УДК 621.391.3.004.14

Кубицкий В. И.

КОДИРОВАНИЕ ДЛЯ НЕПОЛНОГО КОДА ЛАГРАНЖА

ГосНИИ «Аэронавигация» (Россия, Москва)

Рассматриваются процедуры кодирования неполного кода Лагранжа. Выводятся соотношения и предлагаются алгоритмы реализации этих процедур. Определяется сложность реализации данных процедур и дается сравнительная оценка этой сложности.

В работах, посвященных исследованию свойств и разработкам процедур кодирования-декодирования кодов Лагранжа, рассматриваются, в основном, полные коды [1-3]. То есть коды, для которых мощность M множества узлов интерполирования равна порядку P конечного поля $GF(p^m)$. Представляют интерес также неполные коды Лагранжа, где $M<P$. К таким кодам приходим, в частности, при исправлении ошибок, возникающих наряду со стираниями [3].

В [4], [5] разработаны алгоритмы кодирования для полных кодов Лагранжа и введены понятия параллельного, последовательного, параллельно-последовательного алгоритмов кодирования. Здесь рассмотрим аналогичные алгоритмы кодирования для неполного кода Лагранжа.

Пусть $S,\ T, V \subset F_q$ – подмножества поля F_q. Здесь:

$S = \{x_o, x_1, \cdots, x_s\}$ – подмножество информационных узлов мощности k;

$T = \{\beta_1, \beta_2, \cdots, \beta_r\}$ – подмножество контрольных узлов мощности r;

$V = F_q \setminus (S \cup T) = \{v_1, \ldots, v_e\}$.

Докажем следующие леммы.

Лемма 1. Для любых S, T, $V \subset F_q$; $V = F_q \setminus (S \cup T) \neq \emptyset$; $S \cap T = \emptyset$ при параллельном алгоритме кодирования выполняются соотношения:

$$L_S^{(i)}(\beta_j) = -\prod_{\substack{\beta_l \in T \\ l \neq j}} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi} \quad .(1)$$

Доказательство. Запишем отношение:

$$\frac{L_S^{(i)}(\beta_j)}{\prod\limits_{\substack{\beta_l \in T \\ l \neq j}} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod\limits_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi}} =$$

$$= \frac{\prod\limits_{\substack{x_h \in S \\ h \neq i}} (\beta_j - x_h) \prod\limits_{\substack{\beta_l \in T \\ l \neq j}} (\beta_j - \beta_l) \prod\limits_{v_\xi \in V} (\beta_j - v_\xi)}{\prod\limits_{\substack{x_h \in s \\ h \neq i}} (x_i - x_h) \prod\limits_{\substack{\beta_l \in T \\ l \neq j}} (x_i - \beta_l) \prod\limits_{v_\xi \in V} (x_i - v_\xi)} .$$

Умножим числитель и знаменатель этой дроби на величину $(\beta_j - x_i)$. Очевидно, что $(\beta_j - x_i) = -(x_i - \beta_j)$. Так как $x_i \in S$, $\beta_j \in T$ и , то $(\beta_j - x_i) \neq 0$. Тогда в числителе и знаменателе будем иметь произведение всех ненулевых элементов поля F_q, которое равно единице. Тем самым доказана справедливость леммы 1.

Аналогично доказывается следующая лемма:

Лемма 2. Для любых $S,\ T,\ V \subset F_q$; $V = F_q \setminus (S \cup T) \neq \emptyset$; $S \cap T = \emptyset$ при последовательном алгоритме кодирования выполняются соотношения:

$$L_{S_{j-1}}^{(i)}(\beta_j) = -\prod_{\beta_l \in T_j} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi}, \quad (2)$$

где:

$$T_j = T \setminus \{\beta_1, \ldots, \beta_j\}, \quad T_r = \emptyset, \quad S_{j-1} = S \cup \{\beta_1, \ldots, \beta_{j-1}\}.$$

Докажем следующие предложения.

