



4. Хасанов З.М., Хасанов О.З. К синтезу адаптивных САУ многосвязными электроприводами с моделью эталоном и с беспойсковым алгоритмом идентификации. // Вестник УГАТУ, Том 21, №. 4, 2017. С. 163-172.

Я.В. Чесноков, М.А. Кудрина

ИЗУЧЕНИЕ ВЛИЯНИЯ ПАРАМЕТРОВ АЛГОРИТМА СЖАТИЯ LZSS НА ЕГО ХАРАКТЕРИСТИКИ

(Самарский университет)

В настоящее время алгоритмы сжатия используются повсеместно при передаче и хранении информации. Разновидностью алгоритмов сжатия без потерь являются словарные алгоритмы сжатия. Метод сжатия с использованием словаря — разбиение данных на слова и замена их на индексы в словаре. В настоящее время это наиболее распространенный подход для сжатия данных [1]. К словарным алгоритмам относятся алгоритмы LZ77, LZ78, LZSS, LZW, LZR, Deflate и т.д. Рассматриваемый алгоритм сжатия данных LZSS (Lempel-Ziv-Storer-Szymanski) — это алгоритм сжатия данных без потерь, основанный на принципах алгоритма сжатия данных LZ77 [2].

Была поставлена задача разработать программный автоматизированный комплекс изучения алгоритмов сжатия данных, позволяющий оценить характеристики алгоритмов сжатия данных (степень сжатия, быстродействие, объём потребляемой памяти), а также строить диаграммы зависимостей этих характеристик от различных параметров алгоритмов, размеров файлов. Рассмотрен алгоритм LZSS и влияние варьирования его параметров (размер словаря, размер буфера) на характеристики, такие как степень сжатия, время сжатия и объём потребляемой оперативной памяти.

Для исследования было взято 30 текстовых файлов формата .txt. Выбор формата текстовых файлов .txt обусловлен тем, что словарные алгоритмы более всего подходят для сжатия текстовой информации. Размер используемых файлов варьируется от 47 килобайт до 7 мегабайт, но при проведении исследований эти файлы разбиваются на части равного размера, задаваемого в программе, и именно с ними проводятся эксперименты.

На рисунке 1 приведён интерфейс разработанной программной системы. Выбрав из выпадающего списка алгоритм LZSS, задав необходимые параметры в виде списков, и указав папку с файлами, мы можем провести исследование, в результате которого станут известны степень сжатия, количество использованной в ходе сжатия данных оперативной памяти компьютера, и время, затраченное на сжатие данных. Результаты каждого индивидуального эксперимента приводятся в поле «Результаты», а также сохраняются в текстовый файл на жёстком диске. Усреднённые результаты для каждой пары «Размер словаря — Размер буфера» приводятся в виде графиков и гистограмм в программе, а также автоматически записываются в excel таблицу.

