НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Криптография, основанная на помехоустойчивом кодировании.

Лекция 3. Новые направление в кодовой криптографии

Иванов Ф. И.
к.ф.-м.н., доцент

Национальный исследовательский университет
«Высшая школа экономики»

9 августа 2020 г.

Постановка задачи

- Основное достоинство кодовой криптографии - масштабируемость и стойкость (по крайней мере, пока) к квантовым вычислениям
- Основная проблема - очень длинные ключи, необходимость которых обуславливается требованиями криптографической стойкости
- Как уменьшить длину ключа?
  1. Искать коды, которые можно описать компактно
  2. Учиться декодировать за t хотя бы некоторые множества ошибок
  3. В идеале - хотелось бы построить систему, для которой атака по информационным совокупностям/атака Штерна неприменимы.

Сложность ISD атаки

Вероятность того, что выбранные k индексов свободны от ошибок, равна:

$$P_s = \frac{\binom{n-t}{k}}{\binom{n}{k}} = \frac{\binom{n-k}{t}}{\binom{n}{t}}.$$

Тогда среднее число попыток декодирования τ равно

$$\tau = \frac{1}{P_s} = \frac{\binom{n}{t}}{\binom{n-k}{t}}.$$

Это значение τ можно рассматривать как сложность ISD алгоритма. Как увеличить τ?

- Увеличить n
- Увеличить t

Или же сделать ISD алгоритм вообще непригодным...

Криптосистема - прототип

Обозначим через $\mathbf{M}$ некоторую невырожденную $n \times n$ матрицу над $\mathbb{F}_q$, через $\mathbf{P}$ некоторую $n \times n$ матрицу перестановки и через $\mathbf{G}_0$ матрицу, строки которой являются n кодовыми словами секретного кода C. Мы также предположим, что матрица $\mathbf{G}_0 - \mathbf{P}$ вырождена. Построим 2 матрицы, которые будут являться элементами открытого ключа:

$$\mathbf{G}' = \mathbf{G}\mathbf{M},$$

$$\mathbf{G}'_1 = (\mathbf{G}_0 - \mathbf{P})\mathbf{M}.$$

1. *Шифрование:* Вычислим $\phi(\mathbf{u}) = \mathbf{c} = \mathbf{u}\mathbf{G}' + \mathbf{e}\mathbf{G}'_1$, где $\mathbf{e}$ произвольный вектор над $\mathbb{F}_q$ веса t.
2. *Дешифрование:* Вычислим $\phi^{-1}(\mathbf{c})$ следующим образом:
   - $\mathbf{c}\mathbf{M}^{-1} = \mathbf{u}\mathbf{G} + \mathbf{e}(\mathbf{G}_0 - \mathbf{P}) = \mathbf{u}\mathbf{G} + \mathbf{e}\mathbf{G}_0 - \mathbf{e}\mathbf{P}$. Легко заметить, что $\widetilde{\mathbf{c}} = \mathbf{u}\mathbf{G} + \mathbf{e}\mathbf{G}_0 \in C$ и $wt(\mathbf{e}\mathbf{P}) \leq t$.
   - Декодируем $\mathbf{c}\mathbf{M}^{-1}$, используя полиномиальный алгоритм ξ для кода C: $\xi(\mathbf{c}\mathbf{M}^{-1})$ и получим пару $(\mathbf{e}\mathbf{P}, \mathbf{u}\mathbf{G} + \mathbf{e}\mathbf{G}_0)$.

Вырожденность $\mathbf{G}_0 - \mathbf{P}$

Предположим, что $\mathbf{G}_0 - \mathbf{P}$ невырождена. Тогда Ева может применить следующую атаку:

1. Умножим $\mathbf{c} = \mathbf{u}\mathbf{G}' + \mathbf{e}\mathbf{G}'_1$ на матрицу, обратную к $\mathbf{G}'_1$
2. Декодируем слово $\mathbf{c}' = \mathbf{u}\mathbf{G}(\mathbf{G}_0 - \mathbf{P})^{-1} + \mathbf{e}$ по алгоритму ISD

Теорема

$\mathbf{G}(\mathbf{G}_0 - \mathbf{P})^{-1} = -\left(\mathbf{I} - \mathbf{G}\mathbf{P}^{-1}\mathbf{U}\right)^{-1}\mathbf{G}\mathbf{P}^{-1}$, *где* $\mathbf{G}_0 = \mathbf{U}\mathbf{G}$ *и таким образом коды с порождающими матрицами* $\mathbf{G}$ *и* $\mathbf{G}(\mathbf{G}_0 - \mathbf{P})^{-1}$ *эквивалентны с точностью до перестановки.*

Пример вырожденной $\mathbf{G}_0 - \mathbf{P}$

Пусть $\mathbf{c} = (c_1, \ldots, c_n) \in C$ кодовое слово веса w и пусть $c_{j_1}, \ldots, c_{j_w}$ его w ненулевых координат. Построим строки $\mathbf{c}^i$ матрицы $\mathbf{G}_0$ следующим образом: строки, чей номер не из $J = supp(\mathbf{c})$ будут выбираться случайно, а остальные будут выбираться так, что

$$\sum_{j \in J} \mathbf{c}^j = \mathbf{c}.$$

Обозначим через $\mathbf{B}_n = \mathbf{G}_0 - \mathbf{I}$ и пусть $\mathbf{b}^i = \mathbf{c}^i + \delta_i$ — i-ая строка матрицы $\mathbf{B}_n$, где $\delta_i = (\delta_{i,1}, \ldots, \delta_{i,n})$ и $\delta_{i,j}$ — символ Кронекера. Тогда $\sum_{j \in J} \mathbf{b}^j = \mathbf{0}$, а значит $\mathbf{B}_n = \mathbf{G}_0 - \mathbf{I}$ вырождена.

