УДК 004.032.2: 004.932

АЛГОРИТМ ДЕКОДИРОВАНИЯ КОДОВ С МАЛОЙ ПЛОТНОСТЬЮ ПРОВЕРОК НА ЧЕТНОСТЬ С БОЛЬШИМ РАСПАРАЛЛЕЛИВАНИЕМ

Ф. И. Иванов,
аспирант
И. В. Жилин,
аспирант
В. В. Зяблов,
доктор техн. наук, профессор
Институт проблем передачи информации им. А. А. Харкевича, г. Москва

Предложена модификация алгоритма декодирования belief propagation для кодов с малой плотностью проверок на четность, основанных на матрицах перестановок. Представленный в работе алгоритм имеет векторную реализацию. Приведены результаты моделирования данного алгоритма при передаче кодового слова с помощью двоичной фазовой модуляции по каналу с аддитивным белым гауссовым шумом.

Ключевые слова *— МПП-код, векторный декодер, матрица перестановок.*

Введение

Двоичные коды с малой плотностью проверок на четность (МПП-коды) были предложены Галлагером [1]. Данные линейные блочные коды задаются с помощью проверочной матрицы **H**, характеризуемой относительно малым числом единиц в строках и столбцах. Часто проверочную матрицу **H** МПП-кода удобно представлять в виде графа Таннера [2].

Также в работе [1] был предложен итеративный алгоритм декодирования «распространения доверия» (belief propagation). Данный алгоритм основан на декодировании по апостериорным вероятностям на выходе канала и требует порядка $O(n\log(n))$ операций, где n — длина кода.

Помимо случайных МПП-кодов нередко используют алгебраические МПП-коды, основанные на матрицах перестановок специального вида [3–7].

В данной работе рассмотрена модификация алгоритма декодирования «распространения доверия» для случая, когда проверочная матрица **H** кода с малой плотностью проверок на четность состоит из произвольных матриц перестановок. Основное преимущество данного алгоритма заключается в том, что он имеет параллельную реализацию, работая не с отдельными символами, а с векторами.

Структура проверочной матрицы случайного МПП-кода

Для лучшего понимания изложенного в статье материала мы приведем алгоритм построения проверочной матрицы случайного МПП-кода.

В 1960 г. Р. Галлагер предложил алгоритм генерации проверочной матрицы **H** случайного кода с малой плотностью проверок на четность [1]. Ниже приведен алгоритм построения этой матрицы.

Пусть **H** — проверочная матрица кода проверки на четность длины n_0:

$$\mathbf{H}_0 = \underbrace{\begin{pmatrix}1 & 1 & \dots & 1\end{pmatrix}}_{n_0}.$$

Запишем блочно-диагональную матрицу $\mathbf{H}_m$ с m проверочными матрицами $\mathbf{H}_0$ на главной диагонали:

$$\mathbf{H}_m = \underbrace{\begin{pmatrix}\mathbf{H}_0 & \mathbf{0} & \dots & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_0 & \ddots & \vdots\\ \vdots & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{H}_0\end{pmatrix}}_{m},$$

где m достаточно велико. Так как размер матрицы $\mathbf{H}_0$ равен $1 \times n_0$, то размер матрицы $\mathbf{H}_m - m \times mn_0$.

Пусть $\pi(\mathbf{H}_m)$ — случайная перестановка столбцов матрицы $\mathbf{H}_m$. Тогда матрица $\mathbf{H}$, составленная из $l > 2$ таких перестановок в качестве слоев:

$$\mathbf{H} = \begin{pmatrix} \pi_1(\mathbf{H}_m) \\ \vdots \\ \pi_l(\mathbf{H}_m) \end{pmatrix},$$

является разреженной проверочной матрицей размера $ml \times mn_0$, которая определяет ансамбль МПП-кодов Галлагера длины $n = mn_0$. Обозначим этот ансамбль $\varepsilon_G(l, n_0, m)$.

Элементы ансамбля $\varepsilon_G(l, n_0, m)$ получаются путем независимого выбора без возвращений перестановок π_i, $i = 1...l$. Все перестановки выбираются равновероятно, таким образом, на ансамбле $\varepsilon_G(l, n_0, m)$ задано равномерное распределение вероятностей.

Проверочная матрица $\mathbf{H}$ МПП-кода Галлагера, построенная указанным выше способом, содержит l единиц в каждом столбце и n_0 единиц в каждой строке. Такие кодовые конструкции назовем регулярными (l, n_0)-кодами.

