

Рида-Маллера

Код Рида-Маллера

Илья Коннов

Факультет компьютерных наук

Высшая Школа Экономики

12 марта 2022 г.

Код Рида-Маллера

12 марта 2022 г

Кол Рила-Маллера

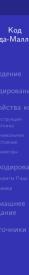
- 1. Существует три различных варианта этого доклада:
 - 1.1 Краткая презентация, которую несложно рассказать, но может быть сложно понять (ReedMuller-trans.pdf).
 - 1.2 Более длинная презентация с ценными комментариями, дополнительными доказательствами и интересными фактами (ReedMuller-slides.pdf). Вы сейчас читаете именно эту версию. Слайды с особенным фоном — не вошедшие в маленькую презентацию.
 - 1.3 Текстовая статья со всем содержимым длинной презентации, комментариями на своих местах, а также бонусным приложением с более подробным описанием алгоритма (ReedMuller-article.pdf).

Их все можно посмотреть здесь: https://sldr.xyz/ReedMuller/

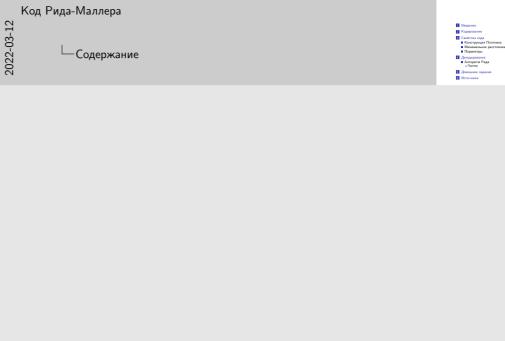
イロト 4周トイミトイミト ヨー かなべ

По любым вопросам: r-m@sldr.xyz или t.me/iliago или vk.com/iliago.









декодирования) в сентябре 1954 года.

 $u \oplus v = (u_1 + v_1, u_2 + v_2, ..., u_n + v_n).$

сообщения длиной $k = \sum_{i=0}^{r} C_{m}^{i}$ при помощи 2^{m} бит.

Соглашение: сложение векторов $u,v\in\mathbb{F}_2^n$ будем обозначать как

Код описан Дзеидом Маллером (автор идеи) и Ирвингом Ридом (автор мето.

лемплипперация) в гентабле 1954 года

Рида-Маллера



Код описан Дэвидом Маллером (автор идеи) и Ирвингом Ридом (автор метода

Традиционно, считается $\stackrel{\circ}{\text{что}}$ коды бинарные и работают над битами, т.е. \mathbb{F}_2 .

Обозначается как RM(r, m), где r — ранг, а 2^m — длина кода. Кодирует

Булевы функции и многочлен Жегалкина

Или при помощи многочлена Жегалкина:

Рида-Маллера

Введение



4□ > 4□ > 4□ > 4□ > 4□ > 4□

f(x,y) = xy + x + y + 1

Всякую булеву функцию можно записать при помощи таблицы истинности:

Код Рида-Маллера -Введение Всякую булкву функцию можно записать при помощи таблицы истинности: Булевы функции и многочлен Жегалкина Или пои помоши миогочлена Жегалина

В общем струке миоточения булут иметь степленияй вил

Рида-Маллера

 $f(x_1, x_2, ..., x_m) = \sum_{S \subseteq \{1, ..., m\}} c_S \prod_{i \in S} x_i$

Например, для m=2: $f(x_1,x_2)=c_{12}\cdot x_{\{1\}}x_2+c_{\{2\}}\cdot x_2+c_{\{1\}}\cdot x_1+c_{\varnothing}\cdot 1$

В общем случае, многочлены будут иметь следующий вид:

Всего $n=2^m$ коэффициентов для описания каждой функции.

-Введение

Код Рида-Маллера



Функции небольшой степени

Каждую можно записать следующим образом:

Сколько тогда всего коэффициентов используется?

















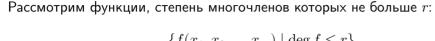












 $k = C_m^0 + C_m^1 + C_m^2 + \dots + C_m^r = \sum_{i=0}^{r} C_m^i$

 $\{f(x_1, x_2, ..., x_m) \mid \deg f < r\}$

 $f(x_1, x_2, ..., x_m) = \sum_{\substack{S \subseteq \{1, ..., m\} \\ |S| < r}} c_S \prod_{i \in S} x_i$

В каждом произведении используется не больше r переменных.

4 D > 4 D > 4 D > 4 D > 1 D 9 Q Q

- -Введение

Код Рида-Маллера

—Ф∨нкции небольшой степени

- $k = C_m^0 + C_m^1 + C_m^2 + ... + C_m^r = \sum_i C_m^i$
- 1. Замечу, что при $S=\varnothing$, мы считаем, что $\prod_{i\in S}x_i=1$, таким образом всегда появляется свободный член.
- 2. Если говорить несколько проще, то для составления многочленов мы сложим сначала одночлены (x+y+z+...), затем произведения одночленов (xy+yz+xz+...) и т.д. вплоть до r множителей (поскольку мы работаем в поле \mathbb{F}_2 , здесь нету x^2, y^2, z^2 , т.к. $a^2 = a$). Тогда легко видеть, почему kименно такое: мы складываем все возможные перестановки сначала для 0 переменных, потом для одной, двух, и так вплоть до r (не не больше, ведь $\deg f < r$).



Идея кодирования

Код Рида-Маллера

Кодирование

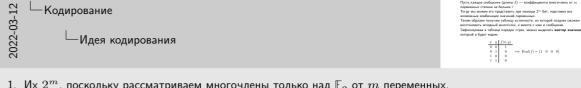
Пусть каждое сообщение (длины k) — коэффициенты многочлена от mпеременных степени не больше r.

Тогда мы можем его представить при помощи 2^m бит, подставив все возможные комбинации значений переменных.

Таким образом получим таблицу истинности, из которой позднее сможем восстановить исходный многочлен, а вместе с ним и сообщение. Зафиксировав в таблице порядок строк, можно выделить вектор значений, который и будет кодом.

x	y	f(x,y)					
0	0	1					
0	1	0	\Longrightarrow	$\mathrm{Eval}(f) = (1$	0	0	0
1	0	0					
1	1	0					





1. Их 2^m , поскольку рассматриваем многочлены только над \mathbb{F}_2 от m переменных.

