \begin{pmatrix} \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} \\ \mathbf{83} & \mathbf{36} & \mathbf{-36} & \mathbf{-83} \\ \mathbf{64} & \mathbf{-64} & \mathbf{-64} & \mathbf{64} \\ \mathbf{36} & \mathbf{-83} & \mathbf{83} & \mathbf{-36} \end{pmatrix} \quad (1)$$

This unit is computed using this equation Eq. (2).

$$DST=M_4*SRC \quad (4)$$

Where SRC [src0...src3] represent the residuals which are the inputs of the transform unit and DST[dst0...dst3] are the outputs.

In order to compute this size of transform, equations used in the HM software are as follows:

STEP_1:
E0=src0+src3 (2a)
O0=src0-src3 (2b)
E1=src1+src2 (2c)
O1=src1-src2 (2d)

STEP_2:
Dst0=64* E0+64* E1 (2e)
Dst1=64* O0-64* O1 (2f)
Dst2=83* E0+36* E1 (2g)
Dst3=36* O0-83* O1 (2h)

The software implementation of this matrix uses 8 multiplications and 8 additions.

Concerning the Transform Unit 8x8, residuals are computed using Eq.3.

$$DST=M8*SRC \quad (3)$$

The same previous method is used to compute the transformed residuals having more inputs SRC [Src0...Src7].

Concerning the internal computations, E[E0..E3] and O[O0..O3] are computed in the first STEP. Then a second STEP is to compute EE[EE0,EE1] and EO[EO0,EO1].

STEP_2:
EE0=E0+E3 (3a)
EO0=E0-E3 (3b)
EE1=E1+E2 (3c)
EO1=E1-E2 (3d)

Finally, the STEP_3 is dedicated to the multiplication stage.

$$\mathbf{M_8} = \begin{pmatrix} \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} & \mathbf{64} \\ \mathbf{89} & \mathbf{75} & \mathbf{50} & \mathbf{18} & \mathbf{-18} & \mathbf{-50} & \mathbf{-75} & \mathbf{-89} \\ \mathbf{83} & \mathbf{36} & \mathbf{-36} & \mathbf{-83} & \mathbf{-83} & \mathbf{-36} & \mathbf{36} & \mathbf{83} \\ \mathbf{75} & \mathbf{-18} & \mathbf{-89} & \mathbf{-50} & \mathbf{50} & \mathbf{89} & \mathbf{18} & \mathbf{-75} \\ \mathbf{64} & \mathbf{-64} & \mathbf{-64} & \mathbf{64} & \mathbf{64} & \mathbf{-64} & \mathbf{-64} & \mathbf{64} \\ \mathbf{50} & \mathbf{-89} & \mathbf{18} & \mathbf{75} & \mathbf{-75} & \mathbf{-18} & \mathbf{89} & \mathbf{-50} \\ \mathbf{36} & \mathbf{-83} & \mathbf{83} & \mathbf{-36} & \mathbf{-36} & \mathbf{83} & \mathbf{-83} & \mathbf{36} \\ \mathbf{18} & \mathbf{-50} & \mathbf{75} & \mathbf{-89} & \mathbf{89} & \mathbf{-75} & \mathbf{50} & \mathbf{-18} \end{pmatrix} \quad (4)$$

Referring to the test model, the computing has been done through these three steps in order to reduce the number of multiplications. The first and second steps are to factorize the inputs before multiplications in the final step.

This matrix is implemented in the test model HM 5.0 using 24 multiplications and 28 additions.

Concerning the larger transform unit which needs more operations, there are three stages for the factorization and a final stage to obtain the final outputs after multiplication.

The Transform Unit 16x16 is fixed by the matrix given in Eq.6 and process is held as Eq.5 shows.

The HEVC 16x16 direct transform is given as Eq. (5).

$$DST=M_{16}*SRC \quad (5)$$

This size needs 16 residuals (SRC[0..15]) and 16 outputs (DST[0..15]).

Referring to the test model, the software implementation of this matrix costs 88 multiplications and 100 additions.

As the number of multiplication is considerable, some considerations have been done in order to simplify the software implementation before proceeding the hardware implementation.

$$M_{16} = \begin{pmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 90 & 87 & 80 & 70 & 57 & 43 & 25 & 9 & -9 & -25 & -43 & -57 & -70 & -80 & -87 & -90 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 & -89 & -75 & -50 & -18 & 18 & 50 & 75 & 89 \\ 87 & 57 & 9 & -43 & -80 & -90 & -70 & -25 & 25 & 70 & 90 & 80 & 43 & -9 & -57 & -87 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 & 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 80 & 9 & -70 & -87 & -25 & 57 & 90 & 43 & -43 & -90 & -57 & 25 & 87 & 70 & -9 & -80 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 & -75 & 18 & 89 & 50 & -50 & -89 & -18 & 75 \\ 70 & -43 & -87 & 9 & 90 & 25 & -80 & -57 & 57 & 80 & -25 & -90 & -9 & 87 & 43 & -70 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 57 & -80 & -25 & 90 & -9 & -87 & 43 & 70 & -70 & -43 & 87 & 9 & -90 & 25 & 80 & -57 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 & -50 & 89 & -18 & -75 & 75 & 18 & -89 & 50 \\ 43 & -90 & 57 & 25 & -87 & 70 & 9 & -80 & 80 & -9 & -70 & 87 & -25 & -57 & 90 & -43 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 & 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 25 & -70 & 90 & -80 & 43 & 9 & -57 & 87 & -87 & 57 & -9 & -43 & 80 & -90 & 70 & -25 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 & -18 & 50 & -75 & 89 & -89 & 75 & -50 & 18 \\ 9 & -25 & 43 & -57 & 70 & -80 & 87 & -90 & 90 & -87 & 80 & -70 & 57 & -43 & 25 & -9 \end{pmatrix} \quad (6)$$

Proposed Algorithm

As the targeted work is the hardware implementation of the transform module, the first step is to ensure algorithms without multiplications replacing all the multiplications by additions and shifts in order to use less hardware size as known multiplications are costly on FPGA resources. The second consideration in the proposed algorithm is to provide more optimization by decomposing each matrix into smaller and less costly matrices. Finally, a slight modification of the matrix M16 has been done in order to be more adequate for decomposition. Note that all the optimizations and modifications were done in the direct transforms in the encoder.

1) Decomposition of the Matrix M4

The transformation matrix in Eq.1 is decomposed as shown in Eq.7 (M.N. Haggag, 2010)(W.-K. Cham, 1989) where P4.1 in Eq.9 is implemented using only additions with a complexity of 4 additions, while M4.1 as Eq.8 shows, can be implemented using 12 additions and 12 shifts, which sum up to a complexity of 16 additions and 12 shifts(M.N. Haggag, 2010).

$$M_4 = M_{4.1} * P_{4.1} \quad (7)$$

Where

$$M_{4.1} = \begin{pmatrix} 64 & 64 & 0 & 0 \\ 0 & 0 & -36 & -83 \\ 64 & -64 & 0 & 0 \\ 0 & 0 & 83 & -36 \end{pmatrix} \quad (8)$$

And

$$P_{4.1} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix} \quad (9)$$

2) Decomposition of the matrix M8

To begin processing this transform, a first factorization is done which already exists in the original algorithm representing the first step.

$$M_8 = M_{8.1} * P_{8.1} \quad (10)$$

Where

$$P_{8.1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (11)$$

P8.1 is the first process for the input data to the matrix multiplication presented by Eq.11, and this process uses only additions. It has a complexity of 8 additions.

The elements of M8.1 as shown in Eq.12 are released as such grouped into two 4x4 independent matrices M8.even in Eq.13 and M8.odd in Eq.14(M.N. Haggag, 2010).

$$M_{8.1} = \begin{pmatrix} 64 & 64 & 64 & 64 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 89 & -75 & 50 & -18 \end{pmatrix} \quad (12)$$

$$M_{8\ even} = \begin{pmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & -64 & -64 & 64 \\ 36 & -83 & 83 & -36 \end{pmatrix} \quad (13)$$

$$M_{8\ odd} = \begin{pmatrix} 89 & 75 & 50 & 18 \\ 75 & -18 & -89 & -50 \\ 50 & -89 & 18 & 75 \\ 18 & -50 & 75 & -89 \end{pmatrix} \quad (14)$$

As for the decomposition of M8.even, it is similar to the decomposition of M4, seeing that it is the same matrix, which costs 16 additions and 12 shifts.

The decomposition of M8.odd is shown in Eq.15.

$$M_{8.odd} = K_1 + K_2 * K_3 \quad (15)$$

Where

$$K_1 = \begin{pmatrix} 17 & -5 & 2 & 0 \\ -5 & 0 & -17 & -2 \\ 2 & -17 & 0 & -5 \\ 0 & -2 & -5 & -17 \end{pmatrix} \quad (16)$$

$$K_2 = \begin{pmatrix} 2 & 0 & 0 & 8 \\ 0 & 2 & 8 & 0 \\ 0 & 8 & -2 & 0 \\ -8 & 0 & 0 & 2 \end{pmatrix} \quad (17)$$

$$K_3 = \begin{pmatrix} 0 & 8 & -8 & 9 \\ 8 & -9 & 0 & 8 \\ 8 & 0 & -9 & -8 \\ 9 & 8 & 8 & 0 \end{pmatrix} \quad (18)$$

As shown in the above Eq.15, the addition between K1 and the product of K2 and K3 has a complexity of 4 additions. The matrix K1 is implemented using 16 additions and 12 shifts. As for the matrix K2 which has a complexity of 4 additions and 8 shifts, the matrix K3 has a complexity of 12 additions and 4 shifts. In summary, M8.odd costs 36 additions and 24 shifts(M.N. Haggag, 2010).

To summarize, M8 is implemented using a totality of 60 additions and 36 shifts.

3) Decomposition of the matrix M16

The first process is always the factorization following the Eq.19.

$$M_{16} = M_{16.1} * P_{16.1} \quad (19)$$

Where P16.1 is presented in Eq.20 and M16.1 is given in Eq.21.

The elements of M16.1 are then grouped into two 8x8 independent matrices M16.even in Eq.18 and M16.odd in Eq. 19.

Concerning the decomposition of M16.even which is the same matrix as M8, M16.even costs 60 additions and 36 shifts.

The matrix M16.odd in Eq.19 is not easily decomposed. In order to be able to decompose this matrix, a technique of modification is used. The new modified integer cosine transformation matrix is based on the concept of the dyadic symmetry which was firstly used by Wai-Kuen Cham in (W.-K. Cham, 1989) and also applied by Haggag in(M.N. Haggag, 2010). The new matrix is obtained by modifying positions of some elements in the matrix. The modified matrix M16.odd_mod is shown in Eq.24 below(W.-K. Cham, 1989) (M.N. Haggag, 2010).

$$P_{16.1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \quad (20)$$

$$M_{16.1} = \begin{pmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9 & -25 & -43 & -57 & -70 & -80 & -87 & -90 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25 & 70 & 90 & 80 & 43 & -9 & -57 & -87 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -43 & -90 & -57 & 25 & 87 & 70 & -9 & -80 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 57 & 80 & -25 & -90 & -9 & 87 & 43 & -70 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -70 & -43 & 87 & 9 & -90 & 25 & 80 & -57 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 80 & -9 & -70 & 87 & -25 & -57 & 90 & -43 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -87 & 57 & -9 & -43 & 80 & -90 & 70 & -25 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 90 & -87 & 80 & -70 & 57 & -43 & 25 & -9 \end{pmatrix} \quad (21)$$

Where

$$M_{16even} = \begin{pmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{pmatrix} \quad (22)$$

And

$$M_{16odd} = \begin{pmatrix} 90 & 87 & 80 & 70 & 57 & 43 & 25 & 9 \\ 87 & 57 & 9 & -43 & -80 & -90 & -70 & -25 \\ 80 & 9 & -70 & -87 & -25 & 57 & 90 & 43 \\ 70 & -43 & -87 & 9 & 90 & 25 & -80 & -57 \\ 57 & -80 & -25 & 90 & -9 & -87 & 43 & 70 \\ 43 & -90 & 57 & 25 & -87 & 70 & 9 & -80 \\ 25 & -70 & 90 & -80 & 43 & 9 & -57 & 87 \\ 9 & -25 & 43 & -57 & 70 & -80 & 87 & -90 \end{pmatrix} \quad (23)$$

$$M_{16odd\_mod} = \begin{pmatrix} 90 & 87 & 80 & 70 & 57 & 43 & 25 & 9 \\ 57 & 43 & 25 & 9 & -90 & -87 & -80 & -70 \\ 80 & 70 & -90 & -87 & -25 & -9 & 57 & 43 \\ 9 & 25 & -43 & -57 & 70 & 80 & -87 & -90 \\ 25 & -9 & -57 & 43 & 80 & -70 & -90 & 87 \\ 87 & -90 & -70 & 80 & -43 & 57 & 9 & -25 \\ 70 & -80 & 87 & -90 & -9 & 25 & -43 & 57 \\ 43 & -57 & 9 & -25 & 87 & -90 & 70 & -80 \end{pmatrix} \quad (24)$$

