# Применение высокопроизводительных вычислений для анализа характеристик корректирующих кодов

к.т.н., доц. П.В. Трифонов
Санкт-Петербургский государственный политехнический университет

# План доклада

- Теория кодирования: области применения
- Методы теории кодирования
- Актуальные задачи
- Применение высокопроизводительных вычислений

# Теория помехоустойчивого кодирования

- 1948 –Шенноном заложены основы цифровой связи
  - Сколь угодно надежная передача данных возможна со скоростью, не превышающей пропускную способность канала
  - Необходимо использование помехоустойчивого кодирования
  - Конкретные методы кодирования и декодирования указаны не были
- Задачи теории кодирования
  - Построение корректирующих кодов для различных приложений
    - Код – некоторое множество векторов
  - Разработка эффективных алгоритмов их декодирования
    - Декодирование подразумевает и исправление ошибок
  - Анализ характеристик корректирующих кодов

# Приложения теории кодирования

- Связь
  - WiFi, WiMAX, Bluetooth, ADSL, GSM/LTE,…,
- Хранение данных
  - Жесткие диски, CD, DVD, BluRay,…
- Устройства оперативной и полуоперативной памяти
  - ECC RAM, флеш-память, кэш-память,...
- Скоростная передача данных по IP-сетям
  - Альтернатива TCP для соединений точка-точка
  - Широковещательная передача
- Сжатие изображений
- Стеганография и криптография
- Автоматическая классификация данных
- ...