Нижняя оценка на скорость R кода из $\varepsilon_G(l, n_0, m)$ определяется формулой $R \geq 1 - l(1 - R_0)$, где $R_0 = (n_0 - 1)/n_0$ — скорость кода проверки на четность. Таким образом, получим оценку на скорость МПП-кода Галлагера

$$R \geq 1 - \frac{l}{n_0}. \quad (1)$$

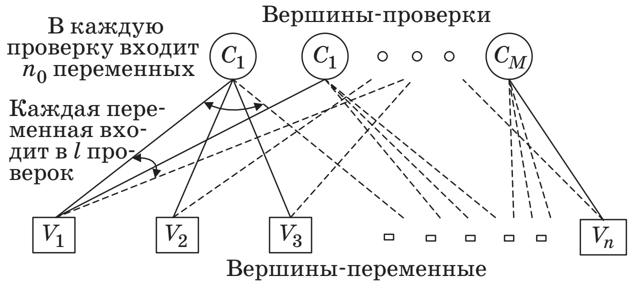
Декодирование случайного МПП-кода Галлагера

Для большего понимания алгоритма декодирования МПП-кодов, основанных на матрицах перестановок, который будет рассмотрен ниже, в данном разделе мы напомним классический алгоритм декодирования, предложенный Галлагером [1]. Описанный здесь декодер относится к классу так называемых вероятностных алгоритмов декодирования. На вход алгоритму передается оценка вероятностного распределения символов, полученная из канала, и далее декодер работает с численными значениями вероятностей.

Рассматриваемый декодер МПП-кодов работает с представлением кода в виде фактор-графа, также известного как граф Таннера.

Граф Таннера — это двудольный граф, состоящий из двух подмножеств вершин: вершин-символов (вершин-переменных) и вершин-проверок (рис. 1). Ребро соединяет вершину-переменную и вершину-проверку в том случае, если соответствующая переменная (символ) входит в проверку.

Рассматриваемый алгоритм является итеративным. Каждая итерация состоит из последовательной обработки сначала данных вершин-проверок, а затем вершин-переменных.



■ *Рис. 1.* Граф Таннера регулярного (l, n_0) МПП-кода длины n

Пусть $1 \leq i \leq n$, $1 \leq j \leq M$, где $n = mn_0$ (длина кода), $M = ml$ (количество проверок на четность). Введем некоторые обозначения:

LLR — логарифм отношения правдоподобия (log likelihood ratio);

α_i — знак LLR i-й переменной из канала;

β_i — модуль LLR i-й переменной из канала;

α'_i — вычисленный знак LLR i-й переменной;

β'_i — вычисленный модуль LLR i-й переменной;

γ_{ji} — сообщение от j-й проверки к i-й переменной;

α_{ji} — знак сообщения от i-й переменной к j-й проверке (принимает значения +1 или −1);

β_{ji} — модуль сообщения от i-й переменной к j-й проверке;

$I(j)$ — набор переменных i_k, которые участвуют в проверке j;

$I(j)\backslash i$ — набор $I(j)$, кроме бита i;

$J(i)$ — набор проверок, в которых участвует i-я переменная;

$J(i)\backslash j$ — набор $J(i)$, кроме бита j.

Декодер включает в себя следующие шаги.

1. *Инициализацию*: присвоим $\forall\ j = 1...M$: $\alpha_{ji}\beta_{ji} = \alpha_i\beta_i$.

2. *Горизонтальный шаг*: вычисление сообщений от вершин-проверок к вершинам-переменным; при использовании LLR оно будет выглядеть следующим образом:

$$\gamma_{ji} = \left(\prod_{i' \in I(j)\backslash i} \alpha_{ji'}\right) \phi\left(\sum_{i' \in I(j)\backslash i} \phi(\beta_{ji'})\right),$$

где функция

$$\phi(x) = \ln\frac{e^x + 1}{e^x - 1}.$$

3. *Вертикальный шаг*: вычисление сообщений от вершин-переменных к вершинам-проверкам:

$$\alpha'_i\beta'_i = \alpha_i\beta_i + \sum_{j \in J(i)} \gamma_{ji};$$

$$\alpha_{ji}\beta_{ji} = \alpha_i\beta_i + \sum_{j' \in J(i)\backslash j} \gamma_{j'i}.$$

Далее по вычисленным $\alpha'_i\beta'_i(1 \le i \le n)$ строится вектор **x**, который является «жестким решением», и проверяется равенство нулю синдрома $\mathbf{Hx}^{\mathrm{T}}$.

Горизонтальный и вертикальный шаги выполняются ограниченное число раз. В случае если все проверки оказались выполнены (синдром равен нулевому вектору), алгоритм может быть остановлен досрочно. Если достигнуто максимальное число итераций, то исполнение алгоритма прерывается и блок считается принятым с ошибкой. Возможны и другие критерии остановки.

Отметим также, что если составить $M \times n$ матрицу

$$\mathbf{H} = \{h_{ji}: h_{ji} = 1 \leftrightarrow i\text{-я переменная входит в } j\text{-ю проверку}, h_{ji} = 0 \text{ иначе}\},$$

то **H** будет являться проверочной матрицей МПП-кода. Таким образом, существует взаимно-однозначное отображение между фактор-графом и проверочной матрицей МПП-кода. Данный факт позволяет дать процессу декодирования матричное описание.