This matrix is decomposed into Tm1 and Tm2 (W.-K. Cham, 1989).

$$M_{16.odd\_mod} = T_{m1} + T_{m2} \quad (25)$$

Where

$$T_{m1} = \begin{pmatrix} 88 & 88 & 88 & 72 & 64 & 48 & 32 & 8 \\ 64 & 48 & 32 & 8 & -88 & -88 & -88 & -72 \\ 88 & 72 & -88 & -88 & -32 & -8 & 64 & 48 \\ 8 & 32 & -48 & -64 & 72 & 88 & -88 & -88 \\ 32 & -8 & -64 & 48 & 88 & -72 & -88 & 88 \\ 88 & -88 & -72 & 88 & -48 & 64 & 8 & -32 \\ 72 & -88 & 88 & -88 & -8 & 32 & -48 & 64 \\ 48 & -64 & 8 & -32 & 88 & -88 & 72 & -88 \end{pmatrix} \quad (26)$$

And

$$T_{m2} = \begin{pmatrix} 2 & -1 & -8 & -2 & -7 & -5 & -7 & 1 \\ -7 & -5 & -7 & 1 & -2 & 1 & 8 & 2 \\ -8 & -2 & -2 & 1 & 7 & -1 & -7 & -5 \\ 1 & -7 & 5 & 7 & -2 & -8 & 1 & -2 \\ -7 & -1 & 7 & -5 & -8 & 2 & -2 & -1 \\ -1 & -2 & 2 & -8 & 5 & -7 & 1 & 7 \\ -2 & 8 & -1 & -2 & -1 & -7 & 5 & -7 \\ -5 & 7 & 1 & 7 & -1 & -2 & -2 & 8 \end{pmatrix} \quad (27)$$

With the modifications made on the matrix M16.odd, the matrix Tm1 is simplified into the product of three sparse matrices (M.N. Haggag, 2010). The decomposition of Tm1 is shown in Eq.28.

$$T_{m1} = M_1 * M_2 * M_3 * 8 \quad (28)$$

Where

$$M_1 = \begin{pmatrix} -2 & 0 & 1 & -1 & -1 & 3 & -1 & 0 \\ 3 & -1 & 1 & 1 & 0 & 2 & 0 & 1 \\ -1 & -3 & 1 & 0 & 1 & 0 & 2 & -1 \\ 0 & 1 & 0 & 1 & 3 & 1 & -1 & -2 \\ 1 & -1 & -3 & -2 & 0 & 1 & 0 & -1 \\ 1 & 1 & 1 & 0 & -2 & 0 & 1 & -3 \\ 0 & -2 & 0 & 1 & -1 & -1 & -3 & -1 \\ -1 & 0 & -2 & 3 & -1 & 1 & 1 & 0 \end{pmatrix} \quad (29)$$

$$M_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix} \quad (30)$$

$$M_3 = \begin{pmatrix} 0 & -2 & 0 & -1 & 0 & 2 & -1 & 1 \\ 0 & -1 & 0 & 2 & 1 & -1 & 0 & 2 \\ 0 & -2 & 1 & 1 & 0 & 0 & 1 & -2 \\ -2 & 0 & -1 & 0 & 2 & 0 & -1 & -1 \\ -1 & 0 & 2 & 0 & -1 & -1 & -2 & 0 \\ -2 & 0 & 1 & -1 & 0 & 0 & 2 & 1 \\ -1 & 1 & 0 & 2 & -1 & 2 & 0 & 0 \\ -1 & -1 & -2 & 0 & -2 & -1 & 0 & 0 \end{pmatrix} \quad (31)$$

M1 has a complexity of 40 additions and 8 shifts, while M2 can be implemented using only additions and has a complexity of 16 additions. As for the matrix M3, it has been implemented using 28 additions and 8 shifts. This sums the complexity of Tm1 to 96 additions and 40 shifts. Tm2 has been implemented using additions and shifts only, and has a complexity of 72 additions and 32 shifts.

The first implemented algorithm has taken into consideration all these optimizations. A second algorithm has been implemented while neglecting Tm2 in order to further reduce the complexity, so the modified matrix M16.odd_mod will be implemented as Eq.32 shows.

$$M_{16.odd\_mod} = M_1 * M_2 * M_3 * 8 \quad (32)$$

All in all, M16 costs for the first algorithm (with Tm2) 232 additions and 92 shifts. While when implementing the second algorithm (without Tm2), it uses on totality 160 additions and 60 shifts.

Software Implementation

Comparison of Computational Complexity

In this section, the evaluation of the computational complexity was done for both algorithms the original and the two proposed. Table I shows that the proposed algorithms are less complex for the three sizes of the transform because they are multiplication -free which will be later useful for hardware implementation. Neglecting Tm2 in the second algorithm provides less

complexity with a gain reaching 30% of the number of additions and shifts, compared to the first proposed algorithm. The difference is only for the transform unit 16x16. As Table I shows, all the operations of multiplications are replaced by additions and shifts in addition to the factorizations and simplifications done. Implementing these two algorithms may generate a slight degradation in the quality. In a second part, an evaluation of quality is done.

Quality Evaluation

In order to evaluate the difference of quality for the three algorithms: the original and the two proposed, tests were done for different videos using different configurations while varying the quantization parameter QP: {22, 27, 32 and 37} according to the test model HM5.0 of the standard HEVC (HM5.0). The evaluation was done through the following criteria given in table II.

TABLE II EVALUATION CRITERIA

Criteria	Description	Formula
ΔPSNR (dB)	PSNR loss	PSNRp-PSNRo
ΔBR (%)	Bitrate increase	(BPp/BPo-1)*100%
BDPSNR (dB)	Bjontegaard average ΔPSNR	Defined in [10]
BDBR (%)	Bjontegaard average ΔBR	Defined in [10]

Where:

-PSNRp and BPp represent respectively the PSNR index and the bitrate of the proposed algorithm.

-PSNRo and BPo represent respectively the PSNR index and the bitrate of the original algorithm. Results in Figure 2 showed that the proposed algorithm was able to achieve comparable values of the PSNR index for the majority of videos. In the worst case, the PSNR loss reached 0.02. In the best case, the PSNR index of the proposed algorithm can exceed that of the original one. The RD-curves shown in Figure 2 for different sequences showed that the RD performance almost remained unaffected compared to the original algorithm, and this can also be readable in table III. The Bjontegaard Delta (BD) results showed useful comparison results for the two algorithms. Compared to the original algorithm, there is an overall reduction of 0.06 dB of BDPSNR for the largest video and this value may increase for smaller sequences. Concerning the BD-Bitrate, there is an increase of just 1% in worst cases. To resume, both algorithms have achieved comparable performances while providing less complexity for the proposed one. That's why we start exploring its hardware implementation.

TABLE III BJONTEGAARD DELTA RESULTS FOR ORIGINAL AND PROPOSED ALGORITHMS

	BD-PSNR	BD-Bitrate
PeopleOnStreet 2560x1600	-0.0610	1.1021
BQTerrace 1080x1920	-0.0170	0.2916
BasketBallDrill 832x480	-0.0285	0.5908
BQMall 832x480	-0.0344	0.5850
BasketBallPass 416x240	-0.0236	0.3919
Vidyo1 720x1280	-0.0358	0.7228

TABLE I COMPLEXITY EVALUATION TABLE FOR ORIGINAL AND PROPOSED ALGORITHMS

Complexity							
	4x4		8x8		16x16		
Operations	Original	Proposed	Original	Proposed	Original	Proposed 1	Proposed 2
Multiplications	8	0	24	0	88	0	0
Additions	8	16	28	60	100	232	160
Shifts	0	12	0	36	0	92	60

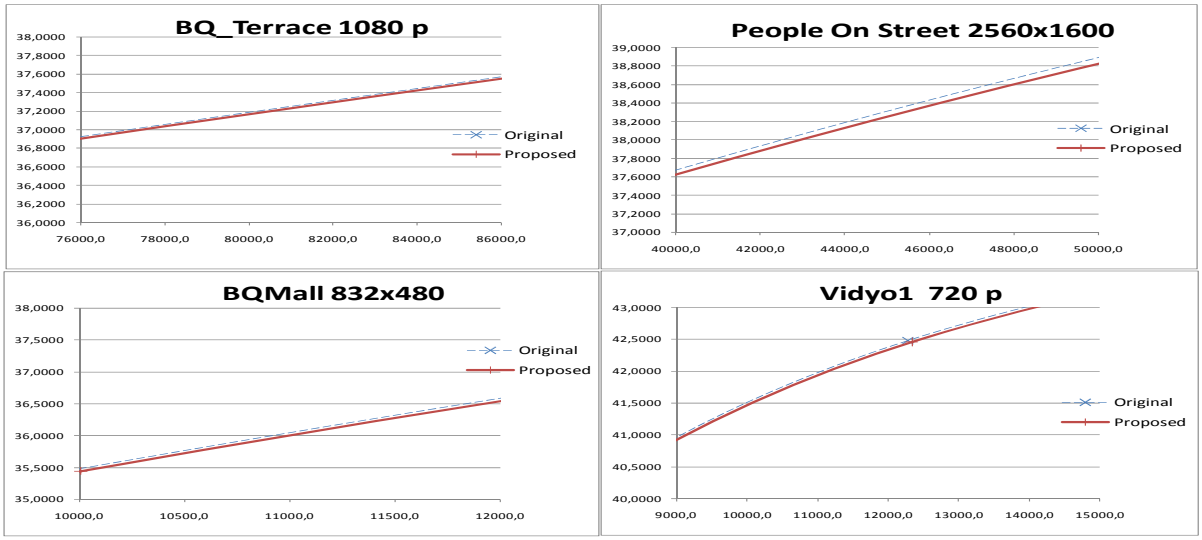


FIG. 2 RD CURVES FOR VIDEO SEQUENCES WITH QP 22,27,32,37

Hardware Implementation

In Figure 3, the diagram of the whole architecture for the integer transform of the HEVC standard is given. This architecture is composed of 4 inputs and 4 outputs, a clock signal, in addition to a reset and a selection input. The multiplexer manages the three different transformations. The transform 1D computes the transformation of each row. The resulting outputs are stored in FIFOS to take advantage of the transposition. In the next step, the outputs of FIFO are fed to the transform 1D block.

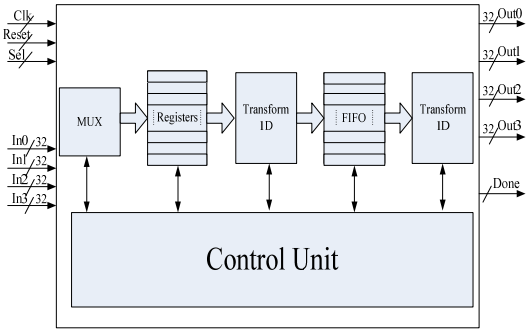


FIG. 3 BLOCK DIAGRAM OF THE PROPOSED ARCHITECTURE

Once the size of the transform unit is chosen, the inputs are read or stored in registers four by four. Whatever the size of the transform is, a unique architecture is used containing always 16 inputs and 16 outputs. Outputs of this one-dimensional transform are stored in FIFOs. Using the same architecture, the second dimension is computed. Outputs are also written four by four. The number of reading and writing cycles changes according to the size of the transform unit.

Transform 1D Architecture

This unit computes the one-dimensional transform which is applied in the horizontal direction then in the vertical direction.

The choice of the number of inputs and outputs has been done considering that the transform 16x16 is the largest implemented.

This block permits to compute one of the three transforms as Figure 4 shows. The three different transforms can be computed using this unified architecture and it takes 11 cycles to compute them regardless of the size of the transform.

To compute the 4-point-DCT, only the first 4th inputs (Src0, Src1, Src2, Src3) are considered and the others are null and the outputs will be (Dst0, Dst4, Dst8, Dst12). Concerning the 8-point-DCT, only 8 inputs (Src0, Src1, Src2, Src3, Src4, Src5, Src6, Src7) are considered and the outputs will be (Dst0, Dst2, Dst4, Dst6, Dst8, Dst10, Dst12, Dst14). To compute 16-point-DCT, we have 16 inputs and 16 outputs.

Merits are considerable due to the maximum reduction of the hardware cost while implementing this shared design for three different sizes. As it proved in Figure 5, the profit was based on the reducing number of adders from 236 for individual implementations to 160 for shared implementation. Concerning shifts, 44% of reduction was reached when implementing the unified design.