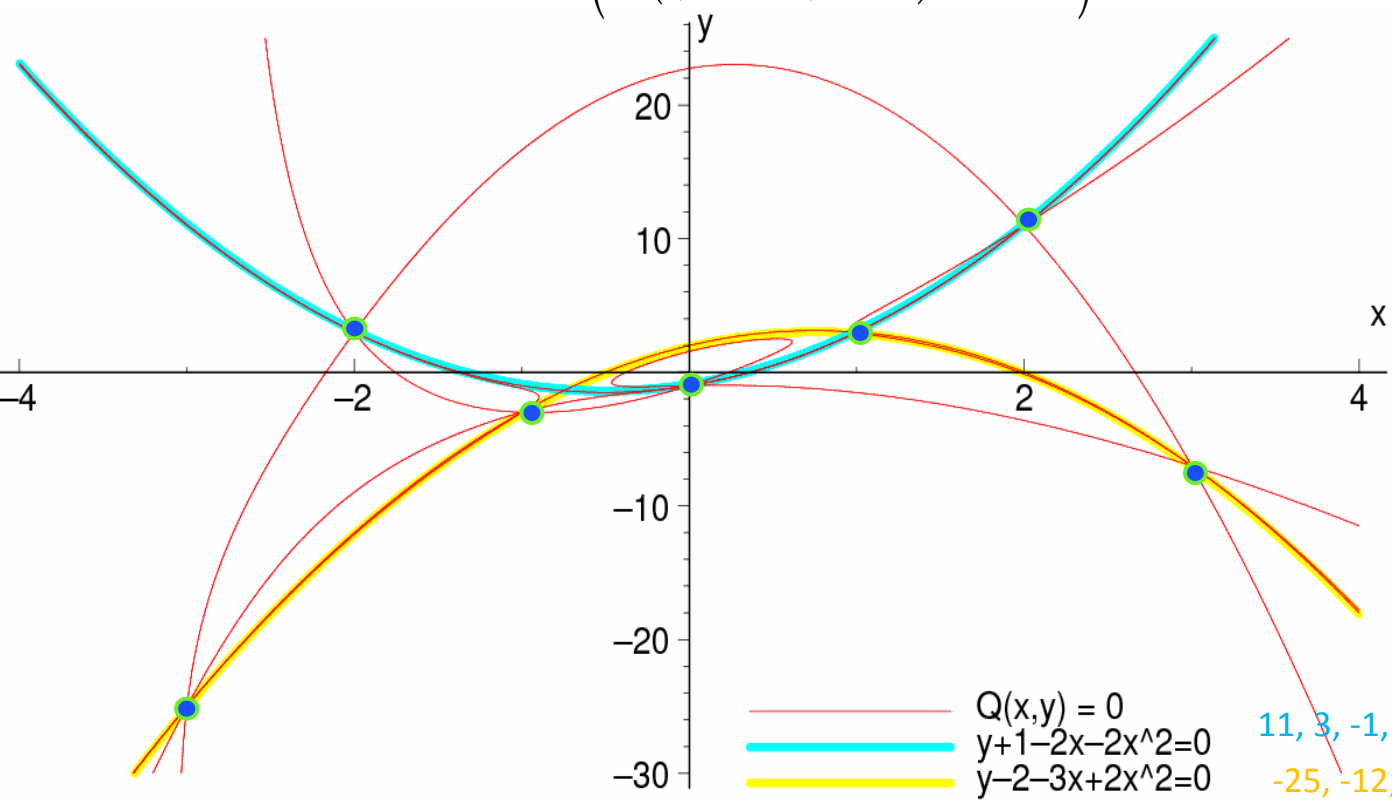
# Методы теории кодирования

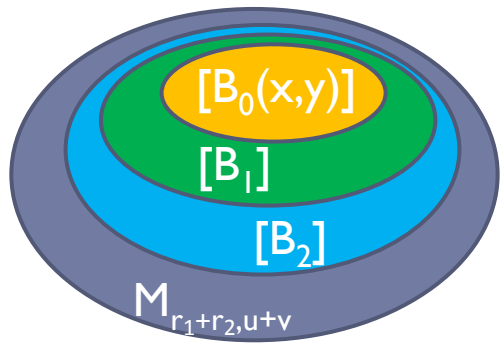
- Комбинаторная ТК
  - Код – множество векторов, удовлетворяющих простым ограничениям
    - Система линейных однородных алгебраических уравнений общего вида
  - Декодирование – перебор всех потенциально подходящих векторов
- Алгебраическая ТК
  - Код – множество векторов, удовлетворяющих некоторым нетривиальным алгебраическим уравнениям
  - Декодирование – поиск решения системы нелинейных алгебраических уравнений
- Вероятностная теория кодирования
  - Декодирование – расчет апостериорных вероятностей того, что было передано то или иное кодовое слово при условии принятия некоторых сигналов на выходе из канала

# Пример: алгебраическое декодирование

- (7,3,5) код Рида-Соломона $\{(f(x_1),\ldots,f(x_7)) \mid x_i \text{ различны}, \deg f(x) < 3\}$
- Задача: найти все кодовые слова, совпадающие с вектором *Y=( -25,3,-3,-1,3,11,-7)* не менее чем в 4 позициях из X=(*-3,-2,-1,0,1,2,3*)
- Построение базиса Грёбнера модуля интерполяционных многочленов

$$I_r = I_1^r = I_1^{\sum_{j=0}^{m} r_j 2^j} = \left(\ldots\left(\left(I_1^2 I_1^{r_{m-1}}\right) I_1^{r_{m-2}}\right)^2 I_1^{r_{m-3}} \ldots\right)^2 I_1^{r_0}, r_j \in \{0,1\}$$