МПП-коды, основанные на матрицах перестановок

Определение: Пусть m, n_0, $l \in N$, причем $n_0 > l$, $m! > ln_0$. Рассмотрим группу $\mathcal{P}_m$ матриц перестановок размерности m, $|\mathcal{P}_m| = m!$. Выберем ln_0 случайных матриц $\{\mathbf{P}_{ji}\} \in \mathcal{P}_m$, $i = 1...l$, $j = 1...n_0$. Потребуем также, что если $\mathbf{P}_{ji} = \mathbf{P}_{ks}$, то $j = k$, $i = s$. Ясно, что такие условия выбора матриц перестановок $\mathbf{P}_{ji}$ соответствуют урновой модели без возвращений. Построим проверочную матрицу **H** следующего вида:

$$\mathbf{H} = \begin{pmatrix} \mathbf{P}_{11} & \dots & \dots & \mathbf{P}_{1n_0} \\ \dots & \dots & \dots & \dots \\ \mathbf{P}_{l1} & \dots & \dots & \mathbf{P}_{l,n_0} \end{pmatrix}. \quad (2)$$

Указанный выше способ построения матрицы **H** гарантирует, что все матрицы в каждой строке и каждом столбце будут различны. Так как $\mathbf{P}_{ji}$ — квадратная $m \times m$ матрица, то размерность $\mathbf{H} - ml \times mn_0$. **H** определяет ансамбль регулярных (l, n_0)-кодов с малой плотностью проверок на четность длины $n = mn_0$, который мы обозначим $\varepsilon_P(l, n_0, m)$. Элементы ансамбля $\varepsilon_P(l, n_0, m)$ получаются путем выбора без возвращений матриц перестановок $\{\mathbf{P}_{ji}\} \in \mathcal{P}_m$, $j = 1...l$, $i = 1...n_0$. Произвольный код $C \in \varepsilon_P(l, n_0, m)$ назовем *кодом, основанным на матрицах перестановок*.

Как и для произвольного кода из ансамбля $\varepsilon_G(l, n_0, m)$, для кода из $\varepsilon_P(l, n_0, m)$ также справедлива оценка на скорость (1).

Одним из наиболее распространенных на практике и простых по структуре кодов из ансамбля $\varepsilon_P(l, n_0, m)$ является квазициклический МПП-код.

Дадим определение ансамбля таких кодов.

Определение: Пусть l, $n_0 \in N$, $n_0 > l$, $\mathbf{I}_{p_{ji}}$ — $m \times m$ матрица p_{ji}-кратного циклического сдвига столбцов единичной $m \times m$ матрицы **I**, $1 \le j \le l$, $1 \le i \le n_0$, $1 \le p_{ji} \le m$. Построим $l \times n_0$ матрицу **H** следующего вида:

$$\mathbf{H} = \begin{pmatrix} \mathbf{I}_{p_{11}} & \cdots & \mathbf{I}_{p_{1n_0}} \\ \cdots & \cdots & \cdots \\ \mathbf{I}_{p_{l1}} & \cdots & \mathbf{I}_{p_{l,n_0}} \end{pmatrix}.$$

Поскольку размерность $\mathbf{I}_{p_{ij}}$ $- m \times m$, то размерность $\mathbf{H} - ml \times mn_0$. **H** определяет ансамбль регулярных (l, n_0) МПП-кодов длины $n = mn_0$. Обозначим этот ансамбль $\varepsilon_Q(l, n_0, m)$. Элементы ансамбля $\varepsilon_Q(l, n_0, m)$ получаются путем равновероятного выбора (возможно, с возвращениями) матриц p_{ij}-кратных циклических сдвигов. Произвольный код $C \in \varepsilon_Q(l, n_0, m)$ назовем квазициклическим МПП-кодом.

Очевидно, что ансамбль $\varepsilon_Q(l, n_0, m)$ является подансамблем ансамбля $\varepsilon_P(l, n_0, m)$. В то же время, поскольку проверочная матрица **H** квазициклического МПП-кода полностью определяется набором из ln_0 чисел p_{ji}, $0 \le p_{ji} \le m - 1$, $1 \le j \le l$, $1 \le i \le n_0$, то для хранения **H** нам достаточно хранить матрицу

$$\tilde{\mathbf{H}} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n_0} \\ \dots & \dots & \dots & \dots \\ p_{l1} & p_{l2} & \cdots & p_{l,n_0} \end{pmatrix}.$$

Хранение данной матрицы вместо **H** позволяет существенно оптимизировать процедуру хранения. Поскольку для представления проверочной матрицы **H** в форме (2) нам потребовалось бы mln_0 чисел, то достигается m-кратная экономия памяти. Матрицы перестановок $\mathbf{P}_{ji}$, использованные в (2), для квазициклического МПП-кода являются матрицами p_{ji}-кратного циклического сдвига.

Отметим, что из работ [8, 9] следует, что коды из ансамблей $\varepsilon_P(l, n_0, m)$, $\varepsilon_Q(l, n_0, m)$ и $\varepsilon_G(l, n_0, m)$ при одинаковых параметрах обладают практически одинаковыми корректирующими свойствами.

Вычисление синдрома для МПП-кода, основанного на матрицах перестановок

Пусть $\mathbf{H} = \begin{pmatrix} \mathbf{P}_{11} & \dots & \dots & \mathbf{P}_{1n_0} \\ \dots & \dots & \dots & \dots \\ \mathbf{P}_{l1} & \dots & \dots & \mathbf{P}_{l,n_0} \end{pmatrix}$ — проверочная матрица регулярного (l, n_0)-кода с малой плотностью проверок на четность, причем размер $\mathbf{P}_{ji}$, $1 \le j \le l$, $1 \le i \le n_0$, равен $m \times m$, тогда матрицу $\mathbf{H}$ можно представить в следующем виде:

$$\mathbf{H} = \begin{pmatrix} \pi_{11} & \dots & \dots & \pi_{1n_0} \\ \dots & \dots & \dots & \dots \\ \pi_{1l} & \dots & \dots & \pi_{l,n_0} \end{pmatrix}, \quad (3)$$

где π_{ji} — перестановка, соответствующая матрице $\mathbf{P}_{ji}$.