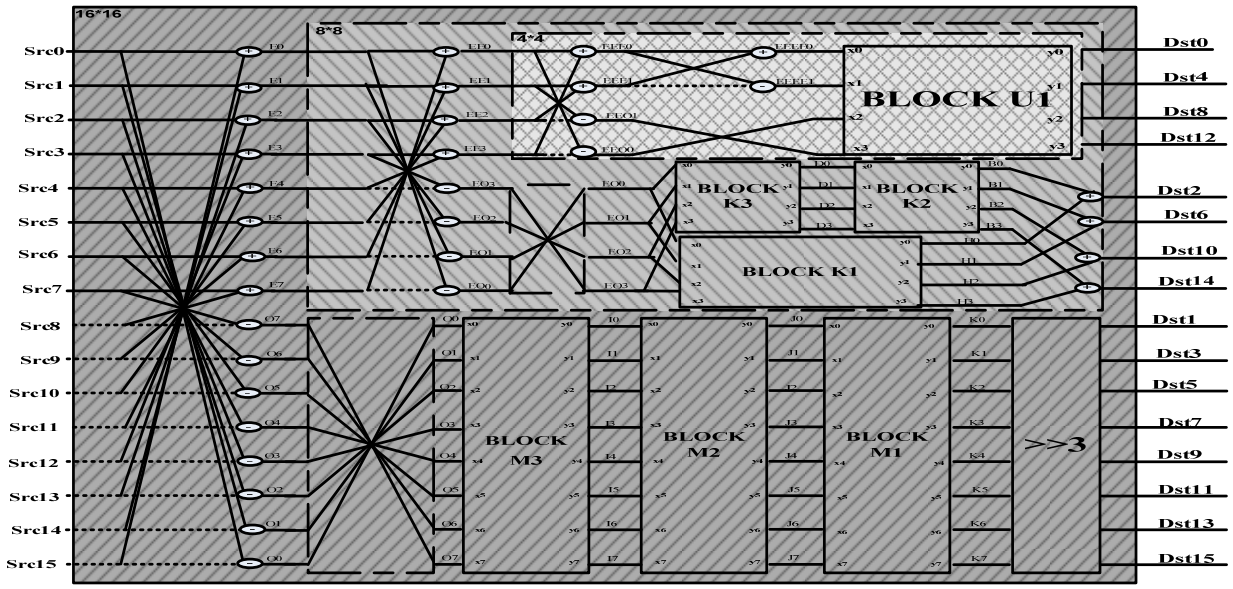


FIG. 4 TRANSFORM 1D ARCHITECTURE

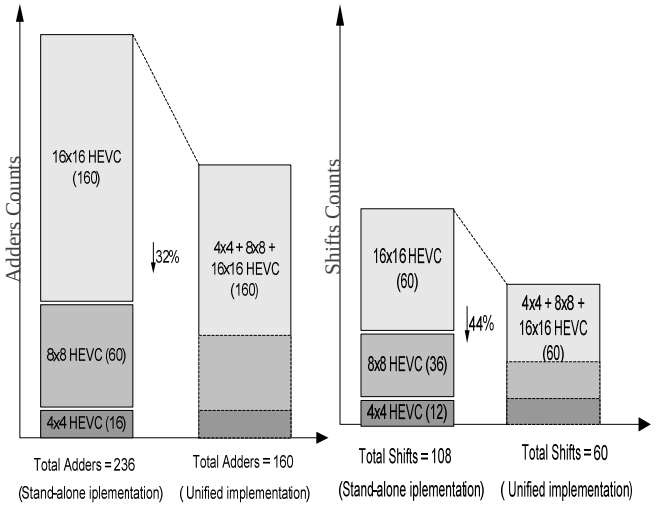


FIG. 5 COST OF THE PROPOSED SCHEME VERSUS THE ORIGINAL

The block U1 represents the basic unit in the transform as is used to compute all the transforms and it needs 10 sums and 12 shifts.

Concerning the subblocks K1, K2, and K3, they are following respectively Eq.16, Eq.17 and Eq.18. These blocks were designed only with adders and shifts.

For the subblocks used to compute the 16-point-DCT, they are designed only using shifts and additions and using the Eq.29, Eq.30 and Eq,31.

FIFOs

Our typical architecture contains 16 FIFOs that are all used when the 16-point-DCT is computed. To compute the 4-point-DCT, only 4 FIFOs are used. Concerning the 8-point-DCT, the first 8th FIFOs are used. In the proposed architecture, the new approach adopted was the use of FIFOs in order to profit from the data transposition. For example, to compute the X-point-2D-DCT ,we should proceed by the first row and each computed row is stocked in a FIFO. The process is done row by row until the end of the block. Once the step I is accomplished, we have in each FIFO a row which is transformed. The step III is responsible for the vertical transformation. These steps are fixed by the control unit as the Figure 6 shows.

Control Unit

Basically, a complete HEVC two-dimensional transform is performed by a three-step approach. The first step is the direct or horizontal transformation; while the second is the results transposition using FIFOs and the last step is the vertical transformation.

The control unit permits to control the decision of the Transform Unit size as the Figure 6 shows. When the 4-point-2D-DCT is computed, we need one cycle for reading and one for writing the outputs.

When the transform size is eight, we have two cycles for reading and two others for writing. The 16-point-2D-DCT needs 4 cycles for reading and 4 cycles for writing. The control unit serves also to have a shared design for the three different sizes of transformation.

Merits are considerable due to the maximum reduction of the hardware cost while implementing this shared design for three different sizes.

Experimental Results

One Dimensional Transform

The proposed architecture was implemented in VHDL. The implementation was verified with RTL simulations using ModelSim ALTERA. The VHDL code was then synthesized. The FPGA family selected for the synthesis was Stratix III EP3SL70F780C2 in order to compare our results to those from similar works in which the HEVC transformation was implemented for the size 16x16 discussed in the draft (Ricardo J., 2012) and an optimized one was proposed using shifts and adders only.

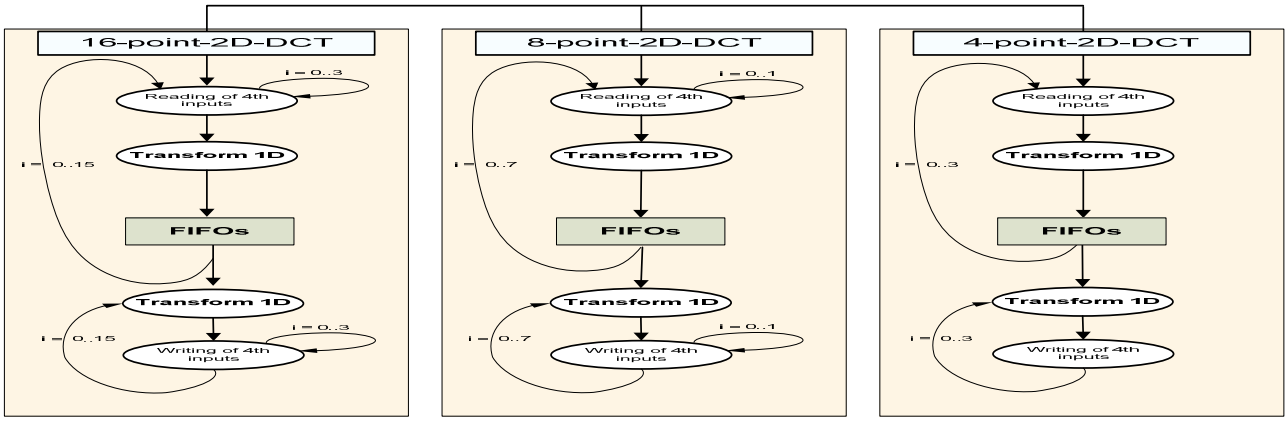


FIG. 6 CONTROL UNIT

As the implementation of the HEVC transforms was not very abundant, our proposed architecture for the three transforms (4, 8 and 16) has been compared to another architecture implementing only the one dimensional 16x16 transform (Ricardo J. ,2012). The results show an improvement concerning the hardware resources savings and the frequency. With these results, we can also evaluate the estimated throughput processed by the different architectures considering the Full HD resolution (1920x1080) because the new standard is intended especially for high resolutions (Antonio J.,2011). Table IV summarizes these results considering that the video uses 4:2:0 color sub sampling (Gary J., 2004). In order to process Full HD resolution at 30 frames per second, considering that it is the lowest frame rate required to establish the sensation of a continuous movement, a throughput of 93.312 million samples per second is required (Ricardo J.,2012). As the results shown in Table IV, our proposed architecture shows obvious improvement compared to the two others architectures with an estimated throughput reaching 4016 Msamples/sec using the Stratix III family. The optimizations done were crucial to allow the proposed architecture to achieve the required throughput in order to process the Full HD in real time.

TABLE IV SYNTHESIS RESULTS FOR 1D TRANSFORM

Architecture	ALUTs	Freq(MHZ)	Throughput (Msamples/sec)
Original HM (Ricardo J, 2012)	18,484	19.66	314.6
Optimized (Ricardo J, 2012)	5,168	87.6	1401.6
Proposed	6,091	251	4016

Two Dimensional Transfrom

As the one dimensional transform is a part from the two dimensional transform, the idea is to divide the complete 2D transform module into several small sub-modules. This architecture was designed to work efficiently in the three different sizes. Considering all the number of cycles to read all the block and transformed horizontally then vertically, Figure 7 resumes the total execution time for each transform. In addition, each transform processing needs 11 cycles. For FIFOs, the cycles number depends on the transform size. If the unit is 4x4, each FIFO needs 4 cycles, if the transform unit is 8x8, we need 8 cycle for each FIFO and for the transform 16x16, 16 cycles are needed to store the outputs in FIFO. All in all, to perform the 4x4 unit, 101 cycles are needed. Concerning the transform 8x8, the number of cycles is 218. As Figure 7 shows, to perform a bloc 16x16 composed of 256 coefficient, 500 cycles are required. To the best of our knowledge, in the literature, this is the first hardware work implementing the 2D transform for the HEVC standard. As the proposed architecture contains only four inputs and four outputs, the number of reading and writing of inputs and outputs depends on the size of the transform. For more gain, the reading cycles of the column n+1 coincides with the writing of the column n. For the transformation processing, the number of cycles is independent of the transform size and it is always equal to 11.

As shown in Figure 7, the time execution is divided into two parts; that is, an horizontal transform and a vertical one. After processing the first row, FIFO0 begins storing the different outputs at the same time of processing the second row. This process is repeated until all the bloc rows are finished. After all the outputs are stored in the different Fifos, the vertical transform begins column by column.

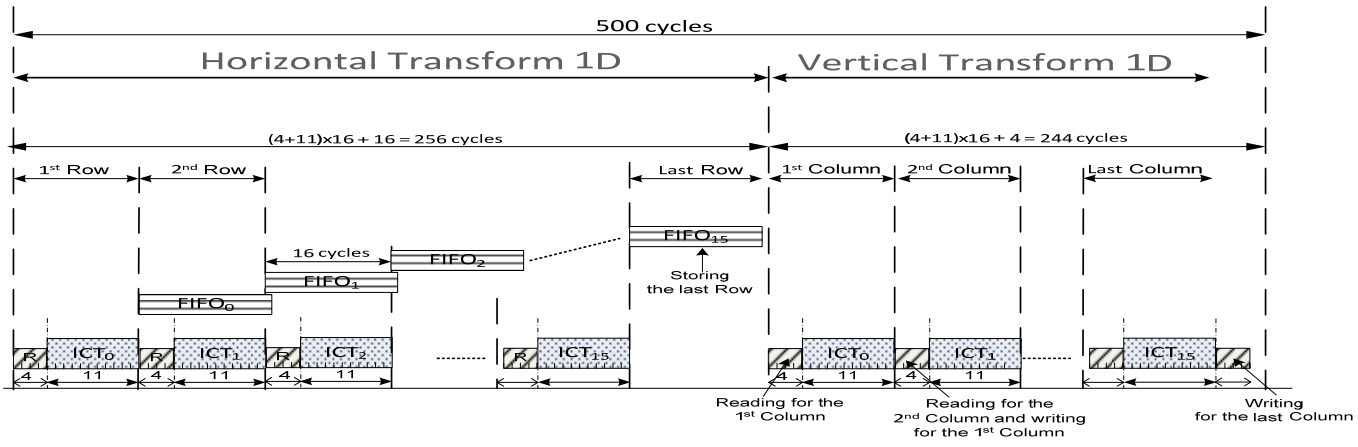


FIG. 7 TIME EXECUTION FOR TRANSFORM 2D 16X16

In order to validate our proposal, this architecture has been synthesized using Alteras Quartus II software adopted as target device the FPGA Stratix III. The following table presents the obtained synthesis results for the proposed architecture.

TABLE V SYNTHESIS RESULTS FOR 2D TRANSFORM

Architecture	ALUTs	MEMORY BITS	Freq
Proposed	14,510	8,192	251MHZ

Conclusion

In this paper, a unified hardware architecture of the two-dimensional transform has been proposed using the Modified Integer Transform which offers less complexity and uses less hardware resources while maintaining the same quality offered by the original algorithm for the modern video codec HEVC. This work is able to reach a considerable throughput even for full HD videos. Therefore, this architecture is capable to process at 240 MHZ in a Stratix III device. In future works, the hardware design for the largest dimension 32x32 is planned.

REFERENCES

Antonio J. Díaz-Honrubia, José Luis Martínez and Pedro Cuenca, "HEVC:A Review, Trends and Challenges" 2nd Workshop on Multimedia Data Coding and Transmission, September, 2011.

