Примітка. Для другої та третьої колонок таблиці використано дані Інституту Інформаційних Технологій Харківського технічного університету радіоелектроніки (в другій колонці, виходячи з обсягу обчислень в $3 \cdot 10^5$ та $3 \cdot 10^{12}$ міпсороків відповідно).При оцінці криптографічної стійкості **першого варіанту** криптоалгоритму на базі СУЛ (шифрує один, розшифровують усі) слід звернути увагу на те, що серед елементів відкритого ключа є константи СЛК (величина діапазону представлення R, модифіковані ортогональні базиси $B_i^{/} = B_i /(m_i \cdot p_j) = R/(p_i \cdot p_j)$, які надають можливість досить просто (шляхом розкладання їх на множники) отримати сукупність основ p_i – основних констант СЛК. Це, в свою чергу, полегшує криптоаналітику можливість розкриття закритих ключів сукупності $p_i^{/} = p_j (i, j = 1, 2, ..., n)$ – величин, які, з метою підвищення криптографічної стійкості перетворення, випадковим чином вибрані із сукупності основ p_i так, що $i \neq j$. Це полегшення полягає в тому, що знання повного набору основ p_i та їх скорочених наборів, з яких обраховується кожен модифікований ортогональний базис $B_i^{/}$, дозволяє для кожного із них визначити пари p_i та p_j, які є відсутніми в даному базисі. Внаслідок цього задача криптоаналітика зводиться до визначення (тепер уже шляхом прямого перебору) з сукупності з *n* таких пар лише одного правильного варіанту (набору) ключів $p_i^{/} = p_j$ (i = 1, 2, ..., n). Неважко упевнитися в тому, що кількість варіантів ключів складає $N_{BK} = 2^n$, наприклад, при n =32 $N_{BK} = 2^{32}$. Звідсіля слід зробити висновок про <u>недостатню криптографічну стійкість даної модифікації криптоалгоритму на базі СУЛ</u>.

При оцінці криптографічної стійкості ***базового варіанту*** криптоалгоритму на базі СУЛ (шифрує один, розшифровують усі) слід звернути увагу на те, що в зв'язку із можливістю розкриття закритих ключів шляхом "статистичного" аналізу слід також зробити висновок про <u>недостатню криптографічну стійкість базового криптоалгоритму на базі СУЛ</u>.

При оцінці криптографічної стійкості будь-якої із запропонованих ***модифікацій базового варіанту*** криптоалгоритму на базі СУЛ (шифрує один, розшифровують усі) слід звернути увагу на те, що в зв'язку із неможливістю розкриття закритих ключів шляхом "статистичного" аналізу чи іншим чином, слід зробити висновок про необхідність застосування для розкриття ключів лише механізму прямого перебору з кількістю варіантів ключів, яка за рахунок більшої кількості секретних величин, ніж в алгоритмі **першого варіанту** криптоалгоритму на базі СУЛ, визначається згідно з даними колонки 6 таблиці, а отже, про <u>достатньо високу криптографічну стійкість базового криптоалгоритму на базі СУЛ,</u> яка не поступається стійкості відомих криптографічних алгоритмів.

Література: ***1.*** *Василенко В. С., Горицький В. М. Варіант завадостійкого криптографічного перетворення // "Современные проблемы телекоммуникаций", збірка доповідей на 6-ій міжнародній науково-технічній конференції, 19 – 22 серпня 2003 р. (ч. 1) // Одеська національна академія зв'язку ім. А. С. Попова – с. 71 – 73.* ***2.*** *Василенко В. С. Варіант завадостійкого криптографічного перетворення // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні, вип.. 8, 2004 р. – с. 101 – 108.* ***3.*** *Коржов В. Асимметричные криптоалгоритмы. // Открытые системы. № 7– 8/2002.*

УДК 621.391

АНАЛИЗ СЛОЖНОСТИ *MAP*, *MAX LOG MAP* И *LOG MAP* АЛГОРИТМОВ ДЕКОДИРОВАНИЯ ТУРБОКОДОВ ПРИ ДЕКОДИРОВАНИИ БИТА ИНФОРМАЦИИ

Сергей Ливенцев, Сергей Зайцев, Борис Горлинский
Специальный факультет СБ Украины ВИТИ НТУУ "КПИ"

Аннотация: **Произведен анализ сложности Map, Max Log Map и Log Map алгоритмов декодирования турбокодов с учетом нормализации при вычислении основных функций, а также количества проверочных символов с выхода компонентного рекурсивного систематического сверточного кода кодера турбокода. В результате анализа получены аналитические выражения, определяющие количество алгебраических операций для основных алгоритмов декодирования турбокодов.**
Summary: **In the article the analysis of complication of the Map, Max Log Map and Log Map decoding**

algorithms of turbocodes taking into account normorlization at the calculation of basic functions is produced, and also quantities of verification characters from the output of component recursive systematic convolutional code of coders of turbocodes. As a result of analysis analytical expressions determining the amount of algebraic operations for the basic decoding algorithms of turbocodes are got.
Ключевые слова: **Помехоустойчивое кодирование, турбокоды, алгоритмы декодирования.**

I Введение

Для обеспечения радиосвязью высших должностных лиц государства применяется сеть правительственной связи с подвижными объектами (СПСПО). Но на современном этапе развития современных специальных систем связи ее характеристики не удовлетворяют требованиям к информационно-телекоммуникационным системам передачи информации с ограниченным доступом. Анализ преимуществ и недостатков этой системы позволяет сделать выводы про необходимость ее усовершенствования и переход на новые современные технологии, которые планируются использовать в мобильной компоненте Национальной системы конфиденциальной связи (НСКЗ).

