Министерство образования и науки Российской Федерации
Московский физико-технический институт (государственный университет)

Физтех-школа прикладной математики и информатики
Кафедра вычислительных технологий и моделирования в геофизике и биоматематике

Выпускная квалификационная работа бакалавра

Блочные методы типа бисопряжённых градиентов

Автор:
Студент 101а группы
Козлов Николай Андреевич

Научный руководитель:
н.с.,к.ф.-м.н.
Желтков Дмитрий Александрович

Москва 2025

Аннотация

Блочный BiCGStab и его друзья
Козлов Николай Андреевич

Краткое описание задачи и основных результатов, мотивирующее прочитать весь текст.

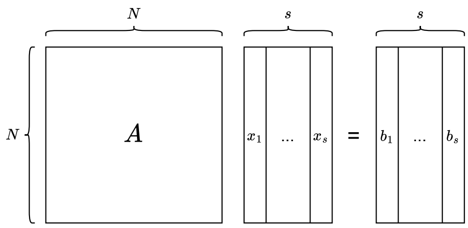
Abstract

Block BiCGStab and his friends

Содержание

1 Введение

В ряде приложений возникают большие линейные системы с многими правыми частями. такую задачу можно записать в блочном виде:



$$AX = B,$$

где A - $N \times N$ невырожденная разреженная матрица системы; B - $N \times s$ невырожденная матрица, столбцы - правые части; X - $N \times s$ матрица, столбцы - решения для соответствующих правых частей. Также еще предполагаем, что $s \ll N$. Такие задачи можно решать прямыми методами, однако они не подходят для больших задач из-за кубической асимптотики. Так что естественным является использование блочных крыловских методов.

В преимущества блочных крыловских методов входят: высокая производительность на вычислительных системах за счет блочных операций, Более быстрая сходимость, по сравнению с неблочными методами [DIANNE O'LEARY]; в задачах со структурированными системами (например МКЭ) БКМ не разрушают структуру, в отличие от прямых методов. Чрезвычайно большие системы, которые не помещаются целиком в оперативную память можно решать с помощью блочных крыловских методов.

<Рассказ про блочные крыловские методы.>

Для наших целей мы хотим построить крыловские методы, отвечающие следующим требованиям: методы должны находить решения систем общего вида, то есть, которые не обязательно являются эрмитовыми; методы не должны требовать сохранения всего крыловского пространства, то есть должны давать короткие итерационные соотношения; методы должны поддерживать оптимальный размер блока.

2 Крыловские методы решения систем уравнений

Здесь надо рассмотреть все существующие решения поставленной задачи, но не просто пересказать, в чем там дело, а оценить степень их соответствия тем ограничениям, которые были сформулированы в постановке задачи.

2.1 Метод сопряженных градиентов

[YOUSEF SAAD ITERATIVE METHODS FOR SPARSE LINEAR SYSTEMS SECOND EDITION]

2.2 Метод бисопряженных градиентов

[YOUSEF SAAD ITERATIVE METHODS FOR SPARSE LINEAR SYSTEMS SECOND EDITION]

2.3 Стабилизированный метод бисопряженных градиентов

[1]

2.4 Блочный метод сопряженных градиентов

[2]

2.5 Блочный метод бисопряженных градиентов

[2]

2.6 Блочный стабилизированный метод бисопряженных градиентов

[3]

2.6.1 Матричнозначные полиномы

Проблемы со сходимостью метода из [3], демонстрация в 4 главе, решение проблемы в 3 главе.

2.6.2 Алгоритм

3 Улучшения блочного метода стабилизированных бисопряженных градиентов

В данной главе предложены изменения, направленные на улучшение стабильности блочного метода бисопряженных градиентов.

3.1 Реортогонализация для поддержания биортогональных соотношений

Для построения базиса в крыловском пространстве и построения невязок алгоритм строится таким образом, чтобы поддерживать следующие соотношения ортогональности:

$$C(\mathbf{Q}_k\mathbf{R}_{k+1}) = 0, \tag{1}$$

$$C^{(1)}(\mathbf{Q}_k\mathbf{P}_k) = 0. \tag{2}$$

Для построения процедуры реортогонализации эти полиномиальные соотношения необходимо перевести в матричный вид. Используя полиномиальное соотношение для $\mathbf{R}_{k+1}$, получаем:

$$\mathbf{Q}_k\mathbf{R}_{k+1} = \mathbf{Q}_k\mathbf{R}_k - t\mathbf{Q}_k\mathbf{P}_k\alpha_k \implies S_k = R_k - AP_k\alpha_k$$