Синдромная атака

Ева вычисляет проверочную матрицу $\mathbf{H}'$ для порождающей матрицы $\mathbf{G}'$, т.е. $\mathbf{G}'\mathbf{H}'^T = \mathbf{0}$. Пусть $\mathbf{H}$ — проверочная матрица кода C, т.е. $\mathbf{G}\mathbf{H}^T = 0$. Тогда легко проверить, что $\mathbf{H}'^T = \mathbf{M}^{-1}\mathbf{H}^T\mathbf{S}^T$, где $\mathbf{S}$ некоторая невырожденная $r \times r$ матрица и $r = n - k$. После того, как Ева умножит $\mathbf{c}$ справа на $\mathbf{H}'^T$, она получит:

$$\tilde{\mathbf{c}} := \mathbf{c}\mathbf{H}'^T = \mathbf{e}\mathbf{P}\mathbf{H}^T\mathbf{S}^T = \mathbf{e}\tilde{\mathbf{H}}^T. \tag{1}$$

Таким образом, уравнение (1) описывает синдромную атаку на $\tilde{C}$ с проверочной матрицей $\tilde{\mathbf{H}} = \mathbf{S}\mathbf{H}\mathbf{P}^T$. Так как коды C и $\tilde{C}$ эквивалентны с точностью до перестановки, то "прототип"эквивалентен криптосистеме Мак-Элиса.

Модификация "прототипа"

Для того, чтобы усилить "прототип усложним структуру матрицы $\mathbf{G}_1'$.
Пусть $\mathbf{G}_1' = \mathbf{WD}(\mathbf{G}_0 - \mathbf{P})\mathbf{M}$, где $(\mathbf{G}_0 - \mathbf{P})\mathbf{M}$ такие же, как у прототипа, $\mathbf{D}$ произвольная диагональная $n \times n$ матрица с t ненулевыми элементами на диагонали, и $\mathbf{W}$ случайная невырожденная $n \times n$ матрица.
Построим 2 открытые матрицы: $k \times n$ матрицу $\mathbf{G}' = \mathbf{GM}$ и $n \times n$ матрицу $\mathbf{G}_1' = \mathbf{WD}(\mathbf{G}_0 - \mathbf{P})\mathbf{M}$.

Шифрование/дешифрование

1. *Шифрование:* Вычислим $\phi(\mathbf{u}) = \mathbf{c} = \mathbf{uG}' + \mathbf{eG}_1'$, где $\mathbf{e}$ произвольный q-ичный вектор произвольного веса.
2. *Дешифрование:* Вычислим $\phi^{-1}(\mathbf{c})$ следующим образом:
   - $\mathbf{cM}^{-1} = \mathbf{uG} + \mathbf{eWD}(\mathbf{G}_0 - \mathbf{P}) = \mathbf{uG} + \mathbf{eWDG}_0 - \mathbf{eWDP}$. Легко заметить, что $\widetilde{\mathbf{c}} = \mathbf{uG} + \mathbf{eWDG}_0 \in C$ and $wt(\mathbf{eWDP}) \leq t$.
   - Декодируем $\mathbf{cM}^{-1}$, используя декодер ξ секретного кода C: $\xi(\mathbf{cM}^{-1})$ и получим пару $(\mathbf{eWDP}, \mathbf{uG} + \mathbf{eWDG}_0)$.
   - Вычислим $\mathbf{e}' = \mathbf{eWDPP}^{-1} = \mathbf{eWD}$.
   - Восстановим $\mathbf{u}$ из $\mathbf{c} + \mathbf{e}'\mathbf{P} = \mathbf{uG} + \mathbf{e}'\mathbf{G}_0$ по известным $\mathbf{eWD} = \mathbf{e}'$, $\mathbf{G}_0$, $\mathbf{G}$ и $\mathbf{P}$.

Синдромная атака

Ева умножает обе части $\mathbf{c} = \mathbf{uG}' + \mathbf{eG}'_1$ справа на $\mathbf{H}^T = \mathbf{M}^{-1}\tilde{\mathbf{H}}^T$, где $\tilde{\mathbf{H}}$ — проверочная матрица кода C. Она получит следующее уравнение:

$$\mathbf{cH}^T = \mathbf{eWDP}\tilde{\mathbf{H}}^T,$$

которое является синдромным уравнением для $\tilde{H}$, а значит Ева может найти вектор $\mathbf{eWDP}$ веса t, но она не сможет "восстановить"вектор $\mathbf{e}$, так как множители $\mathbf{W}$, $\mathbf{D}$ и $\mathbf{P}$ для нее неизвестны.

Атака, основанная на вырожденности $\mathbf{G}_1'$

Так как $rank\,(\mathbf{G}_1') = t$, то существует q^t различных значений $\mathbf{e}\mathbf{G}_1'$ и все эти векторы образуют образ $\mathcal{I}m\,(\mathbf{G}_1')$. Таким образом, вместо перебора всех возможных $\mathbf{e}$ из $\mathbb{F}_q^n$ можно рассмотреть только элементы из $\mathcal{I}m\,(\mathbf{G}_1')$, которых q^t.

Теорема

Для любого $\mathbf{e} \in \mathbb{F}_{\mathbf{q}}^{\mathbf{n}}$ *существует единственное решение* $\mathbf{x}'$*, такое что* $\mathbf{c} - \mathbf{x}' \in C$ *и это решение лежит в* $\mathcal{I}m\,(\mathbf{G}_1')$

Таким образом, Ева может восстановить текст $\mathbf{u}$, используя не более чем q^t попыток.