Поскольку длина МПП-кода с проверочной матрицей $\mathbf{H}$ равна $n = mn_0$, то кодовое слово $\mathbf{c} = (c_1, c_2, ..., c_n)$, $c_i \in GF(2)$, можно представить в следующем виде:

$$\mathbf{c} = (\bar{\mathbf{c}}_1, \bar{\mathbf{c}}_2, \dots, \bar{\mathbf{c}}_{n_0}), \quad (4)$$

где $\bar{\mathbf{c}}_i$ — двоичный вектор длины m. Напомним, что синдром $\mathbf{S}$ для принятого слова $\mathbf{y}$ вычисляется по формуле $\mathbf{S} = \mathbf{H}\mathbf{y}^{\mathrm{T}}$, причем $\mathbf{S} = \mathbf{0}$ тогда и только тогда, когда $\mathbf{y}$ является кодовым словом.

Пусть проверочная матрица задана соотношением (3), а принятое слово $\mathbf{y}$ — соотношением (4), тогда $\mathbf{y}$ является кодовым тогда и только тогда, когда

$$\begin{pmatrix} \pi_{11} & \dots & \dots & \pi_{1n_0} \\ \dots & \dots & \dots & \dots \\ \pi_{1l} & \dots & \dots & \pi_{l,n_0} \end{pmatrix} \begin{pmatrix} \bar{\mathbf{y}}_1 \\ \dots \\ \bar{\mathbf{y}}_{n_0} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \dots \\ \mathbf{0} \end{pmatrix}.$$

Последнее соотношение эквивалентно следующей системе из l уравнений:

$$\begin{cases} \pi_{11}(\bar{\mathbf{y}}_1) + \pi_{12}(\bar{\mathbf{y}}_2) + \dots + \pi_{1n_0}(\bar{\mathbf{y}}_{n_0}) = \mathbf{0} \\ \dots \\ \pi_{l1}(\bar{\mathbf{y}}_1) + \pi_{l2}(\bar{\mathbf{y}}_2) + \dots + \pi_{l,n_0}(\bar{\mathbf{y}}_{n_0}) = \mathbf{0} \end{cases}.$$

Таким образом, доказано следующее **утверждение.**

Вектор $\bar{\mathbf{y}} = (\bar{\mathbf{y}}_1, \bar{\mathbf{y}}_2, \dots, \bar{\mathbf{y}}_{n_0})$, где $\bar{\mathbf{y}}_i$ — двоичный вектор длины m, является кодовым словом кода с малой плотностью проверок на четность длины $n = mn_0$, задаваемого проверочной матрицей (3), тогда и только тогда, когда выполняется l соотношений

$$\sum_{i=1}^{n_0} \pi_{ji}(\bar{\mathbf{y}}_i) = \mathbf{0},\ j = 1 \dots l.$$

Как следует из утверждения, для МПП-кода, основанного на матрицах перестановок, вычисление синдрома ошибки имеет векторный характер: в вычислениях используются не отдельные символы принятого слова, а блоки длины m.

Декодирование МПП-кодов, основанных на матрицах перестановок

Предлагаем модификацию алгоритма belief propagation для кодов с малой плотностью проверок на четность, основанных на матрицах перестановок. Основная идея предложенной модификации заключается в одновременной обработке m символов принятого слова (т. е. алгоритм работает с векторами длины m), в то время как классический алгоритм belief propagation не предусматривает такую возможность. Векторный характер декодирования принятого слова, как будет показано, позволяет распараллелить алгоритм декодирования в m раз, что существенно отразится на скорости обработки данных.

Как и при декодировании случайного МПП-кода Галлагера, на вход алгоритму передается оценка вероятностного распределения символов, полученная из канала. Данная оценка представляет из себя вектор длины n **LLR**, т. е. $\mathbf{LLR} \in R^n$. Так как $n = mn_0$, то для **LLR** справедливо представление (4)

$$\mathbf{LLR} = (\bar{\mathbf{L}}_1, \bar{\mathbf{L}}_2, \dots, \bar{\mathbf{L}}_{n_0}),$$

причем $\bar{\mathbf{L}}_i \in R^m$, $1 \le i \le n_0$.

