Универсальный метод Ланцоша-Паде решения линейных систем над большими конечными полями*

Н.Л. Замарашкин[2], М.А. Черепнев[1]

МГУ[1], ИВМ РАН[2]

Предложен универсальный метод, предназначенный для решения больших разреженных систем линейных уравнений над конечными полями с большим числом элементов, которые возникают при решении задачи дискретного логарифмирования по модулю простого числа. Предложены эффективные способы представления и распределения данных, а также программные реализации параллельных алгоритмов.

1. Введение

Одним из методов, которые используются для решения больших разреженных систем над конечными полями, является метод Ланцоша (см., например [2–9,11,12]). Простота алгоритма и возможность постоянного контроля корректности его работы относятся к главным достоинствам метода. Однако по своей природе метод Ланцоша является последовательными алгоритмом, на каждой итерации которого неизбежно происходит большой обмен данными между различными вычислительными узлами.

Использование блочных версий несколько улучшает масштабируемость параллельных программ (см. например, [12]), но так или иначе не позволяет эффективно использовать вычислительные системы с производительностью более 100 TFlops. Несовершенство метода становится тем заметнее, чем ниже скорость обмена данными в коммуникационной сети вычислительной системы.

В работе описан *универсальный* блочный метод Ланцоша-Паде, предназначенный для решения больших разреженных систем линейных уравнений над конечными полями с большим числом элементов («универсальный» означает «эффективный на параллельных вычислительных системах с различной архитектурой и характеристиками»), а также предлагаются эффективные способы представления данных, распределения данных по вычислительным узлам и параллельные алгоритмы.

Универсальный метод Ланцоша-Паде, сохраняя основные достоинства метода Ланцоша, обладает существенно лучшими параллельными свойствами. Более того в универсальном методе Ланцоша-Паде имеется возможность гибко управлять сложностью вычислений и размером обмениваемых данных.

2. Описание универсального метода Ланцоша-Паде

2.1. A-ортогональный базис пространства Крылова и матричные приближения Паде

Нам потребуется ряд конструкций, чтобы установить связь между матричными приближениями Паде некоторого специального ряда и A-ортогональными базисами пространства Крылова $\mathcal{K}(A, B)$.

Итак, пусть $\alpha(x)$ – формальный матричный ряд

$$\alpha(x) = \sum_{i=0}^{\infty} \alpha_i x^{-i}, \tag{1}$$

*Работа выполнена при финансовой поддержке Минобрнауки России, соглашение от 17 июня 2014 г. N 14.604 .21.0034 (идентификатор соглашения RFMEFI60414X0034).

коэффициентами которого являются матрицы $\alpha_i \in \mathbb{F}^{K\times K}$. Будем говорить, что пара матричных полиномов $\mathcal{Q}^{(s)}(x)$ и $\mathcal{P}^{(s)}(x)$ степени s

$$\begin{aligned}\mathcal{Q}^{(s)}(x) &= \mathcal{Q}_0^{(s)} + \mathcal{Q}_1^{(s)}x + \cdots + \mathcal{Q}_s^{(s)}x^s,\\ \mathcal{P}^{(s)}(x) &= \mathcal{P}_0^{(s)} + \mathcal{P}_1^{(s)}x + \cdots + \mathcal{P}_s^{(s)}x^s,\end{aligned}$$

с коэффициентами $\mathcal{Q}_i^{(s)}, \mathcal{P}_j^{(s)} \in \mathbb{F}^{K\times K}$, задает *матричное приближение Паде степени* s для ряда (1), если выполняется соотношение:

$$\alpha(x)\mathcal{Q}^{(s)}(x) - \mathcal{P}^{(s)}(x) = \sum_{i=s+1}^{\infty} \rho_i^{(s)} x^{-i}. \tag{2}$$

Ряд в правой части (2) будем называть *остаточным рядом* приближения Паде $\mathcal{P}^{(s)}(x)$, $\mathcal{Q}^{(s)}(x)$ степени s и обозначать его через $\mathcal{R}^{(s)}(x)$.

Далее, для ряда (1) и пары произвольных матричных полиномов определим *билинейное отображение*. А именно, пусть $\mathcal{Q}_1(x)$ и $\mathcal{Q}_2(x)$ два матричных полинома. Рассмотрим ряд, полученный формальным произведением $\mathcal{Q}_1^T(x)$, $\alpha(x)$ и $\mathcal{Q}_2(x)$,

$$\mathcal{Q}_1^T(x)\alpha(x)\mathcal{Q}_2(x) = \sum_{i=-t}^{\infty} \gamma_i x^{-i}. \tag{3}$$

Тогда значение билинейного отображения $< \mathcal{Q}_1, \mathcal{Q}_2 >$ для пары полиномов $\mathcal{Q}_1$ и $\mathcal{Q}_2$ определяется как значение коэффициента γ_1 формального ряда (3):

$$< \mathcal{Q}_1, \mathcal{Q}_2 >= \gamma_1 \in \mathbb{F}^{K\times K}. \tag{4}$$

Наконец, установим способ построения блока Q размера $n\times K$ по заданному матричному $\mathcal{Q}$ полиному $\mathcal{Q}(x)$, квадратной $n \times n$ матрице A и $n \times K$ блоку B. Для этого рассмотрим полином $\mathcal{Q}(x)$ степени s с матричными коэффициентами $\mathcal{Q}_0$, $\mathcal{Q}_1$, $\cdots$, $\mathcal{Q}_s$:

$$\mathcal{Q}(x) = \mathcal{Q}_0 + \mathcal{Q}_1 x + \cdots + \mathcal{Q}_s x^s.$$

Определим блок $Q = \mathcal{Q}(A, B) \in \mathbb{F}^{n\times K}$ по следующему правилу (см. например, [9]):

$$Q = \mathcal{Q}(A, B) = B\mathcal{Q}_0 + AB\mathcal{Q}_1 + \cdots + A^sB\mathcal{Q}_s. \tag{5}$$

В дальнейшем нас будут интересовать только те ряды $\alpha(x)$ вида (1), коэффициенты α_i которых задаются некоторым специальным образом, а именно

$$\alpha_i = \alpha_i^T = B^T A^i B,$$

с симметричной матрицей A, и некоторым блоком B.