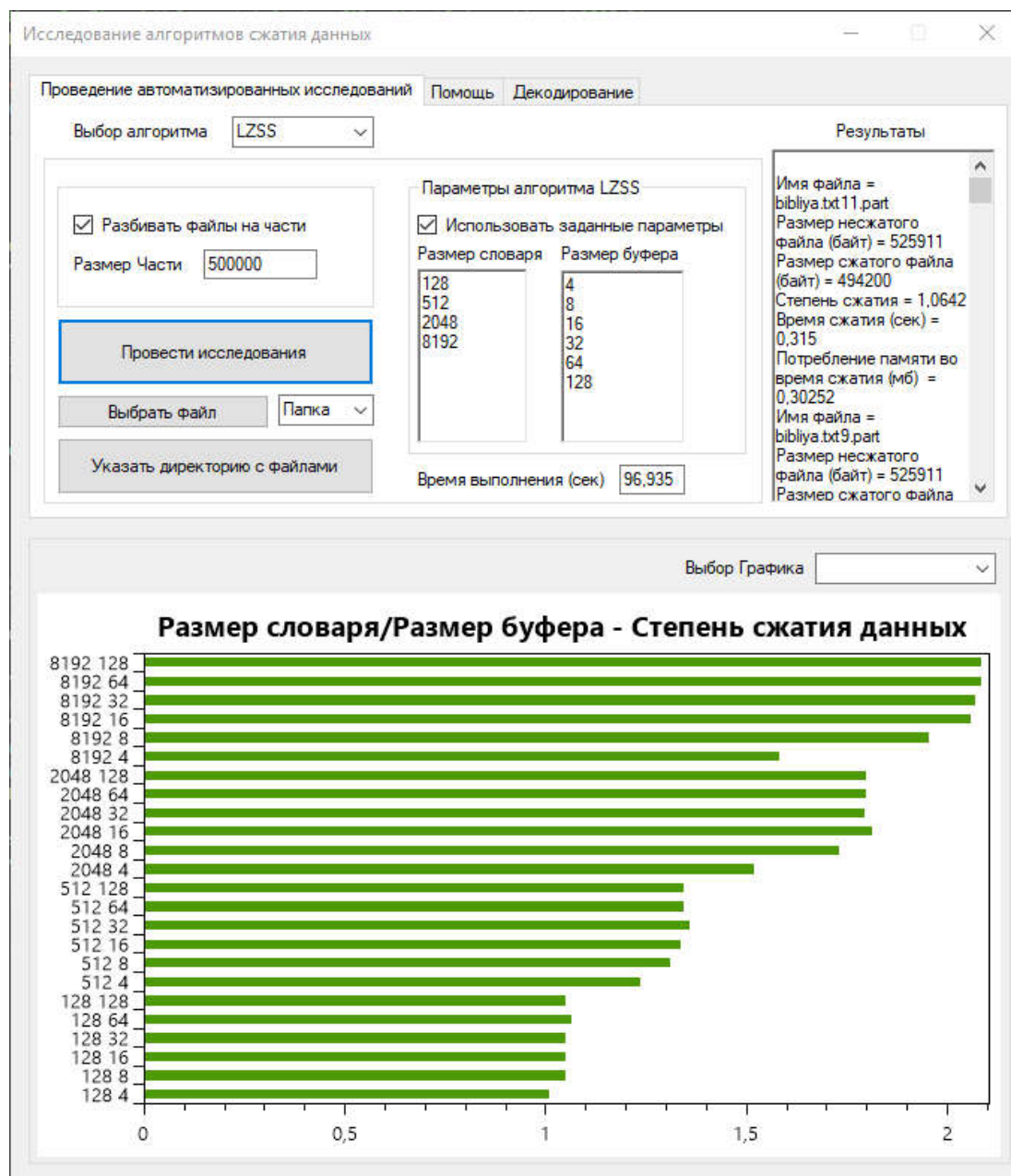


Рисунок 1 – Интерфейс разработанной программной системы

С помощью разработанной системы было проведено исследование со следующими исходными данными:

- рассматриваемые характеристики алгоритма: степень сжатия, время сжатия, объём затраченной оперативной памяти;
- размер файлов: 500000 байт;
- количество файлов: 30;
- рассматриваемые размеры словаря: 64 байт, 128 байт, 256 байт, 512 байт, 1024 байт, 2048 байт, 4096 байт, 8192 байт, 16384 байт, 32768 байт, 65536 байт;
- рассматриваемые размеры буфера: 4 байт, 8 байт, 16 байт, 32 байт, 64 байт, 128 байт, 256 байт, 512 байт.

На рисунках 2, 3 и 4 приведены диаграммы с результатами исследований.