Код Рида-Маллера

2. Вектор значений — обозначается $\operatorname{Eval}(f)$ — столбец таблицы истинности, содержащий значения функции. Имеет смысл только при зафиксированном порядке строк в таблице. У меня он везде самый обычный, как в примере выше.



Кодирование

Это RM(1,2).

■ Тогда наш многочлен: $f(x_1,x_2)=c_{\{2\}}x_2+c_{\{1\}}x_1+c_{\varnothing}$. \blacksquare Сообщение: 101, тогда $f(x_1, x_2) = x_2 + 0 + 1$.

r = 1 (степень многочлена), m = 2 (переменных).

■ Подставим всевозможные комбинации:

$$\begin{array}{c|cc} x_1 & x_2 & f(x_1, x_2) \\ \hline 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ \end{array}$$

■ Получили код: Eval(f) = 1100.

1001 при помощи нескучного шрифта.

Код Рида-Маллера -Кодирование

<u></u>Пример

■ Получили ков: Eval(f) = 1100. 1. Здесь и далее я для краткости и удобства записываю битовые векторы не как $(1 \quad 0 \quad 0 \quad 1)$, а как

2. Для кодирования очень важно понимать, как именно биты сообщения ставятся в соответствие коэффициентам многочлена. Поэтому давайте введём соглашение: если упорядочить элементы

множества у каждого коэффициента по возрастанию, то коэффициенты сортируются в лексиографическом порядке: $c_{1,2}$ раньше $c_{1,3}$, поскольку 2 < 3 и $c_{2,3}$ раньше $c_{3,4}$, поскольку 2 < 3. Пример для m=4:

 $f(x_1, x_2, x_3, x_4) = c_{\{1, 2, 3, 4\}} x_1 x_2 x_3 x_4$ $+ c_{\{1,2,3\}}x_1x_2x_3 + c_{\{1,2,4\}}x_1x_2x_4 + c_{\{1,3,4\}}x_1x_3x_4 +$ $+c_{\{2,3,4\}}x_2x_3x_4$ $+ c_{\{1,2\}}x_1x_2 + c_{\{1,3\}}x_1x_3 + c_{\{1,4\}}x_1x_4 + c_{\{2,3\}}x_2x_3 +$ $+c_{\{2,4\}}x_2x_4+c_{\{3,4\}}x_3x_4$

 $+c_{\{1\}}x_1+c_{\{2\}}x_2+c_{\{3\}}x_3+c_{\{4\}}x_4+c_{\emptyset}$

イロト 4周ト 4 三ト 4 三ト ラ のので

Также можно кодировать множества при помощи битов, используя отношение $x \in A \Longleftrightarrow v_x = 1$



Декодирование когда потерь нет

Рида-Маллера

Кодирование

Кодирован

Свойства ко Конструкция Плоткина

Конструкция Плоткина Минимальное расстояние Параметры

Параметры **Декодирова**Алгоритм Рида

_{Пример} Домашнее

адание

■ Мы получили код: 1100

Подстановками в $f(x_1,x_2)=c_2x_2+c_1x_1+c_0$ получим СЛАУ. $\begin{cases} &c_0=1\\ &c_1+c_0=1\\ &c_2+&c_0=0\\ &c_2+c_1+c_0=0 \end{cases}$

 $c_{\{2\}} = 1, c_{\{1\}} = 0, c_{\varnothing} = 1$, исходное сообщение: 101.

Код Рида-Маллера

— Кодирование

— Декодирование когда потерь нет

— Декодирование когда потерь нет

— Теперь покажем как можно пекспировать когда потерь нет это пример — продолжение

1. Теперь покажем, как можно декодировать когда потерь нет. Этот пример — продолжение предыдущего.





Коды 0-го порядка Для случая RM(0,m) нужна функция от m аргументов, степени не выше 0. $f(x_1, x_2, ..., x_m) = 0$

$q(x_1, x_2, ..., x_m) = 1$

■ Сообщение 1 даст код 11...1

Таблица истинности:

- Вывод: это 2^m -кратное повторение символа ■ Сообщение 0 даст код 00...0
- $\begin{cases} x_1 & x_2 & \dots & x_m & f(x_1, \dots, x_m) & g(x_1, \dots, x_m) \\ 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 & 1 \\ & & \ddots & & \vdots & & \vdots \\ 1 & 1 & 1 & 1 & 0 & 1 \end{cases}$

4 D > 4 A > 4 B > 4 B > B 9 Q P

Коды 0-го порядка ■ Сообщение 1 даст код 11...1

Код Рида-Маллера -Кодирование

единицами.

- 1. Отдельно стоит рассмотреть вариант кода при r=0, он нам в будущем пригодится для доказательств.
- 2. Таких функций существует всего лишь две, поскольку мы можем влиять лишь на свободный член.

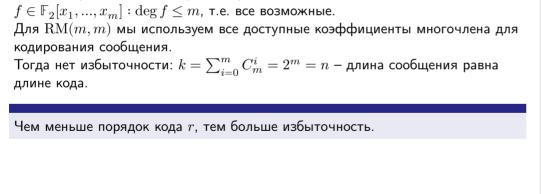
 $g(x_1, x_2, ..., x_m) = 0$ $g(x_1, x_2, ..., x_m) = 1$

Все остальные коэффициенты обнуляются из-за требования $\deg f \leq 0$. 3. Здесь число строк, как и в любой другой таблице истинности, равно 2^m , а колонки со значениями никак не зависят от аргументов функций. Получается две колонки – одна с нулями, другая с

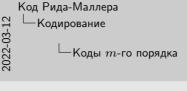
Коды m-го порядка

Рида-Маллера

Кодирование



Есть m переменных, и мы рассматриваем многочлены



1. Есть ещё один тривиальный случай, когда m=r.

イロト 4周トイミトイミト ヨー かなべ

Доказательство линейности

доказательство линеино

Рида-Маллера

....