Предложение 1. Контрольные символы кодового слова неполного кода Лагранжа вычисляются из соотношений:

$$f(\beta_j) = -\sum_{i=0}^{s} f_i \prod_{\substack{\beta_l \in T \\ l \neq j}} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi}, \quad (3)$$

Доказательство. Известно, что при параллельном алгоритме кодирования для полного кода Лагранжа контрольные символы кодового слова вычисляются из соотношений [4]-[6]:

$$f(\beta_j) = -\sum_{i=0}^{s} f_i L_S^{(i)}(\beta_j) \quad , \; j = \overline{1,r} \; , \quad (4)$$

где: f_i – значения кодового полинома в информационных узлах;

$L_S^{(i)}(\beta_j)$ – фундаментальные полиномы Лагранжа в контрольных узлах.

Подставляя (1) в (4), получим (3).

Назовем алгоритм вычисления контрольных символов кода Лагранжа в соответствии с выражением (3) *параллельным алгоритмом кодирования неполного кода Лагранжа.*

Предложение 2. Контрольные символы кодового слова неполного кода Лагранжа вычисляются из соотношений:

$$f(\beta_j) = -\sum_{i=0}^{s+j-1} f_i \prod_{\beta_l \in T_j} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi}, \quad (5)$$

Доказательство. Известно, что при последовательном алгоритме кодирования для полного кода Лагранжа контрольные символы кодового слова вычисляются из соотношений [4]-[5]:

$$f(\beta_j) = -\sum_{i=0}^{s+j-1} f_i L_{S_{j-1}}^{(i)}(\beta_j) \quad , \quad (6)$$

где: f_i – значения кодового полинома в информационных узлах и вычисленных ранее значения f_{s+j-1} этого полинома в предыдущих (j-1)-ых контрольных узлах;

$L_{S_{j-1}}^{(i)}(\beta_j)$ – фундаментальные полиномы Лагранжа в контрольных узлах;

$S_{j-1} = S \cup \{\beta_1, \ldots, \beta_{j-1}\}$;

$x_i \in S_{j-1}$.

Подставляя (2) в (6), получим (5).

Назовем алгоритм вычисления контрольных символов кода Лагранжа в соответствии с выражением (5) *последовательным алгоритмом кодирования неполного кода Лагранжа.*

Очевидна справедливость следующего предложения.

Предложение 3. Контрольные символы кодового слова неполного кода Лагранжа вычисляются из соотношений:

$$f(\beta_j) = -\sum_{i=0}^{s} f_i \prod_{\beta_l \in T} \frac{x_i - \beta_l}{\beta_j - \beta_l} \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_j - v_\xi}, \quad (7)$$

$$j = \overline{1, r-1}$$

$$f(\beta_r) = -\sum_{i=0}^{s+r-1} f_i \prod_{v_\xi \in V} \frac{x_i - v_\xi}{\beta_r - v_\xi}.$$

Алгоритм вычисления контрольных символов кода Лагранжа в соответствии с выражением (7) назовем *параллельно-последовательным алгоритмом кодирования неполного кода Лагранжа.*

Разработанные процедуры кодирования неполного кода Лагранжа могут использоваться в процедурах исправления ошибок, имеющих место наряду со стираниями в случае полного кода Лагранжа. При этом $V = F_q \setminus (S \cup T) = \{v_1, \ldots, v_l\}$ является подмножеством стертых узлов.

Составим алгоритмы вычисления контрольных символов для неполного кода Лагранжа.

1. Параллельный алгоритм кодирования.

Запишем выражение (3) для параллельного алгоритма кодирования в следующем виде:

$$f_{s+1} = -\frac{1}{\prod\limits_{\beta_l \in T_1}(\beta_1 - \beta_l)\prod\limits_{\nu_\xi \in V}(\beta_1 - \nu_\xi)} \sum_{i=0}^{s} f_i \prod_{\nu_\xi \in V}(x_i - \nu_\xi) \prod_{\beta_l \in T_1}(x_i - \beta_l),$$

$$f_{s+2} = -\frac{1}{\prod\limits_{\beta_l \in T_2}(\beta_2 - \beta_l)\prod\limits_{\nu_\xi \in V}(\beta_2 - \nu_\xi)} \left[\sum_{i=0}^{s} f_i \prod_{\nu_\xi \in V}(x_i - \nu_\xi) \prod_{\beta_l \in T_2}(x_i - \beta_l) + \right.$$