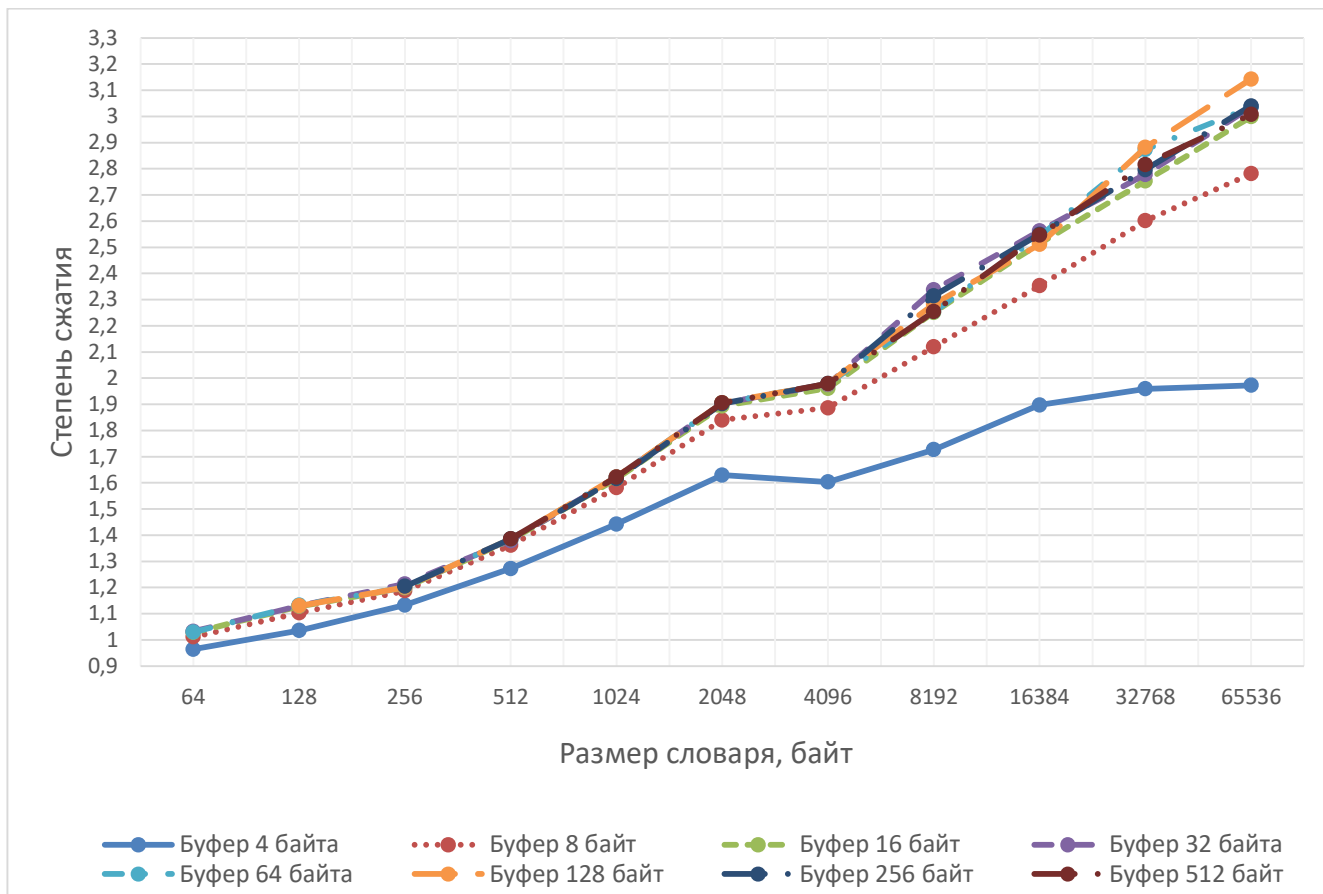


Рисунок 2 – Диаграмма зависимости степени сжатия данных от размера словаря при различных значениях буфера