Лемма 1. [9]*Справедливо следующее равенство:*

$$< \mathcal{Q}_1, \mathcal{Q}_2 >= Q_1^T A Q_2 \tag{6}$$

Утверждение 1. *Пусть $A \in \mathbb{F}^{n\times n}$ симметричная матрица, $B \in \mathbb{F}^{n\times K}$ блок из K линейно независимых векторов, а $\alpha_i \in \mathbb{F}^{K\times K}$ квадратные матрицы вида $\alpha_i = \alpha_i^T = B^T A^i B$. Рассмотрим приближения Паде $\mathcal{P}^{(s)}(x)$, $\mathcal{Q}^{(s)}(x)$ ряда (1) степеней $0, \cdots, l$,*

$$\begin{aligned}\alpha(x)\mathcal{Q}^{(0)}(x) - \mathcal{P}^{(0)}(x) &= \sum_{i=1}^{\infty} \rho_i^{(0)} x^{-i},\\ \alpha(x)\mathcal{Q}^{(1)}(x) - \mathcal{P}^{(1)}(x) &= \sum_{i=2}^{\infty} \rho_i^{(1)} x^{-i},\\ &\cdots \quad \cdots \quad \cdots,\\ \alpha(x)\mathcal{Q}^{(l)}(x) - \mathcal{P}^{(l)}(x) &= \sum_{i=l+1}^{\infty} \rho_i^{(l)} x^{-i},\end{aligned}$$

предполагая, что матрицы $Q^{(s)}(A,B)^T A Q^{(s)}(A,B)$ *являются невырожденными.*

Тогда, для любого s, $0 \le s \le l$ *блоки* $Q_0 = \mathcal{Q}^{(0)}(A,B)$, $\cdots$, $Q_l = \mathcal{Q}^{(l)}(A,B)$, *построенные по* $\mathcal{Q}$*-полиномам приближений Паде, задают* A*-ортогональный базис пространства Крылова* $\mathcal{K}_l(A,B)$

$$\mathcal{K}_l(A,B) = \mathrm{Span}\left(B, AB, \cdots, A^l B\right).$$

Кроме того, $Q^{(s)}(A,B)^T A Q^{(s)}(A,B) = Q_s^{(s)T} \rho_{s+1}^{(s)}$, *и старшие коэффициенты матричных* $\mathcal{Q}$*-полиномов и старшие коэффициенты соответствующих остаточных рядов, стоящие в последнем равенстве справа, являются невырожденными матрицами.* □

Утверждение 1 предоставляет способ описания A-ортогонального базиса пространства Крылова $\mathcal{K}(A,B)$ с помощью $\mathcal{Q}$-полиномов.

2.2. Рекуррентные формулы для приближений Паде

Предположим, что уже известны приближения Паде $\mathcal{P}^{(s-1)}$, $\mathcal{Q}^{(s-1)}$ и $\mathcal{P}^{(s)}$, $\mathcal{Q}^{(s)}$ порядков $s-1$ и s (здесь и везде далее верхний индекс в круглых скобках говорит о том, к какому порядку приближения Паде относится данная величина), соответственно, с остаточными рядами $\mathcal{R}^{(s-1)}(x)$ и $\mathcal{R}^{(s)}(x)$

$$\begin{aligned}
\mathcal{R}^{(s-1)}(x) &= \alpha(x)\mathcal{Q}^{(s-1)}(x) - \mathcal{P}^{(s-1)}(x) = \sum_{i=s}^{\infty} \rho_i^{(s-1)} x^{-i}, \\
\mathcal{R}^{(s)}(x) &= \alpha(x)\mathcal{Q}^{(s)}(x) - \mathcal{P}^{(s)}(x) = \sum_{i=s+1}^{\infty} \rho_i^{(s)} x^{-i}.
\end{aligned}$$

Новое приближение Паде $\mathcal{Q}^{(s+1)}(x)$, $\mathcal{P}^{(s+1)}(x)$ степени $s+1$ будем искать в виде

$$\mathcal{Q}^{(s+1)}(x) = x\mathcal{Q}^{(s)}(x) + \mathcal{Q}^{(s)}(x)\nu_0 + \mathcal{Q}^{(s-1)}(x)\nu_1, \quad (7)$$

$$\mathcal{P}^{(s+1)}(x) = x\mathcal{P}^{(s)}(x) + \mathcal{P}^{(s)}(x)\nu_0 + \mathcal{P}^{(s-1)}(x)\nu_1, \quad (8)$$

$$\mathcal{R}^{(s+1)}(x) = x\mathcal{R}^{(s)}(x) + \mathcal{R}^{(s)}(x)\nu_0 + \mathcal{R}^{(s-1)}(x)\nu_1, \quad (9)$$