$$\left. + f_{s+1} \prod_{\nu_\xi \in V}(\beta_1 - \nu_\xi) \prod_{\beta_l \in T_2}(\beta_1 - \beta_l) \right],$$

$$\vdots$$

$$f_{s+r-1} = -\frac{1}{\prod\limits_{\beta_l \in T_{r-1}}(\beta_{r-1} - \beta_l)\prod\limits_{\nu_\xi \in V}(\beta_{r-1} - \nu_\xi)} \left[\sum_{i=0}^{s} f_i \prod_{\nu_\xi \in V}(x_i - \nu_\xi) \prod_{\beta_l \in T_{r-1}}(x_i - \beta_l) + \right.$$

$$\left. + f_{s+1} \prod_{\nu_\xi \in V}(\beta_1 - \nu_\xi) \prod_{\beta_l \in T_{r-1}}(\beta_1 - \beta_l) + \ldots + f_{s+r-2} \prod_{\nu_\xi \in V}(\beta_{r-2} - \nu_\xi) \prod_{\beta_l \in T_{r-1}}(\beta_{r-2} - \beta_l) \right],$$

$$f_{s+r} = -\frac{1}{\prod\limits_{\nu_\xi \in V}(\beta_r - \nu_\xi)} \left[\sum_{i=0}^{s} f_i \prod_{\nu_\xi \in V}(x_i - \nu_\xi) + f_{s+1} \prod_{\nu_\xi \in V}(\beta_1 - \nu_\xi) + \ldots \right.$$

$$\left. \ldots + f_{s+r-1} \prod_{\nu_\xi \in V}(\beta_{r-1} - \nu_\xi) \right].$$

Или:

$$f_{s+1} = -\frac{1}{(b_{12}b_{13}\ldots b_{1r})\delta_1} \sum_{i=0}^{s} ((\ldots(\alpha_{ji}\, a_{ir})\ldots)a_{i3})a_{i2},$$

$$f_{s+2} = -\frac{1}{(b_{23}b_{24}\ldots b_{2r})\delta_2} \left[\sum_{i=0}^{s} ((\ldots(\alpha_{ji}a_{ir})\ldots)a_{i4})a_{i3} + ((\ldots((f_{s+1}\delta_1)b_{1r})\ldots)b_{14})b_{13} \right], \quad (8)$$

$$\vdots$$

$$f_{s+r-1} = -\frac{1}{b_{r-1,r}\delta_{r-1}} \left[\sum_{i=0}^{s} \alpha_{ji}a_{ir} + (f_{s+1}\delta_1)b_{1r} + \ldots + (f_{s+r-2}\delta_{r-2})b_{r-2,r} \right],$$

$$f_{s+r} = -\frac{1}{\delta_r} \left(\sum_{i=0}^{s} \alpha_{ji} + f_{s+1}\delta_1 + \ldots + f_{s+r-2}\delta_{r-2} + f_{s+r-1}\delta_{r-1} \right).$$

где: $b_{12} = \beta_1 - \beta_2$,

$b_{13} = \beta_1 - \beta_3$, $b_{23} = \beta_2 - \beta_3$,

$\vdots$ $\vdots$ $\ddots$

$b_{1r} = \beta_1 - \beta_r$, $b_{2r} = \beta_2 - \beta_r, \cdots,$ $b_{r-1,r} = \beta_{r-1} - \beta_r$;

$a_{i2} = x_i - \beta_2$, $a_{i3} = x_i - \beta_3$, $a_{i4} = x_i - \beta_4, \ldots,$ $a_{ir} = x_i - \beta_r$;

$\alpha_{ji} = f_i \prod\limits_{\nu_\xi \in V}(x_i - \nu_\xi)$; $\delta_j = \prod\limits_{\nu_\xi \in V}(\beta_j - \nu_\xi)$, $j = \overline{1, r}$.