Выше было введено множество $I(j)$ — набор переменных $\left\{v_1^{(j)}, v_2^{(j)}, \dots, v_{n_0}^{(j)}\right\}$, участвующих в j-й проверке, и множество $J(i)$ — набор проверок $\left\{c_1^{(i)}, c_2^{(i)}, \dots, c_l^{(i)}\right\}$, в которые входит i-я переменная. Рассмотрим произвольный вектор $\bar{\mathbf{L}}_i \in R^m$. Так как размерность $\bar{\mathbf{L}}_i$ равна m, а матрицы $\mathbf{P}_{1i}$, $\mathbf{P}_{2i}$, ..., $\mathbf{P}_{li}$ – $m \times m$ матрицы перестановок (т. е. содержат ровно 1 единицу в каждой строке и каждом столбце), то $\bar{\mathbf{L}}_i$ участвует в ml различных проверках. Таким образом, $|J(\bar{\mathbf{L}}_i)| = ml$. Полученное равенство позволяет нам сделать вывод о том, что элементы $\bar{\mathbf{L}}_i$ участвуют во всех проверках. Таким образом, при декодировании нам не требуется искать $J(\bar{\mathbf{L}}_i)$ для каждого вектора $\bar{\mathbf{L}}_i$. Проводя аналогичные рассуждения, можно показать, что в m проверках участвуют mn_0 переменных, поэтому вычисление $I(j)$ для каждой j-й проверки также не требуется.

Введем необходимые обозначения:

$\mathbf{LLR} = (\bar{\mathbf{L}}_1, \bar{\mathbf{L}}_2, \ldots, \bar{\mathbf{L}}_{n_0})$ — принятый из канала вектор логарифмов отношения правдоподобия, причем $\bar{\mathbf{L}}_i \in R^m$, $\bar{\mathbf{L}}_i = (l_1^i, l_2^i, \ldots, l_m^i)$, $1 \le i \le n_0$;

$\bar{\boldsymbol{\alpha}}_i$ — вектор знаков (+1 или –1) вектора $\bar{\mathbf{L}}_i$, полученного из канала, т. е. $\bar{\boldsymbol{\alpha}}_i = \mathrm{sign}(\bar{\mathbf{L}}_i) = (\alpha_1^i, \alpha_2^i, \ldots, \alpha_m^i)$, где $\alpha_t^i = \mathrm{sign}(l_t^i)$, $t = 1\ldots m$;

$\bar{\boldsymbol{\beta}}_i$ — вектор модулей вектора $\bar{\mathbf{L}}_i$, полученного из канала, т. е. $\bar{\boldsymbol{\beta}}_i = |\bar{\mathbf{L}}_i| = (\beta_1^i, \beta_2^i, \ldots, \beta_m^i)$, где $\beta_t^i = |l_t^i|$, $t = 1\ldots m$;

$\bar{\boldsymbol{\alpha}}'_i$ — вычисленный вектор знаков для вектора $\bar{\mathbf{L}}_i$;

$\bar{\boldsymbol{\beta}}'_i$ — вычисленный вектор модулей для вектора $\bar{\mathbf{L}}_i$;

$\bar{\boldsymbol{\gamma}}_{ji}$ — вектор сообщений от j-й группы из m проверок к $\bar{\mathbf{L}}_i$;

$\bar{\boldsymbol{\alpha}}_{ji}$ — вектор знаков (покомпонентный) сообщений от переменных $\bar{\mathbf{L}}_i$ к j-й группе из m проверок;

$\bar{\boldsymbol{\beta}}_{ji}$ — вектор модулей (покомпонентный) сообщений от переменных $\bar{\mathbf{L}}_i$ к j-й группе из m проверок.

Изложенный ниже алгоритм декодирования применим только для МПП-кодов, основанных на матрицах перестановок, и работает с проверочной матрицей, представленной в форме (3):

$$\mathbf{H} = \begin{pmatrix} \pi_{11} & \cdots & \cdots & \pi_{1n_0} \\ \cdots & \cdots & \cdots & \cdots \\ \pi_{1l} & \cdots & \cdots & \pi_{l,n_0} \end{pmatrix}.$$

Декодирование включает в себя следующие шаги.

1. *Начальную проверку*: по принятому из канала вектору $\mathbf{LLR} = (\bar{\mathbf{L}}_1, \bar{\mathbf{L}}_2, \ldots, \bar{\mathbf{L}}_{n_0})$ строится «жесткое решение» $\mathbf{x}$, вычисляется синдром $\mathbf{Hx}^{\mathrm{T}}$ согласно алгоритму, описанному в предыдущем разделе. Если синдром равен нулевому вектору, то декодирование прекращается и $\mathbf{x}$ является результатом выполнения алгоритма, иначе переходим к шагу 2.