где ν_0 и ν_1 – $K \times K$ матрицы, требующие определения. Из условия равенства нулю коэффициентов $\rho_s^{(s+1)}$ и $\rho_{s+1}^{(s+1)}$ остатка ряда Паде $\mathcal{R}^{(s+1)}(x)$, следует, что матрицы ν_0 и ν_1 удовлетворяют системе линейных уравнений

$$\rho_{s+1}^{(s)} + \rho_s^{(s-1)}\nu_1 = \rho_s^{(s+1)} = 0, \quad (10)$$

$$\rho_{s+2}^{(s)} + \rho_{s+1}^{(s)}\nu_0 + \rho_{s+1}^{(s-1)}\nu_1 = \rho_{s+1}^{(s+1)} = 0. \quad (11)$$

Оставшиеся коэффициенты $\rho_{s+k+1}^{(s+1)}$ остатка $\mathcal{R}^{(s+1)}(x)$ с $k > 0$ получаются по следующей формуле:

$$\rho_{s+k+1}^{(s+1)} = \rho_{s+k+2}^{(s)} + \rho_{s+k+1}^{(s)}\nu_0 + \rho_{s+k+1}^{(s-1)}\nu_1.$$

Следуя (7), запишем для полинома $\mathcal{Q}^{(s+1)}(x)$

$$\mathcal{Q}^{(s+1)}(x) = \mathcal{Q}^{(s)}(x)\left(Ix + \nu_0\right) + \mathcal{Q}^{(s-1)}(x)\nu_1. \quad (12)$$

Применяя данное преобразование t раз, получим представление для полинома $\mathcal{Q}^{(s+t)}(x)$ в виде:

$$\mathcal{Q}^{(s+t)}(x) = \mathcal{Q}^{(s)}(x)\mathcal{H}_{1s}^{(t)}(x) + \mathcal{Q}^{(s-1)}(x)\mathcal{H}_{0s}^{(t)}(x), \quad (13)$$

где $\mathcal{H}_{1s}^{(t)}(x)$ – матричный полином степени t, старший коэффициент которого единичная $K \times K$ матрица, а $\mathcal{H}_{0s}^{(t)}(x)$ – некоторый матричный полином степени не выше $t-1$.

Подход к вычислению A-ортогонального базиса, основанный на приближениях Паде, имеет видимые преимущества с точки зрения параллельных вычислений. В классическом методе Ланцоша итерация алгоритма состоит из умножения матрицы A на блок Q_i и последующей процедуры A-ортогонализации нового блока к предыдущим. Каждая такая итерация требует значительного обмена данными, что приводит к плохой масштабируемости алгоритма на мощных вычислителях. Для метода Ланцоша-Паде требуется только умножать блок на матрицу. Каждый вектор в блоке умножается независимо, а число обменов минимально. Это и есть главное преимущество метода Ланцоша-Паде перед методом Ланцоша.

2.3. Матричные обозначения

Не сложно проверить, что полиномы $\mathcal{Q}^{(0)}(x)$ и $\mathcal{Q}^{(1)}(x)$, взятые в виде $\mathcal{Q}^{(0)}(x) = I$ и $\mathcal{Q}^{(1)}(x) = Ix - \alpha_1^{-1}\alpha_2$, с $\alpha_1 = B^TAB$ и $\alpha_2 = B^TA^2B$, являются $\mathcal{Q}$-полиномами Паде нулевой и первой степени соответственно. Используя частный случай соотношения (13), представим $\mathcal{Q}$-полином Паде степени $s+1$ в виде

$$\mathcal{Q}^{(s+1)}(x) = \mathcal{Q}^{(1)}(x)\mathcal{H}_1^{(s+1)}(x) + \mathcal{Q}^{(0)}(x)\mathcal{H}_0^{(s+1)}(x), \tag{14}$$

с матричным полиномом $\mathcal{H}_1^{(s+1)}(x)$ степени s и старшим коэффициентом, равным единичной матрице, и некоторым матричным полином $\mathcal{H}_0^{(s+1)}(x)$ степени не выше $s-1$. Из (12) и (14) следует, что $\mathcal{H}_1^{(s+1)}(x)$ и $\mathcal{H}_0^{(s+1)}(x)$ удовлетворяют одному и тому же рекуррентному соотношению

$$\mathcal{H}_1^{(s+1)}(x) = \mathcal{H}_1^{(s)}(x)\,(Ix + \nu_0) + \mathcal{H}_1^{(s-1)}(x)\nu_1 \tag{15}$$

$$\mathcal{H}_0^{(s+1)}(x) = \mathcal{H}_0^{(s)}(x)\,(Ix + \nu_0) + \mathcal{H}_0^{(s-1)}(x)\nu_1, \tag{16}$$

отличаясь только выбором начальных условий. А именно, начальные полиномы выбираются в следующем виде: $\mathcal{H}_1^{(0)}(x) = 0$, $\mathcal{H}_1^{(1)}(x) = I$, $\mathcal{H}_0^{(0)}(x) = I$, $\mathcal{H}_0^{(1)}(x) = 0$.