Одно из перспективных направлений усовершенствования существующей СПСПО и мобильной компоненты НСКЗ – применение новой технологии помехоустойчивого кодирования, основанной на турбокодах (ТК) [1, 2]. Для практического использования ТК необходима разработка соответствующего математического аппарата, что свидетельствует о необходимости исследования и усовершенствования их характеристик.

II Постановка задачи

Высокая эффективность ТК обусловлена разработанными для них алгоритмами декодирования *Map*, *Max Log Map*, *Log Map* и *SOVA* [3 – 9]. Анализ сложности реализации декодирования бита информации, приведенных в работах [4, 10, 11], показывает, что следующие особенности реализации этих алгоритмов рассмотрены недостаточно: обязательное использование нормализации при вычислении основных функций и учет проверочных символов с выхода компонентного рекурсивного систематического сверточного кода (РССК). Поэтому возникает задача определения сложности реализации декодирования бита информации в количестве алгебраических операций с учетом перечисленных особенностей.

Таким образом, **целью работы** является: определение числа алгебраических операций, необходимых для декодирования бита информации при использовании *Map*, *Max Log Map* и *Log Map* алгоритмов декодирования ТК с учетом количества ячеек памяти и общего количества символов с выхода РССК, а также использования нормализации основных функций.

III Принцип построения турбокода

Для определения сложности декодирования бита информации рассмотрим принцип построения ТК. На рис. 1, 2 показана структурная схема кодера и декодера ТК (одна итерация) при параллельном соединении двух РССК.

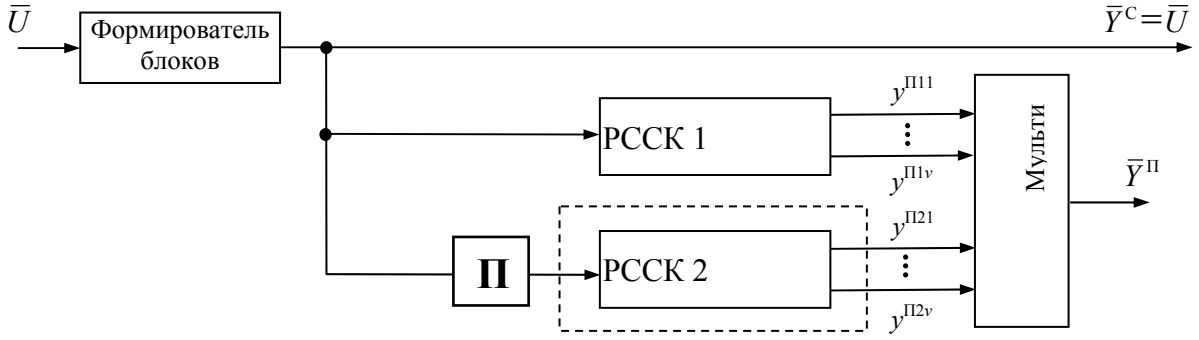

Рисунок 1 – Структурная схема кодера ТК

Информационная последовательность $\overline{U}$ разбивается на блоки длины N символов $\overline{U} = (u_1; u_2; u_3 \text{K}\, u_t \text{K}\, u_N)$, где t – текущий индекс, N – размер информационного блока. Сформированная последовательность поступает на систематический выход кодера, а также параллельно на два РССК, причем на второй через перемежитель (П). Последовательность на выходе кодера ТК имеет вид: $\overline{Y} = (\overline{Y}^{\text{C}}, \overline{Y}^{\text{П}})$. $\overline{Y}^{\text{C}} = \overline{U}$ – систематический выход кодера, а $\overline{Y}^{\text{П}} = (\overline{Y}^{\text{П1}}, \overline{Y}^{\text{П2}})$ – проверочный выход кодера ТК. При этом $\overline{Y}^{\text{П1}} = (\overline{Y}^{\text{П11}}, \text{K}\,, \overline{Y}^{\text{П1}v})$ – проверочный выход РССК1, $\overline{Y}^{\text{П2}} = (\overline{Y}^{\text{П21}}, \text{K}\,, \overline{Y}^{\text{П2}v})$ – проверочный выход РССК 2, v – общее количество проверочных символов каждого РССК кодера ТК. Последовательности проверочных символов для блоков длины N будут иметь следующий вид: $\overline{Y}^{\text{П11}} = (y_1^{\text{П11}}, \text{K}\,, y_t^{\text{П11}}, \text{K}\,, y_N^{\text{П11}})$, …, $\overline{Y}^{\text{П1}v} = (y_1^{\text{П1}v}, \text{K}\,, y_t^{\text{П1}v}, \text{K}\,, y_N^{\text{П1}v})$, $\overline{Y}^{\text{П21}} = (y_1^{\text{П21}}, \text{K}\,, y_t^{\text{П21}}, \text{K}\,, y_N^{\text{П21}})$, **…,** $\overline{Y}^{\text{П2}v} = (y_1^{\text{П2}v}, \text{K}\,, y_t^{\text{П2}v}, \text{K}\,, y_N^{\text{П2}v})$. Последовательность на выходе кодера ТК будет иметь вид:

$\overline{Y} = (y_1^{\text{C}}, y_1^{\text{П11}}, \text{K}\,, y_1^{\text{П1}v}, y_1^{\text{П21}}, \text{K}\,, y_1^{\text{П2}v}; \text{K}\; y_t^{\text{C}}, y_t^{\text{П11}}, \text{K}\,, y_t^{\text{П1}v}, y_t^{\text{П21}}, \text{K}\,, y_t^{\text{П2}v}; \text{K}\; y_N^{\text{C}}, y_N^{\text{П11}}, \text{K}\,, y_N^{\text{П1}v}, y_N^{\text{П21}}, \text{K}\,, y_N^{\text{П2}v})$.

