

Một Số Cải Tiến Kỹ Thuật Trong Chuyển Đổi Mã Hóa Tiếng Nói Băng Rộng Và Băng Hẹp Áp Dụng Trên Mạng Viễn Thông Di Động Viettel

Đinh Văn Phong¹, Nguyễn Thế Hiếu¹, Nguyễn Huy Tinh¹, Đinh Viết Quân¹ và Trần Đức Tân²

¹ Trung Tâm Nghiên Cứu Công Nghệ Mạng Viettel

² Đại Học Công Nghệ, Đại Học Quốc Gia Hà Nội

Email: phongdv6@viettel.com.vn

Tóm tắt — Công nghệ di động sau vài thập niên phát triển đến nay (2018) đã trải qua các thế hệ 2G, 3G, 4G và sắp tới sẽ là 5G. Mỗi thế hệ bao gồm bên trong nó một loạt các công nghệ mới được cải tiến và áp dụng. Một trong những công nghệ đó là công nghệ mã hóa tiếng nói (*speech coding*) cũng được cải tiến dần qua mỗi thế hệ. Trong các thế hệ 2G, 3G, với mục đích tiết kiệm băng thông vô tuyến, việc mã hóa tiếng nói được thực hiện trên cơ sở tín hiệu tiếng nói được lấy mẫu 8kHz, còn được gọi là lấy mẫu băng hẹp. Tuy nhiên, trong các thế hệ 4G, 5G, khi băng thông không còn là bài toán khó khăn, việc mã hóa tiếng nói được thực hiện trên cơ sở tín hiệu tiếng nói được lấy mẫu ở các tần số cao hơn như: 16 kHz, 24 kHz, 32 kHz... còn được gọi là lấy mẫu băng rộng. Cách làm này giúp nâng cao chất lượng thoại, do phổ của tín hiệu tiếng nói được mở rộng hơn, tuy nhiên cũng đặt ra thách thức trong việc xử lý chuyển đổi mã hóa tiếng nói (*transcoding*) giữa các tín hiệu được lấy mẫu băng rộng và băng hẹp. Trong bài báo này, chúng tôi đề xuất một số cải tiến kỹ thuật trong việc xử lý chuyển đổi mã hóa tiếng nói giữa băng rộng và băng hẹp, đã triển khai áp dụng thành công trên mạng viễn thông di động Viettel.

Từ khóa- Mã hóa tiếng nói, băng rộng, băng hẹp, bộ lọc nửa dải, đa tốc, tăng tốc, giảm tốc, nhân chập.

I. GIỚI THIỆU

Công nghệ di động sau vài thập niên phát triển đến nay (2018) đã trải qua các thế hệ 2G, 3G, 4G và sắp tới sẽ là 5G. Mỗi thế hệ bao gồm bên trong nó một loạt các công nghệ mới được cải tiến và áp dụng. Một trong những công nghệ đó là công nghệ mã hóa tiếng nói (*speech coding*) cũng được cải tiến dần qua mỗi thế hệ. Trước thế hệ 2G, chuẩn mã hóa tiếng nói G.711 (PCMA/PCMU) [1] được sử dụng trong mạng PSTN. Trong thế hệ 2G, các chuẩn GSM-FR [2], GSM-HR [3], GSM-EFR [4], GSM-AMR [5] lần lượt được công bố và sử dụng. Trong thế hệ 3G, GSM-AMR được nâng cấp thành các chuẩn UMTS AMR, UMTS AMR2 [5]. Tất cả các chuẩn này đều căn cứ trên dải tần tiếng nói cơ bản từ 300 – 3.400 Hz, và lấy mẫu ở tốc độ 8 kHz đáp ứng yêu cầu cơ bản theo Nyquist/Shannon [6]. Trong thế hệ 4G (LTE) và sắp tới là 5G, các chuẩn mã hóa mới AMR-WB [7], AMR-WB+ [8], EVS [9] sử dụng các tần số lấy mẫu cao hơn, nhằm biểu diễn chi tiết hơn các âm sắc tiếng nói ở ngoài dải tần cơ bản, giúp nâng cao chất lượng thoại. AMR-WB sử dụng tần số lấy mẫu 16 kHz, trong khi AMR-WB+ và EVS có thể sử dụng một trong các tần số lấy

mẫu 16/24/32/48kHz. Các bộ mã hóa tiếng nói này được gọi là các bộ mã hóa băng rộng.