Для удобства дальнейшей записи алгоритмов определим специальные матричные обозначения. Составим из коэффициентов полиномов $\mathcal{H}_1^{(s)}(x)$ и $\mathcal{H}_0^{(s)}(x)$ блоки $H_1^{(s)}$ и $H_0^{(s)}$ размера $n \times K$, расположив коэффициенты меньших степеней в строках с большими номерами:

$$H_1^{(s)} = \begin{bmatrix} 0 \\ \cdots \\ 0 \\ \mathcal{H}_{1(s-1)}^{(s)} \\ \mathcal{H}_{1(s-2)}^{(s)} \\ \cdots \\ \mathcal{H}_{11}^{(s)} \\ \mathcal{H}_{10}^{(s)} \end{bmatrix}, \quad H_0^{(s)} = \begin{bmatrix} 0 \\ \cdots \\ 0 \\ 0 \\ \mathcal{H}_{0(s-2)}^{(s)} \\ \cdots \\ \mathcal{H}_{01}^{(s)} \\ \mathcal{H}_{00}^{(s)} \end{bmatrix}. \tag{17}$$

В таком случае рекуррентные соотношения (15) и (16) эквивалентны следующим матрич-

ным равенствам:

$$H_i^{(s+1)} = \begin{bmatrix} ZH_i^{(s)} & H_i^{(s)} & H_i^{(s-1)} \end{bmatrix} \begin{bmatrix} I \\ \nu_0 \\ \nu_1 \end{bmatrix}, \quad i = 0, 1, \tag{18}$$

где $Z \in \mathbb{F}^{n \times n}$ – матрица блочного сдвига вида

$$Z = \begin{bmatrix} 0 & I_k & 0 & \cdots & 0 \\ 0 & 0 & I_k & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & I_k \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}.$$

Наряду с блоками $H_0^{(s)}$ и $H_1^{(s)}$ определим блок $R^{(s)}$ размера $2n \times K$, который соответствует остаточному ряду Паде степени s (заметим, что размер блоков R вдвое превосходит размер блоков H). Явный вид блока $R^{(s)}$ зададим следующим образом:

$$R^{(s)} = \begin{bmatrix} \star \\ \star \\ \rho^{(s)}_{2\frac{n}{K}-s} \\ \cdots \\ \rho^{(s)}_{s+1} \\ 0 \\ \cdots \\ 0 \end{bmatrix}, \tag{19}$$

где символом $\star$ обозначаются те элементы, которые не являются существенными (другими словами значения этих элементов не будут влиять на вычисления в алгоритме). Число нулевых матриц в $R^{(s)}$ в точности равно $s+1$, и отражает тот факт, что $R^{(s)}$ соответствует остаточному ряду Паде степени s. По аналогии с формулами (18) запишем

$$R^{(s+1)} = \begin{bmatrix} Z^T R^{(s)} & R^{(s)} & R^{(s-1)} \end{bmatrix} \begin{bmatrix} I \\ \nu_0 \\ \nu_1 \end{bmatrix} \tag{20}$$

Заметим, что в соответствии с определением число «несущественных» элементов в $R^{(s+1)}$ увеличивается на 1 по сравнению с числом «несущественных» элементов в $R^{(s)}$. Таким образом, формула (20) может рассматриваться как равенство только для «существенных» компонент $R^{(s+1)}$.

Далее, обозначим через Q_1 и Q_0 блоки $\mathcal{Q}_1(A,B)$ и $\mathcal{Q}_0(A,B)$, соответственно. По определению $Q_0 = B$, а $Q_1 = AB - B\alpha_1^{-1}\alpha_2$. Составим матрицы $K_0,\ K_1 \in \mathbb{F}^{n\times n}$ из блоков вида A^jQ_0 и A^jQ_1 с j от 0 до $\frac{n}{K}-1$ записанных в обратном порядке,

$$\begin{aligned} K_0 &= \left[\begin{array}{cccc} A^{\frac{n}{K}-1}Q_0 & A^{\frac{n}{K}-2}Q_0 & \cdots & AQ_0 & Q_0 \end{array}\right] \\ K_1 &= \left[\begin{array}{cccc} A^{\frac{n}{K}-1}Q_1 & A^{\frac{n}{K}-2}Q_1 & \cdots & AQ_1 & Q_1 \end{array}\right] \end{aligned} \tag{21}$$

Из (13) следует, что для блока $Q_{s+1} = \mathcal{Q}^{(s+1)}(A,B)$ справедливо следующее представление:

$$Q_{s+1} = \sum_{i=0}^{s} A^iQ_1\mathcal{H}_{1i}^{(s+1)} + \sum_{i=0}^{s-1} A^iQ_0\mathcal{H}_{0i}^{(s+1)}, \tag{22}$$

которое удобно записать в виде

$$Q_{s+1} = K_0H_0^{(s+1)} + K_1H_1^{(s+1)}. \tag{23}$$

По построению (см. утверждение 1) блоки Q_s образуют A-ортогональный базис всего пространства, а, следовательно, решение системы $X_{\frac{n}{K}}$ записывается в виде линейной комбинации блоков Q_i

$$X_{\frac{n}{K}} = \sum_{i=0}^{\frac{n}{K}-1} Q_i\left(Q_i^TAQ_i\right)^{-1}Q_i^TB.$$

Вводя обозначение Z_i для матриц

$$Z_i = \left(Q_i^TAQ_i\right)^{-1}Q_i^TB,$$

придем к краткой записи для $X_{\frac{n}{K}}$

$$X_{\frac{n}{K}} = \sum_{i=0}^{\frac{n}{K}-1} Q_iZ_i. \tag{24}$$

Наконец, подставляя в (24) выражение для Q_i из (23), найдем, что

$$\begin{aligned} X_{\frac{n}{K}} &= \sum_{i=0}^{\frac{n}{K}-1} Q_iZ_i = \sum_{i=0}^{\frac{n}{K}-1}\left(K_0H_0^{(i)} + K_1H_1^{(i)}\right)Z_i \\ &= K_0\left(\sum_{i=0}^{\frac{n}{K}-1} H_0^{(i)}Z_i\right) + K_1\left(\sum_{i=0}^{\frac{n}{K}-1} H_1^{(i)}Z_i\right). \end{aligned}$$