2. *Инициализацию*: строим матрицы $\mathbf{A}$ и $\mathbf{B}$ по правилу

$$\mathbf{A} = \begin{pmatrix} \bar{\boldsymbol{\alpha}}_{11} & \bar{\boldsymbol{\alpha}}_{12} & \cdots & \bar{\boldsymbol{\alpha}}_{1,n_0-1} & \bar{\boldsymbol{\alpha}}_{1n_0} \\ \bar{\boldsymbol{\alpha}}_{21} & \bar{\boldsymbol{\alpha}}_{22} & \cdots & \bar{\boldsymbol{\alpha}}_{2,n_0-1} & \bar{\boldsymbol{\alpha}}_{2n_0} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \bar{\boldsymbol{\alpha}}_{l1} & \bar{\boldsymbol{\alpha}}_{l2} & \cdots & \bar{\boldsymbol{\alpha}}_{l,n_0-1} & \bar{\boldsymbol{\alpha}}_{l,n_0} \end{pmatrix} = \begin{pmatrix} \pi_{11}(\mathrm{sign}(\bar{\mathbf{L}}_1)) & \pi_{12}(\mathrm{sign}(\bar{\mathbf{L}}_2)) & \cdots & \pi_{1,n_0-1}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0-1})) & \pi_{1n_0}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0})) \\ \pi_{21}(\mathrm{sign}(\bar{\mathbf{L}}_1)) & \pi_{22}(\mathrm{sign}(\bar{\mathbf{L}}_2)) & \cdots & \pi_{2,n_0-1}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0-1})) & \pi_{2n_0}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0})) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \pi_{l1}(\mathrm{sign}(\bar{\mathbf{L}}_1)) & \pi_{l2}(\mathrm{sign}(\bar{\mathbf{L}}_2)) & \cdots & \pi_{l,n_0-1}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0-1})) & \pi_{l,n_0}(\mathrm{sign}(\bar{\mathbf{L}}_{n_0})) \end{pmatrix};$$

$$\mathbf{B} = \begin{pmatrix} \bar{\boldsymbol{\beta}}_{11} & \bar{\boldsymbol{\beta}}_{12} & \cdots & \bar{\boldsymbol{\beta}}_{1,n_0-1} & \bar{\boldsymbol{\beta}}_{1n_0} \\ \bar{\boldsymbol{\beta}}_{21} & \bar{\boldsymbol{\beta}}_{22} & \cdots & \bar{\boldsymbol{\beta}}_{2,n_0-1} & \bar{\boldsymbol{\beta}}_{2n_0} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \bar{\boldsymbol{\beta}}_{l1} & \bar{\boldsymbol{\beta}}_{l2} & \cdots & \bar{\boldsymbol{\beta}}_{l,n_0-1} & \bar{\boldsymbol{\beta}}_{l,n_0} \end{pmatrix} = \begin{pmatrix} \pi_{11}\left(\left|\bar{\mathbf{L}}_1\right|\right) & \pi_{12}\left(\left|\bar{\mathbf{L}}_2\right|\right) & \cdots & \pi_{1,n_0-1}\left(\left|\bar{\mathbf{L}}_{n_0-1}\right|\right) & \pi_{1n_0}\left(\left|\bar{\mathbf{L}}_{n_0}\right|\right) \\ \pi_{21}\left(\left|\bar{\mathbf{L}}_1\right|\right) & \pi_{22}\left(\left|\bar{\mathbf{L}}_2\right|\right) & \cdots & \pi_{2,n_0-1}\left(\left|\bar{\mathbf{L}}_{n_0-1}\right|\right) & \pi_{2n_0}\left(\left|\bar{\mathbf{L}}_{n_0}\right|\right) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \pi_{l1}\left(\left|\bar{\mathbf{L}}_1\right|\right) & \pi_{l2}\left(\left|\bar{\mathbf{L}}_2\right|\right) & \cdots & \pi_{l,n_0-1}\left(\left|\bar{\mathbf{L}}_{n_0-1}\right|\right) & \pi_{l,n_0}\left(\left|\bar{\mathbf{L}}_{n_0}\right|\right) \end{pmatrix}.$$

3. *Горизонтальный шаг*: строим матрицу $l \times n_0$ сообщений от j-й группы проверок к i-му вектору переменных:

$$\boldsymbol{\Gamma} = \begin{pmatrix} \bar{\boldsymbol{\gamma}}_{11} & \cdots & \bar{\boldsymbol{\gamma}}_{1,n_0-1} & \bar{\boldsymbol{\gamma}}_{1n_0} \\ \cdots & \cdots & \cdots & \cdots \\ \bar{\boldsymbol{\gamma}}_{1l} & \cdots & \bar{\boldsymbol{\gamma}}_{l,n_0-1} & \bar{\boldsymbol{\gamma}}_{l,n_0} \end{pmatrix},$$

где $\bar{\boldsymbol{\gamma}}_{ji} = \left(\prod\limits_{\substack{t=1 \\ t\ne i}}^{n_0} \bar{\boldsymbol{\alpha}}_{jt}\right) \cdot \bar{\boldsymbol{\Phi}}\left(\sum\limits_{\substack{t=1 \\ t\ne i}}^{n_0} \bar{\boldsymbol{\Phi}}\left(\bar{\boldsymbol{\beta}}_{jt}\right)\right)$, причем отображение $\bar{\boldsymbol{\Phi}}: R^m \to R^m$ имеет следующий вид: $\bar{\boldsymbol{\Phi}}(\bar{\mathbf{x}}) = \ln\left(\dfrac{e^{\bar{\mathbf{x}}}+1}{e^{\bar{\mathbf{x}}}-1}\right)$.

