

Thuật toán nén dữ liệu tiếng nói trực tuyến

Nguyễn Văn Xuất, Mai Văn Phú*

Khoa Công nghệ Thông tin, Học viện Kỹ thuật Quân sự, 100 Hoàng Quốc Việt, Hà Nội, Việt Nam

Nhận ngày 19 tháng 2 năm 2009

Tóm tắt. Bài báo đề xuất thuật toán nén dữ liệu tiếng nói trực tuyến. Nội dung của thuật toán là xấp xỉ tuyến tính dữ liệu tiếng nói. Trong bài báo này cũng đề xuất thuật toán giải nén với độ phức tạp tuyến tính. Kết quả thử nghiệm cho thấy thuật toán có thể áp dụng trong các ứng dụng nén và giải nén tiếng nói trực tuyến.

Từ khóa: Nén dữ liệu, tiếng nói, dữ liệu tiếng nói, trực tuyến, dữ liệu tiếng nói trực tuyến.

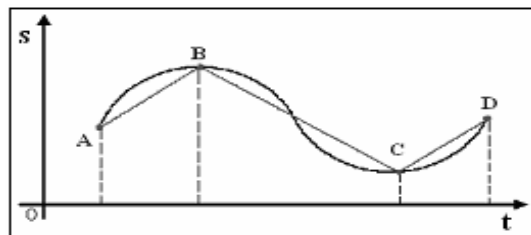
1. Mở đầu

Nén dữ liệu âm thanh nói chung và dữ liệu tiếng nói nói riêng đã và đang được các nhà khoa học, công nghệ trên thế giới quan tâm nghiên cứu, các kết quả đạt được đã được ứng dụng nhiều trong lĩnh vực truyền thông và giải trí. Trong lĩnh vực nén âm thanh, người ta đã từng biết đến các chuẩn nén khá nổi tiếng như dòng họ nén MPEG, MP3, JPEG... [1] Đặc điểm chung của các thuật toán nén trên là: hiệu suất nén cao, chất lượng âm thanh sau khi giải nén tốt. Tuy nhiên do tính phức tạp của các thuật toán đó nên quá trình nén và giải nén luôn phải tách thành hai quá trình riêng biệt. Đặc điểm này làm cho các thuật toán trên không thể sử dụng trong các trường hợp cần nén và giải nén trực tuyến [2]. Vì vậy, việc tìm kiếm và thiết lập các thuật toán nén và giải nén có thể thực hiện trực tuyến đã và đang là vấn đề thời sự. Dưới đây trình bày thuật toán cho phép nén và giải nén trực tuyến trên dữ liệu tiếng nói.

2. Thuật toán

2.1. Ý tưởng

Chúng ta biết rằng đồ thị biểu diễn năng lượng của tiếng nói là một đường cong liên tục hình sin. Đường cong hình sin này được cấu thành từ các khúc cong đơn điệu [5]. Chúng ta dùng các đoạn thẳng thay cho các khúc cong đơn điệu là ý tưởng cơ bản của thuật toán này (hình 1).



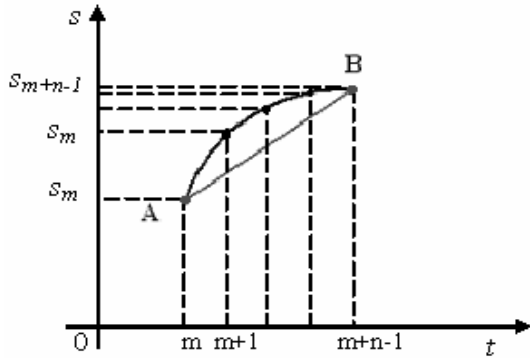
Hình 1. Biểu diễn các khúc cong của hình sin bởi các đoạn thẳng AB, BC, CD.

Để tiện cho việc diễn đạt, ta ký hiệu thuật toán nén mô tả dưới đây là thuật toán TT1.