Чтобы еще немного упростить запись и получить матричное соотношение для $X_{\frac{n}{K}}$, определим систему блоков $G_0^{(s)}$ и $G_1^{(s)}$ размера $n\times K$ следующего вида

$$G_0^{(s)} = \sum_{i=0}^{s} H_0^{(i)}Z_i = G_0^{(s-1)} + H_0^{(s)}Z_s; \tag{25}$$

$$G_1^{(s)} = \sum_{i=0}^{s} H_1^{(i)}Z_i = G_1^{(s-1)} + H_1^{(s)}Z_s, \tag{26}$$

где s меняется в пределах от 1 до $\frac{n}{K}-1$. Используя (25) и (26), окончательно запишем $X_{\frac{n}{K}}$ как

$$X_{\frac{n}{K}} = K_0G_0^{(\frac{n}{K}-1)} + K_1G_1^{(\frac{n}{K}-1)}. \tag{27}$$

Преимущества полученной записи для $X_{\frac{n}{K}}$ состоит в том, что в ней нет явной зависимости от блоков Q_i, а имеется зависимость лишь от систем K_0 и K_1 блочно крыловского типа. Вычисление K_0 и K_1 легко производить на параллельных вычислительных системах. Для этого каждый из $2K$ векторов, входящих в блок Q_0 или Q_1 можно умножать на матрицу A независимо, тем самым достигая максимально возможного эффекта. Чтобы завершить описание метода, необходимо предъявить способы вычисления блоков $G_0^{(\frac{n}{K})}$ и $G_1^{(\frac{n}{K})}$, которые не использовали бы явный вид блоков Q_i.

2.4. Вычисление $Q_i^T B$

Используя рекуррентные соотношения (12)

$$\mathcal{Q}^{(i)}(x) = \mathcal{Q}^{(i-1)}(x)\,(Ix + \nu_0) + \mathcal{Q}^{(i-2)}(x)\nu_1,$$

запишем для блоков Q_i, Q_{i-1} и Q_{i-2}

$$Q_i = AQ_{i-1} + Q_{i-1}\nu_0 + \tilde{Q}_{i-2}\nu_1.$$

Тогда

$$Q_i^T B = Q_{i-1}^T AB + \nu_0^T Q_{i-1}^T B + \nu_1^T Q_{i-2}^T B. \tag{28}$$

В силу свойства A-ортогональности для $i > 2$ первое слагаемое в (28) всегда равно нулю. Окончательно

$$Q_i^T B = \sum_{j=0}^{1} \nu_j^T \left(Q_{i-j-1}^T B\right) = \nu_0 Q_{i-1}^T B + \nu_1 Q_{i-2}^T B. \tag{29}$$

2.5. Вычисление $Q_i^T AQ_i$

Чтобы вычислить $Q_i^T AQ_i$ воспользуемся утверждением 1:

$$Q_i^T AQ_i \;\; = \left(\mathcal{Q}_i^{(i)}\right)^T \rho_{i+1}^{(i)}$$

2.6. Алгоритм Ланцоша-Паде

Выпишем алгоритм Ланцоша-Паде.

Шаг 0 (инициализация алгоритма):

(a) вычислить $\alpha_0 = B^T B$, $\alpha_1 = B^T AB$, $\alpha_2 = B^T A^2 B$;

(b) определить $n \times K$ блок Q_0 и Q_1 по формулам

$$Q_0 = B; \tag{30}$$

$$Q_1 = AB - B\alpha_1^{-1}\alpha_2; \tag{31}$$

(c) определить $n \times K$ блоки $H_0^{(0)}$, $H_0^{(1)}$ $H_1^{(0)}$ и $H_1^{(1)}$;

(d) вычислить квадратные $K \times K$ матрицы ψ_0 и ψ_1 по формулам

$$\psi_0 = Q_0^T B = B^T B, \quad \psi_1 = Q_1^T B;$$

(e) вычислить квадратные матрицы Z_0 и Z_1 как

$$Z_0 = \alpha_0^{-1}\psi_0 = (Q_0^T AQ_0)^{-1}\psi_0, \quad Z_1 = (Q_1^T AQ_1)^{-1}\psi_1;$$

(f) вычислить блоки $G_0^{(1)}$ и $G_1^{(1)}$ по формулам:

$$G_0^{(1)} = H_0^{(0)} Z_0; \tag{32}$$

$$G_1^{(1)} = H_1^{(1)} Z_1; \tag{33}$$

Шаг 1 (вычисление блочно-степенного базиса пространства Крылова и построение ряда Паде линейной системы):

(a) вычислить расширенную матрицу K_0 вида $K_0 = \left[\begin{array}{cccc} Q_0 & AQ_0 & \cdots & A^{\frac{2n}{K}} Q_0 \end{array} \right]$;

(b) вычислить квадратные $K \times K$ матрицы α_i по формулам $\alpha_i = Q_0^T A^i Q_0$, для всех i от 2 до $2\frac{n}{K}$;

(c) присвоить значения элементам блоков $R^{(0)}$ и $R^{(1)}$;

Шаг 2 (построение A-ортогонального базиса): для всех s от 2 до $\frac{n}{K}-1$ повторять (a), (b), (c), (d) и (e)

(a) найти $K \times K$ матрицы ν_0, ν_1, решая систему уравнений,

$$\left[\begin{array}{cc} 0 & \rho_s^{(s-1)} \\ \rho_{s+1}^{(s)} & \rho_{s+1}^{(s-1)} \end{array} \right] \left[\begin{array}{c} \nu_0 \\ \nu_1 \end{array} \right] = - \left[\begin{array}{c} \rho_{s+1}^{(s)} \\ \rho_{s+2}^{(s)} \end{array} \right]. \tag{34}$$

