Прогрессивное кодирование аудио с помощью сингулярного разложения

08, август 2012

DOI: 10.7463/0812.0475576

Свиридов А. А.

УДК 004.627

Россия, Тульский государственный университет

Sviridov86@gmail.com

Введение

В настоящее время очень много внимания уделяется вопросам компрессии аудио, а также передаче аудио по самым разным сетям связи. Эта область исследования затрагивает и обработку видео, поскольку в самых распространённых видео кодеках, таких, например, как DivX, Xvid, WMV, QuickTime, Nero Digital и других, аудио и видео потоки хранятся по отдельности. Все эти кодеки в той или иной степени соответствуют стандарту MPEG-4 [1], который и предполагает раздельное хранение потоков.

Целью работы является разработка кодека для прогрессивного кодирования аудио при потоковой обработке в контексте передачи сигнала по каналу связи в режиме реального времени, создание системы тестирования аудио кодеков, применяемых для решения рассматриваемой задачи, и сравнение предлагаемого подхода с хорошо известными.

Прогрессивное кодирование может использоваться как для передачи, так и для сжатия аудио. Особенный интерес представляет задача непрерывной передачи сигнала в режиме реального времени. Примером такой задачи может служить организация online-конференции по сети Internet.

В предлагаемом подходе используем двумерные методы сжатия, поэтому в работе сначала рассматриваем некоторые известные подходы к компрессии одномерных сигналов двумерными методами, далее описываем новый алгоритм для прогрессивного кодирования аудио и разработанную систему тестирования кодеков, после чего анализируем результаты тестов.

При прогрессивном кодировании фрейм аудио преобразуют в вектор, элементы которого расположены в порядке убывания их значимости, при этом весь фрейм может быть восстановлен с использованием некоторого количества первых элементов этого вектора. Чем большее элементов будет использовано, тем лучше будет качество восстановленного аудио.

На сегодняшний день большинство кодеков для прогрессивного кодирования аудио работает по схожей схеме. В обобщённом виде эта схема показана на рисунке 1.

Рисунок 1. Обобщённая схема кодера для прогрессивного кодирования аудио

Подход, основанный на одномерном дискретном вейвлет-преобразовании (One-Dimensional Discrete Wavelet Transform, 1D DWT или просто DWT, [2]), можно считать классическим в этой задаче. В [3] рассматривают и сравнивают несколько разных 1D DWT применительно к задаче сжатия аудио.

В работе [4] показывают, как можно использовать алгоритм прогрессивного кодирования SPIHT (Set Partitioning In Hierarchical Trees) для одномерного аудио сигнала. Отметим, что изначально SPIHT был предложен для прогрессивного кодирования коэффициентов, полученных с помощью двумерного дискретного вейвлет-преобразования (Two-Dimensional Discrete Wavelet Transform, 2D DWT) изображений [5]. В качестве альтернативы SPIHT может быть использован предложенный ранее алгоритм

EZW (Embedded Zerotree Wavelet) [6], однако в [5] показано, что SPIHT даёт такие же или лучшие степень сжатия и качество восстановленного изображения по сравнению с EZW.

Побитовую прогрессивность, то есть возможность восстанавливать весь фрейм по любому количеству первых закодированных бит, дают SPIHT и EZW, их специально разрабатывали для этого.

Прогрессивность можно обеспечить и с помощью 1D DWT без использования 1D SPIHT, но уже не до отдельного бита. Схема прогрессивного кодирования с помощью 1D DWT показана на рисунке 2.

Рисунок 2. Схема прогрессивного кодирования с помощью 1D DWT

На рисунке 2 за AC N обозначены коэффициенты аппроксимации 1D DWT N-ого уровня (N-Level Approximation Coefficients), а за DC N – коэффициенты, описывающие детали 1D DWT N-ого уровня (N-Level Details Coefficients). После преобразования исходного фрейма в цепочку коэффициентов $AC\ 3 \rightarrow DC\ 3 \rightarrow DC\ 2 \rightarrow DC\ 1$ для последующего восстановления всего фрейма можно использовать любое количество первых звеньев этой цепи.

Предлагали и другие методы для решения задачи прогрессивной передачи аудио, например, [7] и [8]. В этих работах полученные результаты сравнивали с результатами работы ряда кодеков, в том числе и кодеков, основанных на 1D DWT.

В работе предлагаем новый метод, основанный на применении сингулярного разложения (Singular Value Decomposition, SVD) к двумерному представлению аудио сигнала. Аудио уже обрабатывали ранее с помощью SVD. В [9] показана возможность применения SVD в целях обеспечения защиты авторских прав в рамках использования техники водяных знаков. В [10] это разложение применяли на одном из этапов работы аудио кодека, основанного на анализе независимых компонент (Independent Component Analysis, ICA). Тем не менее, предлагаемый в данной работе кодек сильно отличается от кодека, рассматриваемого в [10], кроме того, в [10] не рассматривали особенности прогрессивного кодирования.