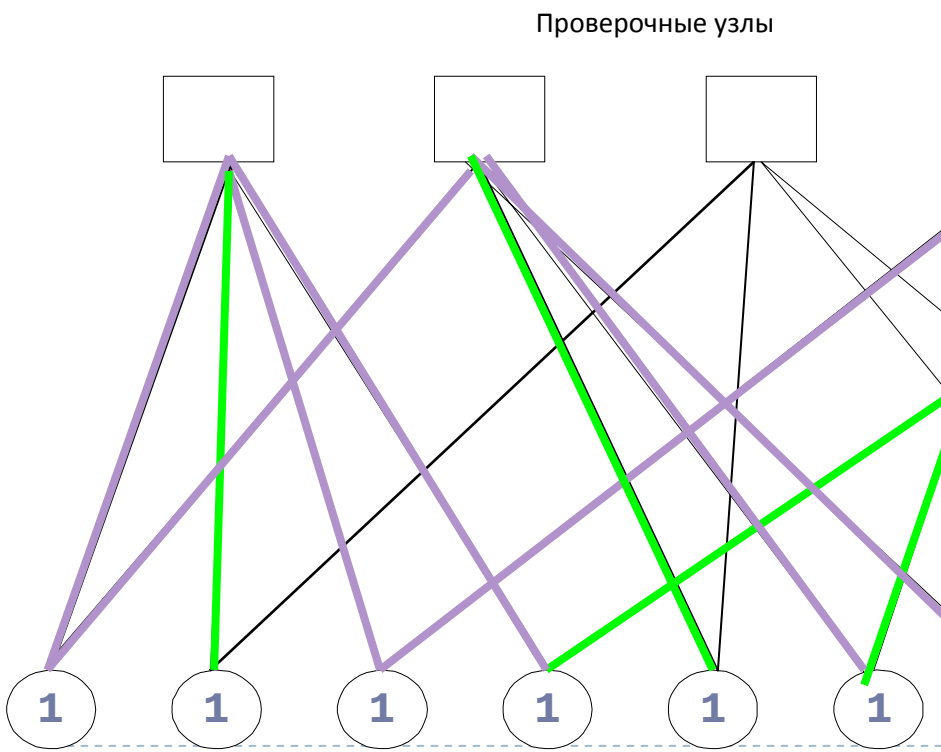




11, 3, -1, -1, 3, 11, 23

-25, -12, -3, 2, 3, 0, -7

# Пример: итеративное декодирование

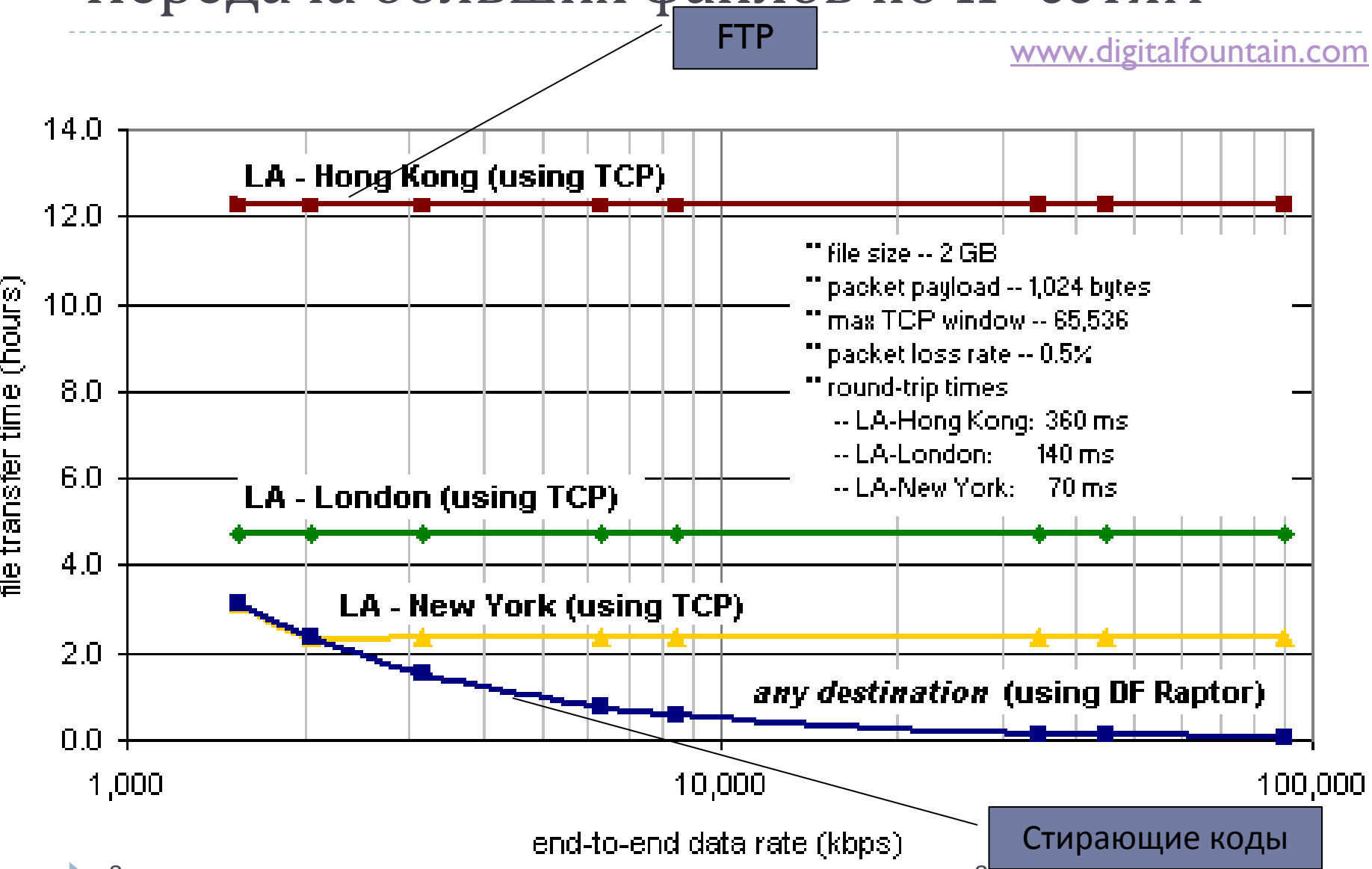
- Линейный код с малой плотностью проверок на четность – множество векторов c: $Hc^T=0$, где H – разреженная матрица, содержащая 0 и 1
  - Арифметика по модулю 2
- Двоичный стирающий канал: символы или принимаются достоверно, или теряются