(b) вычислить $H_0^{(s)}$ и $H_1^{(s)}$ по рекуррентной формуле

$$H_i^{(s)} = Z H_i^{(s-1)} + H_i^{(s-1)} \nu_0 + H_i^{(s-2)} \nu_1, \quad (i = 0, 1); \tag{35}$$

(c) вычислить остаток ряда Паде $R^{(s)}$ для приближения Паде степени s по формуле

$$R^{(s)} = Z^T R^{(s)} + R^{(s)} \nu_0 + R^{(s-1)} \nu_1; \tag{36}$$

(d) вычислить квадратную $K \times K$ матрицу Z_s, определяемую как $Z_s = (Q_s^T A Q_s)^{-1} Q_s^T B$, используя следующий алгоритм:

(a) вычислить квадратную $K \times K$ матрицу $\psi_s = Q_s^T B$, используя рекуррентное соотношение,

$$\psi_s = \nu_0^T \psi_{s-1} + \nu_1^T \psi_{s-2} \tag{37}$$

(b) положить матрицу $Q_s^T A Q_s$ равной $\rho_{s+1}^{(s)}$

(e) вычислить $G_0^{(s)}$ и $G_1^{(s)}$ по рекуррентным формулам:

$$G_0^{(s)} = G_0^{(s-1)} + H_0^{(s)} Z_s, \quad G_1^{(s)} = G_1^{(s-1)} + H_1^{(s)} Z_s. \tag{38}$$

Шаг 3 (построение решения): Вычислить решение $X_{\frac{n}{K}}$ по формуле:

$$X_{\frac{n}{K}} = K_0 G_0^{(\frac{n}{K}-1)} + A K_0 G_1^{(\frac{n}{K}-1)} - K_0 \hat{G}_1^{(\frac{n}{K}-1)}, \tag{39}$$

где $K \times K$ компоненты блока $\hat{G}_1$ есть ни что иное как $K \times K$ компоненты блока G_1, умноженные слева на матрицу $\alpha_1^{-1} \alpha_2$.

2.7. Алгоритм построения блочно-степенного базиса пространства Крылова и ряда Паде линейной системы

Поскольку **Шаг 0 (инициализация алгоритма)** имеет малую алгоритмическую сложность, то при построении параллельного алгоритма мы сразу переходим к анализу **Шага 1 (вычисление блочно-степенного базиса пространств Крылова и ряда Паде линейной системы)**. Будем везде далее считать, что число доступных в алгоритме супер-узлов равно K. На **Шаге 1** все K супер-узлов будут работать независимо и выполнять одинаковую работу.

С супер-узлом с номером i на **Шаге 1** свяжем следующие вычисления: рассмотрим блок $Q_0 \in \mathbb{F}^{n\times K}$; пусть $Q_{0i} \in \mathbb{F}^n$ – i-ый вектор в блоке Q_0. На **Шаге 1** супер-узел с номером i (а) строит последовательность векторов A^jQ_{0i} с параметром j, изменяющимся от 0 до $2\frac{n}{K}$, и (б) вычисляет i-ые столбцы $\alpha_j^i = Q_0^TA^jQ_{0i}$ квадратных матриц $\alpha_j = Q_0^TA^jQ_0$.

Очевидно, для того, чтобы иметь возможность выполнить указанные вычисления в отсутствие обмена данными с другими супер-узлами, перед началом вычислений на супер-узле с номером i необходимо разместить матрицу A, а также блок $Q_0 = B$.

Опишем распределенние данных по вычислительным узлам, относящимся к i-ому супер-узлу. Для простоты будем предполагать, что каждый супер-узел состоит из одинакового числа s вычислительных узлов. Избегая несущественных деталей, будем считать, что матрица A представляется в виде объединенния s блоков-столбцов A_1, A_2, $\cdots$, A_s с равным числом столбцов $\frac{n}{s}$ и равным числом ненулевых элементов в каждом блоке, а блок Q_0 имеет соответствующее блочное строчное разбиение

$$Q_0 = \begin{bmatrix} Q_0^1 \\ Q_0^2 \\ \cdots \\ Q_0^s \end{bmatrix}, \tag{40}$$

где для любого j блоки Q_0^j и имеют размеры $\frac{n}{s} \times K$, и что перед началом вычислений в оперативной памяти узла с номером j, записаны блок-столбец A_j матрицы A и часть Q_0^j блока Q_0.

Предложенная схема хранения естественным образом определяет алгоритмы построения степенного базиса пространства Крылова и алгоритма вычисления коэффициентов ряда Паде.

2.8. Алгоритм построения A ортогонального базиса пространства Крылова

Начнем с элементарного анализа сложности вычислений на **Шаге 2 (построение A-ортогонального базиса)**. Решение одной системы линейных уравнений вида (34) имеет сложность $\mathcal{O}(K^3)$, следовательно, учитывая, что решать систему необходимо $2n/K$ раз, полная решения $2K \times 2K$ линейных систем $\mathcal{O}(nK^2)$.

Рассмотрим вычисления коэффициетов $H_i^{(t)}$ полиномиального представления, задаваемого формулой (35). Сложность этого вычисления оценивается как $\mathcal{O}\left(n^2K\right)$. Аналогичная оценка справедлива и для рекуррентных соотношений (36) и (38). Наконец, элементарная проверка показывает, что все остальные вычисления на **Шаге 2 (построение A-ортогонального базиса)** имеют сложность не выше $\mathcal{O}(nK^2)$.