Научная новизна работы заключается в разработке нового алгоритма прогрессивного кодирования аудио, определении оценки погрешности главного этапа его работы, разработке системы тестирования некоторых аудио кодеков и анализе применения модифицированного оконного преобразования Тьюки. Разработанный кодек показал возможность кодирования при сохранении среднего качества звучания на скорости 171,3 Кбит/с, а для некоторых жанров – на скорости 86,5 Кбит/с. Для сравнения один канал несжатых данных CD-качества закодирован на скорости 706 Кбит/с.

Сравнить этот результат с результатами кодеков, рассматриваемыми в других работах, довольно сложно. Это связано с тем, что не всегда точно и однозначно описан алгоритм работы кодеков, а исходный код, как правило, недоступен. Также недоступны и тестовые композиции. В частности и по этим причинам исходные тексты разработанной системы тестирования, все тестовые композиции и результаты тестов размещены в свободном доступе на вебсайте [11]. Общая оценка, тем не менее, такая: новый подход при определённых параметрах кодирования лучше подходов, основанных на 1D DWT, но при других параметрах хуже.

Если сравнить улучшенные за счёт применения модифицированного оконного преобразования Тьюки 1D DWT кодеки с кодеками из [3], то они будут в некоторых случаях лучше, а в некоторых хуже, но кодекам из [7] и [8] они уступают. Однако здесь следует отметить, что, во-первых, в данной работе при тестировании не использовался этап кодирования с помощью SPIHT, который может значительно улучшить результаты,

во-вторых, было показано, что оконное преобразование улучшает результаты, однако выбора оптимальных параметров этого преобразования не проводилось, и в-третьих, рассматриваемые во всех этих работах подходы могут комбинироваться для получения оптимального результата.

1 Компрессия 1D сигналов с помощью 2D преобразований

Преобразование 1D сигналов в различные 2D формы и обработка полученных представлений разными 2D методами уже проводили ранее.

В [12] показано, что 2D DWT даёт лучшие результаты при компрессии электрокардиограмм (ЭКГ), чем 1D DWT. В качестве 2D формы сигнала в [12] используют PS-представление (Pitch-Synchronous Signal Representation, PSSR) [13]. PSSR используют для представления псевдо периодических сигналов. Понятие псевдо периодичности хорошо иллюстрирует рисунок 3.

Рисунок 3. ЭКГ – пример псевдо периодического сигнала [14]

В [15] рассматривают компрессию данных, полученных с устройств, собирающих информацию о характеристиках напряжения в сети, показано преимущество двумерного

представления над одномерным. Напряжение также можно считать псевдопериодическим сигналом.

В [16] входной сигнал сначала обрабатывают оконной функцией Кайзера-Бесселя, двумерное представление аудио в этой работе основано на спектре сигнала, далее к этому представлению применяют двумерное модифицированное дискретное косинус-или синус-преобразование (two-dimensional modified discrete cosine/sine transform, 2D MDCT/MDST).

В работах [15] и [17] предлагают использовать другой способ для преобразования аудио сигнала в 2D форму. Пусть $x_1, \dots, x_N$ – отсчёты фрейма аудио, тогда матрица $X = (x_{ij})$, $i = 1, \dots, n$, $j = 1, \dots, m$, $n \times m = N$ двумерного представления может быть получена с помощью биективного отображения

$$T_{12} : \mathbf{R}^N \rightarrow \mathbf{R}^{n \times m},$$

$$x_{ij} = x((i-1)m + j), i = 1, \dots, n, j = 1, \dots, m,$$

$$\text{где } x(i) = x_i.$$

Обратное к T_{12} отображение определяют как

$$T_{21} : \mathbf{R}^{n \times m} \rightarrow \mathbf{R}^N,$$

$$x_i = X\left(\left\lfloor \frac{i-1}{m} + 1 \right\rfloor, i - \left\lfloor \frac{i-1}{m} \right\rfloor \cdot m\right), i = 1, \dots, N,$$

$$\text{где } X(i, j) = x_{ij}, \lfloor i \rfloor = \max_{\substack{k \in \mathbf{Z}, \\ k \leq i}} k.$$

Схема преобразования T_{12} показана на рисунке 4.

Рисунок 4. Схема преобразования сигнала отображением T_{12}

На рисунке 5 приведены изображения некоторых матриц фреймов аудио размера 32x32. Вопрос визуализации таких матриц рассматривают в [18].

Рисунок 5. Изображения некоторых матриц фреймов аудио размера 32x32

Идеи по сжатию, предлагаемые в [17], используют в [18] и [19]. В этих работах для перехода к 2D представлению применяют T_{12} . В [18] для компрессии аудио используют SVD, а в [19] 2D DWT. При разработке нового кодека для прогрессивного кодирования аудио возьмём за основу идеи, которые рассматривают в [18].

2 Использование SVD для компрессии аудио

Идея сжатия аудио с помощью SVD основана на двух теоремах. Первая из них определяет само сингулярное разложение [20].

Теорема 1. Пусть $X \in \mathbb{R}^{n \times m}$, U и V – вещественные ортогональные матрицы, тогда X можно представить в виде

$$X = USV^T, \quad (1)$$

где S – диагональная матрица размера $r \times r$

$$S = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r \end{pmatrix},$$

$r = \min(n, m)$ и диагональные элементы матрицы S удовлетворяют неравенствам

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0.$$

Матрица X , разложенная по формуле (1), может быть записана в виде

$$X = \sum_{l=1}^k \sigma_l u_l v_l^T, \quad (2)$$

где k – ранг матрицы X , u_l – столбцы матрицы U (левые сингулярные векторы матрицы X), v_l – столбцы матрицы V (правые сингулярные векторы матрицы X).

