Московский Государственный Университет

им. М.В. Ломоносова
Факультет Вычислительной Математики и Кибернетики.

Практическое задание. Конечные поля и коды БЧХ

Курс: Прикладная алгебра

Герасимов Денис 323
осень 2018

Постановка задачи.

В задании выдаётся список всех примитивных многочленов степени q над полем F_2 для всех q = 2, . . . , 16. В этом списке каждый многочлен представлен десятичным числом, двоичная запись которого соответствует коэффициентам полинома над F_2, начиная со старшей степени. Для выполнения задания требуется:

1. Реализовать основные операции в поле F_2^q.
2. Реализовать основные операции для работы с многочленами из F_2^q[x].
3. Реализовать процедуру систематического кодирования для циклического кода, заданного своим порождающим многочленом
4. Реализовать процедуру построения порождающего многочлена для БЧХ-кода при заданных n и t.
5. Построить графики зависимости скорости БЧХ-кода r = k/n от количества исправляемых кодом ошибок t для различных значений n.
6. Реализовать процедуру вычисления истинного минимального расстояния циклического кода d, заданного своим порождающим многочленом, путем полного перебора по всем 2 k − 1 кодовым словам. Привести пример БЧХ-кода, для которого истинное минимальное расстояние больше, чем величина 2t + 1
7. Реализовать процедуру декодирования БЧХ-кода с помощью метода PGZ и на основе расширенного алгоритма Евклида. Провести сравнение двух методов декодирования по времени работы.

Анализ выполнения:

В результате выполнения были реализованы все описанные в задании функции, а также некоторые вспомогательные, такие как: вычисление степени полинома, добавление и уничтожение незначащих нулей, тестирующие функции.

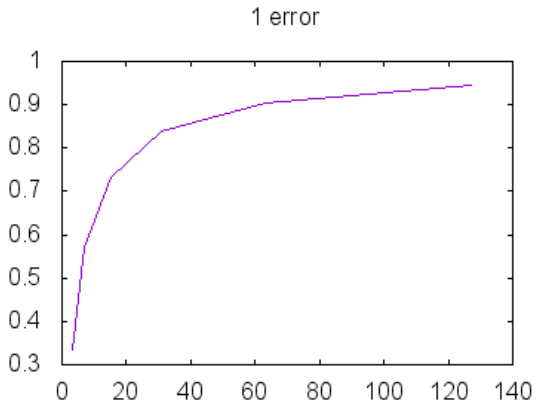
Результаты исследования.

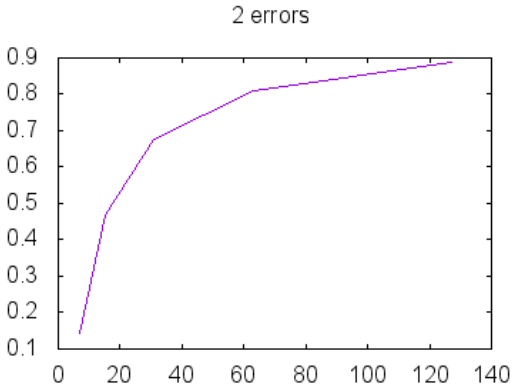
Сравнение двух алгоритмов по времени.

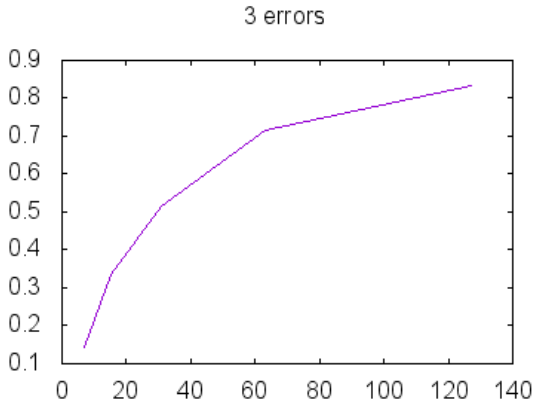
На небольших тестах было выявлено преимущество PGZ-алгоритма. При малом кол-ве возможных ошибок его скорость работы могла превосходить алгоритм Евклида до 10 раз. Но начиная увеличивать длину слов и кол-во возможных ошибок результат совершенно противоположный, что и следовало ожидать, поскольку нам придется перерешивать систему по многу раз.