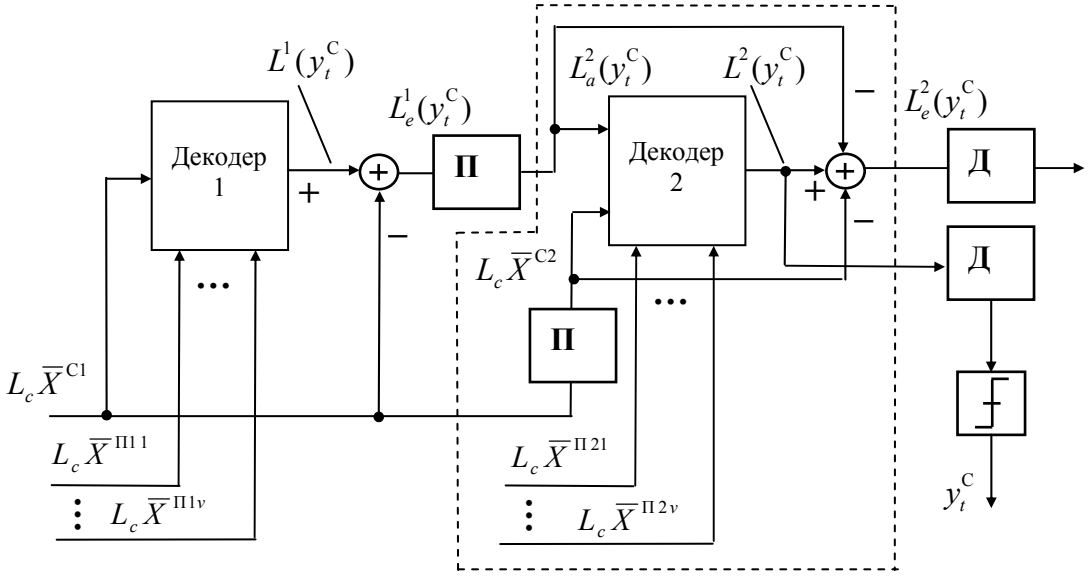

Рисунок 2 – Структурная схема декодера ТК

На каждый декодер поступает информация, полученная с выхода соответствующего РССК, но с учетом прохождения канала с аддитивным белым гауссовским шумом (АБГШ) и канальной “надежности”, а именно (рассматривается такт работы t):

$\overline{X}^1 = (L_c\overline{X}^{\text{C1}}, L_c\overline{X}^{\text{П1}}) = (L_c x_t^{\text{C1}}, L_c x_t^{\text{П11}}, \text{K}\,, L_c x_t^{\text{П1}v})$ – для первого декодера, где $\overline{X}^{\text{П1}} = (\overline{X}^{\text{П11}}, \text{K}\,, \overline{X}^{\text{П1}v})$. Соответственно $\overline{X}^2 = (L_c\overline{X}^{\text{C2}}, L_c\overline{X}^{\text{П2}}) = (L_c x_t^{\text{C2}}, L_c x_t^{\text{П21}}, \text{K}\,, L_c x_t^{\text{П2}v})$ – для второго декодера, где $\overline{X}^{\text{П2}} = (\overline{X}^{\text{П21}}, \text{K}\,, \overline{X}^{\text{П2}v})$. $\overline{X}^{\text{C1}} = \overline{X}^{\text{C}}, \overline{X}^{\text{C2}}$ – последовательности систематических символов с учетом соответствующей операции перемежения. Перемежитель-деперемежитель в схеме декодера ТК преобразует пакеты ошибок на выходе текущего декодера в одиночные ошибки, что значительно облегчает и улучшает работу следующего декодера [12].

Каждый декодер вычисляет функцию правдоподобия $L^1(y_t^{\text{C}}), L^2(y_t^{\text{C}})$, после этого – “внешнюю” информацию: $L_e^1(y_t^{\text{C}}), L_e^2(y_t^{\text{C}})$. С выхода перемежителя “внешняя” информация первого декодера используется в качестве априорной для второго декодера – $L_a^2(y_t^{\text{C}})$. “Внешняя” информация второго

декодера данной итерации $L_e^2(y_t^{\mathrm{C}})$ после операции деперемежения (Д) используется в качестве априорной для первого декодера следующей итерации.

После выполнения определенного числа итераций либо в случае принудительной остановки итеративного декодирования выносятся “жесткие” решения о декодированных символах:

$$y_t^{\mathrm{C}} = \begin{cases} 1, \text{ если } L(y_t^{\mathrm{C}}) \geq 0 \\ 0, \text{ если } L(y_t^{\mathrm{C}}) < 0 \end{cases}.$$

В дальнейшем рассматривается второй кодер и второй декодер структурных схем ТК (на рис. 1, 2 они выделены пунктирными линиями) для момента времени *t*.

Любой РССК описывается решетчатой диаграммой, число возможных состояний которой определяется выражением [3, 4]:

$$\mathrm{S} = 2^{\mathrm{m}}, \tag{1}$$

где *m* – количество ячеек памяти РССК. Обозначим предыдущее состояние решетчатой диаграммы $S_{t-1} = s'$, а текущее $S_t = s$.