4. *Вертикальный шаг*: вычисление сообщений от i-го вектора переменных к j-й группе проверок:

$$\bar{\alpha}'_i\bar{\beta}'_i = \bar{\alpha}_i\bar{\beta}_i + \sum_{t=1}^{l}\pi_{ti}^{-1}(\bar{\gamma}_{ti});$$

$$\bar{\alpha}_{ji}\bar{\beta}_{ji} = \pi_{ji}\left(\bar{\alpha}_i\bar{\beta}_i + \sum_{\substack{t=1\\ t\neq j}}^{l}\pi_{ti}^{-1}(\bar{\gamma}_{ti})\right).$$

5. *Вычисление синдрома*: по вычисленным $\bar{\alpha}'_i\bar{\beta}'_i$, $1 \le i \le n_0$, строится «жесткое решение» $\mathbf{x}$ и вычисляется синдром $\mathbf{S}$; если $\mathbf{S} = 0$, то декодирование прекращается и $\mathbf{x}$ считается результатом работы декодера. Если синдром не нулевой, то возвращаемся на шаг 3.

Вертикальный и горизонтальный шаги выполняются ограниченное число раз. Если достигнуто максимальное число итераций, то алгоритм прерывается и блок считается принятым с ошибкой.

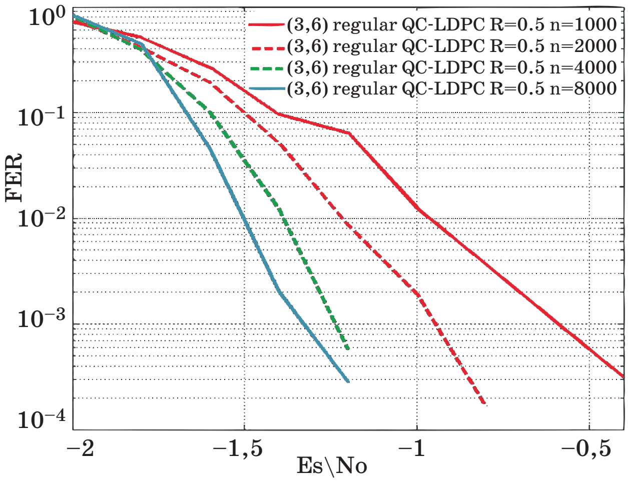
Описанный выше алгоритм оперирует только с векторами длины m, не обращаясь явно к отдельным символам. Таким образом, процесс декодирования можно осуществлять параллельно для m символов.

Результаты имитационного моделирования

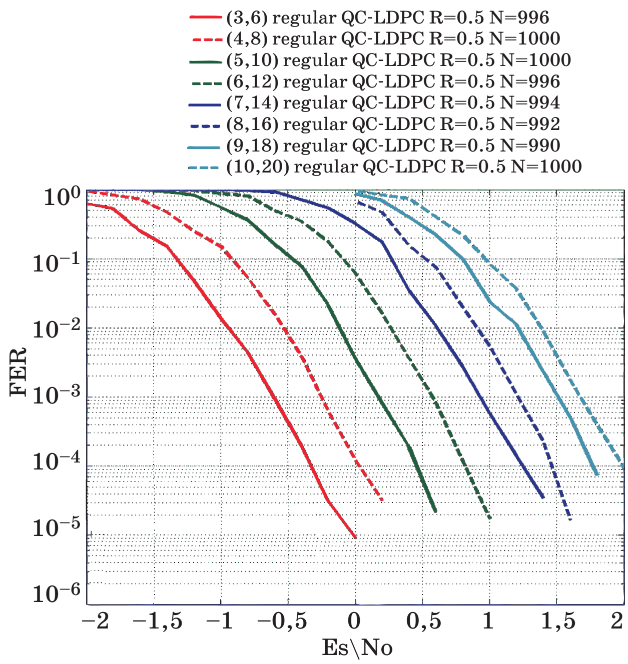
Для практической реализации описанного в статье алгоритма декодирования МПП-кодов, основанных на матрицах перестановок, была написана функция для MatLab. Для рассмотренного в работе декодера производилось имитационное моделирование с использованием среды MatLab. Передача данных осуществлялась по каналу с аддитивным белым гауссовым шумом и двоичной фазовой модуляцией. Максимальное число итераций ограничивалось 50. В качестве МПП-кодов, основанных на матрицах перестановок, были выбраны квазициклические коды QC-LDPC.

Результаты моделирования 4 кодов различных длин из ансамбля $\varepsilon_Q(l, n_0, m)$ при фиксированном числе слоев и столбцов и скорости представлены на рис. 2. Из рисунка следует, что энергетический выигрыш при использовании кода длины 2000 по сравнению с кодом длины 1000 составляет около 0,35 дБ (по уровню вероятности ошибки на блок 10^{-3}); в то же время при переходе от длины 2000 к 4000 выигрыш составляет уже порядка 0,3 дБ при аналогичном уровне вероятности ошибки; увеличение длины кода от 4000 к 8000 уже практически не улучшает корректирующих свойств (выигрыш менее 0,1 дБ).