Ashfaq Ahmed, Muhammad Awais, Martina Maurizio, Guido Masera "VLSI Implementation of 16-point DCT for H.265/HEVC using Walsh Hadamard Transform and Lifting Scheme". IEEE 14th International Multitopic Conference, December, 2011.

Benjamin Bross, Woo-Jin Han Jens Rainer Ohm and Gary J. Sullivan "JCT-VC-G1103 Working Draft 5 WD 5", November, 2011.

Bjontegaard, G, "Calculation of average PSNR difference between RD curves", VCEG-M33, 2001.

Dong Jie "Analysis, Coding, and Processing for High-Definition Videos". A thesis from the Chinese University of Hong Kong, January, 2010.

Gary J.Sullivan, Pankaj Topiwala, and Ajay Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions" SPIE Conference on Applications of Digital Image Processing XXVII. Special Session on Advances in the New Emerging Standard: H.264/AVC, August, 2004.

HM5.0 "High-Efficiency Video coding (HEVC) Test Model HM5.0"(http://hevc.kw.bbc.co.uk/trac/browser/tags/HM5.0?rev=2765).

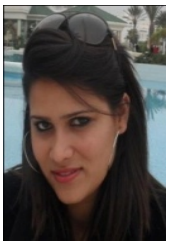
JCT-VC, "Samsung's Response to the Call for Proposals on Video Compression Technology", document JCTVC-A124 of JCT-VC, 2010.

M. N. Haggag, M. E. Sharkawy "Efficient Fast Multiplication Free Integer Transformation for the 1-D DCT of the H.265 Standard". "J. Software Engineering Applications" pp. 784-795, July, 2010

Muhammad Martuza, Khan A.Wahid "Low Cost Design of a Hybrid Architecture of Integer Inverse DCT for H.264, VC-1, AVS, and HEVC" Hindawi Publishing Corporation ,VLSI Design, 2012.

Ricardo Jeske, José Cláudio de Souza Jr., Gustavo Wrege, Ruhan Conceição, Mateus Grellert, Júlio Mattos, Luciano Agostini "Low Cost and High Throughput Multiplierless Design of a 16 Point 1-D DCT of the New HEVC Video Coding Standard" VIII Southern Programmable Logic Conference (SPL), March, 2012.

W.-K. Cham, "Development of Integer Cosine Transformation by the Principle of Dyadic Symmetry", IEEE proceedings, Vol. 136, No.4, pp. 276-282, August, 1989.



Fatma Belghith was born in Sfax, Tunisia, in 1988. She received her degree in Electrical Engineering from the National School of Engineering (ENIS), Sfax, Tunisia, in 2012. Currently, she is working toward her Ph.D in Electronic Engineering at the Laboratory of Electronics and Information Technology (LETI)-ENIS, University of Sfax. Her current research interests include video coding with emphasis on HEVC standard, hardware implementation using FPGA and embedded systems technology

Hassen Loukil Received electrical engineering degree from the National School of Engineering-Sfax (ENIS) in 2004; then M.S. and Ph.D. degrees in electronics engineering from Sfax National School of Engineering in 2005 and 2011 respectively. He is currently an assistant professor at Higher Institute of Electronic and Communication of Sfax (Tunisia). He is teaching Embedded System conception and System on Chip. He is currently researcher in the Laboratory of Electronics and Information Technology and an assistant at the University of Sfax, Tunisia. His main research activities are focused on image and video signal processing, hardware

implementation and embedded systems.

Nouri Masmoudi received his electrical engineering degree from the Faculty of Sciences and Techniques—Sfax, Tunisia, in 1982, the DEA degree from the National Institute of Applied Sciences—Lyon and University Claude Bernard—Lyon, France in 1984. From 1986 to 1990, and Ph.D. degree from the National School Engineering of Tunis (ENIT), Tunisia in 1990. He is currently a professor at the electrical engineering department—ENIS. Since 2000, he has been a group leader 'Circuits and Systems' in the Laboratory of Electronics and Information Technology. Since 2003, he has been responsible for the Electronic Master Program at ENIS. His research activities have been devoted to several topics: Design, Telecommunication, Embedded Systems, Information Technology, Video Coding and Image Processing.

Convergence Architecture for Home Service Communities

Warodom Werapun[*1], Julien Fasson[2], Beatrice Paillassa[3]

[*1]Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University, Thailand
[2 3]University of Toulouse, IRIT Laboratory, INP-ENSEEIHT, Toulouse, France

[*1]warodom@coe.psu.ac.th; [2]julien.fasson@enseeiht.fr; [3]beatrice.paillassa@enseeiht.fr

Abstract

Nowadays, home networks have integrated day to day life through the classical internet access and deliver numerous services to end users. This home entrance is a real opportunity for operators to deploy services directly between homes. However, one major issue is the interconnection between Home Networks (HN) which requires suitable architectures and efficient authentication mechanisms. In this paper, two network architectures were proposed to interconnect HNs in order to support home service delivery and then compared with the IMS as reference architecture. The first architecture was based on a centralized SIP solution and used HTTP digest for authentication purpose; while the second proposition consisted in a distributed architecture based on pure P2P and Identity based cryptography. The study of these two solutions has been undergone through the simulation of a simple photo sharing scenario. As a result, the centralized SIP solution can be relevant for an average number of users and the easiest way to deploy new services. The decentralized solution (pure P2P) can be deployed for small service communities and may be compliant to larger system with improved algorithms.

Keywords

Authentication; Network Architecture; Home Services; P2P; IMS; SIP, Content Network

Introduction

Convergence architectures are elaborated to support multiples services over multiple networks and technologies. According to this concept, operators of ITU networks are elaborating the Next Generation Networks. The basics are to use packet networks to transport services in a common way with various data of services and to offer an integrated level for the management of services (i.e. user profile, authentication, etc.) whatever the access used is. The paper focuses on the service convergence aspectby studying IMS architecture in order to deliver services at home users, in which two alternatives architectures have been proposed.

In the past, network operators handled services with dedicated architectures as voice with the PSTN and Internet data with ADSL. Nowadays, especially when considering the explosion of smartphones, services are more concerned about content delivery. The objective is to establish contact not only between users (e.g. telephone, video conference) but also between a user and content including data, voice, video and any combination of them. Content delivery architectures are proposed by service operators over Internet (i.e. Akamai) through the networks of other operators. In this study, our objective is to study a service delivery architecture that would be directly proposed by a network operator to its users, at their homes. While content services are already available, the originality of this work is to consider them not in the "Internet space" but in the home and operator domains. Data are owned by users at home and their sharing is orchestrated by the network operator.

Users are able to share content among a community, through a home gateway (operator box). The resource location (named index management in the paper) could have been managed as it pleases. However, to deploy such services without a too fussy application control part, a global management of access rights is needed: resource indexation, or at least community management, should be proposed by the global architecture.

Various network operator architectures can be deployed. In this paper, two solutions were presented, P2P and SIP, and their advantages and inconveniences were studied. The first part of the paper determined the main exchanges for the given scenario on a pure P2P architecture and on a SIP based centralized solution. Then, authentication mechanisms are analysed. From this analysis, modelling and evaluation of architectures were conducted. In the last part, the propositions were compared to the actual service architecture of the operators: IMS.

Content Service Architectures

Types of Architectures

Two kinds of solution were studied to deploy the service depending upon the service usage that may be low or important. There are pure P2P [RFC 5694] which stands for original Internet design where end-to-end paradigm is respected, and Session Initiation Protocol (SIP) [RFC 3261] solution which is based on a network paradigm with some centralisation.

Since in a first step small communities has been taken into account, pure P2P architectures with its ease to deploy services may offer a relevant solution. However, centralized solution can also be considered because they are more compliant to the operator point of view. Centralized SIP seems an unavoidable proposition and will be compared with pure P2P

This work utilized a simple content delivery service, photo sharing, as a sample reference scenario to analyse and compare network architectures. In this scenario, Alice shares a photo with Bob.

Three parts are considered for the service. There are:

- Publishing, the sharer maintains resource indexes to downloaders.
- Searching, the downloader lookups resource addresses.
- Retrieving, the downloader gets resources from sharers at the addresses obtained during the Searching step.

Next section detailed the main steps of the scenario for each architecture.

Scenario in Decentralized Pure P2P

In pure P2P, each peer directly keeps the index of their own resources. Thus there is no publishing step. Bob looks for a photo by selecting keywords and flooding them to its neighbours (searching). All peers forward his request until there is a match with local index or until the applicative Time-To-Live (TTL) is reached (e.g., 4 hops for default configuration in a typical Gnutella client). Each matching node answers Bob. Finally, he can select the relevant photo and download it directly (retrieving).

Scenario in Centralized SIP based

In the centralized solution, Alice publishes her resource indexes to the catalogue server (publishing), while Bob sends his searching query to the catalogue server. This one answers Bob that Alice has a matching photo (more than one searching result can be returned). Bob can select the most relevant answers before starting to download it (retrieving).

For publishing and searching data, SIP PUBLISH and OPTIONS were selected, respectively. Retrieving was done by using SIP INVITE.

Authentication Aspects

Authentication mechanisms were detailed along the photo sharing scenario so that signalling exchanges will be evaluated.

Identity based Cryptography in Pure P2P Solution

P2P architectures for resource sharing did not initially provide strong authentication mechanisms. However, the global need of security in the Internet has brought authentication mechanisms to P2P. The challenges are: eliminated certificate server, inside intruder and management cost etc. There are many existing solutions like Web of Trust using Pretty Good Privacy (P. Zimmermann, 1995), Distributed authentication and Byzantine fault tolerance (V. Pathak, 2006), and trusted score mechanisms (Y. Zhang, 2009).

In an HN context, the situation may be a bit different since home gateway could be configured by the operator. It is proposed in this study then to use the Identity Based Cryptography (IBC) (A. Shamir, 1985) to avoid issues related to validation; in which the user's identity (ID) was taken as a public key and so eliminating the need of a certificate server. IBC is based on the elliptic curve encryption, an asymmetric cryptography that offers many advantages in terms of performance and deployment for distributed environment (I. Blake, 2005).

There has a trusted key generator centre (PKG), whose task is to set up the whole system parameters and to extract all user private keys based on user identity and system parameters. After Bob and Alice obtained their private keys and system parameters from PKG, Bob applied Alice's ID as her public key to encrypt his message and sent it to her. Bob did not need to verify Alice's public key since Alice ID is never bogus. Alice can decrypt Bob's message using her private key. Note that ID may be associated to an expiration date for key revocation purposes (e.g., alice@example.com|12mar 2012). In this case, PKG needs to check that there is no user with duplicated ID and different expired dates to prevent ID impersonation. Revocation cost is not more than traditional public key cryptography and gets advantages in the certificateless. It was assumed that

there is not any malicious action during this phase since the goal of this study is to evaluate signaling. Degree of protection can be considered in other research topics.

The precise signaling exchanges are:

Registration phase: Bob, the Source Node (SN), connects to the P2P networks using any kind of bootstrap information to the bootstrap node(s) with its signature and encrypts it with the identity. Each future neighbor decrypts the message and verifies the SN signature. Then a register result is sent back that is signed by its signature and encrypted by the SN Identity. Two signaling messages per neighbor have been exchanged.

Searching phase: SN floods a searching query with its signature to its neighbors. They forward the message according to the specific time-to-live (TTL) value.

Retrieving phase: Once SN found content at Correspondent Node (CN),its ID is taken as a public key to sign and encrypt data that will be sent to CN. SN sends retrieve request. CN returns a retrieve result with session key and its signature. This message is encrypted by SN Identity. SN verifies the message and sends back ACK message to CN. SN sends OPEN to request sharing resource which can also be encrypted with the previously session key. CN responds ACK to SN and session is established. After that session is terminated by sending CLOSE and waiting ACK, the number of signaling messages is 7.

HTTP digest in Centralized SIP based Solution

Many solutions have been proposed to add authentication (W. Werapun, 2009). With the help of SIP proxies, users are able to be more secure by introducing some filtering and also dividing verification tasks from a centralized server. Besides, it is proposed to use HTTP digest (J. Franks, 1999) with pre-share secret key for authentication since it is simple, can be deployed with a centralize AAA server with proxies, and outperforms asymmetric key and is already supported by SIP. HTTP digest is an acceptable method where users negotiate credentials with a server. Pre-shared password is digested with some related text before being sent on the network to authenticated users. The algorithm for HTTP digest authentication is MD5.

Illustration of this solution is shown in Fig. 1. Pre-shared key between all users and AAA is required and links between proxies and AAA are secure tunnel. This assumption is reasonable since all proxies are usually fixed.