Вторая теорема даёт оценку аппроксимации с помощью SVD (это частный случай более общей теоремы для матрицы X над полем $\mathbf{C}$ [21, с. 233]).

Теорема 2. Пусть матрица $X \in \mathbf{R}^{n \times m}$ задана сингулярным разложением (2) и $\sigma_{k+1} = 0$. Пусть задано целое $1 \leq s \leq k$, тогда

$$\min_{\substack{\text{rank}(Y) \leq s, \\ Y \in \mathbf{R}^{n \times m}}} \|X - Y\|_2 = \sigma_{s+1} = \|X - \tilde{X}_s\|_2,$$

$$\text{где } \|X\|_2 = \sup_{\substack{\|y\|_2 = 1, \\ y \in \mathbf{R}^m}} \|Xy\|_2, \quad \tilde{X}_s = \sum_{l=1}^s \sigma_l u_l v_l^T.$$

Сформулируем и докажем два утверждения, которые будем использовать для прогрессивного кодирования аудио с помощью SVD.

Утверждение 1. Пусть $x \in \mathbf{R}^N$ – фрейм аудио сигнала, $T_{12}: x \rightarrow X$, $X = (x_{ij})$, $X \in \mathbf{R}^{n \times m}$. Если в разложении (1) матрицы X оставить s максимальных сингулярных значений, а остальные обнулить, то для приближённого восстановления x потребуется $(n+m)s$ значений.

Доказательство. Согласно (1) $X = USV^T$. Возможно два варианта: $n \geq m$ и $n < m$. Рассмотрим первый. Тогда $r = m$ и (1) в терминах размерностей матриц имеет вид: $(n \times m) \cdot (m \times m) \cdot (m \times m)$. Если отдельно хранить матрицы U , S и V , то для восстановления исходного фрейма потребуется $\tilde{N} = nm + m^2 + m^2 = m(n + 2m)$ значений.

Пусть $\tilde{S}_s^*$ – матрица, полученная из S обнулением всех строк и столбцов с номерами большими s . Если заменить в (1) S на $\tilde{S}_s^*$, то для приближённого восстановления матрицы X можно хранить только первые s столбцов U , так как элементы столбцов U с номерами большими s будут умножаться на соответствующие нулевые элементы строк матрицы $\tilde{S}_s^*$. Аналогично можно хранить только первые s столбцов V . Таким образом, для приближённого восстановления X нужно хранить s первых левых и правых сингулярных векторов и соответствующие сингулярные значения. Обозначим за $\tilde{U}_s$ и $\tilde{V}_s$ матрицы, столбцами которых являются первые s столбцов матриц U и V соответственно, а за $\tilde{S}_s$ – диагональную матрицу размера $s \times s$, диагональные элементы которой совпадают с первыми s диагональными элементами матрицы S .

Теперь (1) можно записать в виде $X \approx \tilde{X}_s = \tilde{U}_s \tilde{S}_s \tilde{V}_s^T$, а в терминах размерностей матриц получим: $(n \times s) \cdot (s \times s) \cdot (s \times m)$. Заметим, что для хранения $\tilde{S}_s$ нужно только s значений диагональных элементов. Тогда $\tilde{N} = ns + s + sm = s(n + m + 1)$. Кроме того, вместо хранения всех матриц $\tilde{U}_s$, $\tilde{S}_s$ и $\tilde{V}_s^T$ по отдельности можно хранить любую пару матриц $\tilde{U}_s \tilde{S}_s$ и $\tilde{V}_s^T$ или $\tilde{U}_s$ и $\tilde{S}_s \tilde{V}_s^T$, тогда $\tilde{N} = ns + sm = (n + m)s$.

Случай $n < m$ рассматривается аналогично. Утверждение доказано.

Замечание. Вообще говоря, от выбора пары матриц $\tilde{U}_s \tilde{S}_s$ и $\tilde{V}_s^T$ или $\tilde{U}_s$ и $\tilde{S}_s \tilde{V}_s^T$ может зависеть общее количество нулевых элементов этих матриц, а от этого будет зависеть степень сжатия, тем не менее, небольшое количество численных экспериментов не выявило предпочтений в этом выборе. В работе при тестировании SVD кодека сохраняем первую пару матриц.