Trong mạng viễn thông di động, một thiết bị di động A khi thực hiện cuộc gọi tới thiết bị di động B có thể sử dụng bất kỳ chuẩn mã hóa tiếng nói nào mà nó hỗ trợ. Tương tự, thiết bị di động B khi nhận cuộc gọi cũng có thể sử dụng bất kỳ chuẩn mã hóa tiếng nói nào mà nó hỗ trợ. Khi chuẩn mã hóa tiếng nói trên thiết bị A và thiết bị B là khác nhau, sẽ cần một bộ chuyển đổi (*transcoder*) [5] giữa chúng đặt tại hệ thống mạng lõi của nhà mạng để giúp thiết bị A giải mã được dữ liệu của thiết bị B và ngược lại. Ngoài ra, khi chuẩn mã hóa là khác nhau giữa băng rộng và băng hẹp, bộ chuyển đổi này cần thực hiện thêm kỹ thuật xử lý đa tốc (*multirate*) [10] để loại bỏ hoàn toàn các phổ tần số không mong muốn xuất hiện trong tín hiệu.

Trong bài báo này, chúng tôi trình bày chi tiết phương pháp chuyển đổi mã hóa tiếng nói (*transcoding*) giữa tín hiệu băng rộng và băng hẹp đồng thời đề xuất một số cải tiến kỹ thuật trong quá trình thực hiện, bao gồm:

- Thiết kế bộ lọc nửa dải phù hợp cho môi trường đầu phẳng đồng hoặc đầu phẳng tĩnh giúp cải thiện hiệu năng tính toán.
- Cải tiến phép nhân chập (*convolution*) khi thực hiện với các gói dữ liệu rời rạc trong môi trường IP.
- Xây dựng qui trình cân bằng năng lượng tín hiệu sau xử lý chuyển đổi mã hóa tiếng nói.

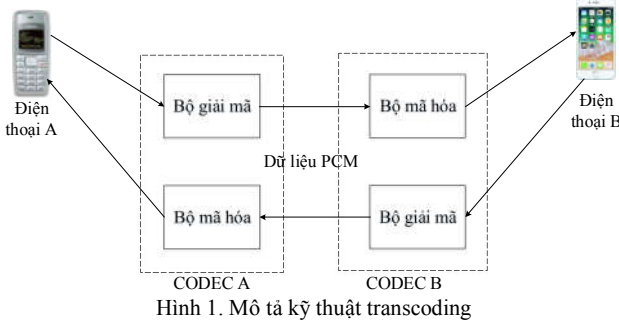
Nội dung tiếp theo của bài báo được trình bày thành 04 phần: trong phần II, chúng tôi trình bày kỹ thuật *transcoding* có xử lý đa tốc. Trong phần III, chúng tôi trình bày chi tiết các vấn đề kỹ thuật gặp phải và đề xuất các cải tiến kỹ thuật của chúng tôi. Phần IV cung cấp các kết quả thực nghiệm và đánh giá chất lượng trên mạng di động Viettel. Cuối cùng, chúng tôi kết luận và định hướng mục tiêu nghiên cứu mới trong phần V.

II. CÁC KỸ THUẬT NỀN TẢNG

1. Kỹ thuật transcoding

Một bộ mã hóa tiếng nói bao gồm 02 thành phần chính: bộ mã hóa và bộ giải mã. Kỹ thuật transcoding luôn được thực hiện bằng việc giải mã với chuẩn mã hóa hiện tại của bản thân dữ liệu và sau đó mã hóa lại theo một chuẩn mã hóa mới. Hình

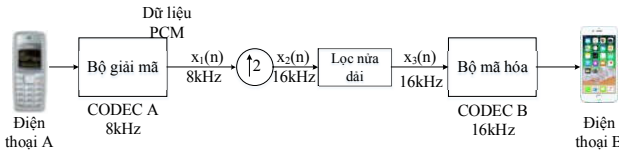
01 minh họa việc thực hiện transcoding giữa 02 thiết bị đầu cuối A và B. Thiết bị A sử dụng chuẩn mã hóa loại A, được giải mã bởi bộ giải mã loại A, dữ liệu sau giải mã là dữ liệu dạng PCM 16 bit, dữ liệu sau đó được mã hóa lại theo chuẩn mã hóa của điện thoại B, sử dụng bộ mã hóa loại B. Chiều ngược lại được thực hiện tương tự. Khi đó thiết bị A và B có thể thực hiện đàm thoại bình thường mặc dù khác nhau chuẩn mã hóa.