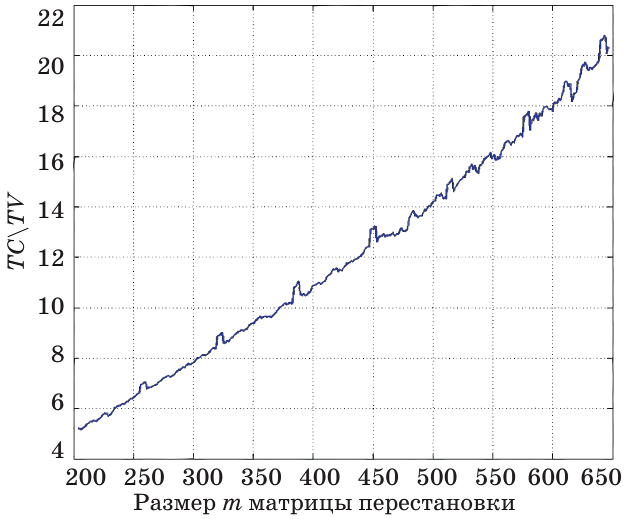
Результаты моделирования 8 квазициклических кодов длин $N_i \sim 1000$ при фиксированной скорости и различном числе слоев l представлены на рис. 3. Как следует из рисунка, наилучшими корректирующими свойствами обладает код с наименьшим числом слоев $l = 3$. Переход от $l = 3$ к $l = 4$ приводит к энергетическому проигрышу порядка 0,3 дБ (по уровню вероятности ошибки на блок 10^{-4}). Дальнейшее увеличение l также приводит к ухудшению корректирующих свойств кода, хотя при $l > 6$ ухудшение становится незначительным.



■ *Рис. 2.* Зависимость вероятности ошибки на блок (FER) от отношения сигнал/шум на кодовый бит (Es\No) для регулярных квазициклических МПП-кодов при различных длинах n



■ *Рис. 3.* Зависимость вероятности ошибки на блок (FER) от отношения сигнал/шум на кодовый бит (Es\No) для регулярных квазициклических МПП-кодов при различном числе слоев l



■ *Рис. 4.* Зависимость отношения $TC\backslash TV$ (TC — время декодирования классическим алгоритмом belief propagation, TV — время декодирования векторным belief propagation) от m для (3, 6) регулярного квазициклического МПП-кода при отношении сигнал/шум на бит Es\No = 2 дБ

В то же время из работ [1, 9, 10] следует, что увеличение l при фиксированной длине кода n и его скорости R приводит к увеличению минимального кодового расстояния d. Следовательно, чем больше число слоев l, тем меньшую часть доли исправляемых ошибок реализует алгоритм belief propagation.

Таким образом, belief propagation хорошо работает только с кодами с наихудшими потенциальными корректирующими свойствами.

Вывод о том, что предложенный в статье алгоритм декодирования уже при сравнительно небольших m дает выигрыш по времени декодирования минимум в 5 раз по сравнению с декодером, предложенным в работе [1], позволяет сделать рис. 4. Отметим, что моделирование обоих алгоритмов проводилось в среде MatLab. При $m = 600$ векторный алгоритм декодирования работает примерно в 20 раз быстрее классического belief propagation. Причем следует отметить практически линейную зависимость отношения TC/TV от m.

Заключение

Предложен векторный алгоритм декодирования МПП-кодов, основанных на матрицах перестановок. Для декодера осуществляется распараллеливание в m раз, где m достаточно велико. Данный подход позволяет существенно увеличить скорость декодирования. Поскольку к современным сигнально-кодовым конструкциям предъявляются достаточно жесткие требования по скорости обработки и передачи данных, то построенный декодер может иметь практическую ценность.

Литература

1. **Галлагер Р. Дж.** Коды с малой плотностью проверок на четность. — М.: Мир, 1966. — 90 с.
2. **Tanner M. A.** Recursive Approach to Low Complexity Codes // IEEE Trans. Inform. Theory. 1981. Vol. 27. N. 5. P. 533–547.
3. **Fossorier P. C.** Quasi-cyclic low-density parity-check codes from circulant permutation matrices // IEEE Trans. Inform. Theory. 2004. Vol. 50. N. 8. P. 1788–1793.
4. **Lu J., Moura M. F., Niesen U.** Grouping-and-shifting designs for structured LDPC codes with large girth // Proc. of IEEE Intern. Symp. on Information Theory (ISIT‘04). 2004. P. 236.
5. **Gabidulin E., Moinian A., Honary B.** Generalized construction of quasi-cyclic regular LDPC codes based on permutation matrices // Proc. of IEEE Intern. Symp. on Information Theory (ISIT‘06). 2006. P. 679–683.
6. **Ivanov F. I., Zyablov V. V., Potapov V. G.** Low-Destiny Parity-Check Codes Based on Galous Field // Information Processes. 2012. Vol. 12. N. 1. P. 68–83.
7. **Иванов Ф. И., Зяблов В. В., Потапов В. Г.** Оценка минимальной длины циклов квазициклических регулярных кодов с малой плотностью проверок на четность // Информационно-управляющие системы. 2012. № 3. С. 42–45.
8. **Иванов Ф. И., Зяблов В. В., Потапов В. Г.** Сравнение различных конструкций двоичных МПП-кодов, построенных на основе матриц перестановок // Информационные процессы. 2012. Т. 12. № 1. С. 31–52.
9. **Шридхаран А.** и др. О минимальном расстоянии низкоплотностных кодов с проверочными матрицами, составленными из перестановочных матриц // Проблемы передачи информации. 2005. Т. 41. № 1. С. 39–52.
10. **Зяблов В. В., Пинскер М. С.** Оценка сложности исправления ошибок низкоплотностными кодами Галлагера // Проблемы передачи информации. 1975. Т. 11. № 1. С. 23–36.