

**СРАВНЕНИЕ АЛГОРИТМОВ СЖАТИЯ ЗВУКОВОЙ ИНФОРМАЦИИ
АЛГОРИТМОМ ХАФФМАНА И АРИФМЕТИЧЕСКИМ
КОДИРОВАНИЕМ**

**COMPARISON OF AUDIO INFORMATION COMPRESSION
ALGORITHMS BY HUFFMAN AND ARITHMETIC CODING**

СЕРГЕЕВ ИГОРЬ СЕРГЕЕВИЧ,

аспирант,

Московский авиационный институт.

БАЛАКИРЕВ НИКОЛАЙ ЕВГЕНЬЕВИЧ,

кандидат технических наук, доцент,

Московский авиационный институт.

SERGEEV IGOR SERGEEVICH,

graduate student,

Moscow Aviation Institute.

BALAKIREV NIKOLAI EVGENIEVICH,

candidate of technical sciences, associate professor,

Moscow Aviation Institute.

Данная работа посвящена сравнению алгоритмов сжатия без потерь (алгоритм Хаффмана и Арифметическое кодирование) с целью выбора оптимального варианта для уплотнения звуковой информации. Указаны и охарактеризованы основные группы алгоритмов сжатия без потерь. Подробно рассмотрены алгоритмы сжатия аудиоданных из второй группы. Описано содержание арифметического кодирования и перечислены основные его шаги. Представлены результаты сравнения работы алгоритма Хаффмана и арифметического кодирования. Проведенный эксперимент по сравнению результатов работы алгоритма Хаффмана и арифметического кодирования с помощью разработанного консольного приложения показал преимущество арифметического кодирования.

This work is devoted to comparing lossless compression algorithms (Huffman algorithm and Arithmetic coding) in order to choose the optimal option for compacting audio information. The main groups of lossless compression algorithms are specified and characterized. The algorithms of compression of audio data from the second group are considered in detail. The content of arithmetic coding is described and its main steps are listed. The results of comparing the work of the Huffman algorithm and arithmetic coding are presented. The conducted experiment comparing the results of the Huffman algorithm and arithmetic coding using the developed console application showed the advantage of arithmetic coding.

Ключевые слова: *сжатие данных без потерь, кодирование Хаффмана, арифметическое кодирование, алгоритм, акустическая информация.*

Key words: *lossless data compression, Huffman coding, arithmetic coding, algorithm, acoustic information.*

В век цифровой трансформации и больших темпов накопления различного рода данных, в том числе аудиоинформации, становятся актуальными проблемы экономии памяти хранения этой информации и низкая пропускная способность каналов передачи цифровой информации на расстояние. Применение сжатия аудиоданных является одним из эффективных способов решения представленных проблем [4, с. 268]. В настоящее время наблюдается возрастание количества практических исследований, посвященных данной теме, постоянная разработка и публикация различных стандартов в этой области [2, с. 32].

Информация сегодня является основой современной жизни человека, и её увеличение требует всё больших ресурсов для своего хранения. Увеличение объемов информации, получаемых и отправляемых с помощью сети Интернет, приводят к увеличению затрат, необходимых для хранения и транспортировки информации. С появлением множества различных мультимедийных форматов информации, в том числе аудио, появилось множество специализированных алгоритмов с высокой степенью сжатия определенных типов файлов.

В области систем связи и радиовещания существует проблема ограниченности возможности выбора свободных частот диапазонов, поэтому возникает необходимость понижения скорости цифровых потоков данных (аудио и видео) без снижения субъективного качества воспроизведения. Для повышения качества воспроизведения звука при цифровой звукозаписи требуется повышение частоты дискретизации и числа разрядов без увеличения размеров носителя и сокращения времени записи [1]. Эти и другие задачи обуславливают потребность создания алгоритмов сжатия аудиоданных, основанных на различных моделях слухового восприятия информации.

В данной статье рассмотрим три алгоритма сжатия акустической информации (алгоритм Хаффмана, алгоритм Шеннона-Фано и арифметическое кодирование), приведем пример сравнения работы изучаемых алгоритмов с помощью разработанного консольного приложения, которое основано на данных алгоритмах сжатия аудиофайлов.

В настоящее время все алгоритмы сжатия данных представлены двумя видами: сжатие без потерь (обратимое искажение, алгоритмы частотного анализа) и сжатие с потерями (необратимое искажение, алгоритмы корреляционного анализа). В обоих случаях речь идет о передаче битового потока, т.е. создаваемый файл имеет бинарную структуру [5, с. 98].

При сжатии без потерь конечные файлы абсолютно идентичны исходным данным. Примечательно, что степень сжатия в этом случае крайне мала, и сильно зависит от содержимого файла, что не позволяет передавать по каналам сети Интернет файлы, сжатые без потерь. При этом примерный объем одной минуты записи составляет 30-40 Мб.