Свойства кода

.воиства ко Конструкция Плоткина

Ілоткина Линимальноє асстояние Іараметры

Декодирован Алгоритм Рида

Домашнее задание

Істочні

Пусть C(x) кодирует сообщение $x \in \mathbb{F}_2^k$ в код $C(x) \in \mathbb{F}_2^m$.

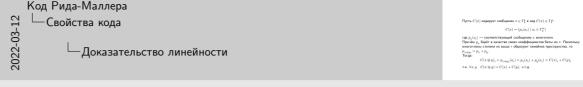
$$C(x) = (p_x(a_i) \mid a_i \in \mathbb{F}_2^m)$$

где $p_x(a_i)$ — соответствующий сообщению x многочлен. Причём p_x берёт в качестве своих коэффициентов биты из x. Поскольку многочлены степени не выше r образуют линейное пространство, то $p_{(x\oplus y)}=p_x+p_y.$

Тогда:

$$C(x \oplus y)_i = p_{(x \oplus y)}(a_i) = p_x(a_i) + p_y(a_i) = C(x)_i + C(y)_i$$

т.е.
$$\forall x,y \quad C(x\oplus y) = C(x) + C(y)$$
, ч.т.д.



- 1. Хотим показать, что этот код является линейным, т.е. что его кодовые слова образуют линейное пространство, и у нас есть изоморфизм из пространства сообщений (\mathbb{F}_2^k) в пространство слов (\mathbb{F}_2^m) . Для этого необходимо немного формализовать всё описанное раньше.
- 2. Пояснение: перебираем все векторы a_i (2^m штук), подставляем каждый в p_x в качестве переменных и таким образом получаем вектор значений (длины 2^m). Именно он и называется кодом.
- 3. Напомню, что базис пространства многочленов выглядит примерно так: 1, x, y, z, xy, yz, xz (для трёх переменных, степени не выше 2). Чтобы преобразовать сообщение в многочлен, мы берём каждый бит сообщения и умножаем его на соответствующий базисный вектор. Очевидно, такое преобразование будет изоморфизмом. Именно поэтому $p_{(x\oplus y)} = p_x + p_y$. Обратите внимание, что сообщение x это не просто число (\mathbb{Z}_2^k) и мы рассматриваем его биты, а реально вектор битов (\mathbb{Z}_2^k). У него операция сложения побитовая.
- 4. Здесь я использую запись $C(x)_i$ для i-го элемента вектора C(x). Поскольку i произвольное, то и весь вектор получился равен. Таким образом, этот код действительно линейный и к нему применимы уже известные теоремы!



Последствия линейности

Рида-Маллера

Введение

Кодировани

Свойства кода
Конструкция
Плоткина
Минимальное

Параметры

Декодирования

Алгоритм Рида

Домашнее задание

задание Лсточникі \blacksquare Существует порождающая матрица G.

$$C(x) = x_{1 \times k} G_{k \times n} = c_{1 \times n}$$

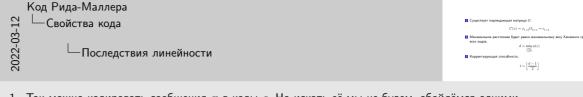
Минимальное расстояние будет равно минимальному весу Хемминга среди всех кодов.

$$d = \min_{\substack{c \in C \\ c \neq 0}} w(c)$$

3 Корректирующая способность:

$$t = \left| \frac{d-1}{2} \right|$$





- 1. Так можно кодировать сообщения x в коды c. Но искать её мы не будем, обойдёмся одними многочленами, это интереснее.
- Вес Хэмминга вектора количество в нём ненулевых элементов.
- 3. Доказательство очень просто: минимальное расстояние вес разности каких-то двух различных кодов, но разность двух кодов тоже будет кодом, т.к. мы в линейном пространстве. Значит достаточно найти минимальный вес, но не учитывая нулевой вектор, т.к. разность равна нулю тогда и только тогда, когда коды равны.
- 4. Однако мы ещё не знаем как выглядят наши коды (как выглядят таблицы истинности функций степени не больше r?). А значит не можем ничего сказать про минимальное расстояние.



Конструкция Плоткина: многочлены

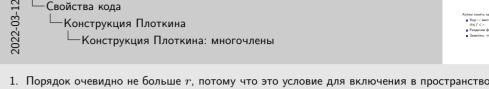
Рида-Маллера

Конструкция Плоткина

Хотим понять как выглядят кодовые слова.

- Код вектор значений функции $f(x_1,...,x_m) \in RM(r,m)$, причём $\deg f < r$.
- Разделим функцию по x_1 : $f(x_1,...,x_m) = g(x_2,...,x_m) + x_1h(x_2,...,x_m)$.
- \blacksquare Заметим, что $\deg f \le r$, а значит $\deg q \le r$ и $\deg h \le r-1$.

4 D > 4 A > 4 B > 4 B > B 9 Q Q



Код Рида-Маллера

- 1. Порядок очевидно не больше r, потому что это условие для включения в пространство кодов RM(r, m).
- 2. Теперь у нас есть две функции от меньшего числа аргументов. Очевидно, так можно сделать всегда, когда m > 1.



Конструкция Плоткина: таблица истинности

Код Рида-Маллера

Введени

Содирован

Конструкция
Плоткина
Минимальное

Декодировани Алгоритм Рида

Домашнее задание

Істочники

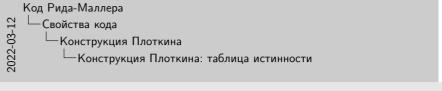
Ранее: $f(x_1,...,x_m) = g(x_2,...,x_m) + x_1h(x_2,...,x_m)$.

■ Заметим, что таблица истинности f состоит из двух частей: при $x_1 = 0$ и при $x_1 = 1$.

$$\operatorname{Eval}(f) = \left(\frac{\operatorname{Eval}^{[x_1=0]}(f)}{\operatorname{Eval}^{[x_1=1]}(f)}\right)$$

- \blacksquare Причём $\operatorname{Eval}^{[x_1=0]}(f) = \operatorname{Eval}(q)$, а $\operatorname{Eval}^{[x_1=0]}(f) \oplus \operatorname{Eval}^{[x_1=1]}(f) = \operatorname{Eval}(h)$.
- Таким образом, $\operatorname{Eval}(f) = \operatorname{Eval}(g)$, а $\operatorname{Eval}(f) \oplus \operatorname{Eval}(h)$.