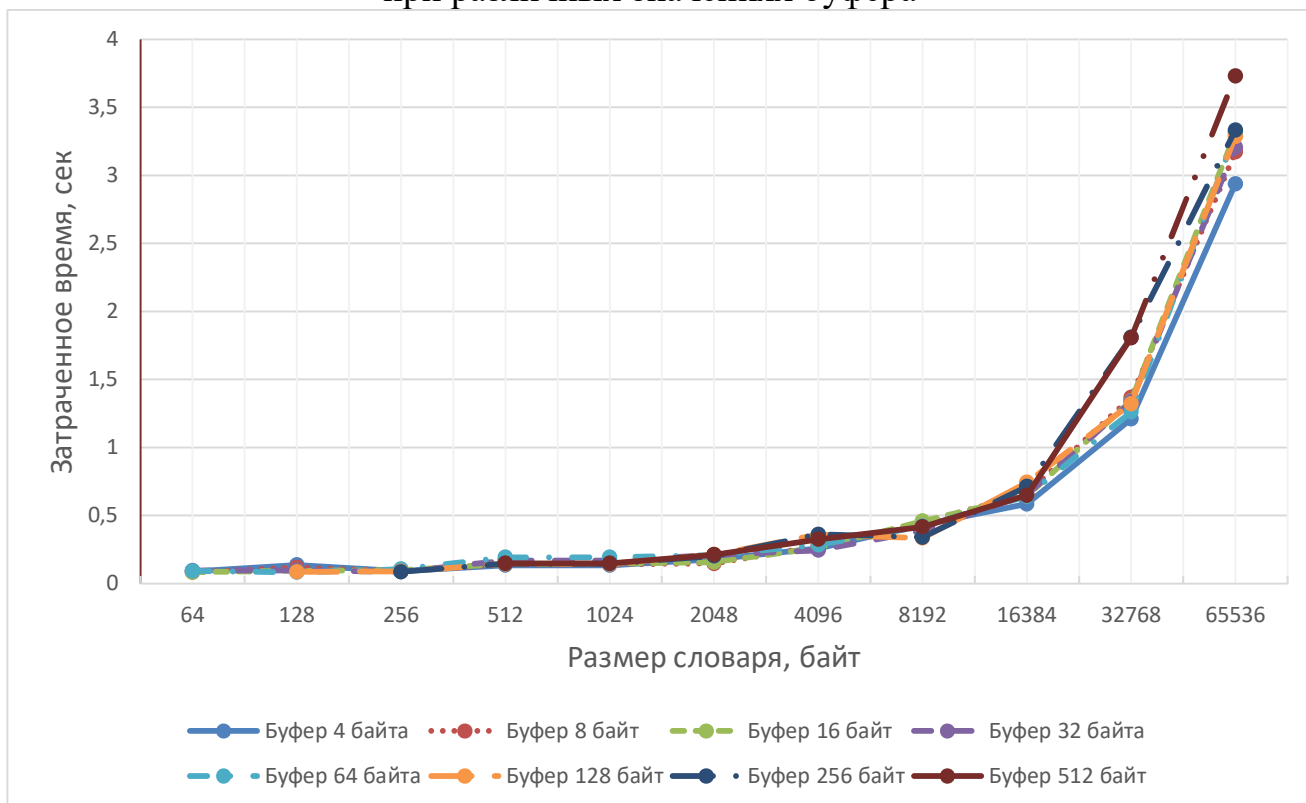


Рисунок 3 – Диаграмма зависимости времени, затраченного на сжатие данных от размера словаря при различных значениях буфера