$$\begin{cases} c_1 + c_2 + c_3 + c_4 = 0 \\ c_1 + c_5 + c_6 + c_7 = 0 \\ c_2 + c_5 + c_8 + c_9 = 0 \\ c_3 + c_6 + c_8 + c_{10} = 0 \\ c_4 + c_7 + c_9 + c_{10} = 0 \end{cases}$$

# Передача больших файлов по IP сетям

www.digitalfountain.com






8

# Пример: итеративное вероятностное декодирование

- На основе анализа принятых зашумленных сигналов $y_1,\ldots,y_n$ можно оценить $P\{c_i=1|y_i\}$ и $P\{c_i=0|y_i\}$
- Какое кодовое слово $(c_1,\ldots,c_n)$ наиболее вероятно при условии получения $(y_1,\ldots,y_n)$?

$$\begin{cases} c_1+c_2+c_3+c_4=0 \\ c_1+c_5+c_6+c_7=0 \\ c_2+c_5+c_8+c_9=0 \\ c_3+c_6+c_8+c_{10}=0 \\ c_4+c_7+c_9+c_{10}=0 \end{cases}$$

При каких условиях эти методы работают хорошо?

$$L(c_i)=\ln\left(\frac{P\{c_i=0|y_i\}}{P\{c_i=1|y_i\}}\right)=\frac{2y_i}{\sigma^2}$$

$$L(r_{ji})=\ln\left(\frac{r_{ji}(0)}{r_{ji}(1)}\right) \qquad L(q_{ij})=\ln\left(\frac{q_{ij}(0)}{q_{ij}(1)}\right)=\ln\left(\frac{P\{x_i=0|y_i\}K_{ij}\prod_{j'\in C_i/\{j\}}r_{j'i}(0)}{P\{x_i=1|y_i\}K_{ij}\prod_{j'\in C_i/\{j\}}r_{j'i}(1)}\right)=L(x_i)+\sum_{j'\in C_i/\{j\}}L(r_{j'i})$$

$$L(r_{ji})=2\prod_{i'\in V_j/\{i\}}\alpha_{i'j}\tanh^{-1}\left(\prod_{i'\in V_j/\{i\}}\tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)=\prod_{i'\in V_j/\{i\}}\alpha_{i'j}2\tanh^{-1}\ln^{-1}\ln\left(\prod_{i'\in V_j/\{i\}}\tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)=$$

$$=\prod_{i'\in V_j/\{i\}}\alpha_{i'j}2\tanh^{-1}\ln^{-1}\left(-\sum_{i'\in V_j/\{i\}}-\ln\tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)=\prod_{i'\in V_j/\{i\}}\alpha_{i'j}\phi^{-1}\left(\sum_{i'\in V_j/\{i\}}\phi\left(\beta_{i'j}\right)\right)=\prod_{i'\in V_j/\{i\}}\alpha_{i'j}\phi\left(\sum_{i'\in V_j/\{i\}}\phi\left(\beta_{i'j}\right)\right)$$

$$L(Q_i)=\ln\frac{P\{c_i=0|y_1,\ldots,y_n\}}{P\{c_i=1|y_1,\ldots,y_n\}}\approx L(c_i)+\sum_{j'\in C_i}L(r_{j'i})$$