Утверждение 2. Пусть $x \in \mathbf{R}^N$ – фрейм аудио сигнала, $T_{12}: x \rightarrow X$, $X = (x_{ij})$, $X \in \mathbf{R}^{n \times m}$, $\tilde{X}_s = \tilde{U}_s \tilde{S}_s \tilde{V}_s^T$ – аппроксимация X по формуле (1) при сохранении s максимальных сингулярных значений, $T_{21}: \tilde{X}_s \rightarrow \tilde{x}_s$, тогда справедлива оценка

$$\|x - \tilde{x}_s\|_2 \leq \sqrt{m}\sigma_{s+1}.$$

Доказательство. В соответствии с теоремой 2 выполняется равенство $\|X - \tilde{X}_s\|_2 = \sigma_{s+1}$, тогда имеем

$$\forall y: y \in \mathbf{R}^m, \|y\|_2 = 1: \|(X - \tilde{X}_s)y\|_2 \leq \sigma_{s+1} \quad (3)$$

Рассмотрим единичные векторы $e_k = (\delta_{k1}, \delta_{k2}, \dots, \delta_{km})$, где δ_{kj} – символ Кронекера, $k=1, \dots, m$. Подставим e_k в (3) вместо y , получим

$$\|(X - \tilde{X}_s)e_k\|_2^2 = \sum_{i=1}^n \left(\sum_{j=1}^m (x_{ij} - \tilde{x}_{ij}^s) \delta_{kj} \right)^2 = \sum_{i=1}^n (x_{ik} - \tilde{x}_{ik}^s)^2 \leq \sigma_{s+1}^2. \quad (4)$$

Просуммируем теперь левые части неравенства (4) для всех $k=1, \dots, m$

$$\begin{aligned} \sum_{k=1}^m \sum_{i=1}^n (x_{ik} - \tilde{x}_{ik}^s)^2 &= \sum_{i=1}^n \sum_{j=1}^m (x_{(i-1)m+j} - \tilde{x}_{(i-1)m+j}^s)^2 = \\ &= \sum_{i=1}^N (x_i - \tilde{x}_i^s)^2 = \|x - \tilde{x}^s\|_2^2 \leq m\sigma_{s+1}^2. \end{aligned} \quad (5)$$

Из (5) следует, что $\|x - \tilde{x}_s\|_2 \leq \sqrt{m}\sigma_{s+1}$. Утверждение доказано.

3 Прогрессивное кодирование аудио с помощью SVD

Предлагаемая в работе схема прогрессивного кодирования аудио показана на рисунке 6.

Рисунок 6. Схема прогрессивного кодирования аудио, основанная на SVD

SVD, как и 1D DWT, также можно использовать для прогрессивного кодирования без SPIHT. Для приближённого восстановления всего фрейма $\forall s: 0 < s \leq \min(n, m)$ достаточно пары матриц $\tilde{U}_s \tilde{S}_s$ и $\tilde{V}_s^T$, то есть $(n+m)s$ значений. Для получения более точного приближения матрицы X или, что то же самое, лучшего качества восстановленного аудио, нужно передать очередной столбец матрицы $\tilde{U}_s \tilde{S}_s$ и очередную строку матрицы $\tilde{V}_s^T$, то есть $n+m$ значений.

На этапе оконного преобразования, заключающегося в умножении отсчётов фрейма аудио на соответствующие значения оконной функции, в работе предлагаем использовать модифицированную оконную функцию Тьюки. Функцию Тьюки определяют следующим образом [22, с. 69]:

$$w_{T,p}(x) = \begin{cases} \frac{1}{2} \left(1 - \cos \frac{2\pi x}{p} \right), & 0 \leq x < \frac{p}{2}, \\ 1, & \frac{p}{2} \leq x < 1 - \frac{p}{2}, \\ \frac{1}{2} \left(1 - \cos \frac{2\pi(1-x)}{p} \right), & 1 - \frac{p}{2} \leq x < 1. \end{cases}$$

При $p=0$ окно Тьюки совпадает с прямоугольным окном, а при $p=1$ – с окном Хана. В работе предлагается использовать $p=0.5$. На рисунке 7 показаны графики этих оконных функций.

Рисунок 7. Графики прямоугольной оконной функции и оконных функций Хана и Тьюки при $p=0.5$

В SVD кодеке используем следующую модификацию $w_{T,p}$

$$\tilde{w}_{T,p}(x) = \frac{w_{T,p}(x) + a}{b}. \quad (6)$$

Параметр a в (6) нужен для того, чтобы не происходило потери значений на концах отрезка $[0,1]$, поскольку перекрытие фреймов не используем. b введен для корректировки максимального значения оконной функции.

В ходе проведения небольшого количества численных экспериментов с целью определения наиболее оптимальных значений параметров a и b для $\tilde{w}_{T,0.5}$ было

установлено, что наилучшее соотношение между степенью сжатия и качеством восстанавливаемого аудио для 1D DWT и SVD кодеков достигается при $a = 10$ и $b = 1000$.

4 Система тестирования 1D DWT и SVD кодеков

Для сравнения результатов работы 1D DWT и SVD кодеков была разработана программная система в математическом пакете MATLAB. Эта система позволяет рекурсивно обработать папку с аудио и отобразить результаты работы кодеков для всех файлов и папок.

Для оценки качества кодирования используем SNR (Signal to Noise Ratio) и RMS (Root Mean Square). Пусть $x = (x_1, x_2, \dots, x_N)$ – отсчёты исходного wav-файла, $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_N)$ – отсчёты восстановленного wav-файла, векторы нормализованы, тогда SNR и RMS можно рассчитать по формулам

$$\text{SNR}(x, \tilde{x}) = 10 \log_{10} \frac{\sum_{i=1}^N x_i^2}{\sum_{i=1}^N (x_i - \tilde{x}_i)^2},$$

$$\text{RMS}(x, \tilde{x}) = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \tilde{x}_i)^2}.$$

При расчёте степени сжатия исходное количество отсчётов всего wav-файла делим на количество ненулевых значений, требуемых для восстановления. Рассчитываем среднее время, за которое кодировщики и декодировщики обрабатывают одну секунду аудио.