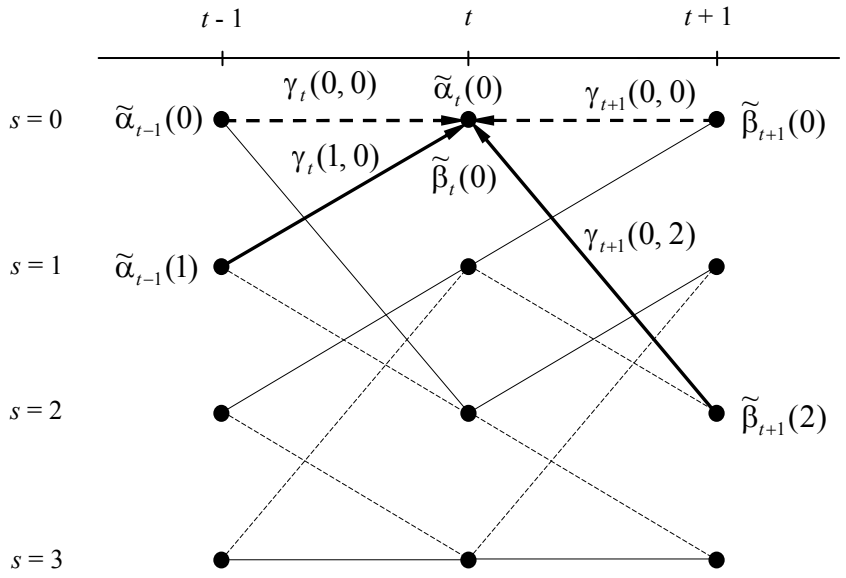

Рисунок 3 – Пример решетчатой диаграммы для РССК

Общее количество переходов (τ) из t –1 такта в такт t определяется по формуле:

$$\tau = 2 \times S = 2 \times 2^m, \tag{2}$$

т. к. в каждое последующее состояние имеется переход из двух предыдущих.

На рис. 3 показан пример решетчатой диаграммы для РССК вида (1, 7/5).

Количество ячеек памяти этого РССК *m* = 2, следовательно, число состояний *S* = 4. Декодирование одного информационного символа u_t начинается с такта t –1 и заканчивается на такте *t*. На диаграмме также показаны основные рекурсии, математическое описание которых приводится ниже.

Анализ аналитических выражений алгоритмов декодирования ТК [3 – 9] показывает, что для декодирования каждого бита информации необходимо вычислить переходную рекурсию, прямую рекурсию, обратную рекурсию, обобщенный параметр, функцию правдоподобия и параметр “внешней” информации.

IV *Map* алгоритм

Для каждого перехода решетчатой диаграммы вычисляется переходная рекурсия $\gamma_t(s', s)$ из состояния s' в состояние s по формуле [4, 8]:

$$\gamma_t(s', s) \sim \exp\left[\frac{1}{2}\cdot\left(y_t^{\mathrm{C}}\cdot\left(L_a^2(y_t^{\mathrm{C}})+L_c\cdot x_t^{\mathrm{C}}\right)+L_c\cdot\sum_{i=1}^{v}x_t^{\Pi 2i}\cdot y_t^{\Pi 2i}\right)\right],$$

где $y_t^{\mathrm{C}}, y_t^{\Pi 2i}, i \in (1, v)$ – соответственно систематический символ кодера ТК и проверочные символы РССК2 до прохождения канала АБГШ; $x_t^{\mathrm{C}}, x_t^{\Pi 2i}, i \in (1, v)$ – систематический символ кодера ТК и проверочные символы РССК2 после прохождения канала с АБГШ; $L_a^2(y_t^{\mathrm{C}})$ – априорная информация второго декодера; L_c – параметр канальной "надежности"; v – количество проверочных символов РССК, $v = q - 1$, где q – общее количество символов РССК (систематический и проверочные).

Количество алгебраических операций в выражении переходной рекурсии следующее: сложений $\xi_c^{\gamma} = q$, умножений $\xi_y^{\gamma} = q + 3$, экспоненцирования $\xi_э^{\gamma} = 1$. С учетом (2) получим общее количество операций, необходимых для вычисления переходных рекурсий для всех переходов решетчатой диаграммы РССК:

$$\xi_c^{\gamma} = 2 \times 2^m \times q; \tag{3}$$

$$\xi_y^{\gamma} = 2 \times 2^m \times (q + 3); \tag{4}$$

$$\xi_э^{\gamma} = 2 \times 2^m. \tag{5}$$

Для каждого состояния решетчатой диаграммы определяется прямая рекурсия $\alpha_t(s)$ (начиная с начала блока – при прямом вычислении) по формуле [4, 8]: $\alpha_t(s) = \sum_{s'} \widetilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s)$, где $\widetilde{\alpha}_{t-1}(s')$ – нормированная прямая рекурсия $t - 1$ такта. Суммирование производится по всем предыдущим состояниям s', для которых существует переход в текущее состояние s.

Количество математических операций в формуле прямой рекурсии следующее: сложений $\xi_c^{\alpha} = 1$, умножений $\xi_y^{\alpha} = 2$. Используя (1), получим общее количество операций сложения и умножения, необходимых для вычисления прямых рекурсий для всех состояний решетчатой диаграммы РССК:

$$\xi_c^{\alpha} = 2^m; \tag{6}$$

$$\xi_y^{\alpha} = 2 \times 2^m. \tag{7}$$

Получив прямые рекурсии для всех состояний данного такта, вычисляем параметр нормализации, который определяется суммой данных рекурсий:

$$\sum_s \alpha_t(s) = \sum_s \sum_{s'} \widetilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s).$$

Количество операций сложения в выражении параметра нормализации определяется выражением вида:

$$\xi_c^{\alpha} = 2^m - 1. \tag{8}$$

С учетом параметра нормализации выражение прямой рекурсии примет вид:

$$\widetilde{\alpha}_t(s) = \frac{\sum_{s'} \widetilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s)}{\sum_s \sum_{s'} \widetilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s)}.$$

Для выполнения вычисления нормализированной прямой рекурсии требуется $\xi_д^a = 1$ деление. Используя (1), получим следующее количество операций деления:

$$\xi_д^{\alpha} = 2^m. \tag{9}$$

Для каждого состояния решетчатой диаграммы определяется обратная рекурсия $\widetilde{\beta}_{t-1}(s')$, начиная с

конца блока при обратном вычислении, по формуле [4, 8]: $\beta_{t-1}(s') = \sum_{s} \tilde{\beta}_t(s) \cdot \gamma_t(s', s)$, где $\tilde{\beta}_t(s)$ – нормированная обратная рекурсия такта *t*. Суммирование производится по всем текущим состояниям S, в которые существует переход из предыдущего состояния s'.