Ранес: $f(x_1,...,x_m) = g(x_2,...,x_m) + x_1h(x_2,...,x_m)$. ■ Замятим, что таблица истинности f состоит из двух частей: при $x_1 = 0$ при $x_1 = 1$. $\text{Evol}(f) = \left(\text{Eval}^{(x_1 = 0)}(f) \right)$

Тричём $\text{Eval}^{[x_1=0]}(f) = \text{Eval}(g)$, а $\text{Eval}^{[x_1]}$

- Таким образом, $\text{Eval}(f) = (\text{Eval}(g) \mid \text{Eval}(g))$
- 1. Теперь рассмотрим те же функции, но со стороны их таблиц истинности. Нам же интересны именно коды, а они как раз очень тесно связаны с этими таблицами.
- 2. Про обозначения: $\operatorname{Eval}(f)$ таблица для всей функции (вектор значений, если точнее), $\operatorname{Eval}^{[x_1=0]}(f)$ кусок таблицы при $x_1=0$, $\operatorname{Eval}^{[x_1=1]}(f)$ кусок таблицы при $x_1=1$. Они нам после этого доказательства больше не понадобятся.
- 3. Это всё следует из ранее полученного утверждения. Если мы подставим $x_1=0$, то останется только g первое равенство очевидно. Если же мы рассмотрим $\operatorname{Eval}^{[x_1=1]}(f)$, то получим $\operatorname{Eval}(g+h)$, но если туда прибавить ещё раз $\operatorname{Eval}(g)$, то останется только $\operatorname{Eval}(h)$ (поскольку 1+1=0 в \mathbb{F}_2) получили второе равенство.
- 4. Палочка по центру конкатенация векторов.



Конструкция Плоткина: вывод

Рида-Маллер:

Конструкция Плоткина

Если дана $f(x_1,...,x_m)$, причём $\deg f < r$, то можно её разделить:

$$f(x_1, ..., x_m) = g(x_2, ..., x_m) + x_1 h(x_2, ..., x_m)$$

Также известно, что $\text{Eval}(f) = (\text{Eval}(g) \mid \text{Eval}(g) \oplus \text{Eval}(h)).$

Заметим, что Eval(f) – кодовое слово (как и для q и h).

Тогда:
$$c=\operatorname{Eval}(f)\in\operatorname{RM}(r,m)$$
 (т.к. $\deg f\leq r$) $u=\operatorname{Eval}(q)\in\operatorname{RM}(r,m-1)$ (т.к. $\deg q\leq r$)

$$v = \operatorname{Eval}(h) \in \operatorname{RM}(r-1, m-1)$$
 (t.k. $\deg h \le r-1$)



- 1. Теперь собираем всё это в одно важное утверждение.
- 2. Причём мы уже знаем, что $\deg q < r$ и $\deg h < r 1$, если $\deg f < r$
- 3. Напомню, что RM(r, m) включает в себя все функции (их таблицы истинности, если точнее) от mаргументов и степени не выше r. Очевидно, наши годятся.

Конструкция Плоткина

Рида-Маллера

Конструкция Плоткина

Теорема

Для всякого кодового слова $c\in\mathrm{RM}(r,m)$ можно найти $u\in\mathrm{RM}(r,m-1)$ и $v \in RM(r-1, m-1)$, такие что $c = (u \mid u+v)$.

-Свойства кода Конструкция Плоткина Конструкция Плоткина

Код Рида-Маллера

1. Что здесь важно отметить — оба наших новых кодовых слова u, v получились «меньше», чем исходное c. Это позволяет, во-первых, устраивать индукцию, чем мы скоро и займёмся. Во-вторых, это позволяет легко строить большие порождающие матрицы, но мы этим не будем заниматься.

Минимальное расстояние

Рида-Маллера

Хотим найти минимальное расстояние для кода $\mathrm{RM}(r,m)$

$$d = \min_{c \in C, c \neq 0} w(c)$$

Предположим, что $d=2^{m-r}$ и докажем по индукции.

База: $\mathrm{RM}(0,m)$ — единственный бит повторён 2^m раз. Очевидно, $w(\underbrace{11...1}) = 2^m = 2^{m-0} \geq 2^{m-r}.$

Гипотеза: Если $v \in \text{RM}(r-1, m-1)$, то $w(v) \geq 2^{m-r}$.

Шаг: Хотим доказать для $c \in \mathrm{RM}(r,m)$.

$$\begin{split} w(c) &\stackrel{(1)}{=} w((u \mid u \oplus v)) \stackrel{(2)}{=} w(u) + w(u \oplus v) \geq \\ &\stackrel{(3)}{\geq} w(u) + (w(v) - w(u)) = w(v) \stackrel{IH}{\geq} 2^{m-r} \blacksquare \end{split}$$

Код Рида-Маллера
СТ-Свойства кода
— Минимальное расстояние
— Минимальное расстояние

отим найти минимальное расстояние для кода $\mathrm{RM}(r,m)$ $d = \min_{c \in \mathcal{C}, r \neq 0} w(c)$ редположим, что $d = 2^{m-r}$ и досажем по индукциях.

адах: $\mathrm{RM}(0,m) = \mathrm{e}_{\mathrm{RM}} \mathrm{e}_{\mathrm{CM}} \mathrm{e}_{\mathrm{CM}}$ $(11...1) = 2^{m-2} - 2^{m-r}$. $2^{m-r} = 2^{m-r} - 2^{m-r}$.

 $w(c) \stackrel{(1)}{=} w((u \mid u \oplus v)) \stackrel{(2)}{=} w(u) + w(u \oplus v) \ge$ $\stackrel{(3)}{\geq} w(u) + (w(v) - w(u)) = w(v) \ge 2^{m-r} \blacksquare$

- 1. Случай $\mathrm{RM}(0,m)$ мы разбирали раньше, но я напомню. Здесь длина сообщения равна $k=\sum_{i=0}^r C_m^i=C_m^0=1$, а длина кода $n=2^m$. Причём мы просто берём один бит и повторяем его 2^m раз (в таблице истинности). Замечу, что не рассматриваю второй случай $w(\mathtt{00...0})$, поскольку он нам не нужен для расчёта
 - Замечу, что не рассматриваю второй случай $w(\mathsf{00...0})$, поскольку он нам не нужен для расчёта минимального расстояния. Вариант с нулевым вектором явно выкидывается, см. определение d выше.
- 2. Теперь немного объяснений.