При сжатии с потерями происходит удаление из исходного аудио сигнала некоторых составляющих, которые не воспринимаются человеческим ухом (психоакустический принцип) и сужение кодируемой полосы частот. Сегодня большинство известных форматов сжатия аудио (MP3, WMA, OGG и др.) базируются на данном подходе. В данном случае объем одной минуты записи может составлять от 2 до 20 МБ. [5, с. 99].

Сегодня существует большое количество алгоритмов сжатия без потерь, их также можно разделить на две группы:

1. Поточные и словарные алгоритмы. Особенность данных алгоритмов состоит в том, что при кодировании используется информация о последовательностях, встречавшихся ранее, а не информация о частотах символов в сообщении.

2. Алгоритмы статистического (энтропийного) сжатия. Особенностью данной группы алгоритмов является то, что при сжатии информации используется неравномерность частот, с которыми различные символы встречаются в сообщении.

Более подробно рассмотрим алгоритмы сжатия аудиоданных, относящихся ко второй группе.

Алгоритм Шеннона-Фано является одним из первых алгоритмов сжатия, сформулированный американскими учёными Шенноном и Фано независимо друг от друга. Данный алгоритм, основанный на частоте повторения, имеет следующие шаги:

1. Выстраивание частоты встречи символа (вероятность встречи).
2. Сортировка списка символов в порядке убывания вероятности, наиболее вероятные слева и наименее вероятные справа.
3. Разделение списка на две части, с полной вероятностью того, что обе части будут как можно ближе друг к другу.
4. Присвоение значения 0 левой части и 1 - правой части.
5. Повторение шагов 3 и 4 для каждой части, пока все символы не будут разделены на отдельные подгруппы.

Представленный алгоритм основывается на построении таблицы вероятностей частоты встречи символа, т.е. на самом первом шаге мы создаём таблицу из символов и сортируем их в порядке убывания. Далее мы строим дерево по этой таблице и присваиваем каждому символу свой код.

Алгоритм Хаффмана был предложен Д. А. Хаффманом в 1952 году, в основе которого положены двоичные кодирующие деревья. Данный алгоритм характеризуется однозначным построением кода и имеет следующие шаги, схожие с шагами алгоритма Шеннона-Фано:

1. Выстраивание таблицы частоты встречи символа (вероятность встречи).
2. Сортировка списка символов в порядке убывания вероятности, наиболее вероятные слева и наименее вероятные справа.
3. Объединение двух символов с наименьшими вероятностями (т.е. сумма вероятностей). Символу с меньшей вероятностью присваивается 0, символу с большей вероятностью – 1.
4. Переход к следующим двум наименее вероятным «узлам» и таким же образом объединение всех последующих «узлов», пока не получится один «узел» наверху.

Арифметическое кодирование является усовершенствованием алгоритма Хаффмана, позволяющим получать оптимальные коды для данных, где частоты появления символов не являются степенью двойки (2^n). В основе идеи данного алгоритма лежит присваивание кодов не отдельным символам, а их последовательностям. При арифметическом кодировании ставится в соответствие символу вещественное число из отрезка $[0;1]$. Шаги арифметического кодирования:

1. Приведение отрезка в соответствие каждому символу, длина отрезка равна частоте его появления.
2. Считывание символа из входного потока и рассмотрение отрезка, соответствующего этому символу. Разделение этого отрезка на части, пропорциональные частотам встречаемости символов.
3. Повторение шага 2 до конца входного потока.
4. Выбор любого числа из получившегося отрезка, которое и будет являться результатом арифметического кодирования.

Арифметическое кодирование показывает очень хороший результат даже при низкой энтропии источника сообщения. В отличие от предыдущих алгоритмов сжатия, этот алгоритм кодирует не один символ, а сразу некую последовательность символов, присваивая ей отрезки (интервалы).

Представленные алгоритмы сжатия позволяют закодировать и оптимизировать код таким образом, чтобы не потерять ничего, например, выделить вместо равномерного кода по 8 бит на символ неравномерный с меньшим количеством бит.

Таким образом, способы сжатия информации без потерь предполагают наличие избыточных данных, например, повторяющихся символов. Эти алгоритмы очень удобны, они

применяются и в настоящее время, поскольку в некоторых условиях нельзя терять информацию. Данные алгоритмы позволяют полностью восстановить исходные данные.

Среди недостатков рассматриваемых алгоритмов можно выделить то, что приходится хранить информацию для восстановления исходных данных, которая уже занимает место. Кроме этого, объём информации уменьшается не на много. Например, рассмотрим самый известный звуковой формат mp3, который выстраивается напрямую из оригинального звукового файла wav.