Введем обозначения:

$$-\frac{1}{(b_{12}b_{13}\ldots b_{1r})\delta_1} = \gamma_1, \quad -\frac{1}{(b_{23}b_{24}\ldots b_{2r})\delta_2} = \gamma_2, \ldots, \quad -\frac{1}{b_{r-1,r}\delta_{r-1}} = \gamma_{r-1}, \quad -\frac{1}{\delta_r} = \gamma_r.$$

Алгоритм, позволяющий определить контрольные символы в соответствии с формулами (8) , будет следующим:

1. Для каждого $i = \overline{0, s}$:

а) задаем значения $\xi = \overline{1, e}$ и вычисляем произведения $\alpha_i = f_i \prod_{v_\xi \in V} (x_i - v_\xi)$, а также сумму $A_r = \sum_{i=0}^{s} \alpha_i$;

б) задавая значения $l = \overline{r, 2}$, вычисляем величины $a_{il} = x_i - \beta_l$ и определяем суммы:

$$A_{r-1} = \sum_{i=0}^{s} \alpha_i a_{ir} \ , \ A_{r-2} = \sum_{i=0}^{s} (\alpha_i a_{ir}) a_{i,r-1}, \ldots,$$

$$A_1 = \sum_{i=0}^{s} ((\ldots((\alpha_i a_{ir}) a_{i,r-1}) \ldots) a_{i3}) a_{i2}.$$

2. Для каждого $j = \overline{1, r}$:

а) задавая значения $\xi = \overline{1, e}$, вычисляем произведения $\delta_j = \prod_{v_\xi \in V} (\beta_j - v_\xi)$ и запоминаем их;

б) находим коэффициенты γ_j ;

в) определяем контрольные символы f_{s+j} ;

г) задаем значения $l = \overline{r, j+2}$ и находим произведения $B_j^{(l-1)} = B_j^{(l)} b_{jl}$, где $B_j^{(r)} = f_{s+1} \delta_j$, $B_j^{(r-1)} = (f_{s+1} \delta_j) b_{jr}$, …,

$B_j^{(j+1)} = (\ldots((f_{s+j} \delta_j) b_{jr}) \ldots) b_{j,j+2}$.

При этом вычисляется сумма $A_l = A_l + B_j^{(l)}$.

Количество операций, которое необходимо выполнить в поле $GF(2^m)$ для последовательного алгоритма кодирования будет следующим:

$N_\oplus = (k+r) + (2r+e-1) - r(r+1)$,

$N_\otimes = (k+r) + (r+e-1)$,

$N_{\circlearrowright} = r$.

Расчет производился с учетом наличия $(r-1)$ m разрядных ячеек памяти для хранения величин b_{il}, 1 ячейки – для хранения δ_i и 1 ячейки – для хранения α_j.

2. Параллельно-последовательный алгоритм

Процедура вычисления контрольных символов в соответствии с выражением (7) для параллельно-последовательного алгоритма кодирования следующая:

1. Для каждого $i = \overline{0, s}$:

а) задавая значения $\xi = \overline{1, e}$, вычисляем величины $\alpha_i = f_i \prod_{v_\xi \in V} (x_i - v_\xi)$ и запоминаем их;

б) вычисляем сумму $\sum_{i=0}^{s} f_i \prod_{v_\xi \in V} (x_i - v_\xi) = A$;

в) задавая значения $l = \overline{1, r}$, вычисляем величины $a_{il} = x_i - \beta_l$ и запоминаем их;

г) для каждого $j = \overline{1, r-1}$ определяем произведение $\alpha_i \prod_{l \neq j} a_{il}$ и сумму $\sum_{i=0}^{s} \alpha_i \prod_{l \neq j} a_{il}$.

2. Для каждого $j = \overline{1, r-1}$:

а) задавая значения $\xi = \overline{1, e}$, вычисляем произведения $\delta_j = \prod_{v_\xi \in V} (\beta_j - v_\xi)$ и запоминаем их;

б) находим коэффициенты $\gamma_j = -\dfrac{1}{\prod_{\beta_l \in T} (\beta_j - \beta_l)\, \delta_j}$;

в) определяем контрольные символы f_{s+j} и сумму $\sum_j f_{s+j} \delta_j = B$.

3. Вычисляем величины $\delta_r = \prod_{v_\xi \in V} (\beta_r - \beta_\xi)$, $\gamma_r = -\dfrac{1}{\delta_r}$ и контрольный символ $f_{s+r} = (A + B)\, \gamma_r$.