2. Теперь немного объяснении. Переход (1): используем конструкцию Плоткина, чтобы разбить c на конкатенацию двух кодовых

слов поменьше. Переход (2): $w((x \mid y)) = w(x) + w(y)$. Вес это всего лишь число ненулевых элементов, поэтому нет разницы как мы будем группировать части вектора.

Переход (3): $w(u\oplus v)\geq w(v)-w(u)$. Если у нас в v стоит w(v) бит, то прибавив к нему u, мы сможем изменить (обнулить) не больше w(u) бит. Возможно появится больше единиц, но нас интересует инжняя граница.

Переход (ІН): предположение индукции в чистом виде.



Свойства и параметры

Рида-Маллера

Параметры

Для бинарного кода RM(r, m): r < m

 \blacksquare Длина кода: 2^m

■ Длина сообщения: $k = \sum_{i=0}^{r} C_{m}^{i}$

■ Минимальное расстояние: $d = 2^{m-r}$

■ Корректирующая способность: $t = 2^{m-r-1} - 1$

 \blacksquare Существует порождающая матрица G для кодирования

 \blacksquare Проверочная матрица H совпадает с порождающей для $\mathrm{RM}(m-r-1,m)$

Код Рида-Маллера -Свойства кода Длина кола: 2⁴⁴ —Параметры Длина сообщения: k = ∑^c , Cⁱ , Свойства и параметры

1. Теперь можно подвести итоги исследования свойств.

2. , поскольку $t=\left|\frac{d-1}{2}\right|=\left|\frac{2^{m-r}}{2}-\frac{1}{2}\right|=\left|2^{m-r-1}-0.5\right|=2^{m-r-1}-1$

3. , она позволяет делать так: C(x) = xG. Но я, как обычно, её избегаю. Рекомендую почитать «Коды Рида-Маллера: Примеры исправления ошибок», если интересно.

4. , но это я это доказывать не собираюсь. Однако доказательство можно найти в «Reed-Muller Codes: Theory and Algorithms», раздел Duality.



Возможные варианты

Рида-Маллера

1	k = 1 $n = 2$ $t = 0$	k = 2 $n = 2$ $t = 0$	_	_	
2	k = 1 $n = 4$ $t = 1$	k = 3 $n = 4$ $t = 0$	k = 4 $n = 4$ $t = 0$	_	
3	k = 1 $n = 8$ $t = 3$	k = 4 $n = 8$ $t = 1$	k = 7 $n = 8$ $t = 0$	k = 8 $n = 8$ $t = 0$	
4	k = 1 $n = 16$ $t = 7$	k = 5 $n = 16$ $t = 3$	k = 11 $n = 16$ $t = 1$	k = 15 $n = 16$ $t = 0$	k = n = t

Код Рида-Маллера -Свойства кода 2022-03-–Параметры \sqsubseteq Возможные варианты 1. У красных кодов минимальное расстояние d равно единице — они совершенно бесполезны, там количество кодов равно количеству сообщений; у желтых кодов d=2 — они могут определить наличие ошибки, но не могут её исправить. Для всех остальных кодов d=2(t+1).

3. И кстати, случай m=0, k=0 (не влез) будет собой представлять колирование единственного бита

2. Напоминание: k — длина сообщения, n — длина кода, а t — количество ошибок, которое код точно сможет исправить. Заодно о параметрах кода: m — количество переменных у функции (очень

совершенно без изменений.

влияет на длину кода), а r — максимальная степень многочлена (очень влияет на длину сообщения, и соотвественно надёжность кода), причём $r \le m$. Конечно, таблицу можно продолжать и дальше.



Как линейный код

Рида-Маллера

Волонио

Колировани

Кодирован

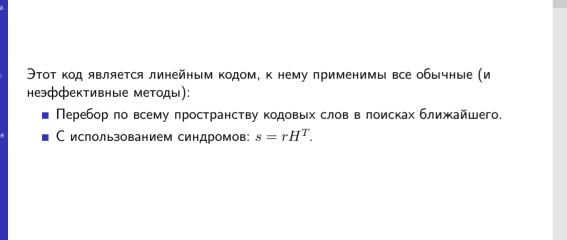
Конструкци Плоткина Минимальн

Минималь расстояни Параметрь

Декодировании Алгоритм Рида

Домашне задание

задани



イロト 4周トイミトイミト ヨー かなべ



1. Этот способ применим ко всем кодам, но никто в здравом уме им не пользуется.

Код Рида-Маллера

- 2. Здесь s синдром, r полученное сообщение, H проверочная матрица. Этот метод обычен для линейных кодов.
- 3. Эти способы нужно иметь в виду, но о них было рассказано и без меня, так что я их пропущу.

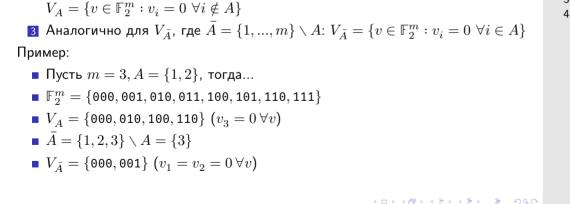


Определения

1 Пусть $A \subseteq \{1,...,m\}$ для $m \in \mathbb{N}$

Рида-Маллера

Алгоритм Рида



2 Подпространство $V_A \subseteq \mathbb{F}_2^m$, которое обнуляет все v_i , если $i \notin A$:

-Декодирование Ananorwino and V_1 , rae $\tilde{A} = \{1, ..., m\} \setminus A$; $V_1 = \{v \in \mathbb{F}^n : v_i = 0 \ \forall i \in A\}$ –Алгоритм Рида Пусть m = 3, A = {1, 2}, тогда... —Определения $V_1 = \{000,001\} (v_1 = v_2 = 0 \forall v)$ 1. Начать стоит с нескольких определений, без которых алгоритм Рида объяснить не получится. 2. — все 8 векторов этого пространства

Код Рида-Маллера

- 3. обнулилась третья позиция, первые две остались
- 4. осталась только третья позиция, остальные обнулились.