Во-первых, здесь могут быть отделены те частоты звука, которые человеческое ухо не воспринимает. Во-вторых, может быть сокращено количество возможных битов в секунду (битрейт), т.е. чем больше бит за секунду времени замеряется, тем качественнее звук, а в формате mp3 максимально укорачивается битрейт до 320 кбит/секунду и, соответственно, качество звука становится немного хуже.

Для сравнения работы изучаемых алгоритмов сжатия аудиоданных нами была разработана компьютерная программа, которая проводит сжатие звукового файла в формате WAV и для сравнения позволяет учесть: размер файла до и после сжатия; коэффициент сжатия (исходный файл/сжатый файл); время сжатия и восстановления [3, с. 279]. Например, при сравнении работы алгоритма Хаффмана и арифметического кодирования нами были получены следующие данные (табл. 1):

Таблица 1. Результаты сравнения работы алгоритма Хаффмана и арифметического кодирования.

Характеристика	Алгоритм Хаффмана	Арифметическое кодирование
Исходный размер файла (Кб)	2933	2933
	391,06	391,06
	46830	46830
Размер после сжатия (Кб)	2409	2398
	205,228	198
	37983	37788
Коэффициент сжатия	1,217	1,22
	1,905	1,975
	1,232	1,239
Время сжатия (сек)	1,9	0,7
	4,28	0,8
	57	19
Время восстановления (сек)	24,1082	16,239
	4,88	4,80
	229,3389	238

Данный пример показывает, что при сжатии исходных аудиофайлов одинакового размера арифметическое кодирование предоставляет лучшие показатели в коэффициенте сжатия, размере и времени, затрачиваемом на сжатие и восстановление файла. Исключением явилось затраченное время восстановления звукового файла, размер которого 46830 кб, где сжатие с применением алгоритма Хаффмана показало результат восстановления меньше на 9 секунд.

Также разработанное нами консольное приложение даёт возможность сжатия аудиофайлов при использовании всех рассмотренных алгоритмов. Кроме этого, приложение

даёт возможность не только кодировать (сжимать), но и декодировать (расжимать) исходные файлы.

Таким образом, преимуществом алгоритмов сжатия с потерями в сравнении с алгоритмами сжатия без потерь является то, что первые предоставляют возможность большей степени сжатия информации, продолжая удовлетворять установленным требованиям, а именно - искажения должны быть в допустимых пределах чувствительности человеческих органов слуха. Однако при использовании сжатия с потерями необходимо учитывать, что повторное сжатие обычно приводит к ухудшению качества информации.

Нами рассмотрены три известных алгоритма сжатия аудиоданных: алгоритм Шеннона-Фано, алгоритм Хаффмана, арифметическое кодирование. Алгоритм Шеннона-Фано, использует алгоритм генерации кодов и не всегда дает оптимальные коды (оптимальный код – код дающий наибольшее сжатие данных из всех возможных для данного типа преобразования). Алгоритм Хаффмана, сходен с алгоритмом Шеннона-Фано и представляет собой замену кода равной длины для символов на коды неравной длины в соответствии с частотой появления символов в данных, коды минимальной длины присваиваются наиболее часто встречающимся символам. Арифметическое кодирование представляет собой усовершенствование алгоритма Хаффмана, позволяющее получать оптимальные коды для данных, где частоты появления символов не являются степенью двойки (2^n). Данный алгоритм и алгоритм Хаффмана являются энтропийными, т.е. длина кода конкретного символа зависит от частоты встречаемости этого символа в данных.

Проведенный эксперимент по сравнению результатов работы алгоритма Хаффмана и арифметического кодирования с помощью разработанного консольного приложения показал преимущество арифметического кодирования, предоставившего более высокие результаты сжатия в сравнении с результатами сжатия с использованием алгоритма Хаффмана. Выявлено, что арифметическое кодирование имеет возможность кодирования каждого символа нецелым числом бит, что эффективнее кода Хаффмана.

СПИСОК ЛИТЕРАТУРЫ

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений // Пер. с англ. М.: Техносфера, 2006. 1072 с.
2. Каргин Р.И., Стаценко Л. Г. Форматы сжатия аудиоданных. Анализ и сравнение // Известия СПбГЭТУ «ЛЭТИ». 2019. №9. С. 31-37.
3. Сергеев И.С. Сравнение алгоритмов сжатия звуковой информации алгоритмом Хаффмана и Арифметическим кодированием // Сборник тезисов работ международной молодёжной научной конференции XLVIII Гагаринские чтения. М.: Издательство «Перо», 2022. С. 278-279.
4. Тазетдинова В.А. Цифровая компрессия аудиоданных // Молодой ученый. 2014. № 4 (63). С. 268-273.
5. Чижов И.И. О новом методе сжатия музыкальных файлов // Научные ведомости БелГУ. Сер. История. Политология. Экономика. Информатика. 2008. №17. Вып.8. С. 97-101.

© Сергеев И.С., Балакирев Н.Е., 2022.