* Tác giả liên hệ. ĐT: 84-4-37580121.
E-mail: phumv06@yahoo.com

2.2. Mô tả thuật toán TT1

Giả sử dữ liệu của tiếng nói là dãy $S = \{s_1, s_2, \dots, s_L\}$, $L \in \mathbb{N}$, $s_i \in \mathbb{R}$. Ta xét dãy con liên tiếp của S : $\{s_m, s_{m+1}, \dots, s_{m+n-1}\}$, $1 \leq m \leq L-n+1$, $n > 0$, (hình 2). Khi đó, thay vì phải lưu trữ n giá trị của dãy con này ta chỉ lưu trữ các giá trị s_m, s_{m+n-1} và n . Sơ đồ khối hình 3 mô tả thuật toán TT1 làm việc theo chế độ trực tuyến.



Hình 2. Biểu diễn dãy con n điểm liên tiếp của một khúc cong đơn điệu.

Đầu vào: Dãy tín hiệu tiếng nói.

Đầu ra: Các véc tơ có dạng (A, B, n) tương ứng là phần tử đầu, cuối và số phần tử của dãy con đơn điệu.

Các biến sử dụng trong thuật toán TT1 mô tả trong hình 3:

Dấu: nhận giá trị $+1$ hoặc -1 biểu thị dãy con không tăng hay giảm,

s: lưu giá trị tín hiệu,

Y1, Y2: lưu hai giá trị liên tiếp của tín hiệu.

A, B: lưu giá trị đầu và cuối của dãy con đơn điệu.

n: lưu số phần tử của dãy con đơn điệu.

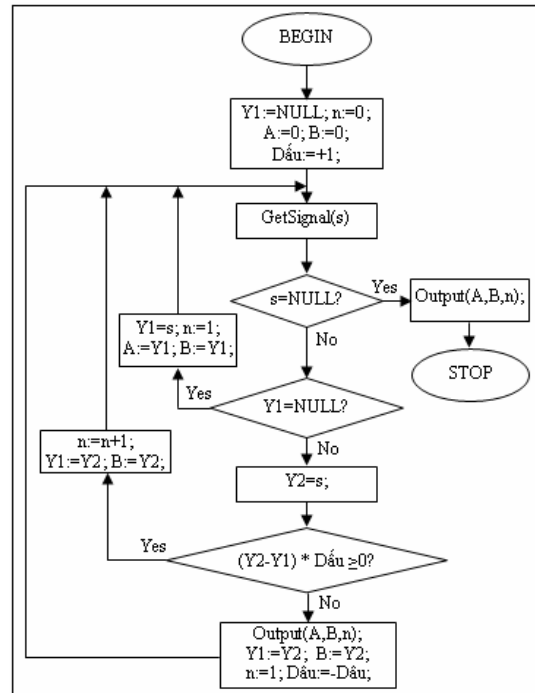
Nhận xét:

1) Thuật toán trên có thể làm việc trực tuyến vì quá trình nén chỉ thực hiện hai thao tác chủ yếu là so sánh hai tín hiệu liên tiếp và đếm.

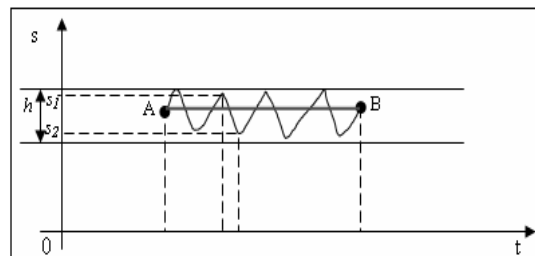
2) Thuật toán mô tả bởi sơ đồ khối hình 3 có thể cải tiến để nâng cao tỷ lệ nén. Chúng ta hãy xét hai lần truyền dữ liệu nén liên tiếp trong

thuật toán trên là Output (A, B, n) và Output $(B, C, n1)$, với A, B là phần tử đầu, cuối của dãy con thứ nhất và B, C là phần tử đầu, cuối của dãy con thứ hai. Rõ ràng ta có thể bớt đi phần tử B ở lần truyền thứ hai.