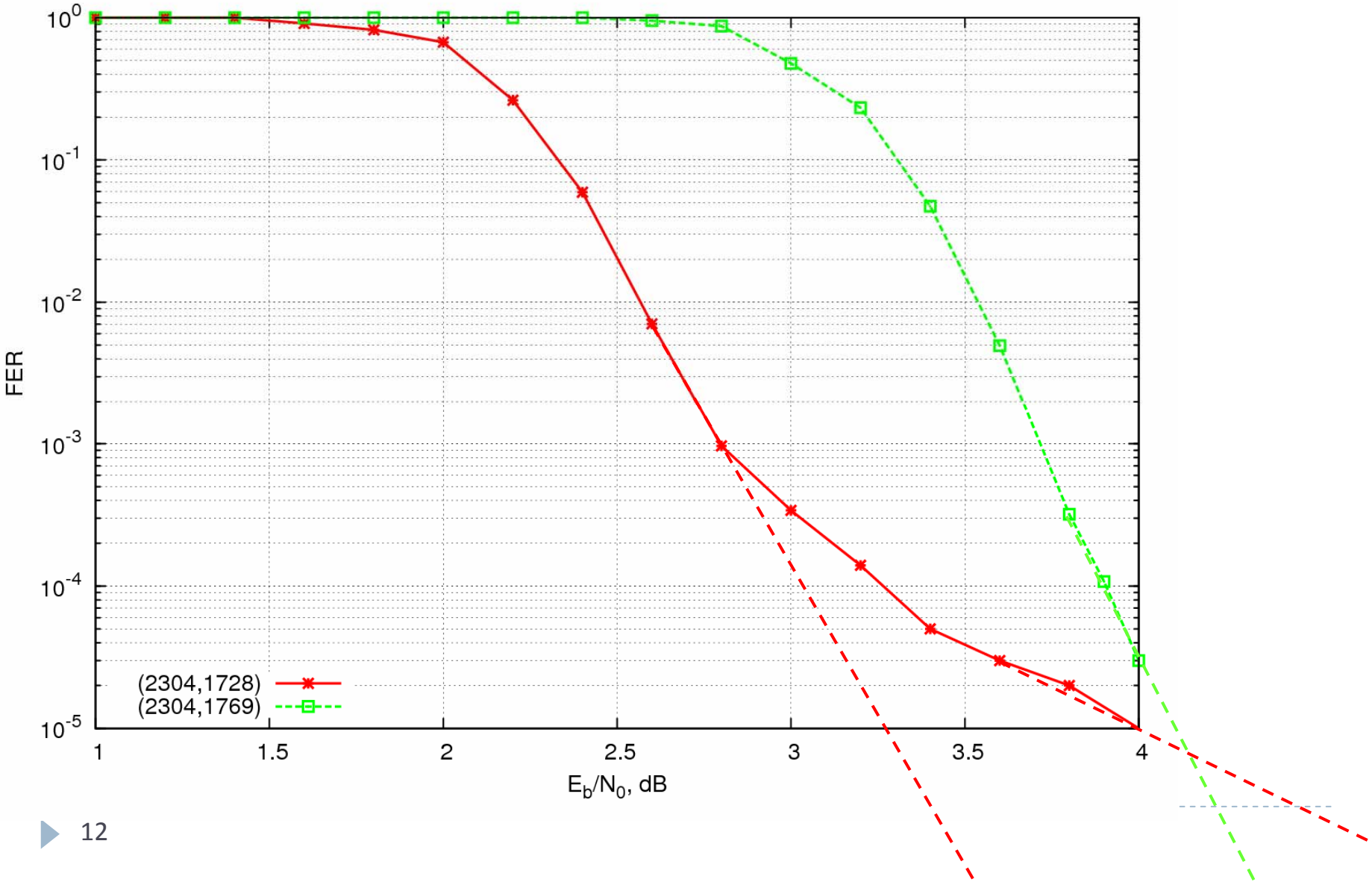
# Анализ характеристик корректирующих кодов

- С 1948 года развивалась алгебраическая теория кодирования
  - Большое число кодов
  - Достаточно точные оценки их параметров
  - Корректирующая способность далека от предела Шеннона
- 1993 год – появление турбо-кодов
- 1998 год – повторное открытие кодов с малой плотностью проверок на четность (1962 г.)
- Точные параметры этих кодов неизвестны
- Эксперименты показывают, что их корректирующая способность близка к пределу Шеннона

# Кодирование в современных системах связи

- 10G-Ethernet over UTP: необходимо обеспечить вероятность ошибки не более $10^{-12}$
- Как проверить выполнение этого требования?
  - Статистическое моделирование
    - Генерация случайных кодовых слов, наложение на них шума, декодирование, подсчет числа ошибок
    - Для получения достоверных оценок необходимо обработать не менее $10^{13}$ кодовых слов
    - Экстраполяция недопустима
      - Необходимо применение высокопроизводительных вычислительных систем
  - Известны приближенные выражения, связывающие вероятность ошибки и весовой спектр кода
    - Весовой спектр – число кодовых слов различного веса
    - Весовой спектр неизвестен; его поиск сводится к перебору кодовых слов
    - Число кодовых слов $\approx 2^{1000}$
      - Алгоритмы с ограничениями перебора
      - Применение высокопроизводительных вычислительных систем