В системе реализованы все этапы кодирования, кроме выполнения SPIHT. Схематически тестируемые этапы показаны на рисунке 8. Отметим, что для корректного

сравнения 1D DWT и SVD анализируем результаты работы обоих преобразований как с применением оконного преобразования, так и без него. Тестирование указанных на рисунке 8 этапов представляет отдельный интерес, поскольку позволяет оценить, насколько хорошо каждый из алгоритмов подходит для рассматриваемой задачи. Кроме того, результаты тестов могут быть полезны при реализации подходов, в которых вместо SPIHT на соответствующем этапе кодирования используют другие алгоритмы.

Рисунок 8. Сравнимые этапы кодирования 1D DWT и SVD кодеков

Рассмотрим 200 аудио файлов, выбранных для тестирования кодеков. Было собрано по 10 композиций следующих 20 разных жанров:

- альтернативный рок;
- блюз;
- классическая музыка;
- кантри;

- электронная музыка:
 1. электро-хаус;
 2. дабстеп;
- речь (английский язык);
- хэви-метал;
- инструментальная музыка:
 1. скрипка;
 2. пианино;
 3. традиционная американская флейта;
 4. гитара;
 5. барабаны;
- джаз;
- латиноамериканская музыка;
- медленная мелодичная музыка;
- опера;
- популярная музыка;
- рэп;
- шотландский фольклор.

Выбирались наиболее распространённые и популярные жанры, при выборе использовали вебсайт [23] и многие другие Internet-ресурсы, посвящённые музыке.

Все 200 аудио файлов имеют одинаковые параметры:

- формат: WAV (PCM);
- количество каналов: 1 (mono);
- количество бит на отсчёт сигнала: 16.

Эти параметры соответствуют “CD-качеству” за исключением наличия только одного канала (mono) вместо двух (stereo), поскольку задачу эффективного кодирования нескольких каналов не рассматриваем. Средняя продолжительность звучания каждого файла составляет 3-4 минуты.

Композиции выбирали так, чтобы наилучшим образом отразить особенности жанра. Например, в случае инструментальной музыки в каждой композиции присутствует только один соответствующий инструмент, для речи отсутствуют любые посторонние звуки, мелодичная музыка остаётся медленной и плавной на протяжении всей композиции и так далее.

Использование большого количества композиций навряд ли рационально, поскольку, во-первых, результаты в большей степени зависят от правильного выбора композиций, а не от их количества, а во-вторых, увеличение количества композиций увеличит и время работы тестов.

5 Сравнение результатов работы 1D DWT и SVD кодеков

В 1D DWT кодеках можно применять разные 1D DWT с различными параметрами. Для сравнения с SVD кодеком были выбраны 1D DWT Хаара (haar) и биортогональные вейвлеты (bior3.5).

haar не очень хорошо подходит для кодирования аудио по сравнению с другими вейвлетами из-за разрывности масштабирующей функции и материнского вейвлета, а bior3.5 напротив должно давать очень хороший результат. Такие выводы можно сделать на основе тестов, проводимых в [2], и анализа масштабирующих функций и материнских вейвлетов этих преобразований. На рисунке 9 показаны: а – масштабирующая функция Хаара, b – материнский вейвлет Хаара, c, d – масштабирующая функция и материнский вейвлет для bior3.5, используемые при выполнении разложения, e, f – масштабирующая функция и материнский вейвлет для bior3.5, используемые для восстановления сигнала.

Рисунок 9. Масштабирующие функции и материнские вейвлеты используемых при тестировании 1D DWT кодеков

Таким образом, для сравнения с SVD кодеком были выбраны один из наименее (haar) и один из наиболее (bior3.5) эффективных вариантов в случае 1D DWT.

В ходе проведения численного эксперимента было выполнено 18 тестов: для каждого кодека (bior3.5, SVD и haar) считали характеристики при сжатии примерно в 2, 4 и

8 раз без применения оконного преобразования и с обработкой фреймов с помощью $\hat{w}_{T,0.5}$ с параметрами $a = 10$ и $b = 1000$.

На рисунке 10 показаны средние значения SNR по всем тестовым композициям для разных степеней сжатия с применением оконного преобразования $\hat{w}_{T,0.5}$ и без него.

Рисунок 10. Средние значения SNR для разных степеней сжатия с применением оконного преобразования (с ОП) и без него

Хорошо видно, что оконное преобразование значительно улучшает SNR. Без выполнения преобразования уже при сжатии в 4 раза качество работы всех кодеков становится плохим. Поэтому далее будем анализировать только работу кодеков с применением предложенного преобразования.