Количество алгебраических операций для реализации обратной рекурсии следующее: сложений $\xi_{\text{с}}^{\beta}$ = 1, умножений ξ_{y}^{β} = 2. Используя (1), получим общее количество операций сложения и умножения, необходимых для вычисления обратных рекурсий для всех состояний решетчатой диаграммы РССК:

$$\xi_{c}^{\beta} = 2^m; \tag{10}$$

$$\xi_{y}^{\beta} = 2 \times 2^m. \tag{11}$$

С учетом параметра нормализации, полученного при вычислении прямой рекурсии, выражение обратной рекурсии примет вид:

$$\tilde{\beta}_{t-1}(s') = \frac{\sum_{s} \tilde{\beta}_t(s) \cdot \gamma_t(s', s)}{\sum_{s} \sum_{s'} \tilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s)},$$

для чего необходимо выполнить ξ_{∂}^{β} = 1 операцию деления. Количество операций деления для всех состояний решетчатой диаграммы следующее:

$$\xi_{\partial}^{\beta} = 2^m. \tag{12}$$

После получения значения переходных рекурсий, прямой и обратной рекурсии для всех состояний, вычисляются обобщенные параметры σ для переходов из предыдущего состояния $S_{t-1} = s'$ в текущее $S_t = s$, вызванных информационным символом $u_t = +1$, и для переходов из предыдущего состояния в текущее, вызванных информационным символом $u_t = -1$ [4, 8]: $\sigma_t(s) = \tilde{\alpha}_{t-1}(s') \cdot \gamma_t(s', s) \cdot \tilde{\beta}_t(s)$.

При реализации обобщенного параметра для одного перехода необходимо выполнить $\xi_{\text{у}}^{\sigma}$ = 2 операции умножения. С учетом выражения (2) общее количество операций умножения определяется из выражения:

$$\xi_{y}^{\sigma} = 4 \times 2^m. \tag{13}$$

Функция правдоподобия вычисляется по формуле [3, 7]:

$$L^2(y_t^{\text{C}}) = \log \frac{\sum_{\substack{(s',s) \\ u_t=+1}} \tilde{\alpha}_{t-1}(s') \cdot \tilde{\beta}_t(s) \cdot \gamma_t(s', s)}{\sum_{\substack{(s',s) \\ u_t=-1}} \tilde{\alpha}_{t-1}(s') \cdot \tilde{\beta}_t(s) \cdot \gamma_t(s', s)} = \log \left[\frac{\sum_{\substack{(s',s) \\ u_t=+1}} \sigma_t(s)}{\sum_{\substack{(s',s) \\ u_t=-1}} \sigma_t(s)} \right],$$

где числитель определяется суммой σ для всех переходов из предыдущего состояния $S_{t-1} = s'$ в текущее $S_t = s$, вызванных информационным символом $u_t = +1$, а знаменатель – аналогично для $u_t = -1$.

Для реализации функции правдоподобия необходимо выполнить

$$\xi_{c}^{L} = 2 \times 2^m - 2, \tag{14}$$

$$\xi_{\partial}^{L} = 1, \tag{15}$$

$$\xi_{л}^{L} = 1 \tag{16}$$

операций сложения, деления и логарифмирования.

Параметр “внешней” информации вычисляется следующим образом [4, 8]:

$L_e^2(y_t^{\mathrm{C}}) = L^2(y_t^{\mathrm{C}}) - Lc \cdot x_t^{\mathrm{C}2} - L_a^2(y_t^{\mathrm{C}})$, для чего необходимо выполнить 2 вычитания:

$$\xi_{в}^{Le} = 2. \tag{17}$$

V Max Log Map алгоритм

Для получения переходной, прямой и обратной рекурсии данного алгоритма необходимо прологарифмировать соответствующие рекурсии алгоритма *Map*: $\mathrm{A}_t(s) = \ln\widetilde{\alpha}_t(s)$, $\mathrm{B}_t(s) = \ln\widetilde{\beta}_t(s)$, $\Gamma_t(s', s) = \ln\gamma_t(s', s)$, а также использовать аппроксимацию: $\ln\left(\sum_{i=1}^{n} e^{a_i}\right) \approx \max_{i=\overline{1,n}} a_i$ [2, 3].

С учетом этих преобразований основные рекурсии будут иметь вид:

$$\Gamma_t(s', s) \sim \frac{1}{2}\cdot\left(y_t^{\mathrm{C}} \cdot \left(L_a^k(y_t^{\mathrm{C}}) + L_c \cdot x_t^{\mathrm{C}}\right) + L_c \cdot \sum_{i=1}^{v} x_t^{\Pi ki} \cdot y_t^{\Pi ki}\right);$$

$$\mathrm{A}_t(s) \approx \max_{s'}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s)\right];$$

$$\widetilde{\mathrm{A}}_t(s) \approx \max_{s'}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s)\right] - \mathrm{A}_t^{\max}(s);$$

$$\mathrm{B}_{t-1}(s') \approx \max_{s'}\left[\widetilde{\mathrm{B}}_t(s) + \Gamma_t(s', s)\right];$$

$$\widetilde{\mathrm{B}}_{t-1}(s') \approx \max_{s'}\left[\widetilde{\mathrm{B}}_t(s) + \Gamma_t(s', s)\right] - \mathrm{A}_t^{\max}(s),$$

где $\mathrm{A}_t^{\max}(s) \approx \max_{s}\left(\max_{s'}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s)\right]\right)$ – параметр нормализации. В выражении прямой рекурсии максимизация производится по всем предыдущим состояниям s', для которых существует переход в текущее состояние s, а для обратной – по всем текущим состояниям s, в которые существует переход из предыдущего состояния s'. Параметр нормализации определяется максимальным значением прямых рекурсий по всем состояниям.