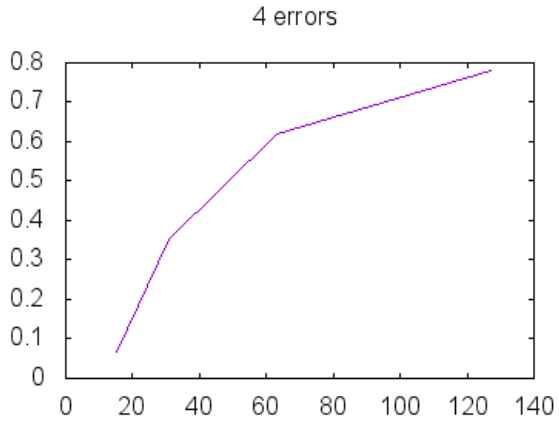
Скорость кода.

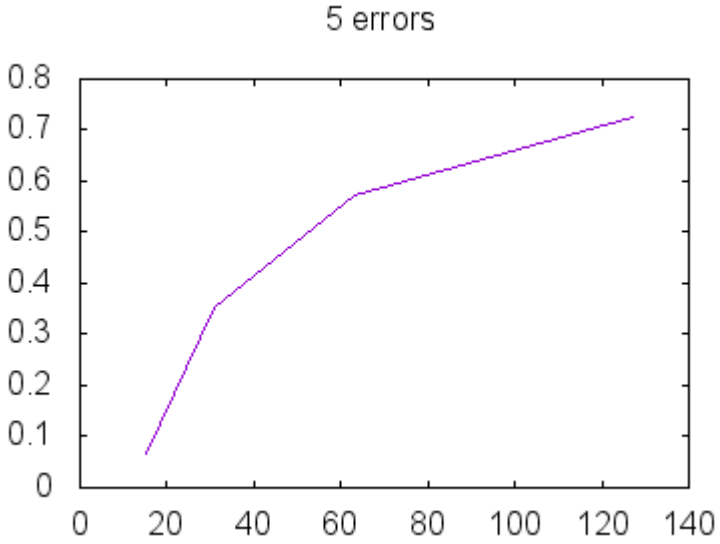
При проведении исследования скорости кода были получены следующие графики, отражающие тот факт, что при любом кол-ве ошибок мы можем сделать кодовую скорость близкой к 1, но при этом n значительно увеличивается. При заданном n естественно выбирать те значение t, при которых кодовая скорость будет максимальной. Это в свою очередь будет означать минимальную избыточность.











Исследование кодов БЧХ.

Для проведения тестирования кодов была реализована функция

testing(times, n = 0, t = 0, met = ‘euclid’)

- times – количество требуемых испытаний
- n, t – параметры создания объекта BCH, и если равны нулю, то генерируются случайно в некотором диапазоне.
- Met – метод декодирования. Допускает параметр ‘both’ означающий, что декодирование будет проводиться 2 раза 2-мя методами. (Использовался для сравнения времени алгоритмов)

Функция создает объект BCH и начинает случайным образом генерировать слово. Затем слово кодируется и принимается за «верный код». После чего этот код подвергается искажению (равномерной мутации). Каждый бит с вероятностью t/n меняет свое значение. Таким образом, в среднем мы должны получать t ошибок. После искажения код декодируется и сравнивается с «верным кодом». Возможно, было бы оптимальнее кодировать и декодировать пачками, но у меня в реализации пачка размером 1. Результат отмечается. Отмечаются так же и случаи, когда кол-во ошибок в передаваемом коде становится выше допустимого. Как показали исследования, именно в таких случаях были возможны случаи отказа или несоответствия декодируемых сообщений. Последнее не означает, что БЧХ-код не может декодировать свыше требуемых t ошибок. В случае, когда истинное минимальное расстояние больше конструктивного, код допускает верное декодирование больше t ошибок. Примерами могут быть коды (31,4) и (63,8) с истинными минимальными расстояниями 11 и 18 соответственно, в то время как конструктивное расстояние у них 9 и 16.

Основные выводы.

В результате исследования мы убедились в том, что БЧХ-код действительно позволяет гарантированно исправить до t ошибок. А так же построили некоторые коды имеющие большее истинное минимальное расстояние большее 2t + 1, которые позволяют правильно декодировать свыше t ошибок.