Смежные классы

 $V_A + b$:

Рида-Маллера

Алгоритм Рида



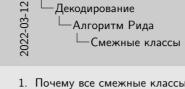
все различны (и это будут все смежные классы).

Если фиксировано $V_A\subseteq \mathbb{F}_2^m$, то для каждого $b\in \mathbb{F}_2^m$ существует смежный класс

 $(V_A + b) = \{v + b \mid v \in V_A\}$

Утверждается, что если брать $b \in V_{\bar{A}}$, то полученные смежные классы будут

イロト 4周トイミトイミト ヨー かなべ



Код Рида-Маллера

Утверждается, что если брать $b \in V_{z}$, то полученные смежные классы будут

Если фиксировано $V_A \subseteq \mathbb{F}_+^m$, то для каждого $b \in \mathbb{F}_+^m$ существует смежный класс

1. Почему все смежные классы (V_A+b) можно получить именно перебором $b\in V_{\bar{A}}$ можно найти в разделе «Дополнительные доказательства» из пдфки



Рида-Маллера

Декодирует сообщение u, если использовался RM(r, m). Для RM(2, 2): $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ **Data:** vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$ for $t \leftarrow r$ to 0 foreach $A \subseteq \{1, ..., m\}$ with |A| = tc = 0foreach $b \in V_{\bar{A}}$ $c \mathrel{+}= \left(\sum\limits_{z\in (V_A+b)} y_z
ight) mod 2$ $y = \operatorname{Eval}\left(\sum_{\substack{A \subseteq \{1, \dots, m\} \ A \subseteq \{A, \dots, M\}}} u_A \prod_{i \in A} x_i \right)$

На вход поступает бинарный вектор u длины 2^{m} . Это вектор значений функции, возможно с ошибками (но их не больше. чем $t = 2^{m-r-1} - 1$).





- 1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает — см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.
- 2. Цель восстановить все коэффициенты при многочлене вида $f(x_1,...,x_m) = u_{\varnothing} + u_1x_1 + x_2x_2 + ... + u_{1,2,...,r}x_{1,2,...,r}$, где $\deg f \leq r$. Обратите внимание, что для индексов при u используются подмножества $A\subseteq \{1,...,m\}, |A|\leq r$, причём каждый u_A умножается на моном $\prod_{i \in A} x_i$.



Декодирует сообщение u, если использовался RM(r,m). Для RM(2,2):

Рида-Маллера

Алгоритм Рида

 $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ **Data:** vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$ for $t \leftarrow r$ to 0 foreach $A \subseteq \{1, ..., m\}$ with |A| = tc = 0foreach $b \in V_{\bar{A}}$ $c \mathrel{+}= \left(\sum_{z \in (V_A + b)} y_z
ight) mod 2$ $y == \operatorname{Eval} \left(\sum_{\substack{A \subseteq \{1, \dots, m\} \\ |A| = \ell}} u_A \prod_{i \in A} x_i \right)$

Будем восстанавливать сначала коэффициенты u_A при старших степенях, потом поменьше и так пока не восстановим их все. Hачинаем с t=r.



Код Рида-Маллера -Декодирование —Алгоритм Рида \square Алгоритм Рида для кода $\mathrm{RM}(r,m)$ $y = \text{Eval}\left(\sum_{A \subseteq \left\{\frac{1}{1-\epsilon}=1\right\}} u_A \prod_{i \in A} x_i\right)$

1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает — см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.



Рида-Маллера

Алгоритм Рида

Декодирует сообщение u, если использовался RM(r, m). Для RM(2, 2): $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ Хотим восстановить все **Data:** vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$ коэффициенты при мономах for $t \leftarrow r$ to 0 степени t. Для этого foreach $A \subseteq \{1, ..., m\}$ with |A| = tперебираем все A, |A| = t и для c = 0каждого восстанавливаем foreach $b \in V_{\bar{A}}$ $c \mathrel{+}= \left(\sum\limits_{z\in (V_A+b)} y_z
ight) mod 2$ коэффициент u_A при $x_{A_1} x_{A_2} ... x_{A_t}$. $u_A \leftarrow 1 \left[c \geq 2^{m-t-1}\right]$ $y = \operatorname{Eval}\left(\sum_{\substack{A \subseteq \{1,\dots,m\} \ |A|=A}} u_A \prod_{i \in A} x_i \right)$ 4 D > 4 B > 4 B > 4 B > 9 Q P Код Рида-Маллера -Декодирование –Алгоритм Рида \square Алгоритм Рида для кода $\mathrm{RM}(r,m)$ $y = \text{Eval}\left(\sum_{A \subseteq \{\frac{1}{2}, \dots, m\}} u_A \prod_{i \in A} x_i\right)$

1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает — см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.



Data: vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$

foreach $b \in V_{\bar{A}}$

foreach $A \subseteq \{1, ..., m\}$ with |A| = t

Декодирует сообщение u, если использовался RM(r, m). Для RM(2, 2):

Рида-Маллера

Алгоритм Рида

 $c \mathrel{+}= \left(\sum_{z \in (V_A + b)} y_z
ight) mod 2$ $u_A \leftarrow \mathbf{1} \left[c \geq 2^{m-t-1} \right]$ $y = \operatorname{Eval}\left(\sum_{\substack{A \subseteq \{1, \dots, m\} \\ |A| = \ell}} u_A \prod_{i \in A} x_i\right)$

c = 0

for $t \leftarrow r$ to 0

 $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ Чтобы восстановить коэффициент, нужно перебрать все смежные классы вида $(V_A + b)$: $V_A = \{v \in \mathbb{F}_2^m \mid v \in \mathbb{F}_2^m \mid v \in \mathbb{F}_2^m \mid v \in \mathbb{F}_2^m \}$ $: v_i = 0 \ \forall i \notin A$ $b \in \{v \in \mathbb{F}_2^m\}$ $: v_i = 0 \ \forall i \in A$



Код Рида-Маллера -Декодирование –Алгоритм Рида Bee embound kracem by, $(V_A+b):$ $V_A=\{v\in\mathbb{F}_2^m \\ : v_i=0 \ \forall i\notin A\}$ $b\in\{v\in\mathbb{F}_2^m \$ \square Алгоритм Рида для кода $\mathrm{RM}(r,m)$

1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает — см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.