Некоторые замечания

- Основным достоинством "улучшенной"криптосистемы является то, что вектор $\mathbf{e}$ произвольный, но основной недостаток состоит в том, что Ева может найти $\mathbf{u}$ из $\mathbf{c}$ за не более чем q^t раундов.
- Этот недостаток проистекает из того, что матрица $\mathbf{G}_1'$ имеет ранг не более чем t.
- Есть ли возможность увеличить ранг матрицы $\mathbf{G}_1'$, оставив при этом вектор $\mathbf{e}$ произвольным?

Улучшение криптосистемы

Покажем, как модифицировать $\mathbf{G}_1'$, чтобы получить ранг $n-k$. Для этого рассмотрим следующую матрицу $\mathbf{G}_2' = \mathbf{Q}\left(\mathbf{G}_0 + \mathbf{T}\right)\mathbf{M}$, где $\mathbf{Q}$ и $\mathbf{T}$ являются $n \times n$ матрицами, $\mathbf{T}, \mathbf{M}$ не вырождены, и матрица $\mathbf{QT}$ имеет вид:

$$\mathbf{QT} = \left(\mathbf{0}_{\mathcal{J}} \;\; \mathbf{R}_{[n]\setminus\mathcal{J}}\right),$$

где $\mathcal{J}$ некоторая информационная совокупность кода C, определенного через порождающую матрицу $\mathbf{G}$, $[n] \setminus \mathcal{J}$ — дополнение к $\mathcal{J}$, $\mathbf{0}$ — $n \times k$ нулевая матрица, $\mathbf{R}$ — $n \times (n-k)$ матрица полного ранга. Это означает, что $\mathbf{QT}$ имеет нулевые столбцы на позициях $\mathcal{J}$ кода C, а остальные $n-k$ ненулевые и образуют матрицу полного ранга $\mathbf{R}$.

Пример конструкции $\mathbf{QT}$

Задача: построить матрицы $\mathbf{Q}$ и $\mathbf{T}$ так, чтобы $\mathbf{QT}$ имела ранг $n-k$ и имела k нулевых столбцов на позициях $\mathcal{J}$.

- Пусть $\mathbf{T}$ произвольная невырожденная матрица, а $\mathbf{T}_{\mathcal{J}}$ состоит из столбцов матрицы $\mathbf{T}$ с индексами $\mathcal{J}$. Рассмотрим матрицу $\mathbf{T}_{\mathcal{J}}^T$ как порождающую матрицу некоторого кода, тогда существует проверочная матрица $\mathbf{H}_J$ размера $(n-k)\times n$, такая что $\mathbf{H}_{\mathcal{J}}\mathbf{T}_{\mathcal{J}} = \mathbf{0}$.
- Построим матрицу $\mathbf{Q} = \mathbf{L}\mathbf{H}_{\mathcal{J}}$, где $\mathbf{L}$ — произвольная невырожденная $(n-k)\times n$ матрица.

Матрица $\mathbf{QT} = \mathbf{L}\mathbf{H}_{\mathcal{J}}\mathbf{T}$ имеет нулевые столбцы на позициях $\mathcal{J}$ а остальные столбцы образуют матрицу полного ранга.

Шифрование/дешифрование

Генерация ключей:

- *Открытый ключ:* $(\mathbf{G}', \mathbf{G}_2') = (\mathbf{GM}, \mathbf{Q}(\mathbf{G}_0 + \mathbf{T})\mathbf{M})$
- *Секретный ключ:* $(\mathbf{G}, \mathbf{M}, \mathbf{T}, \mathbf{Q}, \mathbf{G}_0, \mathcal{J})$

Шифрование/дешифрование:

1. *Шифрование:* Вычислим $\phi(\mathbf{u}) = \mathbf{c} = \mathbf{uG}' + \mathbf{eG}_2'$, где $\mathbf{e}$ — q-ичный вектор произвольного веса.
2. *Дешифрование:* Вычислим $\phi^{-1}(\mathbf{c})$ следующим образом:
   - $\mathbf{y} = \mathbf{cM}^{-1} = \mathbf{uG} + \mathbf{eQ}(\mathbf{G}_0 + \mathbf{T}) = \mathbf{uG} + \mathbf{eQG}_0 + \mathbf{eQT}$.
   - Легко видеть, что $\mathbf{uG} + \mathbf{eQG}_0 \in C$ и вектор $\mathbf{eQT}$ имеет ненулевые позиции только на $[n] \setminus \mathcal{J}$, таким образом, информационная совокупность $\mathcal{J}$ свободна от ошибок. Значит:
     $$\mathbf{y} - \mathbf{y}_{\mathcal{J}}\mathbf{G}_{\mathcal{J}}^{-1}\mathbf{G} = \mathbf{eQT}$$
   - Вычислим $\mathbf{eQTT}^{-1} = \mathbf{eQ}$
   - Вычислим $\mathbf{eQG}_0$
   - Найдем $\mathbf{u}$ из $\mathbf{uG} = \mathbf{y} - \mathbf{eQG}_0 - \mathbf{eQT}$

Синдромная атака

Синдром $\mathbf{S}$ шифрограммы $\mathbf{c}$ имеет следующий вид:

$$\mathbf{S} = \mathbf{c}\mathbf{H}'^T = \mathbf{e}\mathbf{Q}\mathbf{T}\mathbf{H}^T = \mathbf{e}\left(\mathbf{H}\left(\mathbf{Q}\mathbf{T}\right)^T\right)^T.$$

Это значит, что Ева должна решить синдромное уравнение $\mathbf{H}\left(\mathbf{Q}\mathbf{T}\right)^T$, где $\mathbf{Q}\mathbf{T}$ не является перестановкой.