Все тестируемые кодеки при сжатии в 8 раз показали приемлемый результат ($\text{SNR} > 9 \text{ dB}$). Для музыки сильное кратковременное падение качества не так критично, как для речи, поскольку в случае речи нужно понимать говорящего. При сжатии в 8 раз средний SNR для речи у всех кодеков оказался больше 10 dB, при этом сигнал зашумлён, но в большинстве случаев все слова отчётливы и хорошо понятны.

На рисунках 11-13 показаны средние значения SNR по разным жанрам для сжатия примерно в 2, 4 и 8 раз. Хорошо видно, что лучше всего сжимается плавная мелодичная музыка, а хуже – “тяжёлая” музыка.

Рисунок 11. Средние значения SNR для разных жанров при сжатии примерно в 2 раза

Рисунок 12. Средние значения SNR для разных жанров при сжатии примерно в 4 раза

Рисунок 13. Средние значения SNR для разных жанров при сжатии примерно в 8 раз

Для достижения необходимого сжатия (Compression Ratio, CR) в случае SVD выбирали требуемое количество сингулярных значений:

$$CR = \frac{nm}{(n+m)s} \Rightarrow s = \frac{nm}{(n+m)CR}.$$

А для 1D DWT оставляли только AC соответствующего уровня разложения. На рисунке 14 показана средняя степень сжатия, полученная в ходе работы кодеков.

Рисунок 14. Средняя степень сжатия

На рисунках 15 и 16 показано среднее время (в миллисекундах) обработки одной секунды аудио кодировщиками и декодировщиками соответственно. Все тесты выполняли на ноутбуке со следующими характеристиками:

- процессор: Intel Core 2 Duo P8600, 2.40 GHz;
- оперативная память: 3 Gb.

По графикам на рисунках 15 и 16 видно, что кодеки могут работать в режиме реального времени.

Рисунок 15 – Среднее время обработки одной секунды кодировщиками

Рисунок 16. Среднее время обработки одной секунды декодировщиками

Видно, что SVD обладает наилучшей производительностью. Кроме того, время работы кодека не меняется с увеличением степени сжатия, поскольку при выполнении разложения (1) сразу рассчитываются все “уровни” прогрессивного представления, а при декодировании просто умножаются две матрицы. В случае 1D DWT время работы кодека увеличивается с ростом степени сжатия из-за необходимости выполнения дополнительных 1D DWT.

Заключение

В работе рассмотрены общие принципы работы классических кодеков для прогрессивного кодирования аудио, предложен новый метод, основанный на сингулярном разложении, и преобразование, использующее оконную функцию Тьюки, которое улучшило результаты работы всех тестируемых кодеков. Описаны возможности разработанной в математическом пакете MATLAB системы тестирования 1D DWT, 2D DWT и SVD кодеков и используемый для тестирования набор из 200 композиций 20 разных жанров. Проанализированы результаты численных экспериментов, проводимых для оценки работы 1D DWT и SVD кодеков.

В качестве направлений дальнейших исследований можно выделить следующее:

- реализация в системе тестирования кодеков алгоритма SPIHT и его модификации для одномерного случая, проведение численных экспериментов для всех этапов работы кодеков;
- определение оптимального оконного преобразования;
- исследование вопроса о том, какие 1D DWT наиболее эффективны в задаче прогрессивного кодирования аудио;
- объединение наиболее эффективных подходов для достижения оптимального результата.

Статьи автора по обработке аудио, исходный код системы тестирования кодеков, тестовые аудио файлы, примеры работы тестов и другие материалы размещены в свободном доступе на вебсайте [11].

Список литературы

1. Overview of the MPEG-4 Standard. ISO/IEC JTC1/SC29/WG11 N4668. Available at: <http://mpeg.chiariglione.org/standards/mpeg-4/mpeg-4.htm> , accessed 26.09.2012.
2. Добеши И. Десять лекций по вейвлетам : пер. с англ. / под ред. А.В. Широбокова. Ижевск: НИЦ "Регулярная и хаотическая динамика", 2001. 464 с. [Ingrid Daubechies. Ten Lectures on Wavelets. SIAM: Society for Industrial and Applied Mathematics, 1992. 377 p.]

3. Dhubkarya D.C., Sonam Dubey. High Quality Audio Coding at Low Bit Rate using Wavelet and Wavelet Packet Transform // Journal of Theoretical and Applied Information Technology. 2009. Vol. 6, no. 2. P. 194-200. Available at: <http://www.jatit.org/volumes/research-papers/Vol6No2/7Vol6No2.pdf> , accessed 19.10.2012.
4. Strahl S., Zhou H., Mertins A. An Adaptive Tree-Based Progressive Audio Compression Scheme // Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (October 2005, USA, NY, New Paltz), 2005. P. 219-222.
5. Said A., Pearlman W.A. A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees // IEEE Transactions on Circuits and Systems for Video Technology. 1996. Vol. 6, No. 3. P. 243-250. DOI: [10.1109/76.499834](https://doi.org/10.1109/76.499834)
6. Shapiro J.M. Embedded Image Coding using Zerotrees of Wavelet Coefficients // IEEE Transactions on Signal Processing. 1993. Vol. 41, No. 12. P. 3445-3462. DOI: <http://dx.doi.org/10.1109/78.258085>
7. Raad M., Mertins A., Burnett I. Audio Compression using the MLT and SPIHT // Proc. of the 6th International Symposium on DSP for Communication Systems, 2002. P. 128-132.
8. Hansen H., Strahl S., Mertins A. Fine-Grain Scalable Audio Coding Based on Envelope Restoration and the SPIHT Algorithm // Proc. of the 16th International Conference on Digital Signal Processing, 2009. P. 1-5.
9. Hamza Ozer, Bulent Sankur, Nasir Memon. An SVD-based audio watermarking technique // Proc. of the International Multimedia Conference, New York, USA, SESSION: Audio, 2005. P. 51-56.
10. Ben-Shalom A., Werman M., Dubnov S. Improved Low Bit-Rate Audio Compression using Reduced Rank ICA Instead of Psychoscoustic Modeling // Proc. of the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 5. 2003. P. 461-464.
11. Progressive Audio Transmission. Available at: <https://sites.google.com/site/sviridov86> , accessed 26.09.2012.