Количество алгебраических операций в выражении переходной рекурсии следующее: сложений $\xi_c^{\gamma} = q$, умножений $\xi_y^{\gamma} = q + 3$. Определим общее количество операций сложения и умножения, необходимых для вычисления переходных рекурсий для всех переходов решетчатой диаграммы РССК, используя (2):

$$\xi_c^{\gamma} = 2 \times 2^m \times q; \tag{18}$$

$$\xi_y^{\gamma} = 2 \times 2^m \times (q + 3). \tag{19}$$

Для вычисления прямой рекурсии необходимо: сложений $\xi_c^{\alpha} = 2$, определения максимума $\xi_м^{\alpha} = 1$.

Получим общее количество операций, необходимых для вычисления прямых рекурсий для всех состояний решетчатой диаграммы РССК, с учетом (1):

$$\xi_c^{\alpha} = 2 \times 2^m; \tag{20}$$

$$\xi_м^{\alpha} = 2^m. \tag{21}$$

Количество операций определения максимума для определения параметра нормализации определяется выражением вида:

$$\xi_м^{\alpha} = 2^{\boldsymbol{m}} - 1. \tag{22}$$

Для реализации вычисления прямой рекурсии с учетом параметра нормализации необходимо выполнить $\xi_в^{a} = 1$ операции вычитания. Количество операций вычитания для всех состояний решетчатой диаграммы равно:

$$\xi_в^{\alpha} = 2^m. \tag{23}$$

Количество алгебраических операций для реализации обратной рекурсии равно: сложений ξ_c^{β} = 2, определения максимума $\xi_м^{\beta}$ = 1. Определим общее количество операций сложения и максимизации, используя (1):

$$\xi_c^{\beta} = 2 \times 2^m; \tag{24}$$

$$\xi_м^{\beta} = 2^m. \tag{25}$$

Для вычисления обратной рекурсии, учитывая параметр нормализации, необходимо выполнить $\xi_в^{\beta}$ = 1 операцию вычитания для всех состояний решетчатой диаграммы РССК.

Функция правдоподобия вычисляется по следующей формуле [3, 4]:

$$L^2(y_t^C) \approx \max_{\substack{(s',s)\\ u_{t=+1}}}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s',s) + \widetilde{\mathrm{B}}_t(s)\right] - \max_{\substack{(s',s)\\ u_{t=-1}}}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s',s) + \widetilde{\mathrm{B}}_t(s)\right] \approx \max_{\substack{(s',s)\\ u_{t=+1}}}\left[\Sigma_t(s)\right] - \max_{\substack{(s',s)\\ u_{t=-1}}}\left[\Sigma_t(s)\right]$$

где обобщенный параметр Σ равен: $\Sigma_t(s) = \mathrm{A}_{t-1}(s') + \Gamma_t(s',s) + \mathrm{B}_t(s)$.

Для реализации обобщенного параметра для одного перехода необходимо выполнить ξ_c^{σ} = 2 сложения. Общее количество операций сложения, используя (2), определяется следующим образом:

$$\xi_c^{\sigma} = 4 \times 2^m. \tag{26}$$

Числитель выражения функции правдоподобия определяется определением максимального значения Σ для всех переходов из предыдущего состояния $S_{t-1} = s'$ в текущее $S_t = s$, вызванных информационным символом $u_t = +1$, а знаменатель – аналогично для $u_t = -1$. Для этого необходимо выполнить операции определения максимума и вычитания:

$$\xi_м^{L} = 2 \times 2^m - 2; \tag{27}$$

$$\xi_в^{L} = 1. \tag{28}$$

Параметр “внешней” информации вычисляется аналогично алгоритму *Map*:

$$\xi_в^{Le} = 2. \tag{29}$$

VI *Log Map* алгоритм

Данный алгоритм отличается от алгоритма *Max Log Map* добавлением корректирующего слагаемого в выражениях прямой и обратной рекурсии, а также функции правдоподобия. Это корректирующее слагаемое является вторым в выражении логарифма Якобиана: $\ln(e^{a_1} + e^{a_2}) = \max(a_1, a_2) + \ln(1 + e^{-|a_1 - a_2|})$ [4, 6]. Данное корректирующее слагаемое может быть определено с помощью аппроксимации: $f(\delta) = \ln(1 + e^{-\delta})$, где $\delta = |a_1 - a_2|$. Для данного вычисления можно использовать пятиступенчатую (табл. 1) или двухступенчатую (табл. 2) аппроксимацию.

Таблица 1

δ	[0; 0,2)	[0,2; 0,8)	[0,8; 1,4)	[1,4; 2,0)	[2,0; +∞)
$f(\delta)$	0,67	0,52	0,32	0,18	0

Таблица 2

δ	[0; 1,6)	[1,6; +∞)
$f(\delta)$	0,48	0

Выражение для переходной рекурсии определяется аналогично *Max Log Map* алгоритму,

следовательно, количество используемых алгебраических операций также одинаково.