Пример синдромной атаки

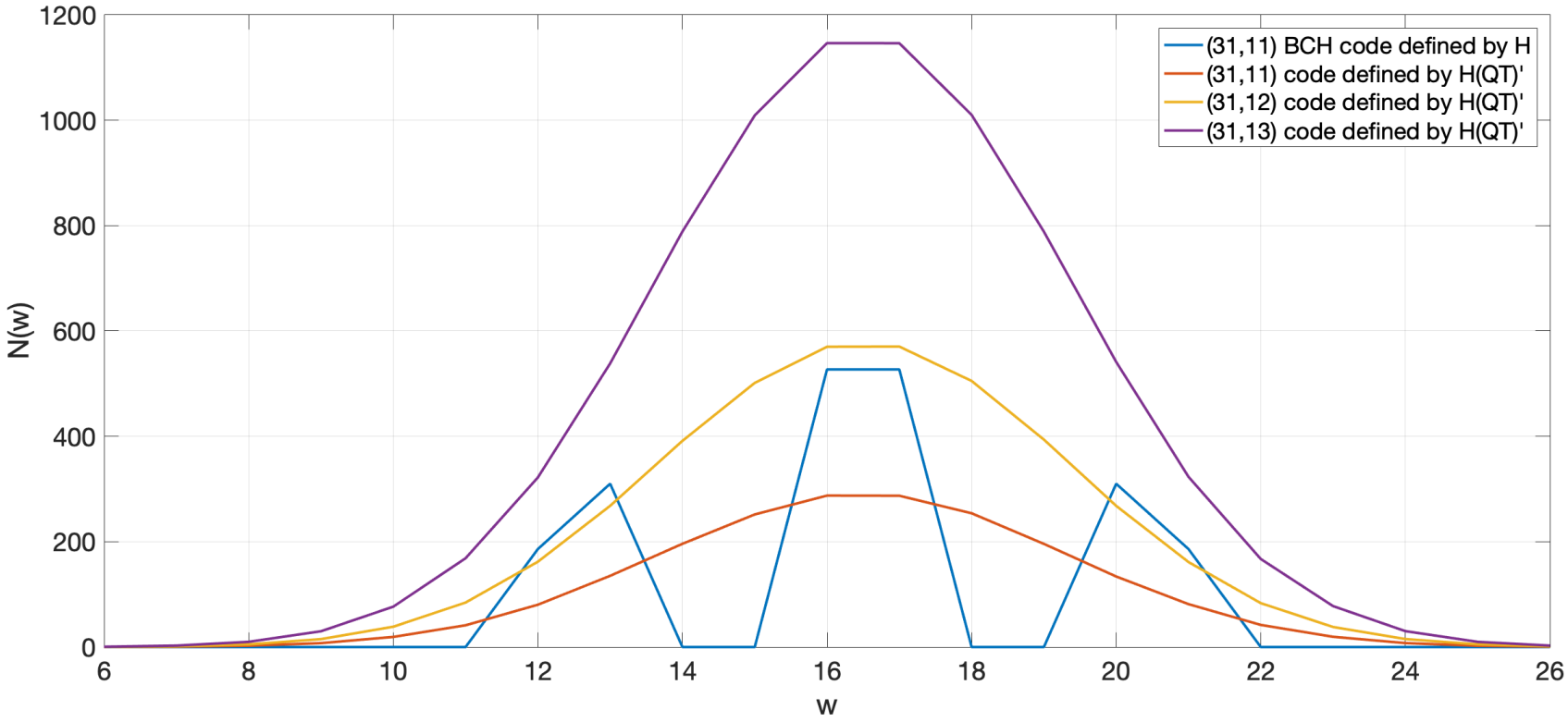

Рис.: Спектр различных $\mathbf{H}\left(\mathbf{QT}\right)^T$ полученных из проверочной матрицы $\mathbf{H}$ БЧХ кода (31,11,11)

Таблица: Минимальное расстояние 500 кодов, полученных из (31,11,11) БЧХ кода

Code	#	минимальное расстояние				
		мин.	макс.	среднее.	дисперсия	ВГ-граница
(31,11)	156	4	8	6,21	0,72	5,09
(31,12)	273	3	7	5,74	0,56	4,68
(31,13)	72	3	7	5,29	0,66	4,29