Код Рида-Маллера

Зведение

Кодировани

Свойства ко Конструкция Плоткина

Конструкция Плоткина Минимальное расстояние

Параметры

Декодирован

Алгоритм Рида

Пример

Астоини

Декодирует сообщение u, если использовался RM(r, m). Для RM(2, 2): $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ Считаем количество (c)**Data:** vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$ смежных классов, в которых for $t \leftarrow r$ to 0 $\sum y_z = 1 \pmod{2}$. foreach $A \subseteq \{1, ..., m\}$ with |A| = tc = 0Пороговое значение (2^{m-t-1}) foreach $b \in V_{\bar{A}}$ здесь — половина от числа $c \mathrel{+}= \left(\textstyle\sum_{z \in (V_A + b)} y_z\right) \bmod 2$ смежных классов. Таким образом, если большинство сумм дало 1, то $u_A = 1$, иначе $y == \operatorname{Eval} \left(\sum_{\substack{A \subseteq \{1, \dots, m\} \\ |A| = t}} u_A \prod_{i \in A} x_i \right)$ $u_{A}=0.$

4 D > 4 D > 4 D > 4 D > 1 D 9 Q Q

- 1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.
- 2. Если это количество больше порогового значения, то считаем, что $u_A = 1$, иначе же $u_A = 0$.



Код Рида-Маллера

Зведение

Кодировани

Свойства ко Конструкция Плоткина

Конструкция Плоткина Минимальное расстояние

Параметры Декодирован Алгоритм Рида

_{Пример} Домашнее

адание

Декодирует сообщение u, если использовался RM(r, m). Для RM(2, 2): $f(x_1, x_2) = u_{\{1,2\}} x_1 x_2 + u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\varnothing}.$ Затем мы вычитаем из y**Data:** vector $y = (y_z \in \mathbb{F}_2 \mid z \in \mathbb{F}_2^m)$ (вектор значений функции) всё for $t \leftarrow r$ to 0 найденное на этой итерации. foreach $A \subseteq \{1, ..., m\}$ with |A| = tпосле чего переходим к c = 0мономам меньшей степени. foreach $b \in V_{\bar{A}}$ Повторять до восстановления всех коэффициентов. $u_A \leftarrow \mathbf{1} \left[c \geq 2^{m-t-1} \right]$ $y = \operatorname{Eval}\left(\sum_{\substack{A \subseteq \{1,\dots,m\} \ |A|=t}} u_A \prod_{i \in A} x_i
ight)$





1. Теперь, наконец, сам алгоритм Рида с объяснением, что тут происходит. Почему он именно такой и почему это работает — см. раздел (на русском) «Reed's Algorithm: Unique decoding up to half the code distance» [??] в пдфке.



Ранее: 101 кодируется как 1100 при помощи ${
m RM}(1,2)$

Рида-Маллера

 $101 \leadsto (f(x_1, x_2) = x_1 + 1) \leadsto \begin{vmatrix} x_1 & x_2 & f \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{vmatrix} \leadsto \begin{cases} y_{00} = 1 \\ y_{01} = 1 \\ y_{10} = 0 \\ y_{11} = 0 \end{cases} \Longrightarrow 1100$ イロト 4周トイミトイミト ヨー かなべ

Код Рида-Маллера —Декодирование —Алгоритм Рида □Пример

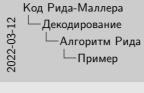
1. Как происходит кодирование, схематически:



Код Рида-Маллера

Ранее: 101 кодируется как 1100 при помощи RM(1,2)Положим $y_{00} = 1, y_{01} = 1, y_{10} = 0, y_{11} = 0$ Здесь m=2, значит $A \subset \{1,2\}$. Причём r=1, т.е. |A| < 1. Шаг 1/3: $t=1, A=\{1\}$ lacktriangle Здесь $V_A=\{\mathtt{00},\mathtt{10}\}$, $V_{ar{A}}=\{\mathtt{00},\mathtt{01}\}.$ Нужно рассмотреть два смежных класса. $(V_A + 00) = \{00, 10\}, \text{ cymma: } y_{00} + y_{10} = 1 + 0 = 1$ $(V_A + 01) = \{01, 11\}, \text{ cymma: } y_{01} + y_{11} = 1 + 0 = 1$ ■ Итого: $u_A = u_{\{1\}} = 1$





- Здась m=2. значит $A \subseteq \{1,2\}$. Причём r=1. т.е. $|A| \le 1$ Har 1/3: t = 1: $A = \{1\}$
- $(V_1 + 00) = \{00, 10\}, \text{ cyama: } v_{-1} + v_{-1} = 1 + 0 = 1$

Повором $v_{rr} = 1$ $v_{rr} = 1$ $v_{rr} = 0$ $v_{rr} = 0$

■ Utoro: $u_A = u_{(1)} = 1$

- 1. Теперь начинаем декодирование.
- 2. по одному на каждый вектор из $V_{ar{A}}$



Код Рида-Маллера

 $(V_A + 10) = \{10, 11\}, \text{ cymma: } y_{10} + y_{11} = 0 + 0 = 0$ ■ Итого: $u_A = u_{\{2\}} = 0$

Ранее: 101 кодируется как 1100 при помощи RM(1,2)

Здесь m=2, значит $A \subset \{1,2\}$. Причём r=1, т.е. |A| < 1.