Выражения прямой и обратной рекурсии вычисляются с учетом корректирующего слагаемого, а именно [4, 6]:

$$\mathrm{A}_t(s) = \max_{s'}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s)\right] + f(\delta);$$

$$\widetilde{\mathrm{A}}_t(s) = \mathrm{A}_t(s) - \mathrm{A}_t^{\max}(s);$$

$$\mathrm{B}_{t-1}(s') = \max_{s'}\left[\widetilde{\mathrm{B}}_t(s) + \Gamma_t(s', s)\right] + f(\delta);$$

$$\widetilde{\mathrm{B}}_{t-1}(s') = \mathrm{B}_{t-1}(s') - \mathrm{A}_t^{\max}(s),$$

где $\mathrm{A}_t^{\max}(s) = \max_{s}(\max_{s'}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s)\right])$ – параметр нормализации, аналогичен предыдущему алгоритму.

Для реализации выражения прямой рекурсии с учетом корректирующего слагаемого необходимо выполнить: сложений – 3, определения максимума – 1, сравнений – 5 (для пятиступенчатой аппроксимации) и 2 (для двухступенчатой аппроксимации), определения абсолютного значения – 1.

Подставляя эти значения в (1), получим общее количество операций определения максимума, сложения, сравнения (для пятиступенчатой аппроксимации) и определения модуля, необходимых для вычисления прямых рекурсий для всех состояний решетчатой диаграммы РССК:

$$\xi_{м}^{\alpha} = 2^m; \tag{30}$$

$$\xi_{с}^{\alpha} = 3 \times 2^m; \tag{31}$$

$$\xi_{ср}^{\alpha} = 5 \times 2^m; \tag{32}$$

$$\xi_{мод}^{\alpha} = 2^m. \tag{33}$$

В случае использования двухступенчатой аппроксимации количество операций сравнения определяется выражением:

$$\xi_{ср}^{\alpha} = 2 \times 2^m. \tag{34}$$

Расчет параметра обратной рекурсии и прямой аналогичны, поэтому количество алгебраических операций одинаково.

В данном алгоритме декодирования ТК функция правдоподобия определяется выражением вида [4, 6]:

$$L^2(y_t^{\mathrm{C}}) \approx (\max_{\substack{(s',s)\\ u_{t=+1}}}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s) + \widetilde{\mathrm{B}}_t(s)\right] + f(\delta)) - (\max_{\substack{(s',s)\\ u_{t=-1}}}\left[\widetilde{\mathrm{A}}_{t-1}(s') + \Gamma_t(s', s) + \widetilde{\mathrm{B}}_t(s)\right] + f(\delta))$$

$$\approx (\max_{\substack{(s',s)\\ u_{t=+1}}}\left[\Sigma_t(s)\right] + f(\delta)) - (\max_{\substack{(s',s)\\ u_{t=-1}}}\left[\Sigma_t(s)\right] + f(\delta)).$$

Для расчета функции правдоподобия количество операций сложения, определения максимального значения, вычитания, сравнения и определения абсолютного значения равно:

$$\xi_{с}^{\alpha} = 2 \times 2^m - 2; \tag{35}$$

$$\xi_{м}^{\alpha} = 2 \times 2^m - 2; \tag{36}$$

$$\xi_{в}^{\alpha} = 1; \tag{37}$$

$$\xi_{ср}^{\alpha} = 10 \times 2^m - 10; \tag{38}$$

$$\xi_{мод}^{\alpha} = 2 \times 2^m - 2. \tag{39}$$

Параметр “внешней” информации аналогичен алгоритму *Map* и *Max Log Map.*

Выражения (3) – (39) приведены в табл. 3, 4, 5, где m – количество ячеек памяти РССК, q – общее количество символов (систематический и проверочные) с выхода РССК, $\widetilde{\alpha}$, $\widetilde{\mathrm{A}}$, $\widetilde{\beta}$, $\widetilde{\mathrm{B}}$ - нормализация соответственно α, A, β, B. Основными операциями являются: операции сложения (ADD), умножения

(MULT), деления (DIV), вычитания (SUB), определение максимального значения (MAX), сравнения (COMP), определения абсолютного значения (ABS) двух чисел, взятия логарифма (LOG) и определения экспоненты (EXP) от определенного числа.

Таблица 3

Операции	Параметры алгоритма декодирования *Map*							
	γ	α	β	$\tilde{\alpha}$	$\tilde{\beta}$	σ	L	L_e
ADD	$2 \times 2^m \times q$	2^m	2^m	$2^m - 1$			$2 \times 2^m - 2$	
MULT	$2 \times 2^m \times (q + 3)$	2×2^m	2×2^m			4×2^m		
DIV				2^m	2^m		1	
SUB								2
LOG							1	
EXP	2×2^m							

Таблица 4

Операции	Параметры алгоритма декодирования *Max Log Map*							
	Γ	A	B	$\tilde{A}$	$\tilde{B}$	σ	L	L_e
ADD	$2 \times 2^m \times q$	2×2^m	2×2^m			4×2^m		
MULT	$2 \times 2^m \times (q + 3)$							
SUB				2^m	2^m		1	2
MAX		2^m	2^m	$2^m - 1$			$2 \times 2^m - 2$	

Таблица 5

Операции	Параметры алгоритма декодирования *Log Map*							
	Γ	A	B	$\tilde{A}$	$\tilde{B}$	σ	L	L_e
ADD	$2 \times 2^m \times q$	3×2^m	3×2^m			4×2^m	$2 \times 2^m - 2$	
MULT	$2 \times 2^m \times (q + 3)$							
SUB				2^m	2^m		1	2
MAX		2^m	2^m	$2^m - 1$			$2 \times 2^m - 2$	
COMP		5×2^m	5×2^m				$10 \times 2^m - 10$	
ABS		2^m	2^m				$2 \times 2^m - 2$	

VII Выводы

1. В работе установлена функциональная зависимость между алгоритмами декодирования, видом РССК и алгебраическими операциями.
2. Формализирован процесс декодирования одного бита информации с учетом проверочных символов с выхода компонентного РССК и нормализации основных функций.
3. На основе формальной модели получены аналитические выражения, определяющие количество алгебраических операций для основных алгоритмов декодирования ТК.
4. Полученные аналитические выражения могут быть использованы для дальнейшего анализа сложности ТК при его аппаратно-программной реализации.