Тогда выражение (1) можно представить в виде:

$$\tilde{R}_0^* S_k = 0. \tag{3}$$

В точной арифметике это соотношение выполняется строго, однако при вычислениях на компьютере соотношение (3) выполняется с какой-то погрешностью. Для существенного уменьшения этой погрешности можно произвести ортогонализацию еще раз, взяв S_k в качестве блока, к которому производится ортогонализация:

$$S_k^r = S_k - AP_k\alpha_k^r. \tag{4}$$

При этом мы стремимся поддерживать соотношение $\tilde{R}_0^* S_k^r = 0$ с уточненным блоком S_k^r. Тогда, домножая обе части выражения (4) слева на $\tilde{R}_0$, получим уравнение для поправки α_k^r:

$$(\tilde{R}_0^T AP_k)\alpha_k^r = \tilde{R}_0^T S_k.$$

Поработаем теперь над выражением (2). Используя полиномиальное соотношение для P_{K+1}, получаем следующее выражение:

$$t\mathbf{Q}_k\mathbf{P}_{k+1} = t\mathbf{Q}_k\mathbf{R}_{k+1} + t\mathbf{Q}_k\mathbf{P}_k. \tag{5}$$

Введем обозначение $W_k \equiv (t\mathbf{Q}_k\mathbf{R}_{k+1})(A) \circ R_0$. Тогда выражение (5) можно записать в матричном виде:

$$W_k = AS_k + AP_k\beta_k. \tag{6}$$

Тогда выражение (2) можно представить в виде:

$$\tilde{R}_0^* W_k = 0. \tag{7}$$

Аналогично получаем соотношения реортогонализации для (7):

$$W_k^r = W_k + AP_k\beta_k^r$$

Поправка β_k определяется уравнением:

$$(\tilde{R}_0^* AP_k)\beta_k^r = -\tilde{R}_0^* W_k$$

Следующим шагом получим формулу для вычисления P_{k+1} с учетом введённых обозначений

$$\begin{aligned}\mathbf{Q}_{k+1}\mathbf{P}_{k+1} &= \mathbf{Q}_{k+1}(\mathbf{R}_{k+1} + \mathbf{P}_k\beta_k) = \\ &= \mathbf{Q}_k\mathbf{R}_{k+1} - \omega_k t\mathbf{Q}_k\mathbf{R}_{k+1} + (1-\omega_k t)\mathbf{Q}_k\mathbf{P}_k\beta_k = \\ &= \mathbf{Q}_k\mathbf{R}_{k+1} - \omega_k t\mathbf{Q}_k\mathbf{P}_{k+1} + \mathbf{Q}_k\mathbf{P}_k\beta_k\end{aligned}$$

В матричном виде это выражение записывается как:

$$P_{k+1} = S_{k+1} + P_k\beta_k - \omega_k W_k$$

Для дополнительной минимизации нормы невязки поддерживается следующее соотношение:

$$\langle AS_k, R_{k+1}\rangle_F = 0$$

Для этого выражения также можно выписать процедуру реортогонализации:

$$R^r_{k+1} = R_{k+1} - \omega^r_k T_k$$

$$\omega^r_k = \frac{\langle R_{k+1}, T_k\rangle_F}{\langle T_{k+1}, T_k\rangle_F}$$

3.2 Ортогонализация векторов направлений и проверочных невязок

Вспомним вид алгоритма [3], в блоке Алгоритм 1 красным отмечены все места где используется блок векторов направлений P_k. Легко видеть, что он везде входит в алгоритм вместе матрицей коэффициентов (α_k и β_k). Так что если сделать замену $P_k \leftarrow P_k U$, где U - $s\times s$ матрица, то изменятся лишь сами матрицы коэффициентов, в то время как сами выражения в алгоритме останутся неизменными. Так что можно попробовать

Algorithm 1 Блочные стабилизированные бисопряженные градиенты

X_0 - начальное приближение
$R_0 = B - AX_0$
$P_0 = R_0$
$\tilde{R}_0$ - произвольная $N \times s$ матрица
for $k = 0, 1, 2, \ldots$ **do**
  решить $\tilde{R}_0^T A P_k \alpha_k = \tilde{R}_0^T R_k$
  $S_k = R_k - AP_k\alpha_k$
  $T_k = AS_k$
  $\omega_k = \frac{\langle T_k, S_k\rangle_F}{\langle T_k, T_k\rangle_F}$
  $X_{k+1} = X_k + P_k\alpha_k + \omega_k S_k$
  $R_{k+1} = S_k - \omega_k T_k$
  решить $\tilde{R}_0^T A P_k \beta_k = -\tilde{R}_0^T T_k$
  $P_{k+1} = R_{k+1} + P_k\beta_k - \omega_k A P_k \beta_k$
end for

подобрать такую U, чтобы вычисления стали более устойчивыми. Например, можно сделать QR-разложение матрицы P_k:

$$P_k = Q_P R_P,$$

и в качестве U взять R_k^{-1}. Такой выбор U повлечет ортогонализацию P_k, что должно улучшить стабильность операций проектирования на вектора направлений.