Учитывая, что $n \gg K$ и, следовательно, $n^2K \gg nK^2$, основное внимание будем уделять параллельной реализации рекуррентных соотношений (35), (36), (38). Ключевое наблюдение, которое позволяет реализовать эффективную параллельную процедуру пересчета ре-

куррентных соотношений, заключается в возможности хранить строки, блоков H, R и G, в виде, допускающем сбалалансированные вычисления на всех s узлах любого супер-узла.

Итак, будем считать, что на узле с номером 1 супер-узла с номером i располагаются векторы-строки $R_i^{(0)} = \alpha_{0i}$, $R_{si}^{(0)} = \alpha_{si}$, $R_{(2s)i}^{(0)} = \alpha_{(2s)i}$, $\cdots$, $R_{(2n/K)i}^{(0)} = \alpha_{(2n/K)i}$ (напомним, что s – это параметр, определяющий число узлов, относящихся к данному супер-узлу). Аналогично, на узле с номером 2 того же супер-узла с номером i будут сохраняться векторы $R_{1i}^{(0)} = \alpha_{1i}$, $R_{(s+1)i}^{(0)} = \alpha_{(s+1)i}$, $R_{(2s+1)i}^{(0)} = \alpha_{(2s+1)i}$ и т.д. Таким образом, на каждом из s узлов каждого супер-узла будет храниться примерно поровну, а именно по $\frac{2n}{Ks}$ векторов длины K, представляющих блок $R^{(0)}$. Аналогичным образом реализуется схема хранения блоков $R^{(s)}$, $H_0^{(s)}$, $H_1^{(s)}$, а также $G_0^{(s)}$ и $G_1^{(s)}$.

Рассмотрим преимущества, которые дает описанная нами схема хранения. Выпишем алгоритм, соответствующий вычислению $s+1$ остатка ряда Паде по формуле

$$R^{(s+1)} = Z^T R^{(s)} + R^{(s)}\nu_0 + R^{(s-1)}\nu_1, \tag{41}$$

считая что матрицы ν_0 и ν_1 являются известными.

Обозначим через $\tilde{R}_{il}^{(s)}$ блок размера $\frac{2n}{Ks}$, представляющий набор векторов $R^{(s)}$ на l-ом узле i-ого суперузла. Не сложно проверить, что (41) переходит в следующее соотношение:

$$\tilde{R}_{il}^{(s+1)} = \tilde{Z}^T \tilde{R}_{i(l+1)}^{(s)} + \tilde{R}_{il}^{(s)}\nu_0 + \tilde{R}_{il}^{(s-1)}\nu_1, \tag{42}$$

где $\tilde{Z}$ – жорданова клетка размера $\frac{2n}{Ks}$ с нулевым значением на диагонали. Первое слагаемое в правой части (42) имеет нижний индекс $l+1$, а значит эта часть блока $R^{(s)}$ располагается в памяти узла с номером $l+1$. В тоже время остальные слагаемые и предполагаемый результат находятся на узле с номером l. Таким образом, вычисления по формуле (42) подразумевают обмен данными между узлами супер-узла с номером i. Запишем алгоритм.

Алгоритм для (41):

1. на каждом из s узлов вычислить произведение всех $\frac{2n}{Ks}$ строк от блока $R^{(s)}$ на матрицу ν_0;
2. на каждом из s узлов вычислить произведение всех $\frac{2n}{Ks}$ строк от блока $R^{(s-1)}$ на матрицу ν_1;
3. сложить соответствующие строки;
4. переслать строки с узла с номером i на узел с номером $(i \bmod s) + 1)$ и сложить с предыдущим результатом.

Для того, чтобы закончить описание параллельной реализации **Шага 2 (построение A-ортогонального базиса)**, нам необходимо описать процесс глобальных пересылок между различными супер-узлами и связанные с ними вычислениями. Действительно, еще ничего не было сказано о том, как получать матрицы ν_0 и ν_1. Поскольку данные матрицы являются результатом решения систем (34), то необходимо учесть время, необходимое на сборку и решение таких систем. Начнем со сборки. Учитывая выбранную нами схему хранения данных, сборка (34) является результатом элементарного обмена данными между супер-узлами.

2.9. Организация параллельных вычислений на Шаге 3 (построение решения)

Подход к построению данного алгоритма описан в разделе 4.3 книги [12]. Идея подхода весьма проста. Рассмотрим сначала несколько упрощенную ситуацию. Считая заданными

блок $B \in \mathbb{F}^{n\times K}$, матрицу $A \in \mathbb{F}^{n\times n}$ и вектор G, будем искать вектор X в виде

$$X = K_0 G = \sum_{i=0}^{\frac{n}{K}-1} A^i B G_i, \tag{43}$$

где G_i – подвекторы вектора G размера K.

Пусть имеются K узлов, и на каждом из K узлов находится последовательность Крылова вида B_i, AB_i, A^2B_i, $\cdots$, $A^{n/K-1}B_i$ и весь вектор G. Наша ближайшая цель предложить алгоритм получения X, минимизирующий число обменов.

Запишем (43) более подробно, расскрывая неявное суммирование в соотношении ABG_i,

$$X = \sum_{i=0}^{\frac{n}{K}-1} A^i B G_i \tag{44}$$

$$= \sum_{i=0}^{\frac{n}{K}-1} \left(\sum_{j=1}^{K} \left(A^i B\right)_j G_{ij} \right) \tag{45}$$

$$= \sum_{j=1}^{K} \left(\sum_{i=0}^{\frac{n}{K}-1} \left(A^i B\right)_j G_{ij} \right) \tag{46}$$

$$= \sum_{j=1}^{K} \left(\sum_{i=0}^{\frac{n}{K}-1} A^i B_j G_{ij} \right). \tag{47}$$

Не трудно заметить, что в (47) выражение в скобках может вычисляться на всех узлах независимо. Таким образом, если бы требовалось получить X на узле с номером 1, то количество необходимых обменов для получения X не превосходило бы $\mathcal{O}(n)$.