12. Dominic Mathew, Devassia V.P., Tessamma Thomas. Compression of Pseudo-periodic Signals Using 2D Wavelet Transform // American Journal of Signal Processing. 2011. Vol. 1, No. 2. P. 46-50. DOI: 10.5923/j.ajsp.20110102.08
13. Evangelista G. Pitch-Synchronous Wavelet Representations of Speech and Music Signals // IEEE Transactions on Signal Processing. 1993. Vol. 41, No. 12. P. 3313-3330. DOI: <http://dx.doi.org/10.1109/78.258076>
14. Суворов А. В. Клиническая электрокардиография. Нижний Новгород: Изд-во НМИ, 1993. 124 с.
15. Omer Nezih Gerek, Dogan Gokhan Ece. Compression of Power Quality Event Data using 2D Representation // Electric Power Systems Research. 2008. Vol. 78, No. 6. P. 1047-1052.
16. Vinton M.S., Atlas L.E. A Scalable and Progressive Audio Codec // Proc. of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 5. 2001. P. 3277- 3280.
17. Свиридов А.А. Использование методов сжатия изображений для компрессии потокового аудио // Современные проблемы математики, механики, информатики: материалы Региональной научной студенческой конференции. Тула: Изд-во ТулГУ. 2010. С. 167-174.
18. Свиридов А.А. Использование сингулярного разложения для компрессии потокового аудио // Известия ТулГУ. Технические науки. 2012. Вып. 3. С. 175-183.
19. Свиридов А.А. Использование двумерного дискретного вейвлет-преобразования для компрессии аудио // Научно-технический вестник Поволжья. 2012. № 4. С. 186-190.
20. Anderson E., Bai Z., Bischof C., Blackford S., Demmel J., Dongarra J., Du Croz J., Greenbaum A., Hammarling S., McKenney A., Sorensen D. LAPACK User's Guide. Third Edition. SIAM, 1999. Available at: <http://www.netlib.org/lapack/lug/node53.html>, accessed 26.09.2012.
21. Тыртышников Е.Е. Матричный анализ и линейная алгебра. М.: ФИЗМАТЛИТ, 2007. 480 с.

22. Bloomfield P. Fourier Analysis of Time Series. An Introduction. Second Edition. A Wiley-Interscience Publication, 2000. 269 p.
23. AllMusic : Music Search, Recommendations, Videos and Reviews. Available at: <http://www.allmusic.com> , accessed 26.09.2012.

Progressive coding of audio by means of singular decomposition

08, August 2012

DOI: 10.7463/0812.0475576

Sviridov A.A.

Russia, Tula State University
Sviridov86@gmail.com

The author considers the current best approaches to progressive audio coding which use one-dimensional and two-dimensional transformations. The author proposes a new method for progressive coding which is based on singular decomposition and gives the error estimate of the main stage of the algorithm. The author describes the new codec which uses the proposed ideas. The author considers the system of testing audio codecs which was developed in MATLAB; it allows comparison of classical codecs with the offered codec in numerical experiments on 200 compositions of 20 different genres. In some cases the new approach is better than other existing ones, and in some cases it is worse. During the numerical experiments it was determined that the proposed modification in the Tukey window conversion improved coding parameters of all the tested codecs. The new approach can be combined with the conventional ones. The author gives directions of future research.

Publications with keywords:[singular value decomposition](#), [progressive audio coding](#), [Tukey window function](#)

Publications with words:[singular value decomposition](#), [progressive audio coding](#), [Tukey window function](#)

References

1. *Overview of the MPEG-4 Standard. ISO/IEC JTC1/SC29/WG11 N4668.* Available at: <http://mpeg.chiariglione.org/standards/mpeg-4/mpeg-4.htm> , accessed 26.09.2012.