Hình 1. Mô tả kỹ thuật transcoding

2. Kỹ thuật chuyển đổi tần số lấy mẫu giữa băng rộng và băng hẹp

Tiếp tục với ví dụ bên trên, chúng ta giả thiết rằng codec A là codec băng hẹp được lấy mẫu 8kHz và codec B là codec băng rộng được lấy mẫu 16kHz. Lúc này, việc transcoding cần bổ sung thêm kỹ thuật xử lý đa tốc (*multirate*) [10]. Xét chiều dữ liệu từ điện thoại A tới điện thoại B, tín hiệu tiếng nói lấy mẫu 8kHz được chuyển đổi thành tín hiệu lấy mẫu 16kHz thông qua một bộ tăng tốc (*upsample*) cơ số 2. Mô hình thực hiện như trong hình 2.



Hình 2. Mô hình transcode kết hợp tăng tốc dữ liệu

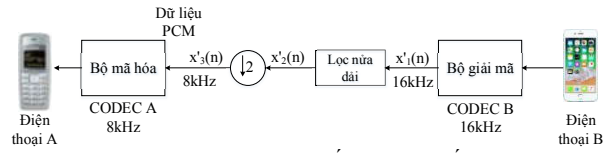
$$x_2(n) = \begin{cases} x_1\left(\frac{n}{2}\right), & n = 0, 2, 4, 6, \dots \\ 0 & n = 1, 3, 5, 7, \dots \end{cases} \quad (1)$$

Sau bộ tăng tốc, dữ liệu bắt buộc đi qua một bộ lọc nửa dải để loại bỏ phổ ảnh không mong muốn, đồng thời nó cũng hoạt động như một bộ nội suy những giá trị 0 vừa được chèn thêm:

$$x_3(n) = \sum_{k=0}^{L-1} h(k)x_2(n-k) \quad (2)$$

Trong đó $h(k)$ là bộ lọc nửa dải (half-band filter) độ dài L.

Ở chiều ngược lại từ điện thoại B tới điện thoại A, tín hiệu tiếng nói lấy mẫu 16kHz được chuyển đổi thành tín hiệu lấy mẫu 8kHz thông qua một bộ giảm tốc (*downsample*) hệ số 2. Mô hình thực hiện như trong hình 3.



Hình 3. Mô hình transcode kết hợp giảm tốc dữ liệu

Theo lý thuyết xử lý đa tốc [10], trước khi thực hiện giảm tốc cần sử dụng bộ lọc thông thấp để chống hiện tượng chồng phổ có thể xảy ra.

$$x'_2(n) = \sum_{k=0}^{L-1} h(k)x'_1(n-k) \quad (3)$$

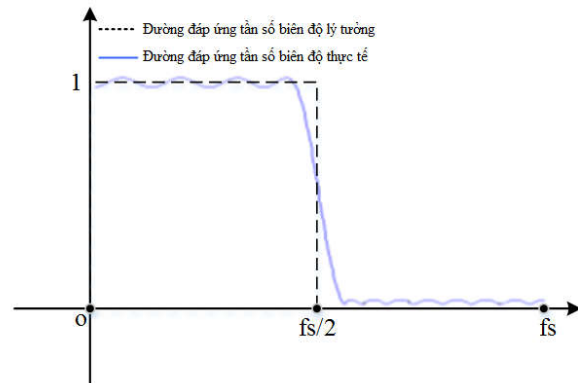
Trong đó $h(k)$ là cũng bộ lọc nửa dải độ dài L.

$$x'_3(n) = x'_2(2n) \quad (4)$$

Như vậy, trong cả mô hình của việc tăng tốc và giảm tốc đều cần sử dụng một bộ lọc nửa dải $h(k)$. Hình 4 mô tả đáp ứng tần số của bộ lọc nửa dải, trong đó đường nét liền mô tả đáp ứng tần số biên độ lý tưởng, đường nét đứt mô tả đáp ứng tần số biên độ thực tế có thể thiết kế. Việc thiết kế và tính toán các hệ số bộ lọc nửa dải có thể được thực hiện bằng một trong các phương pháp như: phương pháp của số, phương pháp Parks-McClellan... hoặc các công cụ như *Fdatool* của Matlab. Kết quả sau thiết kế, bộ lọc thường có khuôn dạng của một mảng 1 chiều với các hệ số dạng dấu phẩy động.