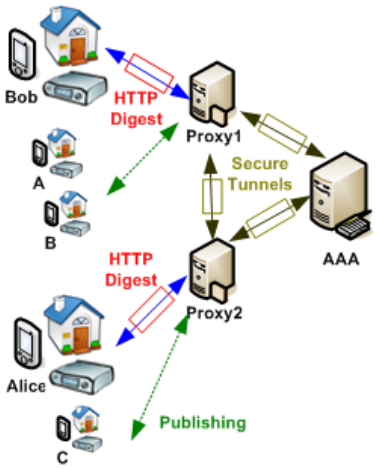


FIG. 1 SIP BASED AUTHENTICATION

The main steps of this authentication are:

Registration phase: All communications are done via Proxy. SN sends a SIP register to AAA server as shown in Fig. 2. AAA server returns 401-Unauthentication to SN that then uses its pre-shared key and the previous parameters (nonce) from server to generate authentication information and returns to AAA a Register-AuthInfo. AAA server verifies the returned authentication information and sends back SIP 200 OK with parameters for a session key generation and a session key. The proxy removes the session key from the message and forwards the 200 OK with only the session key parameters. Finally, SN generates session key from received parameters with pre-shared key and starts to create secure tunnel with proxy by using this session key. There are 4 messages exchanged between SN and Proxy and 4 messages between Proxy and AAA.

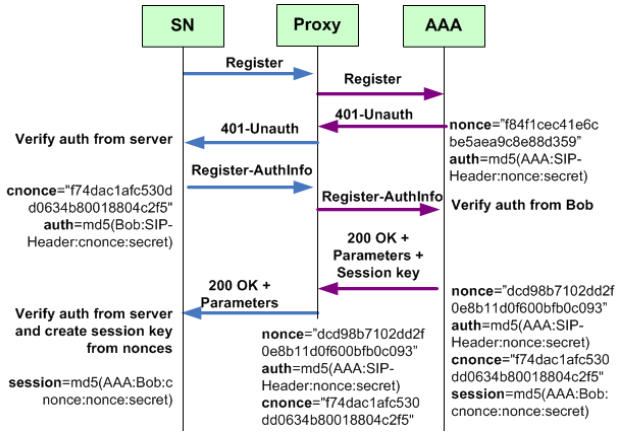


FIG. 2 HTTP DIGEST WITH SIP REGISTRATION

Publishing phase: CN publishes its resource indexes to a catalogue server (Cat) via its proxy server (Proxy). Indexes are embedded in SIP PUBLISH. Cat replies

with a 200 OK message. There are 2 messages from CN to Proxy and 2 messages between Proxy and Cat.

Searching phase: SIP OPTIONS is used as querying information from nodes. SN searches indexes from Cat by sending SIP OPTIONS including its keywords. Searching results are embedded in SIP 200 OK. 2 messages are exchanged between SN and Proxy and 2 between Proxy and Cat.

Retrieving phase: SN opens a session with the resource owner by sending a SIP INVITE with application specific query (e.g., photo URI). This invitation passes through its proxy and the CN proxy. CN returns 200 OK to SN that then sends back ACK to CN. Then, it downloads content directly and tears down the session by sending BYE and receiving OK. 5 messages are exchanged.

Modelling and Evaluation

In this part, the signalling cost of architecture, eventually compare the solutions was firstly analyzed.

Evaluation Reference: IMS Architecture

IP Multimedia Subsystem (IMS) (3GPP, 2003) is an operator view of service architecture which can provide strong security mechanisms. Therefore, it was considered as an architecture reference. IMS uses a long-term sharing secret key which is embedded in ISIM (IMS Subscribe Identity Module) to establish a secure channel. This process, called Authentication and Key Agreement (AKA) is based on HTTP digest authentication [RFC 3310]. The security parameters are generated during AKA process for having a session key used to secure communication channels.

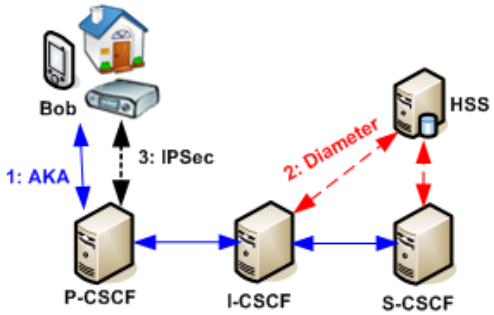


FIG. 3 IMS AKA

The main IMS entities are presented in Fig. 3. There are three types of Call Session Control Function (CSCF): Proxy (P-CSCF), Interrogating (I-CSCF) and Server (S-CSCF). Home Subscriber Server (HSS) is the client database.

Bob sends SIP REGISTER to its P-CSCF which acts as an outbound/inbound SIP proxy server for the IMS terminal (arrow 1: AKA). It communicates with DNS (Domain Name Server) to resolve SIP server domain. Then, P-CSCF forwards the I-CSCF. This one searches Bob's corresponding S-CSCF from HSS. Then, it forwards the REGISTER message to this S-CSCF. Communications between CSCFs and HSS use Diameter protocol. S-CSCF asks an authentication vector from HSS and challenges Bob through I-CSCF and P-CSCF consequently. If successful, Bob uses authentication parameters to establish IPSec [RFC 2401] with P-CSCF (arrow 3: IPSec). Once the two pairs of IPsec ESP Security Associations (SAs) have been done, the traffic is sent through the encrypted tunnel and P-CSCF will identify all the SIP requests coming through the corresponding SA as pertaining to the authenticated user. Thus, no more user authentication is needed. SIP signaling is clearly sent in the encrypted tunnel.

Signaling evaluation is summarized for the different steps:

Registration phase: Fig. 4 (left) shows registration signaling, and there are 3 parts: create credential (with SIP REGISTER and some diameter protocol with HSS), IPSec setup and challenge.

Publishing/Searching phase: Fig. 4 (right) shows publishing (or searching) signaling. There is always SIP INVITE to open a session. Note that publishing and searching have the same signaling in IMS.

Retrieving phase: Fig. 5 shows retrieving signaling. SN creates a session with CN using SIP INVITE. After a session is established, SN sends retrieving query through CN via IMS elements. Data is transferred from CN to SN. SN tears down a session.

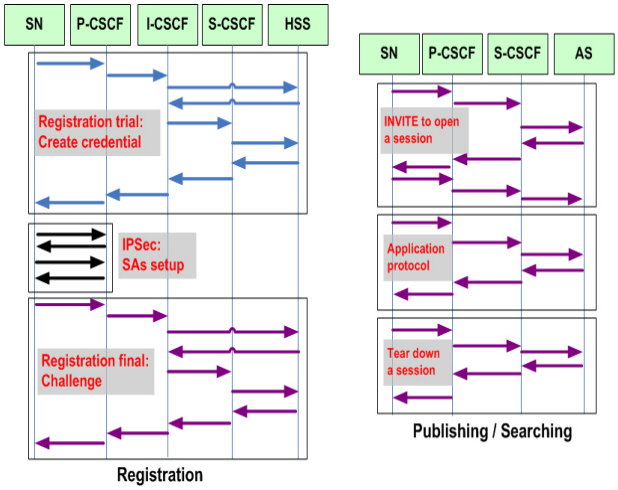


FIG. 4 IMS REGISTRATION AND PUBLISHING/SEARCHING SIGNALING

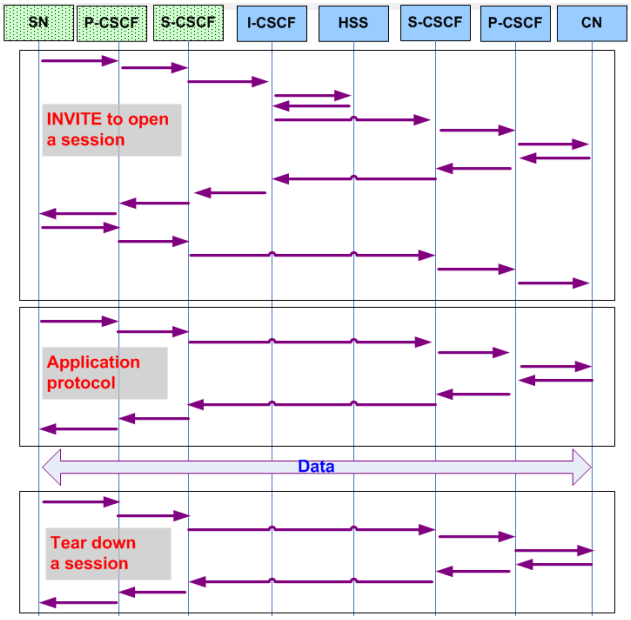


FIG. 5 IMS RETRIEVING SIGNALING

Topology Modeling and Signaling Cost Analysis

Considering the searching phase, the signaling in centralized solution does not depend on number of users whereas the pure P2P solution does. Thus, the first step of the study consists in analyzing pure P2P topology.

1) Topology Modelling and Evaluation

Characteristics of P2P model are the number of nodes and the interconnections distribution. In pure P2P overlay, the distribution of node degrees may be modeled from a power law (M. Jovanovic, 2001). A number of real-life P2P networks have compliant topologies with this distribution (M. Jovanovic, 2001). When a node *i* joins the network, the probability that it connects to a node *j* already belonging to the network is given in (1), where *d* is the degree of the target node, *V* is the set of nodes that have joined the network and $\sum_{k \in V} d_k$ is the sum of degrees of all nodes that have previously joined the network.

$$P(i,j) = \frac{d_j}{\sum_{k \in V} d_k} \quad (1)$$

The size of physical network is assumed to be 1,000 connected nodes (home gateways and routers) and the size of participated users is 100 connected users. Pure P2P topology is generated by using BRITE (A. Medina, 2001) tool with Barabasi-Albert model (A. Barabasi, 1999)which creates a graph having 1,000 nodes for the physical network and *n* nodes for an overlay network following the power law distribution. From this physical network, 100 overlay topologies with minimum degree equal to 1 and 100 others with minimum degree equal to 2 are generated per each value of *n*. These nodes are randomly selected among the nodes of degree 1 of the physical network.

The flooding signaling and the average hop distance have been evaluated. For the flooding TTL has been valuated to be 4 as default configuration of typical Gnutella client. An average search signaling was computed by testing communication among all nodes (changing source and destination nodes until reaching all nodes). As a result, S_n^{cost} values were obtained which are denoted as the signaling searching cost for *n* connected nodes in the overlay network (*n* varying from 10 to 100 and increase 10 nodes for each step). Concerning the hop distance, senders and receivers have been randomly selected in the simulated overlay network, then routing based on Dijkstra algorithm is applied resulting in an average hop distance between SN and CN of 7.

2) Signaling Cost Analysis

The cost considers the number of signaling exchange, expressed in the previous section and the distance between the source and the destination of the exchange. A distance of *n* induces n transmission cost in the network.

Active users (*N*) are 50% of *n* connected nodes. *N* varies from 5 to 50 in simulations. *R* is the average user transaction rate for each phase. It is set to be 1 and 5 in simulations. γ is the unit packet transmission cost and $d_{SN\text{-}CN}$ denotes the average hop distance between SN and CN.

Pure P2P with IBC

Total transmission signaling cost by using pure P2P is:

$$C_{total}^{IBC} = C_{register}^{IBC} + C_{publish}^{IBC} + C_{search}^{IBC} + C_{retrieve}^{IBC}$$

Where, each transmission signaling phase is given by:

$$C_{register}^{IBC} = N^{*}R_{reg}\,(\gamma\,(2\,d_{SN\text{-}CN}))$$

$$C_{publish}^{IBC} = 0 \text{ (Nodes keep indexes, no publishing)}$$

$$C_{search}^{IBC} = N^{*}R_{sea}\,(\gamma\,(S_n^{cost}\,d_{SN\text{-}CN}))$$

$$C_{retrieve}^{IBC} = N^{*}R_{ret}\,(\gamma\,(7\,d_{SN\text{-}CN}))$$

SIP based with Http Digest

Total transmission signaling cost by using SIP based is:

$$C_{total}^{SIP} = C_{register}^{SIP} + C_{publish}^{SIP} + C_{search}^{SIP} + C_{retrieve}^{SIP}$$

Where, each transmission signaling phase is given by:

$$C_{register}^{SIP} = N^{*}R_{reg}\,(\gamma\,(\,4d_{SN\text{-}Proxy} + 4d_{Proxy\text{-}AAA})\,)$$

$$C_{publish}^{SIP} = N^{*}R_{pub}\,(\gamma\,(\,2d_{CN\text{-}Proxy} + 2d_{Proxy\text{-}Cat}\,)\,)$$

$$C_{search}^{SIP} = N^{*}R_{sea}\,(\gamma\,(\,2d_{SN\text{-}Proxy} + 2d_{Proxy\text{-}Cat})\,)$$

$$C_{retrieve}^{SIP} = N^{*}R_{ret}(\gamma\,(5d_{SN\text{-}Proxy} + 5d_{Proxy\text{-}Proxy} + 5d_{Proxy\text{-}CN})\,)$$

$d_{x\text{-}y}$ denotes hop distances from x to y where they are elements of SIP architecture

IMS with AKA

Total transmission signaling cost by using IMS is:

$$C_{total}^{IMS} = C_{register}^{IMS} + C_{publish}^{IMS} + C_{search}^{IMS} + C_{retrieve}^{IMS}$$

Where, each transmission signaling phase is given by:

$$C_{register}^{IMS} = N^{*}R_{reg}\,(\gamma\,(\,8d_{SN\text{-}Pcscf} + 4d_{Pcscf\text{-}Icscf} + 4d_{Icscf\text{-}HSS} + 4d_{Icscf\text{-}Scscf} + 4d_{Scscf\text{-}HSS})\,)$$

$$C_{publish}^{IMS} = N^{*}R_{pub}\,(\gamma\,(7d_{CN\text{-}Pcscf} + 7d_{Pcscf\text{-}Scscf} + 7d_{Scscf\text{-}AS}\,)\,)$$

$$C_{search}^{IMS} = N^{*}R_{sea}\,(\gamma\,(\,7d_{SN\text{-}Pcscf} + 7d_{Pcscf\text{-}Scscf} + 7d_{Scscf\text{-}AS}\,)\,)$$

$$C_{retrieve}^{IMS} = N^{*}R_{ret}\,(\gamma\,(7d_{SN\text{-}Pcscf1} + 7d_{Pcscf1\text{-}Scscf1} + 2d_{Scscf1\text{-}Icscf} + 5d_{Scscf1\text{-}Scscf2} + 2d_{Icscf\text{-}HSS} + 2d_{Icscf\text{-}Scscf2} + 7d_{Scscf2\text{-}Pcscf2} + 7d_{Pcscf2\text{-}CN}\,)\,)$$

$d_{x\text{-}y}$ denotes hop distances from x to y where they are elements of IMS architecture

Simulation and Comparison Evaluation

The trunked Pareto distribution is assumed for packet length with an average equal to 480 bytes (D. Tarchi, 2006). Data bit rate is 10 Mbps, and γ is equal to 3.84 x 10^{-4} sec.