# Проблема «насыщения»



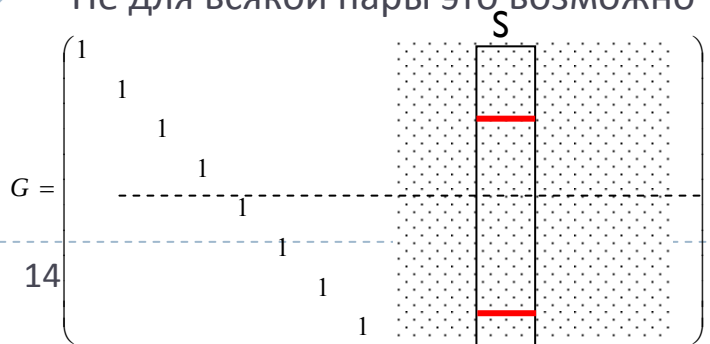
# Оценка характеристик корректирующих кодов

- Наименьший из весов ненулевых кодовых слов (минимальное расстояние), а также число кодовых слов с таким весом, определяют корректирующую способность кода
- Вычисление минимального расстояния произвольного кода является NP-полной проблемой
  - Равно как и задача его декодирования
- Для некоторых кодов существуют эффективные субоптимальные алгоритмы их декодирования
- Можно попытаться продекодировать нулевое кодовое слово с различными шумовыми векторами и зарегистрировать найденные ненулевые кодовые слова
- Метод ошибочного импульса
  - Декодируются вектора вида $(I+\eta_1, I+\eta_2, I+\eta_3, I+\eta_4, \ldots, I+\eta_i, -A, I+\eta_{i+2}, \ldots, I+\eta_n)$
    - A – большое число
    - $\eta_n$ – нормально распределенные случайные величины
  - Результатом ошибочного декодирования, как правило, являются кодовые слова малого веса

# Поиск кодовых слов малого веса в линейных кодах

- Число кодовых слов малого веса определяет корректирующую способность кода
- Перебор: вероятность нахождения слова малого веса исчезающе мала
  - Для многих кодов распределение числа слов различного веса близко к нормальному
- Необходимо заранее отсеять кодовые слова большого веса
- Алгоритм Канто-Шабада

1. Выбрать произвольную информационную совокупность и диагонализировать порождающую матрицу на ней
2. Разбить произвольным образом множество строк на два подмножества $Z_1$, $Z_2$
3. Выбрать произвольное $\sigma$-элементное подмножество S проверочных символов
4. Рассмотреть все p-элементные комбинации строк $Z_1$, $Z_2$ и вычислить значения позиций из S, занеся результаты в таблицы $T_1$ и $T_2$
5. Выявить равные элементы в $T_1$ и $T_2$, сложить соответствующие им кодовые слова и вычислить их вес
6. Выбрать случайным образом один информационный и один проверочный символы и перейти к новой информационной совокупности, поменяв их
   - Не для всякой пары это возможно




$$W = O(2^{0.12\ nh_2(R+0.031)+10})$$

# Вопросы реализации

- Алгоритм Канто-Шабада поиска кодовых слов малого веса
  - XOR двоичных векторов длины 500 – 1000
    - Вектор int64 длины ~ 16
  - Вычисление веса двоичного вектора
  - Операция извлечения битов
  - SIMD
- Статистическое моделирование, метод ошибочного импульса
  - Векторное вычисление функций exp, log, tanh
  - Сложение векторов чисел с плавающей запятой

# Реализация с использованием MPI

- Обработка одного кодового слова/информационной совокупности занимает время менее $10^{-4}$ секунд
- Кодовые слова обрабатываются независимо
- Сеть передачи сообщений используется только для распределения заданий и сбора результатов
  - Менее 1 сообщения в секунду

# Применение высокопроизводительных вычислений

- Использование технологии OpenMP и 4-ядерных процессоров позволяет надежно оценивать минимальное расстояние кодов длиной ~ 250
- Применение самодельного кластера позволило увеличить длину анализируемых кодов до ~500
  - Рабочие станции получены в рамках гранта HP Innovation in Education
- Планируется использовать ресурсы программы «Университетский кластер» для анализа кодов длиной ~2000

# Заключение

- Разработка методов помехоустойчивого кодирования для современных систем связи требует точной оценки параметров корректирующих кодов
- Задача оценки минимального расстояния линейного кода является NP-полной
- Применение систем высокопроизводительных вычислений позволит упростить процесс выбора корректирующих кодов