

Задание 4

И. Герасимов

1 Использование программы

```
python3.8 main.py [-h] [-g -r [R] -n [N] -p [PROBABILITY]]  
[-c -f [FILE] -m [MESSAGE] [-e [ERROR]]] [-d -f [FILE] -y [Y]]
```

1.1 Генерация кода (указан флаг -g)

- **-r** — число проверочных символов;
- **-n** — желаемая максимальная длина блока сообщения, передаваемая по каналу связи;
- **-p** — вероятность ошибки в канале связи для двоичного симметричного канала.

Если указанные параметры не позволяют сформировать код, удовлетворяющий прямой теореме Шеннона, то будет выполняться понижение n , пока не требования не будут соблюдены или не будет исчерпано множество возможных n . Во втором случае работа программы закончиться с выводом того, что невозможно найти подходящие параметры относительно прямой теоремы Шеннона.

Будут созданы 3 файла (в конце каждого файла указывается индекс, чтобы избежать перезаписываний):

- **info** — полная информация о построенном коде;
- **code** — информация для кодера;
- **decode** — информация для декодера.

1.2 Режим кодирования (указан -c)

- **-f** - файл для кодера (например, второй файл **code**, получаемый при генерации кода);
- **-m** - кодируемое сообщение;
- **-e** - ошибка передачи по каналу связи. Если не указана, то генерируется случайная.

Будет выполнен вывод сначала результата кодирования, затем самой ошибки.

1.3 Режим декодирования (указан -d)

- -f - файл для декодера (например, третий файл `decode`, получаемый при генерации кода);
- -y - сообщение из канала связи;

Если получено сообщение с ошибкой меньше веса t определяемого кодом, то будет выведен результат и утверждение с ошибкой и её весом о том, что ошибки декодирования нет.

Иначе выводится сообщение о декодировании через стандартное расположение (лидера смежного класса, полученного по синдрому) и вероятность ошибки декодирования.

2 Описание работы генерации кода

1. Определяется максимальная скорость кодирования и соответствующие k, n через прямую теорему Шеннона;
2. Определяется максимально возможное минимальное расстояние кода d через неравенство Варшамова-Гильберта;
3. Определяется количество исправляемых ошибок t через d ;
4. Вычисляются проверочная и порождающая матрицы H, G в систематическом виде (*sigma* — тождественная подстановка);
5. Строится стандартное расположение;
6. Строится таблица синдромов для векторов веса не больше t , а также таблица синдромов для лидеров смежных классов по стандартному расположению;
7. Вычисляется ошибка декодирования по стандартному расположению. Из-за выбора параметров кода вероятность ошибки декодирования для получаемых из канала сообщений с вектором ошибки веса не больше t равна 0;
8. Формируются файлы, описанные в 1.1.

Замечание: Поскольку в задании рассматривается информация относительно векторов ошибки веса не больше t и стандартное расположение, вообще говоря, можно было бы не строить. Достаточно взять все вектора, веса не больше t , посчитать их синдромы и сохранять только эту таблицу. Однако, в режиме кодирования e может быть любым, поэтому необходимо использовать стандартное расположение, чтобы снизить ошибку декодирования.

3 Описание работы кодера

Выполняется умножение $x = mG$ и накладывается ошибка $y = x \oplus e$.

3.1 формат файла кодера

1. значение r ;
2. значение n ;
3. значение ошибки декодирования;
4. порождающая матрица G ;

4 Описание работы декодера

1. Вычисляется синдром полученного сообщения: $S(y) = S(e) = H^T e$.
2. Выполняется поиск синдрома в таблице синдромов для векторов ошибки веса не больше t ;
3. Для найденного синдрома выполняется проверка, что запись не пуста и содержит только один вектор ошибки. Если это так, то вектором ошибки берется эта запись;
4. Если же условия не выполнены, то реализуется декодирование по стандартному расположению и берется лидер смежного класса, соответствующий этому синдрому;
5. Вычисляется вектор $x = y \oplus e$ и берется его начальная часть по причине того, что используется матрица в систематическом виде.
6. Если в качестве вектора ошибки была взята запись из таблицы синдромов для векторов веса не больше t , то выводится результат и информация о том, что вероятность ошибки равна 0. Иначе выводится результат и вероятность ошибки декодирования.