In P2P, hop distance between SN and CN is averaged on 7, in SIP and in IMS, the distance from the user to the service network (i.e. $d_{S/CN\text{-}Proxy}$ in SIP, $d_{S/CN\text{-}P\text{-}CSCF}$ in IMS) is 3, counting on 2 routers between path on average; while the distance inside the service network (i.e. $d_{Proxy\text{-}Proxy}$ for SIP, and $d_{x\text{-}CSCF}$ $d_{y\text{-}CSCF}$ for IMS) is set to be 2 (A. Munir, 2008). Note that the value range is representative of real situation but of course another modeling, especially of P2P, would produce different values. Meanwhile, the result trends would be similar.

Fig. 6 and Fig. 7 indicate the transmission cost of communication that corresponds to active nodes and establishment in function of connected node number. There are 1 register, publishing, searching and retrieving per active node (R=1 for all phases) in Fig. 7, and 5 searching (R_{sea} = 5) in Fig. 8. Conformal to formula, the transmission cost grows with active users. SIP based has the lowest transmission cost. Pure P2P has the highest one, due to the flooding algorithm in searching phase (more sophisticated algorithm may be adopted). Moreover, it has more hop distance than others. The gap between IMS transmission cost and SIP one decreases with the number of searching per user. Thus, for a frequently used service, IMS and SIP do not have very different performance compared to pure P2P.

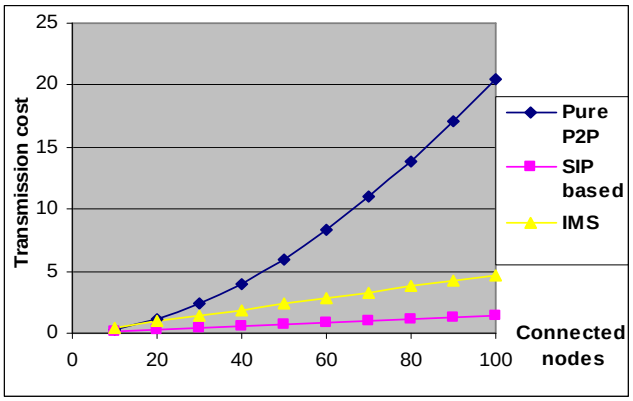


FIG. 6 TRANSMISSION COST WITH 1 REGISTRATION, PUBLISHING, SEARCHING RETRIEVING/USER

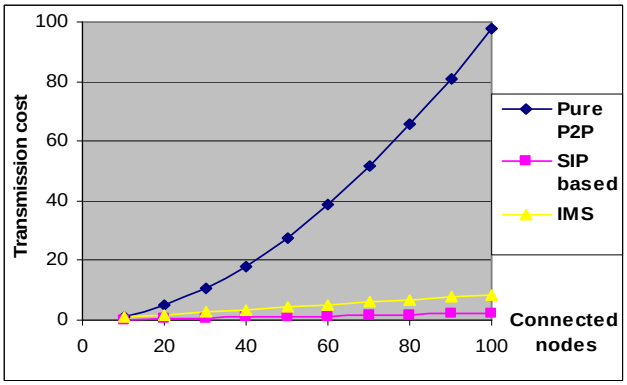


FIG. 7 TRANSMISSION COST WITH 1 REGISTRATION, PUBLISHING, RETRIEVING/USER AND 5 SEARCHING/USER

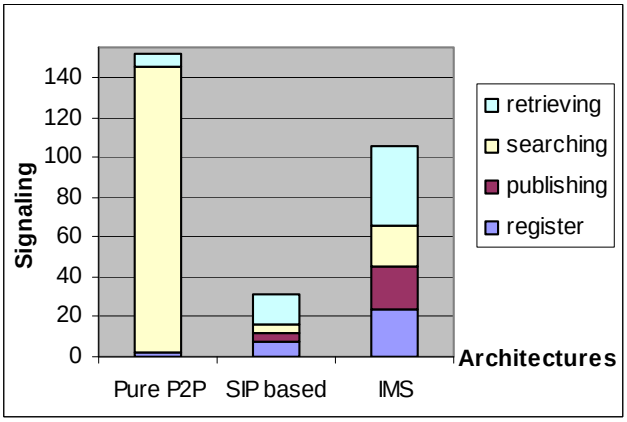


FIG. 8 SIGNALING IN NETWORK ARCHITECTURES FOR ONE SEARCHING

Signaling is detailed on Fig. 8. SIP has least signaling because it does not treat all SIP IMS standard signaling, (information is included in the SIP messages during publishing and searching) and it has less architecture components than IMS. Furthermore, IMS must open the session with SIP INVITE when it starts any connected phase (e.g., Publishing, Searching). The major cost in P2P is due to research. With a more sophisticated algorithm, a factor 2 of improvement could be obtained so that P2P could reach same level of performance as IMS. Although pure P2P has more signaling compared with SIP, each node does not handle too much signaling.

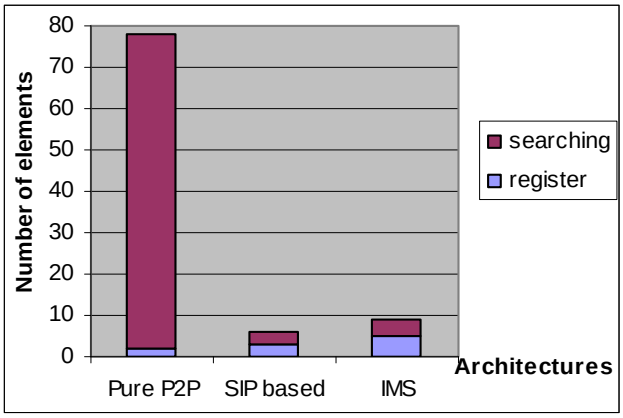


FIG. 9 PARTICIPATED NUMBER OF ELEMENTS IN ARCHITECTURES

When considering the number of entities which forward signaling in Fig. 9, it is seen that there are many nodes which work together in decentralized fashion for sharing resources. Signaling per node might not be very different with SIP based solution. On the other hand, the centralized catalogue server in SIP and in IMS handle many searching queries compare with pure P2P. Thus P2P is not so costly for a large number of service users in each node.

Deployment Assumption

Assumption for deployment is shown in Table1. IBC with pure P2P does not need to have pre-shared key, however, it also needs to have private key which correspond with its ID and system parameter. IBC can be well integrated in distributed environment while SIP based uses pre-shared to do HTTP digest with AAA server for session key generation and uses between its proxy. In addition, the proxy can be a firewall to protect the system from other intruders attacking the system (e.g., DoS attack). In the computation view, symmetric cryptography is used in SIP based that takes less computing time than asymmetric cryptography used in IBC too.

TABLE 1 : SECURITY ASSUMPTION FOR DEPLOYMENT

	Pure P2P	**SIP Based**	**IMS**
Setup	Private key correspond with ID and parameters	Pre-shared key between user and AAA	Pre-shared key between user and HSS
Trusted elements	PKG	Between proxy and AAA	CSCFs, AS and HSS
Security protocol	IBC Session key	HTTP Digest Session key	HTTP Digest IPSec

IMS uses centralized user profiles. All users must authenticate themselves through P/I/S-CSCFs components and establish IPSec tunnel between UEs and P-CSCF. Actually, our SIP based is quite similar to IMS architecture since they use a centralized catalogue and have a proxy to manage and protect a network. The main different between them are user profile location and CSCF components (e.g., I/S-CSCF, HSS).

All of the proposed network architectures provide authentication mechanisms to secure home services. These authentication mechanisms do not tie with a specific cryptograph algorithm. Network administrator freely chooses a suitable protocol (e.g., AES, 3DES, RSA, MD5 and SHA-1).

Conclusions

In this work, two network architectures were proposed for delivery home content to home users and then compared to IMS reference architecture, considering the scaling factor.

The focus was placed on authentication mechanisms which can be used for user identification to secure communities. In terms of security aspects and deployment, all considered architectures have some similarities since they required previously setup which can be done through home gateway configuration.

In terms of performance, architectures have been studied from their signaling. Photo sharing service is used to illustrate the function of presented architectures. It has been shown that SIP based has least signaling. Indeed, even if it presents a less complex architecture, it can still provide effective user authentication. In addition, the SIP solution can be improved by the use of SIP proxy (kind of firewall against intruders) but with a cost: more signaling and more complexity.

Considering the scaling factor, the analysis showed that for a successful service (frequently used), IMS performances are improved. For the P2P solution, signaling is too heavy and can be deployed only in

small communities. Nevertheless, the P2P solution can be highly improved by the use of an efficient research mechanism like, Distributed Hash Table. Experiments have been done with DHT P2P by implementing a SIP RELOAD architecture and this assumption has been confirmed.

REFERENCES

3GPP, "IP Multimedia Subsystem (IMS)," TS 23.228, Release 8, Version 8.7.0, Dec 08 Peer-to-Peer Lookup Protocol for Internet Applications, IEEE/ACM Transactions on Networking, Vol 11, Feb 2003

A. L. Barabasi and R. Albert, Emergence of Scaling in Random Networks, Science, pages 509-512, Oct 1999

A. Medina, A. Lakhina, I. Matta, and J.Byers, "BRITE: Boston university representative internet topology generator", Boston University, Apr 2001

A. Munir, W. Vincent and S. Wong, Interworking Architectures for IP Multimedia Subsystems, Computer science mobile networks and applications, Volume 12, Numbers 5-6, Springer Science, Apr 2008

A. Shamir, "Identity-based cryptosystems and signature schemes", Advances in Cryptology (Proceedings of Crypto '84), Lecture Notes in Computer Science, vol 196, Springer-Verllag, 1985

D. Tarchi, R. Fantacci, M. Bardazzi, Quality of service management in IEEE 802.16 wireless metropolitan area networks, In Proc of IEEE International conference on communication (ICC), Turkey, Jun 2006

I. F. Blake, G. Seroussi and N. P. Smart, Advances in Elliptic Curve Cryptography, Cambridge University Press, London mathematical Society Lecturer Note Series 317, 2005

J. Franks, P. Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen and L. Stewart, HTTP Authentication: Basic and Digest Access Authentication, RFC 2617, IETF Network Working Group, Jun 1999

M. Jovanovic, F. Annexstein, and K. Berman: Modeling peer-to-peer network topologies through "small-world" models and power laws, IX Telecommunications Forum TELFOR 2001, Belgrade, 2001

M. Jovanovic, Modeling large-scale peer-to-peer networks and a case study of Gnutella, MS. Thesis, University of Cincinnati, Cincinnati, Ohio, USA, 2001

P. Zimmermann, The Official PGP User's Guide. MIT Press, Cambridge, Massachusetts, 1995

V. Pathak and Iftode, Byzantine fault tolerant public key authentication in peer-to-peer systems, Computer Networks, Special Issue on Management in Peer-to-Peer Systems: Trust, Reputation and Security, 50(4): 579-596, Mar 2006

W. Werapun, A. Abou El Kalam, B. Paillassa, J. Fasson, "Solution Analysis for SIP Security Threats", International Conference on Multimedia Computing and Systems, ICMCS 2009, Apr 2009

Y. Zhang, S. Chen and G. Yang, SFTrust: A double trust metric based trust model in unstructured P2P system, 2009 IEEE International Symposium on Parallel & Distributed Processing, May 2009

Warodom Werapun received the Ph.D. degree from Institute National Polytechnique of Toulouse (INPT), France. During 2008-2010, he did a joint Feel@Home research project supported by European Celtic Project. He is currently an associate department head for administration at the department of computer engineering, Prince of Songkla University, Phuket Campus, Thailand; meanwhile, he is a deputy head of INFAR research group. His current areas of research are network architecture, P2P, SIP, routing protocol and security.