2. Daubechies I. *Ten Lectures on Wavelets*. SIAM: Society for Industrial and Applied Mathematics, 1992. 377 p. (Russ. ed.: Dobeshi I. *Desiat' lektsii po veivletam*. Izhevsk, NITs "Reguliarnaia i khaoticheskaia dinamika" Publ., 2001. 464 p.).
3. Dhubkarya D.C., Sonam Dubey. High Quality Audio Coding at Low Bit Rate using Wavelet and Wavelet Packet Transform. *Journal of Theoretical and Applied Information Technology*, 2009, vol. 6, no. 2, pp. 194-200. Available at: <http://www.jatit.org/volumes/research-papers/Vol6No2/7Vol6No2.pdf> , accessed 19.10.2012.
4. Strahl S., Zhou H., Mertins A. An Adaptive Tree-Based Progressive Audio Compression Scheme. *Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 2005, USA, NY, New Paltz, 2005, pp. 219-222.
5. Said A., Pearlman W.A. A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 1996, vol. 6, no. 3, pp. 243-250. DOI: [10.1109/76.499834](https://doi.org/10.1109/76.499834)
6. Shapiro J.M. Embedded Image Coding using Zerotrees of Wavelet Coefficients. *IEEE Transactions on Signal Processing*, 1993, vol. 41, no. 12, pp. 3445-3462. DOI: <http://dx.doi.org/10.1109/78.258085>
7. Raad M., Mertins A., Burnett I. Audio Compression using the MLT and SPIHT. *Proc. of the 6th International Symposium on DSP for Communication Systems*, 2002, pp. 128-132.
8. Hansen H., Strahl S., Mertins A. Fine-Grain Scalable Audio Coding Based on Envelope Restoration and the SPIHT Algorithm. *Proc. of the 16th International Conference on Digital Signal Processing*, 2009, pp. 1-5.
9. Hamza Ozer, Bulent Sankur, Nasir Memon. An SVD-based audio watermarking technique. *Proc. of the International Multimedia Conference*, New York, USA, SESSION: Audio, 2005, pp. 51-56.
10. Ben-Shalom A., Werman M., Dubnov S. Improved Low Bit-Rate Audio Compression using Reduced Rank ICA Instead of Psychoscoustic Modeling. *Proc. of the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 5, 2003, pp. 461-464.
11. *Progressive Audio Transmission*. Available at: <https://sites.google.com/site/sviridov86> , accessed 26.09.2012.
12. Dominic Mathew, Devassia V.P., Tessamma Thomas. Compression of Pseudo-periodic Signals Using 2D Wavelet Transform. *American Journal of Signal Processing*, 2011, vol. 1, no. 2, pp. 46-50. DOI: 10.5923/j.ajsp.20110102.08
13. Evangelista G. Pitch-Synchronous Wavelet Representations of Speech and Music Signals. *IEEE Transactions On Signal Processing*, 1993, vol. 41, no. 12, pp. 3313-3330. DOI: <http://dx.doi.org/10.1109/78.258076>

14. Suvorov A.V. *Klinicheskaiia elektrokardiografiia* [Clinical electrocardiography]. Nizhnii Novgorod, NMI Publ., 1993. 124 p.
15. Omer Nezih Gerek, Dogan Gokhan Ece. Compression of Power Quality Event Data using 2D Representation. *Electric Power Systems Research*, 2008, vol. 78, no. 6, pp. 1047-1052. DOI: <http://dx.doi.org/10.1016/j.epsr.2007.08.006>
16. Vinton M.S., Atlas L.E. A Scalable and Progressive Audio Codec. *Proc. of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 5, 2001, pp. 3277- 3280.
17. Sviridov A.A. Ispol'zovanie metodov szhatiia izobrazhenii dlia kompressii potokovogo audio [The use of methods of image compression for compressing streaming audio]. *Sovremennye problemy matematiki, mekhaniki, informatiki: materialy Regional'noi nauchnoi studencheskoi konferentsii* [Modern problems of mathematics, mechanics, computer science: Proc. of the Regional scientific student conference]. Tula, TulSU Publ., 2010, pp. 167-174.
18. Sviridov A.A. Ispol'zovanie singuliarnogo razlozheniia dlia kompressii potokovogo audio [The use of the singular decomposition for compression streaming audio]. *Izvestiia TulGU. Tekhnicheskie nauki* [Bulletin of the TulSU. Technical Sciences], 2012, iss. 3, pp. 175-183.
19. Sviridov A.A. Ispol'zovanie dvumernogo diskretnogo veivlet-preobrazovaniia dlia kompressii audio [The use of two-dimensional discrete wavelet transform for compression of audio]. *Nauchno-tekhnicheskii vestnik Povolzh'ia* [Scientific and technical Bulletin of the Volga region], 2012, no. 4, pp. 186-190.
20. Anderson E., Bai Z., Bischof C., Blackford S., Demmel J., Dongarra J., Du Croz J., Greenbaum A., Hammarling S., McKenney A., Sorensen D. *LAPACK User's Guide*. Third Edition. SIAM, 1999. Available at: <http://www.netlib.org/lapack/lug/node53.html> , accessed 26.09.2012.
21. Tyrtysnikov E.E. *Matrichnyi analiz i lineinaia algebra* [Matrix analysis and linear algebra]. Moscow, FIZMATLIT, 2007. 480 p.
22. Bloomfield P. *Fourier Analysis of Time Series. An Introduction*. Second Edition. A Wiley-Interscience Publication. 2000. 269 p.
23. *AllMusic : Music Search, Recommendations, Videos and Reviews*. Available at: <http://www.allmusic.com> , accessed 26.09.2012.