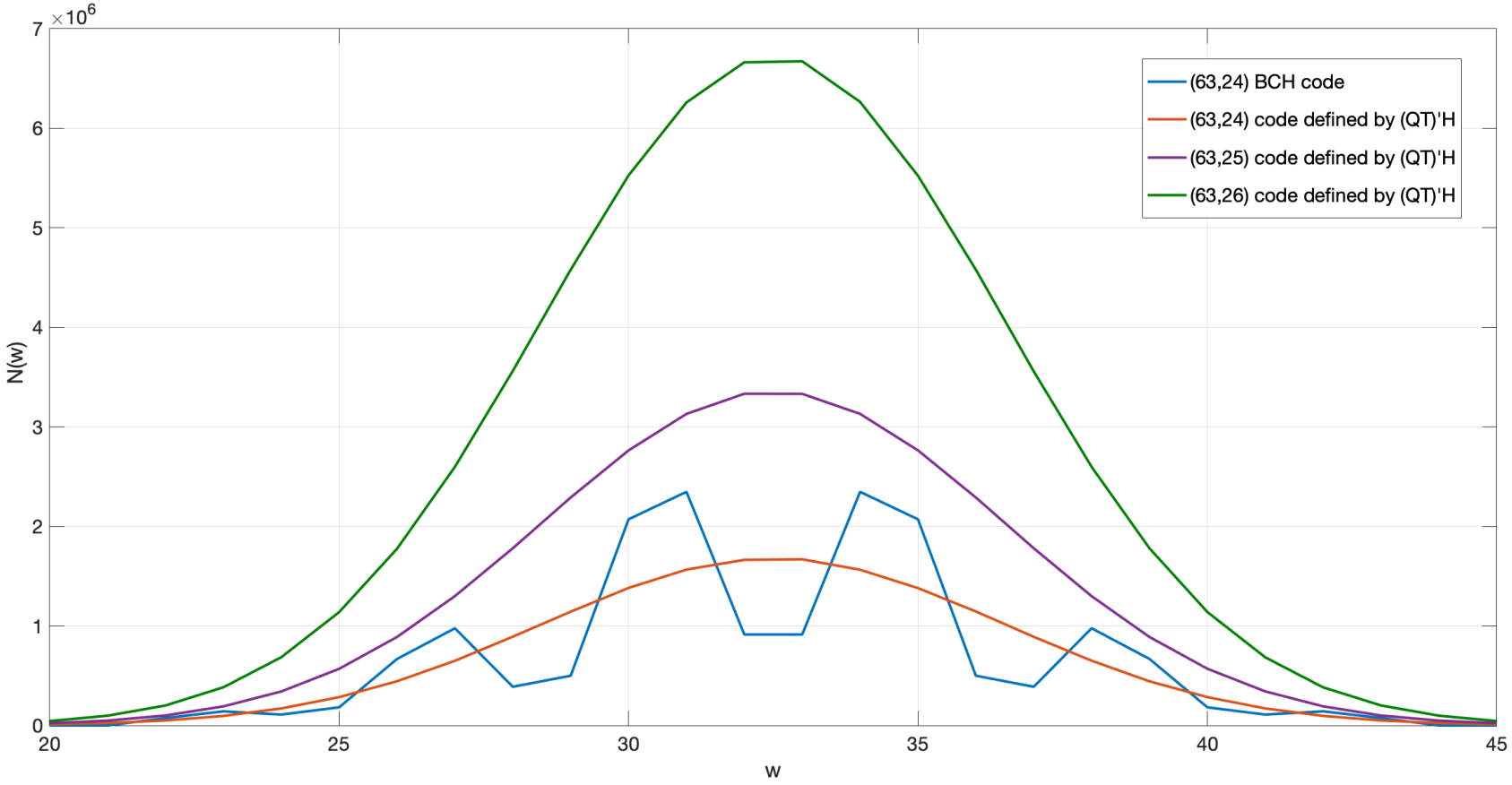

Рис.: Спектр кодов с проверочными матрицами $\mathbf{H}(\mathbf{QT})^T$, полученными из проверочной матрицы $\mathbf{H}$ БЧХ кода (63,24,17)

Таблица: Минимальное расстояние 100 кодов, полученных из (63,24,17) БЧХ кода

Code	#	минимальное расстояние				
		мин.	макс.	среднее	дисперсия	ВГ-граница
(63,24)	29	10	13	11,28	0,56	9,66
(63,25)	60	8	12	10,67	0,56	9,26
(63,26)	11	9	11	10,18	0,56	8,87

Примеры параметров криптосистемы

Предположение: сложность атаки $q^{min\{k,n-k\}}$. Тогда:

Пример

Исходные параметры криптосистемы Мак-Элиса: $n = 1024$, $k = 524$, $t = 50$,*тогда* $\log_2 \tau \approx 53$. *Для того, чтобы получить такие же параметры безопасности для новой криптосистемы, нужно найти такой код* C' *длины* n' *и размерности* k', *чтобы* $\min\{k',n' - k'\} \approx 53$.
БЧХ-код с параметрами $(n',k',t') = (127,71,9)$ *имеет* $\log_2 \tau' \approx 56$. *Для* $(1024,524,50)$ *кода размер откытого ключа* $k(n-k) = 262000$ *бит, а для предложенной системы размер ключа* $n'^2 + k'(n' - k') = 20105$ *бит, что более чем в* 13 *раз меньше.*

Другая постановка задачи

- Основное достоинство кодовой криптографии - масштабируемость и стойкость (по крайней мере, пока) к квантовым вычислениям
- Основная проблема - очень длинные ключи, необходимость которых обуславливается требованиями криптографической стойкости
- Как уменьшить длину ключа?
  1. Искать коды, которые можно описать компактно
  2. Учиться декодировать за t хотя бы некоторые множества ошибок
  3. В идеале - хотелось бы построить систему, для которой атака по информационным совокупностям/атака Штерна неприменимы.

Некоторые предварительные замечания

- Идея системы Мак-Элиса состоит в том, что после обратного преобразования $\mathbf{P}^{-1}$, примененного к шифрограмме $\mathbf{y}$, вектор ошибки $\mathbf{eP}^{-1}$ не меняет своего веса, то есть принадлежит тому же множеству векторов, к которому принадлежал исходный вектор ошибки $\mathbf{e}$, с помощью которого шифровалось исходное сообщение. Поэтому для расшифрования легальному пользователю не надо было знать конкретный вектор ошибки — любой вектор ошибки веса до t декодировался в коде C_0 одним и тем же алгоритмом.
- Большое количество усилий, которые предпринимались в теории кодирования на протяжении нескольких десятков лет позволили получить алгоритмы, имеющие небольшой показатель экспоненты сложности (лучшая на сегодняшний день оценка имеет порядок $0.049n$). Поэтому для достижения требуемой стойкости криптосистемы Мак-Элиса приходится выбирать коды большой длины n.
- Все описанные в ранее указанных работах алгоритмы, так или иначе описаются на то, что они исправляют легкие ошибки, вес которых значительно меньше длины кодового слова. Однако любой линейный код способен исправлять значительное количество ошибок большого веса.

Неформальная постановка задачи

Как генерировать множество ошибок, которые мы предполагаем декодировать с помощью кода C с порождающей матрицей $\mathbf{G}_0$? Фактически мы должны задать множество ошибок $\mathcal{E}$, из которого будет случайным образом выбираться вектор ошибок, используемый в процедуре шифрования.