Julien Fasson has received a Ph.D. degree in Computer Systems and Telecommunications from the University of Toulouse in 2004. Since then, he has been an Assistant Professor at IRIT laboratory of Toulouse. His main interests are focused on the integration of satellite systems in terrestrial networks and the network architecture for service integration. In this context, he has worked on NGN architectures (IMS) and mobility issues in heterogeneous, including the use of MIH in a satellite/LTE context.

Beatrice Paillassa is Professor in computer networks at the ENSEEIHT engineer school. She earned a Ph.D in computer sciences and an "Habilitation a Diriger les Recherches" in computer networks from Toulouse university; in addition, member of the research team IRT "Network and Telecommunication Engineering" at the IRIT laboratory of Toulouse. Her research interests include communication protocol design and analysis, modeling, and architecture specification. She currently works on convergence architecture, satellite networks and wireless protocols.

A Novel Scheme of Destination-based Regulation for Ad Hoc Networks

Hassene Bouhouch

SecNum Laboratory Research, Sup'Com
High School of Human Sciences of Tunis, ISSHT.
University of Tunis El Manar, Tunisia.

hassene_bouhouche@yahoo.fr

Abstract

Today, the rise of wireless technologies offers new prospects in the telecommunications field. Ad hoc networks (AHNs) which include mobile nodes with limited communication resources are facing a new challenge: a regulation mechanism oriented quality of service that has received much attention in research. In this paper, a destination based regulation mechanism for AHNs has been proposed, in which a novel scheme was considered for the regulation of the Real Time flows allowing a preventive and recovery behaviours to the applications. The advantages of this proposition have been shown thanks to some simulation results detailed in the end of this paper.

Keywords:

Regulation Mechanism; Destination Based Regulation; Quality of Service; Ad Hoc Networks

Introduction

The emergence of new mobile and wireless networks, with novel capabilities, offers opportunities to widen traditional internet-based applications. In fact, the recent evolution of wireless communication means allows the handling of information through portable units with particular characteristics such as modest flows and communication resources, autonomous and limited source of energy and frequent disconnections.

Because the QoS provision constitutes a real challenge for current networks and this aspect has not been sufficiently dealt with in AHNs, we have chosen to focus our attention on this issue. Particularly, this paper aims to propose a regulation mechanism in AHNs QoS oriented i.e. allowing customized treatments for the Real Time traffics.

In this paper, a mechanism of reaction was defined in case of variability of load allowing the management of variable application, where variability is measured in terms of QoS requirements. The adaptability allows the network to accept, by regulating Real Time flows, the application that can be normally rejected in case of congestion.

The remaining part of this paper is organized as follows: the Ad Hoc environment's is very particular and the AHNs unit's constraints are very strong, which was studied in the second section of this paper. The third section dealt with the concept of QoS in AHNs and the classes of traffic were detailed. In section 4 the process of regulation of Real Time flows was defined. Section 5, a novel proposition of a destination based regulation was exposed that introduced a set of resources allocation scheme allowing different behaviour to the applications (preventive and recovery). In section 6, the simulation results were made, followed by the conclusions in the end.

The Ad Hoc Networks Environment

In this section, the Ad Hoc networks environment's and the mobile unit's characteristics were studied.

Recently, new mobile and wireless networks types emerged, offering novel and interesting capabilities, among which, Ad Hoc networks represent an immediate example. In fact, such networks do not need any infrastructure or management entity. However, this absence of infrastructure has led to some problems. In particular, in order to ensure the transmission of information through the network, the mobile node performs the role of a station and a router. Moreover, the radio link specificities as well as the mobility of the users make unacceptable the routing protocols used in the usual networks. (T.Imienlinski, B. R. Badrinath, 1994).

Generally, AHNs used in any application where the deployment of a wired infrastructure network is too constraining (difficult to set up, duration of installation of the network does not justify its wiring...) allow a group of persons to communicate in a spontaneous way, anywhere and at anytime, while

being free to join and leave the network at any time.

In addition to a dynamic topology and nodes arbitrary mobility, AHNs, based on IEEE 802.11 standard (M. S. Gast, O'Reilly, 2002), are characterized by several constraints such as limited and shared bandwidth, limited energy, poor security, and so on, making big differences between AHNs functioning and the one of traditional wired networks (G. Pujolle, 2002). The resources allocation constitutes one of the more illustrative examples of such differences.

The Concept of Quality of Service in Ad Hoc Networks

The term 'Quality of Service' (M. Mirhakkak, N. Schult, D. Thomson, 2000) includes a large number of concepts and complementary techniques. The network offers a QoS to his users since it sets up a preferential treatment. The IETF has defined the QoS as a set of services requirements to be provided by the network to the application while transporting a packet from source to destination and which can be characterized by various criteria of performance which include the availability, the rate of errors, the response time, the delay of the connection establishment, the throughput, the loss of connection or data because of the network's congestions and the speed of the errors detection and correction …

For the QoS in the AHNs, we distinguished the following characteristics (M. A. Remiche, 2005): reliability, availability, bandwidth, latency (delay of crossing of the network), regularity of the variation of the transmission's delay, error's rate.

Researches concerning QoS in AHNs are classified into four categories: QoS models, medium access control protocols, QoS routing protocols and finally signalling protocols which try to offer a resources' reservation and a regulation mechanisms.

In (H. Bouhouch, S. Guemara El Fatmi, 2007), the authors have considered three classes of traffic according to the applications QoS requirements that can be detailed as follows:

- Class 0: Delay sensitive Real Time traffic (CBR) generated by applications having strong temporal constraints: each bandwidth allowing the requested delay is acceptable by such traffic class.
- Class 1: Bandwidth sensitive Real Time traffic (VBR_RT) generated by applications having strong temporal constraints in addition to strong constraints in terms of bandwidth.
- Class 2: Best Effort traffic (BE) generated by applications having no QoS constraints.

Regulation of Real Time Flows

A mechanism of dynamic regulation was introduced in answer to the constraints imposed by the dynamicity of Ad Hoc networks such as the nodes mobility and the phenomenon of False Admission. It is interesting to illustrate the impact of these two problems on the network's resources.

- Impact of the mobility: Fig. 1 shows that a transmitted Real-time flow between a source S and a destination D can be redirected from a node n1 to another node n2 because of the mobility of the network. In fact, the node n1, by moving, goes out of the zone covered by the source and, at the same time, the node n2 makes its appearance in this coverage area. However, this rerouting of the flow to the node n2 can engender a degradation of the QoS for the traffic seeing that the network does not operate any admission control in the intermediate nodes.

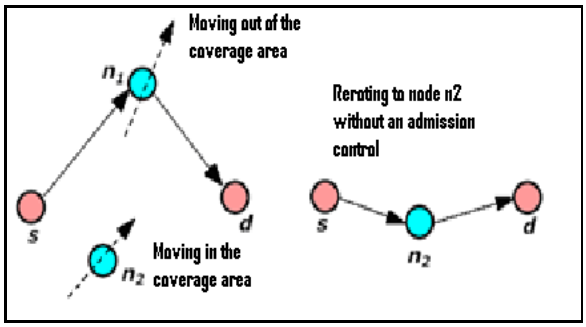


FIG. 1 CONGESTION AND DEGRADATION OF THE QOS DUE TO THE MOBILITY

If this rerouting of flows causes congestion, this one would be called congestion due to the mobility.

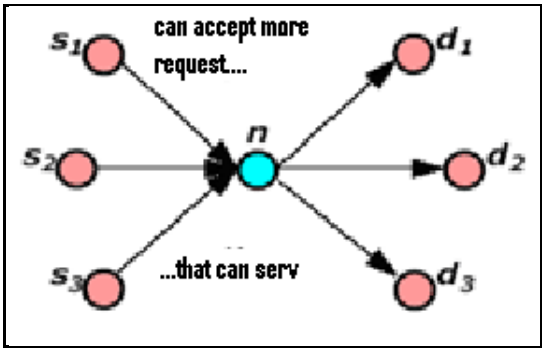


FIG. 2 CONGESTION DUE TO THE PHENOMENON OF FALSE ADMISSION

- Impact of the phenomenon of False Admission: as illustrated in Fig. 2, the nodes S1, S2 and S3 can launch transmission requests of a new Real Time flow to

nodes D1, D2 and D3 (respectively) through the same node n. If this node takes into account these three requests in a small lapse of time (before none of these three flows begins its transmission), then the admission controller will accept these three flows supposing that the node n can satisfy the requirements in terms of QoS of each of these flows. This situation is called congestion due to the phenomenon of False Admission.

Finally, it is important to indicate that the mobility and the phenomenon of False Admission represent only two aspects of the dynamicity of the network. We can also note the deterioration of the radio links caused by the interferences and the batteries limited lives used by stations.

The Notion of Explicit Congestion Notification

As already explained, the dynamicity of the network can cause violations on the requirements of accepted Real Time traffic, so a QoS degradation of some flows. To fix this problem, the ECN mechanism (Explicit Congestion Notification) (Ramakrishnan, Floyd, Black, 2001) is used to regulate the Real Time traffic. This regulation is realized when a node notices the violation of the required QoS for a Real Time flow: when a node suffers from QoS degradation, it marks every Real Time transmitted flow by an ECN notification. The destination of a marked flow has to inform, necessarily, the source of this flow. This one is going to either interrupt the transmission, or adapt it to the new conditions of the network.

The use of ECN for the traffic control concerns only the Real Time traffic. By regulation, it is meant that the ECN mechanism obliges these flows to revise their QoS requirements during the degradation of service or in the case of congestion.

The Dynamic Regulation Solutions

The decision to mark the congested Real Time flows by ECN is very delicate seeing that these flows are going to lose a part or all their privileges in term of QoS.

The literature has suggested two types of regulation (B. Reddy, I. Karthigeyan, B.S. Manoj, C. Siva Ram Murthy, 2006) called source based regulation and network based regulation. These both approach mark congested flows by ECN by different manners but adopt, then, the same treatment to solve the congestion.

1) Source Based Regulation

In the source based regulation, a congested node marks all the Real Time flows by ECN. When the destination receives a marked packet, it sends to the source a regulation request message. The source is then going to treat the flow to make it adapt to the new conditions of the network until the disappearance of the conditions having caused the congestion. From the restoring of the network, the node having been congested will stop its operation of marking ECN.

The source based regulation forces, thus, all the Real Time flows transiting by a congested node to be regulated. The excessive aspect of this approach makes that flows requiring a limited QoS can maintain their connections, on the other hand, flows demanding a high QoS will have to reinitialize their transmissions from the restoring of the network conditions.

2) Network Based Regulation

In this type of regulation, a node suffering from congestion select, randomly, a part of the Real Time flows to form a series of victims flow. The congested node marks only these flows by ECN.

When the marked series reaches its destination, the same procedure described previously for the source based regulation is launched.

If after a while the congested node does not notice improvement in the network conditions, then it proceeds to the election of a new series of victims flow.

A Destination Based Regulation Proposition

The proposed regulation solutions do not take into account the end-to-end delay for the Real Time flows as measure of the quality of service, but use rather the inter-nodes delay to detect the possible congestions of the network (M. Cherif, S. Guemara el Fatmi, N. Boudriga, M.S. Obaidat, 2007).

However, when the source and the destination are distant, in number of jumps, the end-to-end delay becomes big what makes flows sensitive to the dynamic changes of the network. Therefore, the destination can have a better judgment (that in the source) on the quality of received flows by using the deadline end-to-end delay. For this reason, the use of an algorithm based on the destination to solve the problem of dynamic regulation seems more interesting.

Thus, the use of a destination based regulation mechanism is recommended instead of both

mechanisms used, classically, for the regulation of the Real Time flows during the network's congestion. This mechanism plays an important role in the choice of victim's flows and implements a preventive behaviour to decrease the occurrence of the congestions of the network.

Indeed, during a congestion case, the destination based regulation selects a part of congested Real Time flows with the aim of regulation. This series is selected based on the initial quality of service required by flows. In this sense, and flows requiring limited QoS will be the first selected. This can be explained by the fact that flows demanding an important quantity of resources find many more difficulties during the admission control and, thus, more chances are given to finish their transmissions. On the other hand, flows demanding limited quality of service will always have the possibility of being again admitted from the improvement of the conditions of the network.

The destination based regulation mechanism is based on two behaviours.

The first one, preventive, is realized by the destination with flows requiring limited QoS even before detecting the existence of congestion is detected. Indeed, if a destination receives a number of flows having delays superior to the MAPD delay (Maximum Acceptable packet Delay), then it is detected that the resources of the network become limited and, consequently, a message of regulation is sent at the source of the traffic. This source will realize a procedure of reset to localize, if possible, another path having a better quality.

The second behaviour, of recovery, intervenes when an intermediate node suffers from congestion by regulating flows requiring limited QoS before flows demanding important QoS is attacked.