Bảng 1. Ví dụ bộ lọc nửa dải chiều dài $L = 21$ được thiết kế

```
float H1[21] = {0, 0.0340, 0, -0.0333, 0, 0.0545, 0, -0.1004, 0, 0.3164, 0.5000, 0.3164, 0, -0.1004, 0, 0.0545, 0, -0.0333, 0, 0.0340, 0};
```



Hình 4. Đáp ứng tần số biên độ của bộ lọc nửa dải

III. CÁC KỸ THUẬT ĐƯỢC CẢI TIẾN CHO MẠNG VIỄN THÔNG VIETTEL

1. Thực thi hệ số bộ lọc theo dấu phẩy tĩnh

Bộ lọc nửa dải và kỹ thuật transcoding có thể được triển khai trên các môi trường sau:

- Môi trường dấu phẩy động: Unix, Linux sử dụng các dòng CPU Intel

- Môi trường dấu phẩy tĩnh: Các dòng chip DSP của Texas Instrument, FreeScale...

Tuy nhiên, các thuật toán được triển khai dưới dạng dấu phẩy tĩnh đã được chứng minh chạy nhanh hơn khi triển khai dưới dạng dấu phẩy động [11]. Trong các thử nghiệm của chúng tôi, chúng tôi xây dựng bộ lọc trên môi trường Linux, Intel CPU, thuật toán với đầu vào là khung dữ liệu PCM 16bit, chiều dài 20ms, được lấy mẫu 8kHz, bộ lọc có chiều dài $L = 21$. Chúng tôi chạy thử thuật toán trong 02 trường hợp: hệ số bộ lọc dạng dấu phẩy tĩnh và hệ số bộ lọc dạng dấu phẩy động sau đó đo số xung nhịp CPU MCPS (Mega Cycles Per Second) cần để chạy hết thuật toán lọc. Kết quả được thể hiện như trong bảng 2.

Bảng 2. Kết quả so sánh bộ lọc dấu phẩy động và dấu phẩy tĩnh

Lần chạy	MCPS (dấu phẩy động)	MCPS (dấu phẩy tĩnh)
1	32.586	21.635
2	32.784	21.475

Có thể nhận thấy rằng, khi hệ số bộ lọc ở dạng dấu phẩy tĩnh, thuật toán đạt hiệu năng cao hơn 33% so với khi sử dụng dấu phẩy động. Do đó, trong các trường hợp cần thực thi hiệu năng tính toán, việc chuyển đổi hệ số bộ lọc từ dấu phẩy động sang dạng dấu phẩy tĩnh là cần thiết.

Việc chuyển đổi hệ số bộ lọc từ dạng dấu phẩy động sang dạng dấu phẩy tĩnh được thực hiện bằng việc dịch phải dấu “.” trong hệ số của bộ lọc, tương đương việc nhân hệ số bộ lọc với 10^N . Trong đó N là số bước dịch phải.

$$h'(n) = h(n)10^N \quad (5)$$

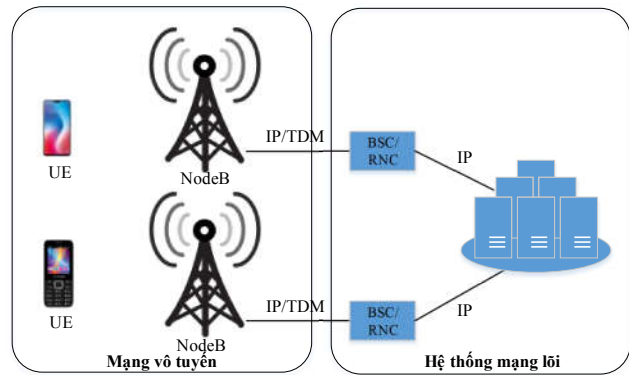
Tuy nhiên, cách làm này có thể gây tràn số tại $x_3(n)$ khi thực hiện phép nhân chập được mô tả trong (1). Để loại bỏ điều này phép nhân chập (1) có thể được thay đổi như sau:

$$x_3(n) = \frac{1}{10^N} \sum_{k=0}^{L-1} h(k)x_2(n-k) \quad (6)$$

2. Thử nghiệm phép nhân chập cho xử lý các gói dữ liệu miền IP

Các công thức nhân chập (2)(3)(6) mô tả $x(n)$ dưới dạng một tín hiệu liên tục kéo dài vô hạn trên miền thời gian. Đối với các hệ thống viễn thông di động, giả thiết này là không thực tế. Một hệ thống viễn thông di động rút gọn được mô tả như trong Hình 5. Tại giao diện mạng vô tuyến, tín hiệu tiếng nói là liên tục trên miền thời gian, tuy nhiên khi được truyền dẫn và xử lý trong hệ thống mạng lõi, tín hiệu được đóng gói thành các gói tin theo chuẩn TCP/IP.

Kỹ thuật transcoding thường được thực hiện tại hệ thống MSC (Mobile Switching Center) hoặc GMSC (Gate Mobile Switching Center) thuộc hệ thống mạng lõi. Tại các hệ thống này, tín hiệu tiếng nói $x(n)$ được chia thành các gói dữ liệu IP tương ứng với từng loại codec.



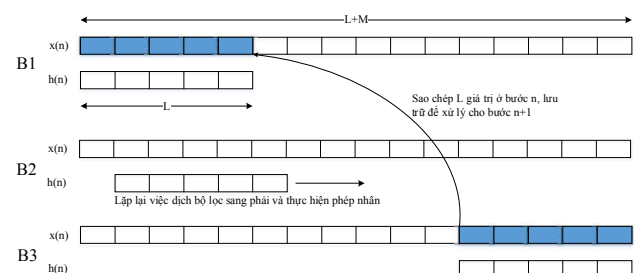
Hình 5. Mô hình giản lược hệ thống viễn thông di động

Bảng 3. Một số codec trong mạng lõi viễn thông

Codec	Tần số lấy mẫu (kHz)	Chiều dài gói (ms)	Kích thước gói (byte)
G711(PCMA/PCMU)	8	10	80
GSM-FR	8	20	33
GSM-HR	8	20	14
GSM-EFR	8	20	31
AMR	8	20	12 - 31
AMR-WB	16	20	17 - 60
AMR-WB+	8/16/24/32	20	15 - 120
EVS	8/16/24/32	20	15 - 320

Việc thực hiện nhân chập khi dữ liệu là các gói rời rạc không thể áp dụng trực tiếp các công thức (2)(3)(6) mà cần cải tiến kỹ thuật xử lý phù hợp. Chúng tôi đề xuất cải tiến phép nhân chập cho các gói rời rạc với 03 bước sau:

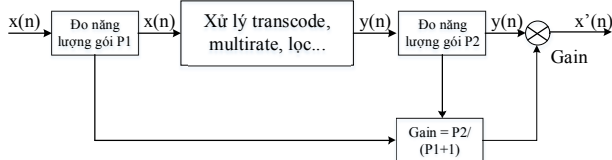
- Bước 1: Thực hiện lưu trữ giá trị bộ lọc chiều dài L và chuỗi giá trị tín hiệu $x(n)$ chiều dài M trong 02 buffers riêng biệt có kích thước L và $L + M$ tương ứng. Đối với buffer chứa $x(n)$, L giá trị đầu tiên được khởi tạo bằng 0.
- Bước 2: Thực hiện phép nhân vector các giá trị trong 02 buffers, dịch buffer bộ lọc sang phải và lặp lại bước 2 M lần đến khi buffer chứa $x(n)$ chỉ còn L giá trị.
- Bước 3: Không thực hiện việc nhân chập hết chiều dài tín hiệu $x(n)$ mà giữ lại L giá trị cuối cùng, L giá trị này được sao chép vào L giá trị đầu tiên của buffer chứa $x(n)$ chuẩn bị cho việc xử lý gói dữ liệu tiếp theo.



Hình 6. Quy trình thực hiện nhân chập cho các gói rời rạc

3. Quy trình cân bằng năng lượng tín hiệu sau chuyển đổi mã hóa tiếng nói

Việc chuyển đổi hệ số bộ lọc từ dấu phẩy động sang dấu phẩy tĩnh, và áp dụng phép nhân chấp lên tín hiệu trong quá trình xử lý có thể làm thay đổi biên độ và năng lượng giữa tín hiệu trước và sau khi được xử lý. Sự thay đổi này có thể tác động trực tiếp đến cảm nhận của người nghe như: âm thanh trở nên to hoặc nhỏ hơn. Để xử lý vấn đề này, chúng tôi thực hiện đo năng lượng các gói dữ liệu trước và sau khi xử lý, từ đó tính ra hệ số khuếch đại cần thiết để điều chỉnh mức năng lượng tín hiệu đầu ra. Hình 7 mô tả cách thức thực hiện của qui trình này.