Как указано в алгоритме 1, $\tilde{R}_0$ - произвольная матрица, обычно ее выбирают равной R_0. Аналогично для улучшения стабильности предлагается сделать QR-разложение матрицы R_0:

$$R_0 = Q_R R_R,$$

и сделать замену $\tilde{R}_0 \to Q_R$.

3.3 Выбор правых частей

Алгоритм перестает сходиться, если блок невязок становится почти вырожденным, поэтому предлагается на этапе инициализации алгоритма сделать RRQR-разложение блока правых частей, рассмотреть получившуюся перестановку, и выбрать несколько правых частей с номерами, соответствующим первым номерам в перестановке. Благодаря такому выбору рассматривается более линейно-независимый набор столбцов, что положительно сказывается на сходимости.

3.4 Алгоритм

Algorithm 2 Регуляризованный блочный метод стабилизированных бисопряженных градиентов

X_0 - начальное приближение;
$R_0 = B - AX_0$;
$P_0 = R_0$;
$R_0 = \tilde{Q}\tilde{U}$ - QR-разложение R_0;
$\tilde{R}_0 = \tilde{U}$;
for $k = 0, 1, ...$ **do**
  $P_k = QU$ - QR-разложение P_k;
  $P_k \to P_k U^{-1}$;
  $V_k = AP_k$;
  решить $(\tilde{R}_0^* V_k)\hat{\alpha}_k = \tilde{R}_0^* R_k$;
  $\hat{S}_k = R_k - V_k\hat{\alpha}_k$;
  решить $(\tilde{R}_0^* V_k)\alpha_k = \tilde{R}_0^* \hat{S}_k$;
  $S_k = \hat{S}_k - V_k\alpha_k$;
  $T_k = AS_k$;
  $\hat{\omega}_k =< S_k, T_k >_F / < T_k, T_k >_F$;
  $\hat{R}_{k+1} = S_k - \hat{\omega}_k T_k$;
  $\omega_k =< \hat{R}_{k+1}, T_k >_F / < T_k, T_k >_F$;
  $R_{k+1} = \hat{R}_{k+1} - \omega_k T_k$;
  $X_{k+1} = X_k + P_k(\hat{\alpha}_k + \alpha_k) + (\hat{\omega}_k + \omega_k)S_k$;
  решить $(\tilde{R}_0^* V_k)\hat{\beta}_k = -\tilde{R}_0^* T_k$;
  $\hat{W}_k = T_k + V_k\hat{\beta}_k$;
  решить $(\tilde{R}_0^* V_k)\beta_k = -\tilde{R}_0^* \hat{W}_k$;
  $W_k = \hat{W}_k + V_k\beta_k$;
  $P_{k+1} = S_k + P_k(\hat{\beta}_k + \beta_k) - (\hat{\omega}_k + \omega_k)W_k$;
end for

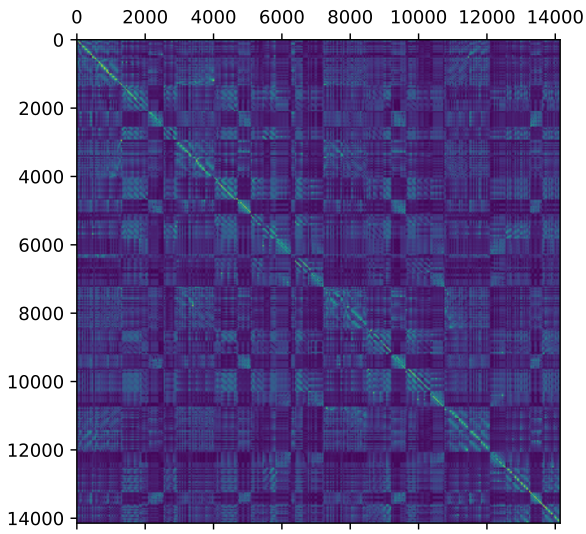
3.5 Проблемы

В данном алгоритме возможны брейкдауны, в случаях, когда матрица $\tilde{R}_0^* AP_k$ становится вырожденной. В такой ситуации авторы [3] предлагают провести рестарт с другой $\tilde{R}_0$.