Требования на множество $\mathcal{E}$

Обозначим через $\mathcal{E}_0$ множество векторов ошибок $\mathbf{e}$, исправляемых кодом с помощью полиномиального алгоритма декодирования ξ, а через $\phi(C_0)$ – некоторое обратимое преобразование кода. В результате этого преобразования получается код $C = \phi(C_0)$. Пусть, кроме того, ω — допустимая стойкость криптосистемы (число операций, необходимых для ее вскрытия). Тогда мы можем сформулировать свойства, предъявляемые к множеству $\mathcal{E}$:

1. Множество $\mathcal{E}$ вложено в множество $\mathcal{E}_0$ ошибок, исправляемых кодом C_0: $\mathcal{E} \subset \mathcal{E}_0$ и существует полиномиальный алгоритм V генерации случайного вектора $\mathbf{e} \in \mathcal{E}$.
2. $|\mathcal{E}| \geq \omega$ — т. е. мощность множества должна препятствовать перебору по всем его элементам с целью вскрытия криптосистемы
3. Существует обратимое преобразование $\phi(C_0)$ полиномиальной сложности, отображающее код C_0 в эквивалентный код C
4. $\phi^{-1}(\mathcal{E}) \subset \mathcal{E}$, причем $|\phi^{-1}(\mathcal{E})| \geq \omega$.
5. Для кода C_0 существует полиномиальный алгоритм ξ исправления ошибок из множества $\mathcal{E}$.
6. Сложность декодирования ошибок из множества $\mathcal{E}$ в коде C не меньше, чем ω

Общий концепт криптосистемы

С учетом введенных обозначений предлагаемая схема шифрования $\mathbb{S}$ может быть описана следующим образом:

- Публичный ключ в предлагаемой системе: код $C = \phi(C_0)$ и алгоритм V.
- Секретный ключ: обратное преобразование ϕ^{-1}.
- Алгоритм шифрования:
  1. С помощью алгоритма V выбирается случайный вектор $\mathbf{e} \in \mathcal{E}$.
  2. По сообщению $\mathbf{x}$ вычисляется шифрограмма

$$\mathbf{y} = \mathbf{x}\mathbf{G} + \mathbf{e}$$

- Алгоритм рашифрования:
  1. Вычисляется $\mathbf{y}' = \phi^{-1}(\mathbf{y}) = \mathbf{x}\phi^{-1}(\mathbf{G}) + \phi^{-1}(\mathbf{e}) = \mathbf{x}\mathbf{G}_0 + \mathbf{e}'$, где $\mathbf{e}' \in \mathcal{E}_0$.
  2. С помощью алгоритма ξ находится $\mathbf{x}$.

Стойкость криптосистемы

Стойкость системы, с учетом условий (1)-(6) определяется при прямой атаке, (т.е. атаке, основанной на декодировании ошибок из $\mathcal{E}$ в коде C) величиной ω. В описанных примерах ошибки из $\mathcal{E}$, как правило, не являются лидерами смежных классов (самыми легкими в смежном классе векторам кода C), а значит их декодирование возможно только перебором либо по словам кода C либо по множеству ошибок из $\mathcal{E}_0$. Таким образом:

$$\omega = \min\{2^k, |\mathcal{E}_0|\}.$$

По построению $|\mathcal{E}| \leq |\mathcal{E}_0| \leq 2^{n-k}$. Поэтому естественно называть оптимальной криптосистему, для которой $|\mathcal{E}_0| = 2^{n-k}$, и оценивать качество системы величиной

$$\tau = \frac{\log_2 |\mathcal{E}_0|}{n-k}.$$

Задача построения преобразования

Пусть задан код C_0 и множество ошибок $\mathcal{E}_0$, декодируемых в этом коде некоторым декодером ξ имеющим полиномиальную сложность. Пусть, кроме того, алгоритм V , имеющий полиномиальную сложность, порождает равномерное распределение $U(\mathcal{E}_0)$ на множестве $\mathcal{E}_0$. Требуется найти обратимое преобразование ϕ кода C_0 в код $C = \phi(C_0)$ такое, что:

1. Множество векторов $\mathcal{E} = \phi(\mathcal{E}_0)$ является множеством различных представителей смежных классов кода C. Это обеспечивает исправление кодом C ошибок из $\mathcal{E}$.
2. Алгоритм V порождает равномерное распределение $U(\mathcal{E})$ на множестве $\mathcal{E}$. В этом случае декодирование кода может быть осуществлено только перебором по словам кода C или по векторам из $\mathcal{E}$.

Пример - двоичный образ РС кода

Пример 1:

Пусть $\mathbf{G}'$ — порождающая матрица (n,k)–кода Рида Соломона с кодовым расстоянием $d = n - k + 1$ над полем $GF(q) = GF(2^m)$ длины $n = 2^m - 1$. Рассмотрим двоичное представление такого кода, т.е. код, слова которого получаются из слов кода Рида Соломона в результате замены q-ичных символов их двоичным представлением. Очевидно, в результате мы получим двоичный (nm,km)–код с порождающей матрицей $\mathbf{G}$ размера $km \times nm$. Разобъем позиции вектора ошибки $\mathbf{e}$ длины nm на n фиксированных отрезков длины m. Говорят, что вектор $\mathbf{e}$ является вектором t-кратных фиксированных пакетов ошибок, если все его ненулевые элементы накрываются ровно t фиксированными отрезками. Поскольку код Рида Соломона исправляет любые t q-ичных ошибок, то двоичный код с порождающей матрицей $\mathbf{G}$ исправляет любые t-кратные фиксированные пакеты. Множество фиксированных пакетов кратности, не превышающей t обозначим через $\mathcal{E}$ — это множество исправляемых кодом ошибок. Определим множество $\mathcal{V}$ как множество перестановок η, осуществляемых в два шага:

1. Производится перестановка ρ длины m на множестве символов внутри в отрезков длины m.
2. Производится перестановка β длины n на множестве из n отрезков длины m

То есть любая перестановка $\phi \in \mathcal{V}$ имеет вид $\phi = \beta \cdot \rho$.
Очевидно, для любого вектора $\mathbf{e} \in \mathcal{E}$ вектор $\phi(\mathbf{e}) \in \mathcal{E}$, а поскольку $\phi^{-1} \in \mathcal{V}$, то и $\phi^{-1}(\mathbf{e}) \in \mathcal{E}$.