3) Thuật toán mô tả trên còn có thể cải tiến để nâng cao tỷ lệ nén bằng cách thay các khúc đường cong AB (bao gồm cả các khúc cong không giảm, không tăng) bằng đoạn thẳng AB nếu mọi điểm trên khúc đường cong AB thỏa mãn điều kiện $|s_j - s_2| < h$ ($h > 0$), ở đây s_1, s_2 nằm trên khúc cong AB, minh họa ở hình 4.



Hình 3. Sơ đồ khối thuật toán TT1.



Hình 4. Minh họa ý tưởng cải tiến thuật toán TT1.

2.3. Thuật toán giải nén ứng với thuật toán TT1

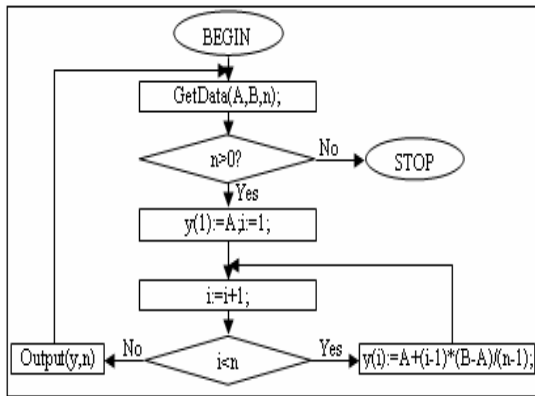
Dữ liệu nén dạng (A, B, n) do thuật toán TT1 thực hiện được giải nén bằng cách nội suy tuyến tính n điểm cách đều với giá trị đầu, cuối tương ứng là A, B . Ký hiệu (y_1, y_2, \dots, y_n) là n giá trị được giải nén. Khi đó:

$$y_i = A + (i-1) * (B-A) / (n-1), \text{ với } n > 1, i = 1..n.$$

Ký hiệu thuật toán giải nén ứng với thuật toán TT1 là thuật toán TT2. Thuật toán TT2 được mô tả trong sơ đồ khối hình 5.

Đầu vào: Các véc tơ chứa dữ liệu nén dạng (A, B, n) , tương ứng là phần tử đầu, cuối và số phần tử của dãy con đơn điệu.

Đầu ra: n giá trị đã được giải nén chứa trong mảng y .



Hình 5. Sơ đồ khối thuật toán TT2.

3. Độ phức tạp tính toán của thuật toán

Thao tác chủ yếu trong thuật toán TT1 là so sánh hai phần tử liên tiếp. Vì vậy nếu dãy tín hiệu vào có n tín hiệu thì số phép so sánh sẽ là $n-1$. Do vậy, độ phức tạp của thuật toán TT1 là $O(n)$.

Tương tự như vậy, thuật toán TT2 cũng có độ phức tạp tính toán tuyến tính, $O(n)$.

4. Đánh giá chất lượng nén

Chúng ta xét một dãy dữ liệu trên một đoạn đơn điệu: (s_1, s_2, \dots, s_n) .

Giả sử dữ liệu sau khi giải nén bởi thuật toán TT2 tương ứng với dãy trên là (y_1, y_2, \dots, y_n) .

Tỷ lệ sai lệch (%) giữa dữ liệu gốc (s_1, s_2, \dots, s_n) và dữ liệu giải nén (y_1, y_2, \dots, y_n) được tính bằng công thức sau:

$$\sigma = \frac{\sum_{i=1}^n |s_i - y_i|}{\sum_{i=1}^n s_i} * 100 (\%)$$

Giả sử thuật toán TT1 đã nén được K đoạn dữ liệu đơn điệu, tỷ lệ sai lệch giữa dữ liệu gốc và dữ liệu giải nén trên từng đoạn tương ứng là $\sigma_1, \sigma_2, \dots, \sigma_K$, khi đó, tỷ lệ sai lệch trung bình giữa dữ liệu gốc và dữ liệu giải nén của thuật toán TT1 được tính bằng công thức sau:

$$\sigma_{TB} = \frac{\sum_{i=1}^K \sigma_i}{K} (\%)$$

Chất lượng của thuật toán TT1 có thể đánh giá qua σ_{TB} . Nếu σ_{TB} càng thấp thì chất lượng sau khi giải nén càng cao và ngược lại.