Simulation Results

Once the proposed regulation's concepts have been completely defined, it becomes necessary to check their feasibility and evaluate their benefits. Simulation analysis is used here to evaluate the proposed scheme. The goal of this section is to present the main aspects of the simulation model and its results.

Simulations Environment

Two categories of parameters have been considered in our study: the input and the output parameters.

These parameters are as follows:

Input parameters: These include:

- Two types of applications: CBR & VBR_RT.
- Three kinds of regulation: Source based regulation, Network based regulation, and, Destination based regulation.
- The number of flows varies between 10 and 600.

Output parameters: These include:

- Goodput,
- Throughput,
- Delay, and,
- Fairness.

Simulations Scenarios

For the applications distribution, CBR and VBR_RT applications are employed. We chose to distribute fairly the applications between those requiring a limited QoS, those demanding an average QoS and finally, those requiring high QoS.

For the regulation, we use:

- Source based regulation,
- Network based regulation, and,
- Destination based regulation.

The number of flows varies between 10 and 600.

The mobility model used is the Random WayPoint (RWP) model.

Simulations Results

In this sub-section, we try to evaluate the performances provided by our regulation mechanism. These performances are expressed in terms of goodput, throughput, delay and fairness.

3) Comparison in Term of Goodput

In computer networks, goodput is the application level throughput, i.e. the number of useful bits per unit of time forwarded by the network from a certain source address to a certain destination, excluding protocol overhead, and excluding retransmitted data packets. For example, if a file is transferred, the goodput that the user experiences corresponds to the file size in bits divided by the file transfer time. The goodput is always lower than the throughput (the gross bit rate that is transferred physically), which generally is lower than network access connection speed (the channel capacity or bandwidth).

Examples of factors that cause goodput lower than

throughput are:

- Protocol overhead; typically, transport layer, network layer and sometimes data link layer protocol overhead is included in the throughput, but excluded from the goodput.
- Transport layer flow control and congestion avoidance, for example, TCP slow start, may cause a goodput lower than the maximum throughput.
- Retransmission of lost or corrupt packets due to transport layer automatic repeat request (ARQ), caused by bit errors or packet dropping in congested switches and routers, is included in the data link layer or network layer throughput but not in the goodput

In a general manner, it is noticed that the goodput offered by our proposed destination based regulation is always higher than that offered by the source based regulation and the network based regulation, as shown in the Fig. 3, whatever the number of flows transmitted in the network is.

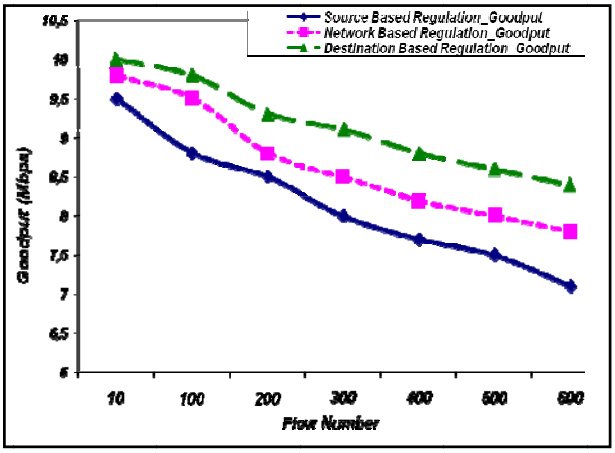


FIG. 3 GOODPUT EVOLUTION FOR SOURCE, NETWORK AND DESTINATION BASED REGULATIONS

We note that:

- The goodput decreases with the increase of the number of flows transited in the network.
- On average, for a number of flows equal to 10, representing the minimum number of flows in the network, the goodput is for its maximum and is almost equal to 10 Mbps.
- For an intermediate value of the number of flows (between 200 and 300), the goodput is equal to 9 Mbps.
- Finally, for a maximal value of the number of flows equal to 600, the goodput is for its minimum and is equal to 8 Mbps. This can be explained by the fact that the more significantly the number of Real Time flows transited in the network increases, the more difficultly all the requirements of the applications is satisfied.

It is also noticed that the offered goodput is always higher with our destination based regulation proposition whatever the number of flows in the network is. The use of this kind of regulation offers the applications the exact quantities of the required resources, so the network can accept a more important number of applications because there is no wasting of resources.

4) Comparison in Term of Throughput

This parameter allows us an evaluation of the efficiency of the network.

In a general manner, the throughput offered by the destination based regulation is always higher than that by the source based regulation and the network based regulation, as shown in Fig. 4, whatever the number of flows transited in the network is.

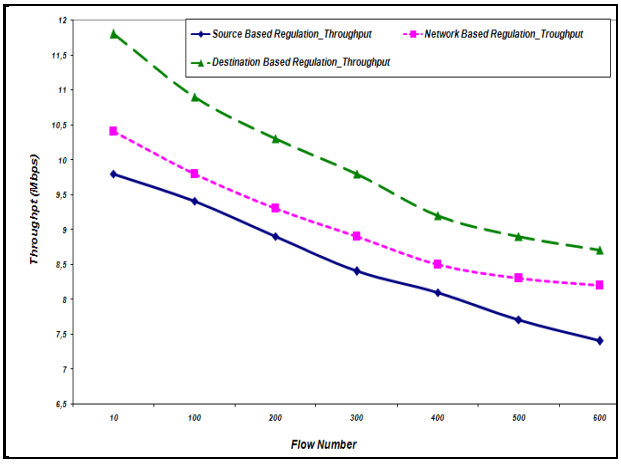


FIG. 4 THROUGHPUT EVOLUTION FOR SOURCE, NETWORK AND DESTINATION BASED REGULATIONS

Indeed, the fact that this proposition offers a higher throughput allows decreasing the rate of losses of the Real Time flows, which represents the first advantage for our proposition compared with the two other types of regulation.

It is worth noting that:

- The throughput decreases with the increase of the number of flows in the network, which represents the second advantage of our proposition. This is due to the fact that the more the number of flows transiting in

the network increases, the more the risk of having concentrations of buffers in nodes becomes importing, which implies more risk of regulation.

- For a number of flows equal to 10, representing the minimum number of Real Time flow transited in the network, the offered throughput is for its maximum and is equal, on average, to 11 Mbps.
- For an intermediate value of the number of flows (between 200 and 300), the throughput is equal, on average, to 9.5 Mbps.
- Finally, for a number of flows equal to 600, the offered throughput is equal, on average, to 8 Mbps.

5) Comparison in Term of Delay

The delay, expressed in second, represents the necessary time to transmit a packet from a source to a destination.

In a general manner, the delay offered by the destination based regulation is always lower than that produced by the source based regulation and the network based regulation, as shown in Fig. 5, whatever the number of flows in the network is.

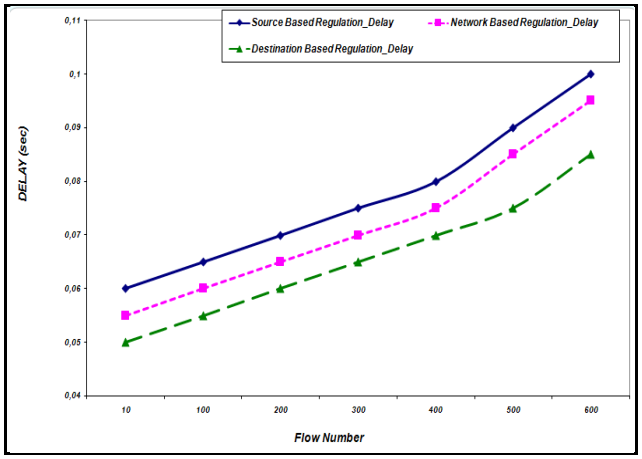


FIG. 5 DELAY EVOLUTION FOR SOURCE, NETWORK AND DESTINATION BASED REGULATIONS

This figure shows that:

- The delay increases with the increase of the number of flows.
- On average, for a minimum number of flows (equal to 10), the delay is for its minimum and is equal to 0.05 seconds. This is due to the fact that for this number of flows, the network will not have to regulate, almost, the Real Time flows.
- The delay is equal to 0.07 seconds for an average number of flows considering that more resources are reserved with this method.
- A maximal value of delay is obtained when the number of flows becomes its maximum. This is due to that for a significant number of flows transited in the network; this one finds more difficulties to satisfy the requirements of all the Real Time applications, which causes more degradation of the network conditions and afterward more regulation. In this sense, the applications will observe a delay more important than that observed for a small or average number of flows.

6) Comparison in Term of Fairness

For every throughput set of the users (x1, x2…xn), the fairness index is defined as follows:

$$\text{Fairness Index } f(x1, x2\ldots xn) = \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n\sum_{i=1}^{n} x_i^2}$$

With 0<= f <=1

In a general manner, the index of equity produced by our destination based regulation proposition is always superior to that produced by the source based regulation and the network based regulation, as shown in Fig. 6, whatever the number of flows transited in the network in.

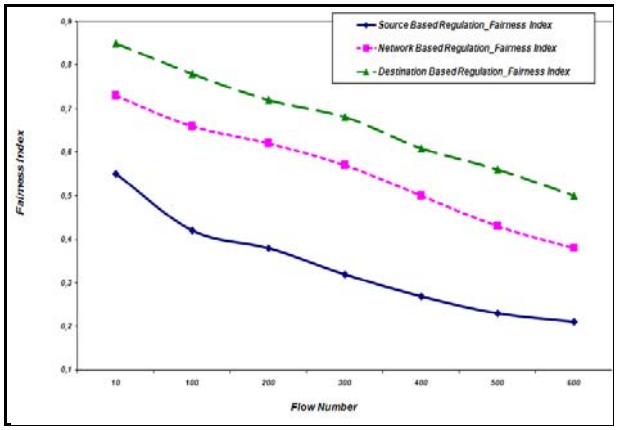


FIG. 6 FAIRNESS INDEX EVOLUTION FOR SOURCE, NETWORK AND DESTINATION BASED REGULATIONS

This figure shows that:

- - The fairness index decreases with the increase of the number of flows transited in the network.
- - For a number of flows equal to 10, representing the minimum number of Real

Time flow transiting in the network, the produced fairness index is for its maximum and is equal, on average, to 0.9. This is due to the fact that for a small number of flows, the network can assign fairly the quantities of resources required by the applications.

- - For an intermediate value of the number of flows (between 200 and 300), the fairness index is equal, on average, to 0.65 seeing that more resource are reserved with this number of flows.
- - Finally, for a number of flows equal to 600, the produced fairness index value is equal, on average, to 0.4 seeing that the network will have more difficulties to satisfy the requests of all the applications with this important number of transiting flow. In this sense, the network will proceed to more regulation than that with a limited or average number of flows transiting in the network.

It is noticed that all the regulation solutions are unfairness. However, the fairness index is always higher with our destination based regulation proposition, whatever the number of Real Time flows transited in the network is. This is can be justified by the fact that within the destination based regulation, the network offers its users exactly the required resources quantities and so we have no wasting of resources.

Conclusion

In this paper, a novel destination based regulation scheme was proposed to react in front of a degradation of the network conditions, allowing the applications to adapt their requirements in terms of QoS not to risk their rejections in the case of congestion.

This new scheme of destination based regulation presents an innovative approach for the choice of the Real Time victim's flows which must be regulated. In fact, with this proposition, it is the destination which takes care to detect and to start a regulation because it always has a better judgment (that in the source) on the quality of flows receipt and as a result, a preventive behaviour can be implemented, to limit the wasting of the resources, as soon as the destination notices a small degradation of the QoS. The second contribution of this proposition is in the choice of flows to be regulated. Indeed, flows requiring a limited QoS will be the first selected to give the possibility to the Real Time flows requiring an important QoS to finish their transmissions.

Simulation analyses have been conducted and produced some performance evaluation results showing that an AHN implementing this destination based regulation mechanism provides excellent performance in terms of goodput, throughput, delay and fairness index.

REFERENCES

B. Reddy, I. Karthigeyan, B.S. Manoj and C. Siva Ram Murthy. "Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions", Ad Hoc Networks 4, pp 83–124, 2006.

G. Pujolle. "Les Réseaux", Troisième Edition, Eyrolles, 2002.

H. Bouhouch and S. Guemara El Fatmi. "QoS Routing In Ad Hoc Networks by Integreting Activity in the OLSR Protocol", ICSNC, pp.1, Second International Conference on Systems and Networks Communications (ICSNC 2007), 2007.

M. A. Remiche. "ELEC 379 : Réseaux II", Belgium, 2005.

M. Cherif, S. Guemara el Fatmi, N. Boudriga and M.S. Obaidat. "Burst Variability Management and Modeling in OBS Networks", SPECTS, 2007.

M. Mirhakkak, N. Schult and D. Thomson. "Dynamic Quality of Service for Mobile Ad Hoc Networks", In IEEE MobiHoc 2000, Boston, Massachusets, USA, August 2000.

M. S. Gast and O'Reilly. "802.11 Wireless networks: The definitive guide", 2002.

Ramakrishnan, Floyd and Black. "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, Proposed Standard, September 2001.

T. Imienlinski and B. R. Badrinath. "Mobile wireless computing: solutions and challenges in data management", CACM, October 1994.