Пример - двоичный образ РС кода с умножением на произвольные матрицы

Рассмотрим (27,9) укороченный код Рида-Соломона (РС), полученный из РС-кода длины 63 над полем $GF(2^6)$. Данный код имеет расстояние 19 и исправляет 9 любых независимых ошибок. Рассмотрим далее двоичный образ этого кода, представив каждый элемент поля $GF(2^6)$ в виде двоичного вектора длины 6. В результате получим двоичный (162,54) код C, исправляющий до 9 пакетов ошибок длины до 6. Рассмотрим матрицу $\mathbf{M}$ размера 6×162, имеющую вид

$$\mathbf{M} = [\mathbf{M}_1\ \mathbf{M}_2\ \ldots \mathbf{M}_{29}]\,,$$

где $\mathbf{M}_i$ — невырожденная 6×6 матрица.
Рассмотрим код C_M такой, что любое слово $\mathbf{c} \in C_M$ имеет вид:

$$\mathbf{c} = (\mathbf{c}_1\mathbf{M}_1\ \mathbf{c}_2\mathbf{M}_2\ \ldots \mathbf{c}_{29}\mathbf{M}_{29})\,,$$

где $\mathbf{c}_i$ — двоичный вектор длины 6 такой, что $(\mathbf{c}_1\ \mathbf{c}_2\ \ldots\ \mathbf{c}_{29}) \in C$.
Множество фиксированных пакетов кратности, не превышающей 9 обозначим через $\mathcal{E}$ — это множество исправляемых кодом C ошибок. Определим множество $\mathcal{V}$ как множество перестановок η, осуществляемых в два шага:

1. Производится перестановка ρ длины 6 на множестве символов внутри в отрезков длины 6.
2. Производится перестановка β длины 29 на множестве из 29 отрезков длины 6

Продолжение примера

То есть любая перестановка $\phi \in \mathcal{V}$ имеет вид $\phi = \beta \cdot \rho$, где операция $(\cdot)$ - операция суперпозиции перестановок.
Публичной порождающей матрицей $\mathbf{G}$ криптосистемы является:

$$\mathbf{G} = \phi\left(\mathbf{S}\mathbf{G}'\right), \tag{2}$$

где перестановка ϕ подразумевается как перестановка столбцов матрицы $\mathbf{S}\mathbf{G}'$, а $\mathbf{S}$ - случайная невырожденная 54×54 матрица, а $\mathbf{G}'$ - секретная порождающая матрица кода C.
Легко заметить, что так как $\mathbf{M}_i$ - произвольные невырожденные матрицы, то код C_M не эквивалентен коду $\mathbf{C}$, как это было в случае, описанном в предыдущем примере.

Пример - сверточные коды

$$\mathbf{G}(D) = \begin{pmatrix} D^2 + D + 1 & D^2 + 1 \end{pmatrix} = \begin{pmatrix} 7 & 5 \end{pmatrix} .$$