Количество операций в конечном поле при вычислении контрольных символов неполного кода параллельно-последовательным алгоритмом равно:

$N_\oplus = (k+r) + (2r+e) - r(r+1) - 1$,

$N_\otimes = (r-1)\left[r(k+1) - 1\right] + (k+r)\, e$,

$N_{\circlearrowright} = r$.

Для хранения величин α_i и δ_j необходимо иметь по одной m разрядной ячейке памяти и r ячеек – для хранения величин α_{il}.

Следует отметить, что алгоритмы и выражения для расчета количество операций в конечном поле приведены для $L^{(i)}(x) \neq const$.

При $L^{(i)}(x) = const$ для данных алгоритмов кодирования неполного кода количество операций в конечном поле будет следующим:

1) Для параллельного алгоритма:

$$N_{\oplus} = (n-r-1)\,r,$$

$$N_{\otimes} = \begin{cases} (n-r)\,r, & \text{при } r > 1, \\ 0, & \text{при } r = 1. \end{cases}$$

2) Для последовательного алгоритма:

$$N_{\oplus} = (2n-r-3)\,r/2,$$

$$N_{\otimes} = (2k+r-2)(r-1)/2 + (k+r-1).$$

Для параллельно-последовательного алгоритма:

$$N_{\oplus} = r(n-r)-1,$$

$$N_{\otimes} = r(k+1)-1.$$

Отметим, что при $L^{(i)}(x) = const$ для алгоритмов кодирования неполного кода количество операций сложения (а для параллельного алгоритма – и операций умножения) такое же, как для соответствующих алгоритмов полного кода [5].

Для хранения коэффициентов $L^{(i)}(x)$ необходимо иметь регистров памяти в количестве:

1) kr – для параллельного алгоритма;

2) $r(2k+r-1)/2$ – для последовательного алгоритма;

3) $r(k+1)-1$ – для параллельно-последовательного алгоритма.

Исследования показывают, что кодирование для неполного кода Лагранжа можно выполнять на устройствах кодирования полного кода, внеся в их схемы некоторые изменения.

Проведя сравнение алгоритмов кодирования неполного кода, приходим к следующим основным выводам:

1) При $L^{(i)}(x) \neq const$ меньшее количество операций в конечном поле имеет процедура кодирования, использующая последовательный алгоритм.

2) При $L^{(i)}(x) = const$ меньшее количество операций в конечном поле и количество регистров для хранения коэффициентов $L^{(i)}(x)$ имеет процедура кодирования, использующая параллельный алгоритм.

Список литературы

1. *Бияшев Р. Г.*, *Белова М. Н.* Вычисление фундаментальных многочленов для кодов Лагранжа, исправляющих одиночную ошибку // Вопросы кибернетики: Сб. науч. трудов. – Ташкент, 1975. – вып. 81. – С. 141-147.

2. *Нугманов Р. Н.* Процедура исправления многократных ошибок кодом Лагранжа // Электронная техника. Серия 10. 1979, вып. 1 (13). – С. 7-9.

3. *Кубицкий В. И.* Исправление стираний и ошибок кодами Лагранжа. // Материалы Московской научно-технической конф. молодых специалистов и ученых «Управление-82».– М., 1982. – Ч. 2. – С. 125-130.

4. *Кубицкий В. И.* Модификация процедуры кодирования полиномиальными кодами. (Деп. в ВИНИТИ 10.09.86, № 422 ГА-86, 16 с.). – Библиографический указатель ВИНИТИ «Депонированные научные работы», №1, 1987. – С. 128.

5. *Кубицкий В. И.* Процедуры кодирования и декодирования для полиномиальных кодов. – В сб.: Эксплуатация программного обеспечения систем реального времени, построенных на базе микро- и мини-ЭВМ. – К.: КИИГА, 1989. – С. 67-71.

6. *Амербаев В. М.*, *Бияшев Р. Г.* Интерполяция и коды, исправляющие ошибки. – В кн.: Теория кодирования и информационное моделирование. – Алма-Ата: Наука, 1973. – С. 57-63.