Применим данный подход к построению оптимального по сложности алгоритма получения решения линейной системы в виде линейной комбинации блоков блочно-степенного базиса пространства Крылова. Начнем с обсуждения того, чем реальный алгоритм отличается от только что рассмотренной упрощенной ситуации. Во-первых, в реальной ситуации имеется K супер-узлов, каждый из которых состоит из s узлов. Во-вторых, векторы вида A^jB_i хранятся на отдельных узлах супер-узлов в виде подвекторов размера $\frac{n}{s}$, что определяется выбранной нами схемой хранения (см. пункт 2.7). Как будет видно из дальнейшего, эти отличия не повлияют существенно на построение параллельной процедуры.

Отметим еще, что в рассматриваемом нами алгоритме G это не вектор, а $n \times K$ блок векторов. Схема хранения блока G была описана нами ранее. Не трудно проверить, что после всех вычислений на **Шаге 2 (построение A-ортогонального базиса)** строки блоков $G_0^{\frac{n}{K}-1}$, $G_1^{\frac{n}{K}-1}$ и соответствующие столбцам A^jB_i крыловских систем, располагаются в памяти узлов одного и того же супер-узла с номером i (см. пункт 2.8). Этот факт позволяет нам рассмотреть алгоритм вычисления решения X, который в минимальной степени использует обмен между отдельными супер-узлами, а, следовательно, не зависит от типа параллельной архитектуры.

Алгоритм вычисления X:

1. собрать на каждом узле каждого супер-узла $\frac{n}{K} \times K$ подблоки $\tilde{G}_0$ и $\tilde{G}_1$ блоков $G_0^{(\frac{n}{K}-1)}$ и $G_1^{(\frac{n}{K}-1)}$ (напомним, что до начала работы алгоритма на узлах располагаются лишь $\frac{n}{Ks} \times K$ подблоки блоков $\tilde{G}_0$ и $\tilde{G}_1$);

2. провести независимо на каждом узле вычисления вида

$$\tilde{X}_{ti} = \left(\sum_{j=0}^{\frac{n}{K}-1} A^j B_i^t \tilde{G}_j \right), \tag{48}$$

здесь t обозначает номер узла в супер-узле с номером i, а $\tilde{X}_{ti}$ – блок размера $\frac{n}{s} \times K$, расположенный в памяти t-ого узла супер-узла с номером i;

3. переслать блоки $\tilde{X}_{ti}$ на t-ый узел супер-узла с номером 1 (выделенного супер-узла);
4. на узле с номером t супер-узла 1 вычислить

$$\tilde{X}_t = \sum_{i=1}^{K} \tilde{X}_{ti}. \tag{49}$$

5. собрать решение на узле 1 супер-узла 1.

Литература

1. Lanczos C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators // J. Res. Nat. Bur. Standards. 1950. Vol. 45, P. 255–282.
2. Coppersmith D. Solving linear equations over $GF(2)$: Block Lanczos algorithm. // Linear Algebra Appl. 1993. Vol. 193. P. 33–60.
3. Gutknecht M.H. A completed theory of the unsymmetric Lanczos process and related algorithms. Part I. // SIAM J. Matrix Anal. Appl. 1992. Vol. 13. N. 2. P. 594–639.
4. Gutknecht M.H. A completed theory of the unsymmetric Lanczos process and related algorithms. Part II. // SIAM J. Matrix Anal. Appl. 1992. Vol. 15. P. 15–58.
5. Montgomery P. A block Lanczos algorithm for finding dependencies over GF(2) // Springer-Verlag, 1995. Vol. 921.
6. Peterson M., Monico C. $\mathbb{F}_2$ Lanczos revisited. // Linear Algebra Appl. 2008. Vol. 428. P. 1135–1150.
7. Дорофеев А.Я. Вычисление логарифмов в конечном простом поле методом линейного решета. // Труды по дискретной математике. 2002. Т. 5. P. 29–50.
8. Дорофеев А.Я. Решение систем линейных уравнений при вычислении логарифмов в конечном простом поле. // Математические вопросы криптографии. 2012. Vol. 3. N. 1. P. 5–51.
9. Черепнев М.А. Блочный алгоритм типа Ланцоша решений разреженных систем линейных уравнений. // Дискретная математика. 2008. Vol. 20. N. 1.
10. М.А. Черепнев. Version of block Lanczos-type algorithm for solving sparse linear systems. *Bull.Math.Soc.Sci.Math.Roumanie*, V.53(101), N.3, 2010, p.225-230, http://rms.unibuc.ro/bulletin.
11. И.А. Поповян Ю.В. Нестеренко, Е.А. Гречников. Вычислительно сложные задачи теории чисел. Учебное пособие. Издательство Московского Университета, 2012.
12. Н.Л. Замарашкин. Алгоритмы для разреженных систем линейных уравнений в GF(2). Учебное пособие. Издательство Московского Университета, 2013.

Universal block Lanczos-Pade method for linear systems over large finite fields

Nikolay Zamarashkin and Mihail Cherepnev

In this paper we propose a universal algorithm designed for solving large sparse linear systems over finite fields with large prime number of elements. Such systems arise in the solution of the discrete logarithm problem modulo a prime number. Parallel algorithms and effective data distributions are proposed.