5. Kết quả, thảo luận

Thử nghiệm thuật toán TT1 và TT2 6 lần trên 6 tệp dữ liệu tiếng nói dạng file wave. Kết quả được thể hiện trong bảng 1.

Bảng 1. Kết quả thử nghiệm thuật toán TT1 và TT2.

Lần thử	Kích thước trước khi nén (f bytes)	Kích thước sau khi nén (g bytes)	Tỷ lệ nén $\frac{100.g}{f} (\%)$	$\sigma_{TB} (\%)$
1	467712	182024	38.92	4.53
2	654336	213407	32.62	6.11
3	787968	271967	34.51	5.78
4	1487616	512879	34.48	6.15
5	2609664	908306	34.81	6.08
6	5806080	1990121	34.28	6.02

Kết quả trong bảng 1 cho thấy tỷ lệ nén trung bình mà thuật toán TT1 đạt được xấp xỉ 35%, điều đó có nghĩa là thuật toán nén được ≈ 3 lần. Tỷ lệ sai lệch trung bình giữa dữ liệu gốc và dữ liệu giải nén $\approx 6\%$.

6. Kết luận

Hiệu suất nén của thuật toán TT1 không thực sự cao như một số thuật toán nén thông dụng trên thị trường, nhưng với ưu điểm đơn giản trong cài đặt và có độ phức tạp tuyến tính nên phù hợp với các bài toán xử lý trực tuyến. Mặt khác, có thể thấy dữ liệu nén của một đoạn dữ liệu tiếng nói cũng có thể coi như một đặc trưng mô tả đoạn dữ liệu đó. Vì vậy, ý tưởng này còn có thể sử dụng vào một số bài toán khác.

Tài liệu tham khảo

- [1] David Salomon, *Data Compression*, Springer - Verlag New York, Inc., 2004.
- [2] Lajos Hanzo, F. Clare Somerville, Jason Woodard, *Voice compression for wireless communication*. Antony Rowe Ltd, Chippenham, England., 2007
- [3] C. Greg Plaxton, Yu Sun, Mitul Tiwari, Harrick Vin, "Online Compression Caching", Proceedings of the 11th Scandinavian workshop on Algorithm Theory, Gothenburg, Sweden, ISBN: 978-3-540-69900-2, vol. 5124, (2008) 414.
- [4] Konstantinos G. Kyriakopoulos, David J. Parish, "A system for online compression of high-speed network measurements", International Journal of Internet Protocol Technology, Springer-Verlag Berlin, Heidelberg, vol. 3, No.2, (2008) 95.
- [5] R. John Deller, Jr., John H.L. Hansen, G. John Poakis, *Discrete - Time Processing of Speech Signals*. The Institute of Electrical and Electronics Engineers Inc., New York, 2000.
- [6] G. Richard Lyons, *Understanding Digital Signal Processing*. Prentice Hall, 2001.

Algorithm for compressing and decompressing online voice data

Nguyen Van Xuat, Mai Van Phu

Faculty of Information Technology, Military Technical Academy, 100 Hoang Quoc Viet, Hanoi, Vietnam

This article proposed an algorithm for compressing and decompressing online voice data. The content of this algorithm is linear approximations voice data. This article also addressed an algorithm for unpacking measures with complex linear. The test results showed that this algorithm can be used for compressing and decompressing application of online voice data.

Keywords: Data compression, voice, voice data, online, online voice data.