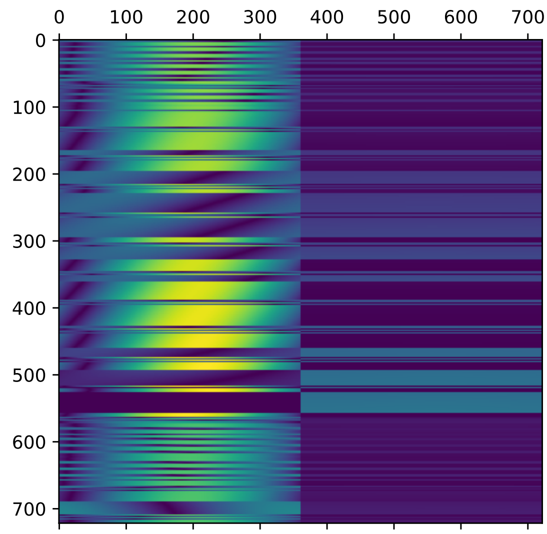
Но главным недостатком алгоритма [3] является выбор ω в виде скалярной матрицы, из-за этого чем больше размер блока мы берем для расчета, тем меньше по модулю становится ω, что в свою очередь ведет к стагнации алгоритма. Наша модификация алгоритма также страдает от этой проблемы. Была надежда, что получится обобщить метод на случай, когда ω_k - произвольная $s \times s$ матрица, но в ходе исследования выяснилось, что это невозможно из-за некоммутативности матричнозначных полиномов, которая не позволяет получить короткие итерационные формулы.

4 Численные эксперименты

Тесты производились на интересующей нас задаче – линейной системе с многими правыми частями, возникающей при решении задачи электромагнитного рассеяния методом интегральных уравнений [STAVTSEV]. Порядок системы - 14144, всего правых частей - 722. Каждая правая часть соответствует разным углам падения, а также первая половина правых частей отличается от правой типом поляризации.



(a)



(b)

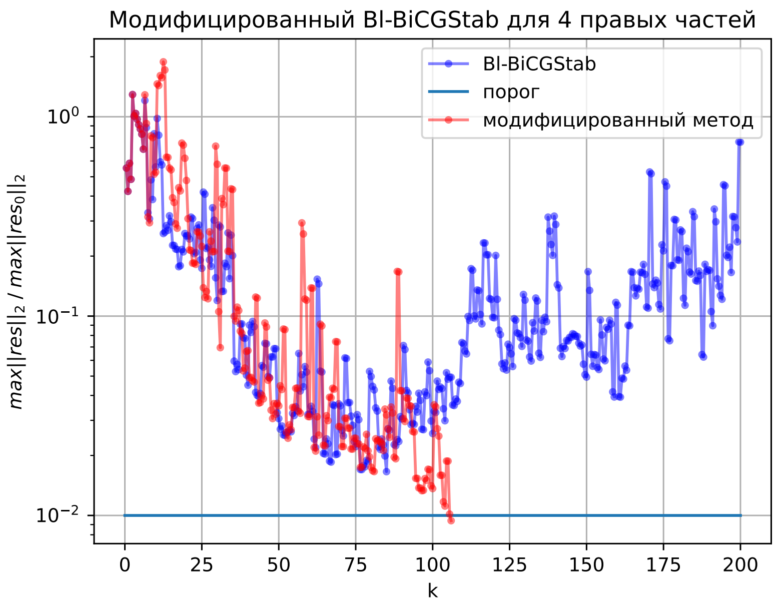


Рис. 2

Первый тест демонстрирует, что метод из статьи [GUENNOUNI] не сходится с требуемой точностью, в то время как версия с улучшениями, описанными в главе 3, сходится линейно без проблем. Эксперимент проводился в одинарной точности для четырех правых частей с номерами: 0, 90, 180, 270. Его результаты представлены на рис.2