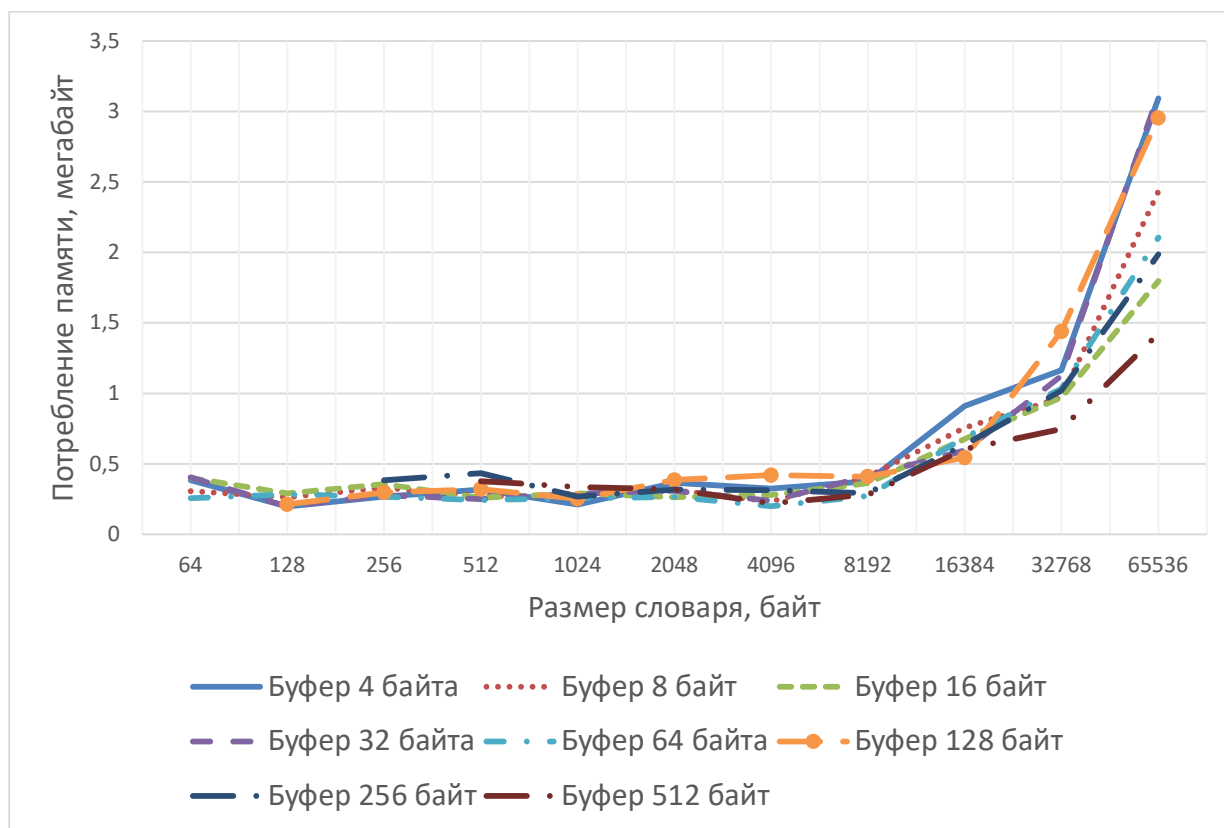


Рисунок 4 – Диаграмма зависимости объёма оперативной памяти, потребляемой во время сжатия данных от размера словаря при различных значениях буфераасбапов

На основе полученных результатов, были сделаны следующие выводы:

- степень сжатия данных в алгоритме LZSS напрямую зависит от размера словаря, зависимость имеет линейный характер;
- при малых размерах буфера (4 байта, 8 байт) алгоритм работает значительно менее эффективно, при размере словаря 64 и буфера 4 показывая степень сжатия менее единицы. В связи с этим, рекомендуется всегда устанавливать размер буфера не менее 16 байт;
- увеличение буфера больше 16 байт не гарантирует увеличения степени сжатия;
- время, затраченное алгоритмом LZSS на сжатие данных, напрямую зависит от размера словаря;
- размер буфера не оказывает существенного влияния на время, затрачиваемое алгоритмом LZSS на сжатие данных;
- объём оперативной памяти, потребляемой алгоритмом LZSS, напрямую зависит от размера словаря;
- размер буфера не оказывает существенного влияния на объём потребляемой памяти;
- учитывая, что объём потребляемых ресурсов ЭВМ(оперативной памяти и процессорного времени) при размерах словаря от 64 байт до 8192 байт растёт незначительно, рекомендуется использовать размеры словаря в 4096 или 8192 байт в тех ситуациях, когда используется процессорное время и потребляемая



оперативная память являются критичными ресурсами, а местом на физическом носителе можно пожертвовать. Степень сжатия, достигаемая в таком случае (около двух) вполне приемлема;

– если объём потребляемой алгоритмом во время работы оперативной памяти и процессорного времени не являются критичными, а важно максимальное сжатие данных, рекомендуется использовать размеры словаря более 16384 байт. Это позволит существенно увеличить степень сжатия данных.

Литература

1 Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. – М.: ДИАЛОГ-МИФИ, 2003. – 348с.

2 Д.Сэломон. Сжатие данных, изображений и звука - Москва: Техносфера, 2004. - 368с.

И.В. Чеховских, Е.В. Симонова

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПОИСКА СХОДСТВА ИССЛЕДОВАНИЙ В НАУЧНЫХ СТАТЬЯХ

(Самарский национальный исследовательский университет
имени академика С.П. Королёва)

Научная статья – это произведение, отражающее результаты исследовательской деятельности автора (авторов). Информация излагается четко, конкретно, детально [1].

В век информационных технологий исследователь может получить доступ к огромному количеству научных статей. Проблемой является поиск действительно полезной информации во множестве доступных статей, находящихся в интернете, для подготовки к различным конференциям или для написания собственных исследовательских работ.

Как правило, при поиске научных статей используют два основных подхода: по автору, если необходимо ознакомиться с полным списком исследований конкретного автора, или по ключевым словам, которые могут как содержаться, так и отсутствовать в названии статьи. Искать научные статьи двумя этими способами можно на большинстве сайтов. Но, к сожалению, данные методы поиска не дают полного представления об исследовании. Необходимо выполнять более продвинутый поиск научных статей на основании сходства текстового содержания научных исследований.

Для поиска похожих научных статей на основании контекста самой статьи необходимо первоначально выполнить канонизацию текста исследования – приведение оригинального текста к единой нормальной форме. Канонизация текста статьи выполняется с помощью Natural Language Processing (NLP).