Литература: ***1.*** *Сергей Зайцев, Сергей Ливенцев, Дмитрий Алексеев. Применение турбокодов в специальных телекоммуникационных системах // Правове, нормативне та метрологічне забезпечення систем захисту інформації в Україні. – 2005. – № 11. – С. 162-167.* ***2.*** *Berrou C., Glavieux A., Thitimajshima P. Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes // Proc. Int. Conf. On Commun., ICC-93. – Geneva, 1993. – May. – P. 1064 – 1070.* ***3.*** *Woodard J., Hanzo L. Comparative Study of Turbo Decoding Techniques: An Overview // IEEE Transactions on Vehicular Technology, Vol. 49, No. 6, 2000. - November. P. 2208 2232.* ***4.*** *Прокопов С. Д. Анализ и оптимизация характеристик помехоустойчивости турбокодов // Диссертация на соискание научной степени кандидата технических наук. Одесса – 2002. – С. 133-143.* ***5.*** *Robertson P., Villebrun E., Hoeher P. Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding // Institute of communications technology. – Oberpfaffenhofen, Germany. – P. 4-9. 14.* ***6.*** *Robertson*

P., Villebrun E., Hoeher P. A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain // in Proc. Int. Conf. on Commun., ICC-95. – 1995. – June. – P. 1009-1013. ***7.*** *William E. Ryan Concatenated Convolutional Codes and Iterative Decoding // Department of Electrical and Computer Engineering - The University of Arizona, 2001. – July 10. – P. 8-23.* ***8.*** *Qi J. Turbo code in IS-2000 code division multiple access communications under fading // Wichita State University. – 1999. P. 38-59.* ***9.*** *Jeong Woo Lee The Study of Turbo Codes And Iterative Decoding // Dissertation for the degree of Doctor of Philosophy in Electrical Engineering in Graduate College of the University of Illinois at Urbana-Champaign, 2003. P. 86-96.* ***10.*** *Malardel F. Simulation and Optimisations of the Turbo Decoding Algorithm // Signal Processing Research Institute – University of South Australia, 1996. – July-November. – P. 23-26.* ***11.*** *Золотарев В. В., Овечкин Г. В. Помехоустойчивое кодирование. Методы и алгоритмы: Справочник // Москва: Горячая линия – Телеком. – 2001. С. 104.* ***12.****Ливенцев С. П., Алексеев Д. А., Зайцев С. В. Анализ характеристик перемежителей, используемых в турбокодах // Зв'язок. – 2005. – № 3. – С. 57-61.*

УДК 621.391.7

РОЗПОДІЛ СЕКРЕТНИХ КЛЮЧІВ НА ОСНОВІ РЕКУРЕНТНИХ ПОСЛІДОВНОСТЕЙ

Юрій Яремчук
Вінницький національний технічний університет

Анотація: **Пропонується метод розподілу секретних ключів, в основі якого лежить використання властивостей класу рекурентних послідовностей, для обчислення елементів яких вірні рекурентні співвідношення з коефіцієнтами, що пов'язані з початковими елементами послідовностей.**
Summary: **In the given work the method of secret keys distribution is offered, in which base use of properties of the class recurrent of sequences lays, at calculation of which elements the recurrences with coefficients coupled to initial elements of sequences are used.**
Ключові слова: **Захист інформації, криптографія, розподіл ключів, рекурентні послідовності.**

І Вступ

Шифрування інформації може забезпечувати ефективний захист лише за умови вирішення проблеми керування ключами. Розподіл ключів є однією з фундаментальних задач криптографії. Найбільш гостро проблема розподілу ключів стоїть в симетричних криптосистемах, де перед початком роботи необхідно передати секретний ключ обом сторонам.

Існує декілька шляхів вирішення проблеми розподілу ключів [1].

Фізичний розподіл. Розподіл ключів традиційним фізичним шляхом за допомогою довірених кур'єрів або озброєної охорони ключів. До 70-х років це був чи не єдиний безпечний шлях передавання ключів. Однак, такий спосіб передавання залежав від кур'єра: якщо його буде вбито, або підкуплено, то система шифрування буде скомпрометована.

Для двох сторін A та B фізичний розподіл ключів може організовуватись такими способами [2]:

1) сторона A вибирає ключ і фізично доставляє стороні B;

2) ключ вибирає третя сторона C і фізично доставляє його учасникам A і B;

3) якщо учасники обміну ключів A і B використовують деякий загальний ключ, то одна з сторін може передати новий ключ іншій стороні в шифрованому вигляді, використовуючи старий ключ;

4) якщо учасники обміну ключів A і B мають криптографічно захищені канали зв'язку з третьою стороною C, то остання може доставити ключ учасникам A і B цими захищеними каналами.

Розподіл за допомогою методів з секретним ключем. Генерування ключів і розподіл між будь-якими двома користувачами здійснюється на основі довгострокових секретних ключів, що розподілені між цими користувачами та третьою стороною – певним центром довіри. Такий шлях розподілу є достатньо ефективним, однак має недоліки. Зокрема, обидва користувача та центр довіри повинні працювати у режимі онлайн. Крім того, необхідно забезпечувати розподіл довгострокових ключів. Найбільш відомими протоколами розподілу ключів, що базуються на симетричному шифруванні, є протоколи Барроуза [3], Нідхейма-Шредера [4], Отвей-Ріса [5] та Kerberos [6].