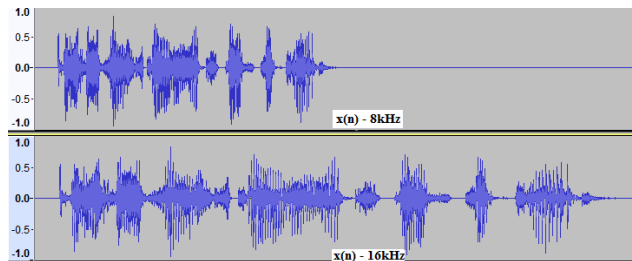


Hình 7. Quy trình xử lý cân bằng năng lượng tín hiệu

Tuy nhiên, nếu mọi gói dữ liệu đến đều trải qua qui trình đo năng lượng có thể làm giảm hiệu năng của hệ thống, vì vậy việc đo năng lượng các gói dữ liệu được thực hiện cứ mỗi N giây. Hiện chưa có căn cứ logic về mặt học thuật để xác định giá trị của N, tuy nhiên giá trị này bị ràng buộc trong việc đạt hiệu năng hệ thống và sự cân bằng trong năng lượng tín hiệu, và được xác định dựa trên kết quả thực nghiệm. Trong các bài test thực nghiệm của chúng tôi, việc đặt $N = 3$ là đạt yêu cầu về hiệu năng hệ thống đặt ra.

IV. KẾT QUẢ

Trong hình 8 mô tả dạng tín hiệu thu được trên mạng lưới Viettel sau quá trình transcoding giữa các codec UMTS AMR2 (8kHz) và UMTS AMR-WB (16kHz).



Hình 8. Tiếng nói trước 8kHz và sau chuyển đổi sang 16kHz (so sánh số lượng mẫu của 02 tín hiệu trước và sau chuyển đổi)

Từ tín hiệu thu được trên mạng lưới có thể dễ dàng nhận thấy số lượng các mẫu dữ liệu đối với dữ liệu lấy mẫu 16kHz nhiều gấp 02 lần so với trước khi được chuyển đổi (lấy mẫu 8kHz). Các tín hiệu đều cho cảm nhận nghe tốt từ người nghe trên mạng lưới Viettel.

Để định lượng chính xác chất lượng tín hiệu tiếng nói, chúng tôi sử dụng phần mềm đo kiểm chất lượng thoại GL VQT (Voice Quality Testing) [12] theo tiêu chuẩn ITU P.862/P863 [13]. Phần mềm sẽ xếp loại chất lượng tiếng nói theo điểm MOS (mean opinion square) với các mức từ cao

xuống thấp: Rất tốt, Tốt, Khá, Xấu. Chất lượng tiếng nói được gọi là “Đạt” nếu có điểm đánh giá từ mức “Khá” trở lên. Chúng tôi bắt thử một số mẫu tiếng nói trên mạng lưới cho việc thử nghiệm, điểm MOS được đo tại thời điểm trước và sau khi thực hiện chuyển đổi mã hóa tiếng nói từ 8kHz sang 16kHz, kết quả được cho như trong bảng 4.

Bảng 4. Kết quả so sánh MOS trước và sau khi chuyển đổi tần số lấy mẫu

Mẫu thử	MOS (trước chuyển tốc độ 8kHz)	MOS (sau chuyển tốc độ sang 16kHz)
test1_8kHz_20ms.pcma	Rất tốt	Rất tốt
test2_8kHz_40ms.pcma	Rất tốt	Rất tốt
test3_8kHz_60ms.pcmu	Rất tốt	Rất tốt
test4_8kHz_80ms.pcmu	Rất tốt	Rất tốt
test5_8kHz_100ms.amr	Tốt	Tốt
test6_8kHz_160ms.amr	Tốt	Tốt

Từ điểm MOS đo được có thể thấy rằng, chất lượng tiếng nói được bảo toàn và không bị thay đổi trong quá trình thực hiện các kỹ thuật xử lý chuyển đổi tiếng nói từ băng hẹp (8kHz) sang băng rộng (16kHz) và ngược lại.

