Минеев Р.Р.
Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
Email: minirorus7@gmail.com

Исследование тенденций в алгоритмах сжатия данных без потерь

Сжатие без потерь — это метод сжатия, при котором не теряются данные в процессе сжатия. Сжатие без потерь «упаковывает» данные в файл меньшего размера, используя своего рода внутреннее сокращение для обозначения избыточных данных. Например, если размер исходного файла составляет 1,5 МБ, сжатие без потерь может уменьшить его примерно до половины этого размера в зависимости от типа сжимаемого файла. Это делает методы сжатия без потерь удобным для передачи файлов по сети, так как файлы меньшего размера передаются быстрее. Сжатие без потерь также удобно для хранения файлов, поскольку они занимают меньший объем [1].

При дальнейшем развитии алгоритмов сжатия данных стоит несколько задач:

1. Необходимо представить входной поток данных в виде, более удобном для кодирования существующими алгоритмами.

2. Закодировать входной поток таким образом, что выходной поток будет занимать места меньше и сможет декодироваться обратно в исходную строку.

3. Алгоритм должен быть производительным и выполнять операции кодирования и декодирования быстро [2].

Обзор существующих алгоритмов и анализ принципов работы данных алгоритмов:

1. Преобразование Барроуза — Уилера (Burrows-Wheeler transform, BWT, историческое называние - блочно-сортирующие сжатие [3]). Данный алгоритм меняет порядок символов во входной строке таким образом, что повторяющиеся подстроки образуют на выходе идущие подряд последовательности одинаковых символов.

2. LZSS (Lempel-Ziv-Storer-Szymanski) [4] - алгоритм, являющийся модификацией стандартного метода LZ77. Основная разница между исходным LZ77 и LZSS состоит в том, что в методе LZ77 запись ссылки на словарь может быть длиннее, чем строка, которую она замещает (то есть запись такой ссылки делает сжатый фрагмент длиннее, чем несжатый). В методе LZSS подобные ссылки опускаются в случае, если длина строки меньше некоторой настройки («break even»). LZSS применяет однобитный флаг для обозначения того, является ли следующий фрагмент сжатого потока литералом (байтом) или ссылкой в словарь (парой значений длина и смещение) [5].

3. Алгоритм LZRW1 (Lempel–Ziv Ross Williams) [6] является модификацией LZSS и разработан с целью обеспечения максимальной скорости компрессии и декомпрессии. Степень сжатия LZRW1 равна примерно 1,5-2.

4. В ноябре 2021 года мировому сообществу в открытом виде был представлен новый алгоритм сжатия изображений без потерь. Quite OK Image Format [7] (QOI) – это алгоритм компрессии RGB и RGBA изображений, по размеру результирующего изображения близок к PNG, но в скорости превосходит его в 20-50 раз для компрессии и в 3-4 раза для декомпрессии [8].

5. Finite State Entropy (FSE) [9] – алгоритм энтропийного кодирования, похожий на алгоритм Хаффмана, и на арифметическое кодирование. При этом он взял лучшее от них обоих: работает так же быстро, как кодирование Хаффмана, и со степенью сжатия как у арифметического кодирования. FSE принадлежит семейству кодеков ANS.

6. Метод rANS [10] также, как и FSE, является членом семейства кодеков ANS. Он позволяет достичь практически оптимального сжатия при очень высокой скорости работы. В этом rANS ничем не уступает FSE, так как оба алгоритма построены на общей теоретической базе. Однако алгоритм rANS значительно проще в реализации, чем FSE.

Подводя итоги, можно сделать выводы о производительности и необходимости использования данных подходов при реализации алгоритмов сжатия данных без потерь. Для повышения степени сжатия можно использовать алгоритм преобразования входного потока, чтобы увеличить возможность более сильного сжатия путем сохранения повторяющихся блоков данных. С данной задачей хорошо справляется алгоритм BWT. Для повышения скорости сжатия в алгоритмах LZRW1 и QOI используется механизм запоминания блоков данных в hash-таблицы по различным формулам hash-функций, а в алгоритмах rANS и FSE (так же, как и в одном из условий QOI) используется запись блока в виде его смещения от предыдущих блоков данных и нет необходимости записывать данные целиком.

Литература

1. Recommendation T.45 (02/00): Run-length colour encoding. International Telecommunication Union. 2000.

2. Jacob Ziv, Abraham Lempel. A Universal Algorithm for Sequential Data Compression IEEE Transactions on Information Theory, May 1977. - pp. 337—343.

3. Huffman, D. «A Method for the Construction of Minimum-Redundancy Codes». Proceedings of the IRE. 40 (9), 1952. - 1098–1101c. doi:10.1109/JRPROC.1952.273898.

4. J. Duda, K. Tahboub, N. J. Gadil, E. J. Delp, The use of asymmetric numeral systems as an accurate replacement for Huffman coding, Picture Coding Symposium, 2015.

5. Bell T.C. 1986. Better OPM/L test compression. IEEE Trans. Commun. COM-34. 12, 1176-1182.

6. Williams, R.N., «An Extremely Fast Ziv-Lempel Data Compression Algorithm», Data Compression Conference 1998 (DCC'98), 8–11 April 1998, Snowbird, Utah, pp.362-371

7. Dominic Szablewski, The Quite OK Image format, Specification Version 1.0, 2022.01.05.

8. Witten, Ian H.; Neal, Radford M.; Cleary, John G. «Arithmetic Coding for Data Compression» (PDF). Communications of the ACM. 30 (6): 1987. - 520–540. doi:10.1145/214762.214771

9. http://fastcompression.blogspot.com/2013/12/finite-state-entropy-new-breed-of.html [электронный ресурс] Дата обращения: 02.03.2022

10. https://fgiesen.wordpress.com/2015/12/21/rans-in-practice/ [Электронный ресурс] Дата обращения: 02.03.2022