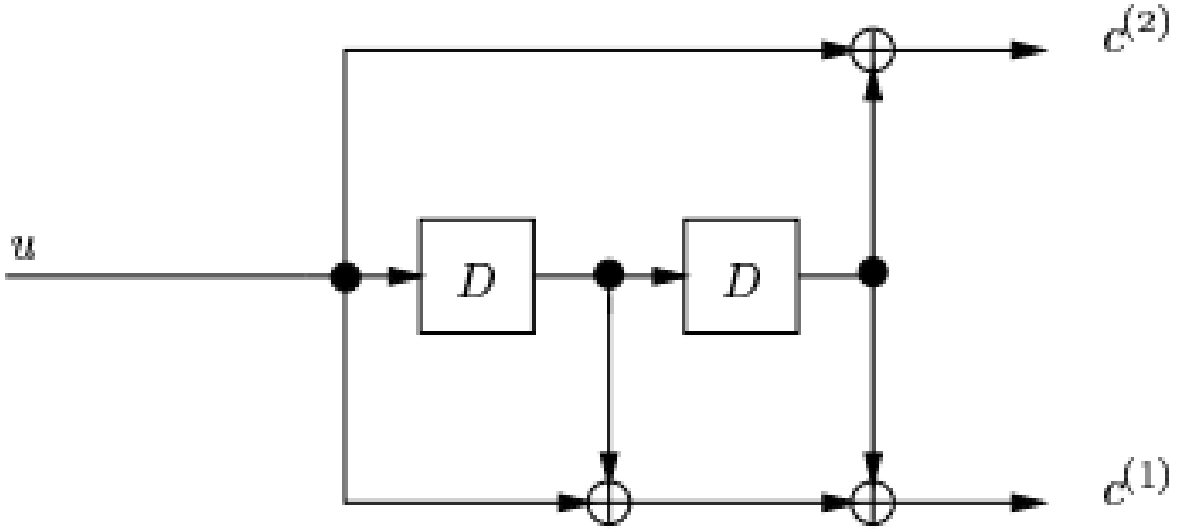

Рис.: (7,5) сверточный код

Решетка сверточного кода

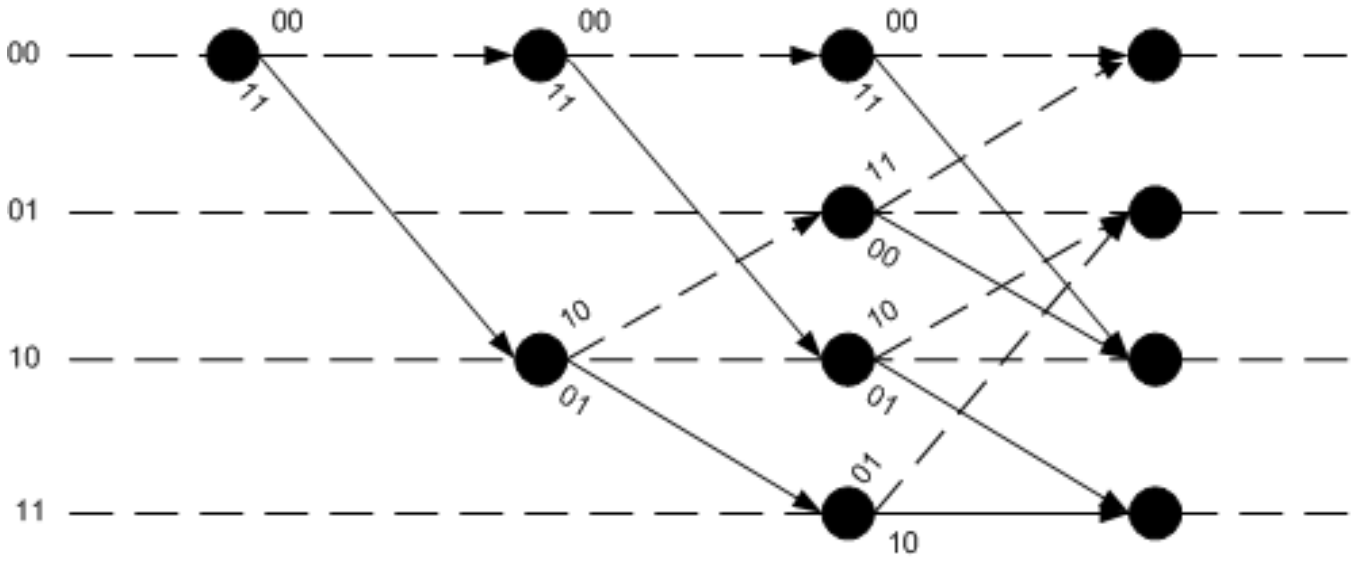

Рис.: Решетка (7,5) сверточного кода

Свободное расстояние этого кода равно $d_{free} = 5$, а значит он исправляет любые $t = 2$ независимые ошибки на последовательности из $m + 1 = 3$ кортежей длины $n = 2$.

Блочный код из сверточного

Рассмотрим (228,112)-терминированный сверточный код. Он может исправить по крайней мере $23 * 2 = 46$ ошибок, расположенных на 22 последовательностях (частично перекрывающих друг друга) $\mathbf{s}$ из 14 символов таких, что $\mathbf{s}[1:4]$ и $\mathbf{s}[11:14]$ свободны от ошибок, а на $\mathbf{s}[5:10]$ расположено не более двух ошибок и на последней последовательности $\mathbf{s}'$ из 12 символов таких, что $\mathbf{s}[1:4]$ и $\mathbf{s}[11:12]$ свободны от ошибок, а на $\mathbf{s}[5:10]$ расположено не более двух ошибок. Для исправления этих ошибок можно использовать алгоритм Витерби.

Ошибки, исправимые сверточным кодом

Множество из $\binom{6}{2}^{22} \approx 2^{86}$ различных конфигураций в 22 последовательностях **s** обозначим через $\mathcal{E}$ — это множество исправляемых кодом ошибок. Определим множество множество $\mathcal{V}$ как множество перестановок η, осуществляемых в два шага:

1. Производится перестановка ρ длины 10 на множестве символов внутри в отрезков длины 10, причем последние 4 символа являются неподвижными точками перестановки. Также производится перестановка ρ' длины 8, причем последние 2 символа являются неподвижными точками перестановки.
2. Производится перестановка β длины 23 на множестве из 22 последовательностей длины 10 и 1 последовательности длины 8.

То есть любая перестановка $\phi \in \mathcal{V}$ имеет вид $\phi = \beta \cdot \rho$, где операция $(\cdot)$ - операция суперпозиции перестановок.
Очевидно, для любого вектора $\mathbf{e} \in \mathcal{E}$ вектор $\phi(\mathbf{e}) \in \mathcal{E}$, а поскольку $\phi^{-1} \in \mathcal{V}$, то и $\phi^{-1}(e) \in \mathcal{E}$.

Криптосистема на базе сверточных кодов

Публичной порождающей матрицей криптосистемы является:

$$\mathbf{G} = \phi\left(\mathbf{SG}'\right),$$

где перестановка ϕ подразумевается как перестановка столбцов матрицы $\mathbf{SG}'$, а $\mathbf{S}$ - случайная невырожденная 112×112 матрица.

Таким образом, мы получили все компоненты схемы шифрования $\mathbb{S}$.

Отметим, что для прямой атаки требуется либо перебрать все 2^{112} кодововых слов кода (228,112), либо перебрать все 2^{86} различных элементов множества $\mathcal{E}$ — векторов ошибок.

Краткое резюме

- Для сокращения длины ключа можно либо подбирать код, позволяющий описать его компактно, либо перейти к исправлению ошибок большого веса
- Описать множество исправимых кодом ошибок (даже подмножества) - достаточно трудная задача, зависящая от структуры кода
- Еще более сложная задача - это представить полиномиальный алгоритм генерации множества ошибок, исправимых кодом.
- Для некоторых классов кодов (образ обобщенных РС, сверточные) можно описать некоторые подмножества множества исправимых ошибок, однако описанные множества далеко не полны
- И тут мы, кажется, переходим к задачам, интересным не только в кодовой криптографии...