V. KẾT LUẬN

Trong bài báo này, chúng tôi đã đề xuất một số cải tiến trong việc thực hiện kỹ thuật transcoding giữa các tín hiệu băng rộng và băng hẹp, đã áp dụng thành công trên mạng viễn thông Viettel cho phép hệ thống thực hiện chuyển đổi các chuẩn mã hóa tiếng nói giữa mạng 2G, 3G, 4G và tiếp theo là 5G. Việc thực thi trong thực tiễn cho kết quả tốt và phù hợp với những những kết quả trong bài báo này. Tuy nhiên, các kết quả mới được thực hiện giữa 02 tần số lấy mẫu 8kHz và 16kHz. Xa hơn, chúng tôi có kế hoạch tiếp tục nghiên cứu cho việc chuyển đổi giữa nhiều tần số lấy mẫu khác như 24kHz, 32kHz, 48kHz... Những cải tiến kỹ thuật này có thể kết hợp với những kỹ thuật mã hóa thông tin cho những ứng dụng đặc thù [14][15].

Lời cảm ơn: Kết quả của bài báo là một phần trong đề tài nghiên cứu của Tập đoàn Viettel “Nghiên cứu phát triển hệ thống Gate MSC hỗ trợ TDM” mã số 002-18-TĐ-RĐP-DS. Chúng tôi xin gửi lời cảm ơn tới Trung tâm nghiên cứu công nghệ mạng Viettel (VTTEK) và Tổng Công ty mạng lưới Viettel (VTNET) đã hỗ trợ thực hiện và thử nghiệm các kết quả đề tài.

TÀI LIỆU THAM KHẢO

- [1] ITU G.711 : Pulse code modulation (PCM) of voice frequencies; ITU-T Recommendation (11/1988), Retrieved on 2009-07-08.
- [2] ETSI EN 300 961 V8.1.1 (2000-11) - (GSM 06.10 version 8.1.1 Release 1999), Retrieved on 2009-07-08.
- [3] ETSI, EN 300 969 - Half rate speech transcoding (GSM 06.20 version 8.0.1 Release 1999), Retrieved on 2009-07-11.
- [4] RFC 3551 - GSM-EFR (GSM 06.60), Retrieved on 2009-07-08.
- [5] 3GPP TS 26.090 - Mandatory Speech Codec speech processing functions; Adaptive Multi-Rate (AMR) speech codec; Transcoding functions". 3GPP. Retrieved 2010-07-21.

- [6] Marks, R.J.(II): Introduction to Shannon Sampling and Interpolation Theory, Springer-Verlag, 1991.
- [7] ITU-T (2003) ITU-T Recommendation G.722.2 Page i. Retrieved on 2009-06-17.
- [8] 3GPP TS 26.290; Audio codec processing functions; Extended Adaptive Multi-Rate - Wideband (AMR-WB+) codec; Transcoding functions, Page 10, Retrieved on 2009-07-08.
- [9] Atti, V.; Sinder, D. J.; Subasingha, S.; Rajendran, V.; Dewasurendra, D.; Chebiyyam, V.; Varga, I.; Krishnan, V.; Schubert, B. (2015-04-01). "Improved error resilience for volte and VoIP with 3GPP EVS channel aware coding". 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 5713–5717. doi:10.1109/ICASSP.2015.7179066.
- [10] Ljiljana milic, "Multirate filtering for digital signal processing: matlab applications", 2009.
- [11] Steven W.Smith, "The Scientist and Engineer's Guide to Digital Signal Processing".
- [12] Voice Quality Testing (VQT) Software (POLQA, PESQ), <https://www.gl.com/voicequality-testing-pesq-polqa.html>, access: June 26, 2018.
- [13] ITU P.863 "Perceptual objective listening quality prediction", March 2018.
- [14] Tam Vu Van, Tran Duc-Tan, Phan Trong Hanh (2017). Data embedding in audio signal using multiple bit marking layers method. Multimedia Tools and Applications, 76(9), 11391-11406.
- [15] Vu, V. T., Tran, D. T., Nguyen, D. T., Nguyen, T. T., & Phan, T. H. (2015). Data embedding in audio signal by a novel bit marking method. International Journal of Advancements in Computing Technology, 7(1).