 $(V_A + 00) = \{00, 01\}, \text{ cymma: } y_{00} + y_{01} = 1 + 1 = 0$

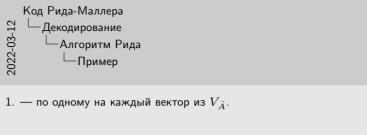
Положим $y_{00} = 1, y_{01} = 1, y_{10} = 0, y_{11} = 0$

 $lacksymbol{\blacksquare}$ Здесь $V_A=\{\mathtt{00},\mathtt{01}\}$, $V_{ar{A}}=\{\mathtt{00},\mathtt{10}\}.$

Нужно рассмотреть два смежных класса

Шаг 2/3: $t=1, A=\{2\}$

イロト 4周トイミトイミト ヨー かなべ



Здась m=2. значит $A \subseteq \{1,2\}$. Причём r=1. т.е. $|A| \le 1$ IIIar 2/3: t = 1 $A = \{2\}$

 $v(V_1 + 00) = \{00, 01\}$, cyama: $v_{rr} + v_{rr} = 1 + 1 = 0$ $(V_4 + 10) = \{10, 11\}, \text{ cymma: } y_{10} + y_{11} = 0 + 0 = 0$

Повором $v_{rr} = 1$ $v_{rr} = 1$ $v_{rr} = 0$ $v_{rr} = 0$

■ Mroro: u_A = u_{PN} = 0



Ранее: 101 кодируется как 1100 при помощи RM(1,2)Положим $y_{00} = 1, y_{01} = 1, y_{10} = 0, y_{11} = 0$ Здесь m=2, значит $A\subseteq \{1,2\}$. Причём r=1, т.е. |A|<1.

Тогда $y \leftarrow y - \text{Eval}(g) = 1100 \oplus 0011 = 1111.$ 4 D > 4 B > 4 B > 4 B > 9 Q P

функции: $g(x_1, x_2) = u_{\{2\}}x_2 + u_{\{1\}}x_1 = 0x_2 + 1x_1 = x_1$ Вычислим $\operatorname{Eval}(g)$: x_1 $x_2 \mid g(x_1, x_2)$

Перед переходом к t=0, нужно вычесть из y вектор значений следующей

Код Рида-Маллера -Декодирование

–Алгоритм Рида

<u></u>Пример

3. Полезно заметить, что в \mathbb{F}_2 сложение и вычитание — одно и то же.

- 1. Здесь мы берём все u, полученные при t=1, домножаем каждую на соответствущие ей x-ы и
- получаем функцию от m переменных. 2. Очень важно, чтобы у вас во всех таблицах истинности (в т.ч. той, которая использовалась при
- кодировании для получения y) был одинаковый порядок строк. Иначе чуда не выйдет.

 $g(x_1, x_2) = u_{231}x_2 + u_{211}x_1 = 0x_2 + 1x_1 = x_1$

Продолжение примера: t=0

Теперь $y_{00} = 1, y_{01} = 1, y_{10} = 1, y_{11} = 1$

lacktriangle Здесь $V_A=\{00\}$, но $V_{ar A}=\{00,01,10,11\}$. Нужно рассмотреть четыре смежных класса.

Рида-Маллера

Шаг 3/3: $t = 0, A = \emptyset$

 $(V_A + 00) = \{00\}, \text{ cymma: } y_{00} = 1$ $(V_A + 01) = \{01\}, \text{ cymma: } y_{01} = 1$ $(V_A + 10) = \{10\}, \text{ cymma: } y_{10} = 1$ $(V_A + 11) = \{11\}, \text{ cymma: } y_{11} = 1$ ■ Итого: $u_A = u_{\emptyset} = 1$





Продолжение примера: t=0

Теперь $y_{00} = 1, y_{01} = 1, y_{10} = 1, y_{11} = 1$

Это значит, что исходный многочлен был таков:

а исходное сообщение: 101, как и ожидалось.

Получили $u_{\{2\}}=1, u_{\{1\}}=0, u_{\varnothing}=1.$

Код Рида-Маллера

Время работы

Утверждается, что время работы алгоритма — $O(n \log^r n)$, где $n = 2^m$ длина кода.

 $f(x_1, x_2) = u_{\{2\}} x_2 + u_{\{1\}} x_1 + u_{\emptyset} = \frac{x_2 + 1}{2},$

4 D > 4 A > 4 B > 4 B > B 9 Q P

Код Рида-Маллера –Декодирование —Алгоритм Рида \square Продолжение примера: t=0

 $f(x_1, x_2) = u_{22}x_2 + u_{22}x_3 + u_{23} = x_2 + 1$



Рида-Маллера

Домашнее задание

Домашнее задание

RM(1,3)

Вариант 1

1 Закодировать сообщение: 1001.

2 Декодировать код, если ошибок нет: 1010, использовался RM(1,2).

3 Декодировать код, полученный с ошибками: 1101 1010, использовался RM(1,3)

Вариант 2

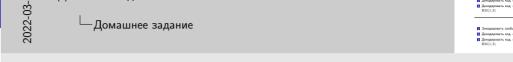
1 Закодировать сообщение: 0101.

2 Декодировать код, если ошибок нет: 0110, использовался RM(1,2).

3 Декодировать код, полученный с ошибками: 1111 0100, использовался

4 D > 4 A > 4 B > 4 B > B 9 Q P

Код Рида-Маллера -Домашнее задание



1. Замечание: каких-либо требований на методы решения нет, но если используете код — приложите его. Различных способов решить существует больше одного. Номер варианта можете определять как $1 + ((5n + 98) \bmod 2)$, но главное напишите его и своё имя.



Источники

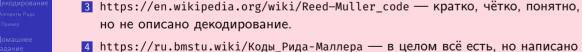
рекомендую.

это не весело.

но не описано декодирование.

Рида-Маллера

Источники



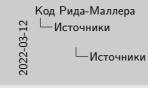
4 https://ru.bmstu.wiki/Коды Рида-Маллера — в целом всё есть, но написано очень непонятно:

1 https://arxiv.org/pdf/2002.03317.pdf — великолепный обзор, очень

подробно, но используется подход через матрицы, а не через полиномы, а

2 http://dha.spb.ru/PDF/ReedMullerExamples.pdf — очень хорошо и





- https://arxiv.ore/odf/2002.03317.odf великолепный обасо, очень ## http://dbs.sch.ru/DDE/SpedMollerEverples.ndf -- nasus, wonoun #
- https://ru.bmstu.wiki/Kogw_Pwga-Mannepa в целом воё есть, но написано