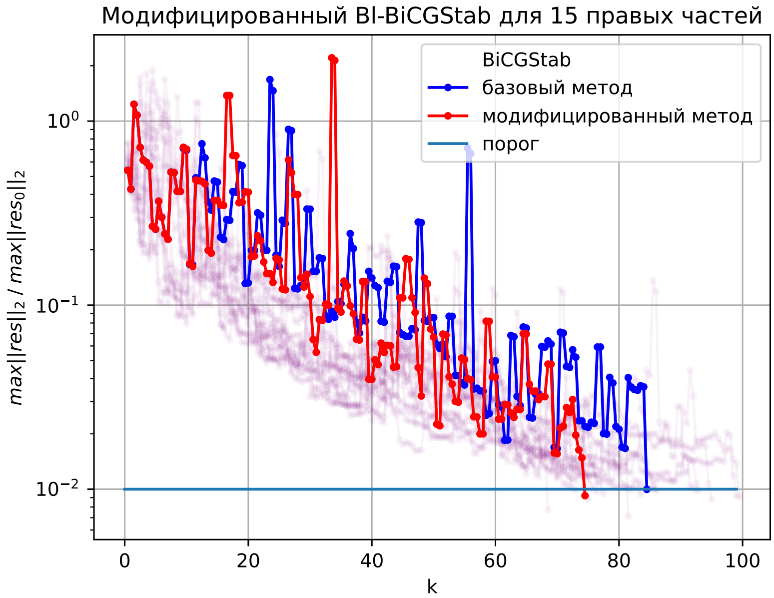


Рис. 3

15 правых частей, уменьшения числа итераций, считаем в двойной точности

Второй тест демонстрирует, что улучшения, описанные в главе 3 позволяют получить выгоду по количеству матвеков, по сравнению с решением систем с каждой правой частью в отдельности. Эксперимент проводился в двойной точности с 15 правыми частями, выбранными с помощью RRQR. Его результаты представлены на рис.3. По оси абцисс - количество иттераций, по оси ординат - относительная максимальная невязка в блоке. Фиолетовым изображено падение невязки при решении задачи с каждой правой частью в отдельности, синим - метод из статьи [GUENNOUNI], красным - метод с улучшениями из главы 3. Для решения этой задачи стабилизированными бисопряженными градиентами было потрачено 2525 матвеков, для решения методом из статьи [GUENNOUNI] - 2535, модифицированный метод сошелся за 2235 матвеков, таким образом выгода составляет 12% по сравнению с неблочной версией.

более 30 правых частей, демонстрация отсутствия взрыва невязки

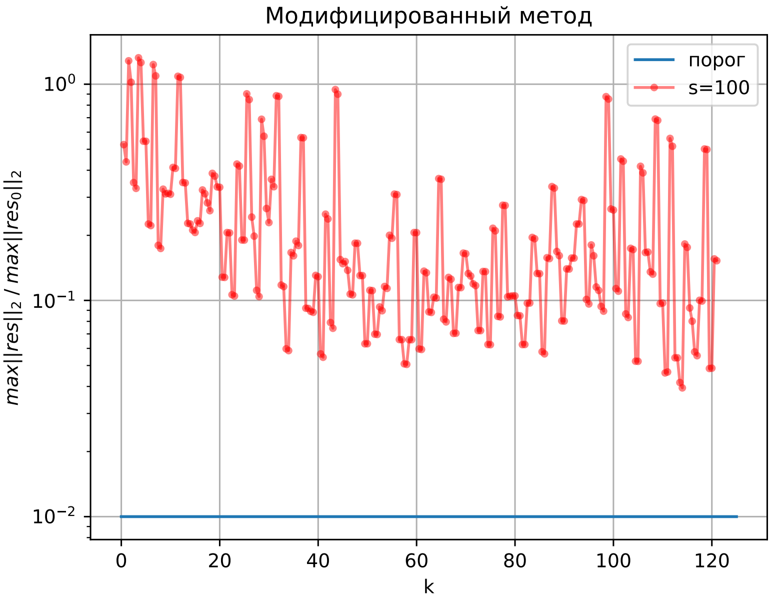

Модифицированный метод
порог
s=100
10^0
10^{-1}
10^{-2}
$max||res||_2$ / $max||res_0||_2$
0
20
40
60
80
100
120
k

Рис. 4

5 Заключение

Результаты

Нерешенные проблемы редукции блока

Список литературы

[1] *van der Vorst, H. A.* Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems / H. A. van der Vorst // *SIAM Journal on Scientific and Statistical Computing.* — 1992. — Vol. 13, no. 2. — Pp. 631–644. https://doi.org/10.1137/0913035.

[2] *O'Leary, Dianne P.* The block conjugate gradient algorithm and related methods / Dianne P. O'Leary // *Linear Algebra and its Applications.* — 1980. — Vol. 29. — Pp. 293–322. — Special Volume Dedicated to Alson S. Householder. https://www.sciencedirect.com/science/article/pii/0024379580902475.

[3] *el Guennouni A., Jbilou K. Sadok H.* A block version of BiCGSTAB for linear systems with multiple right-hand sides. / Jbilou K. Sadok H. el Guennouni, A. // *ETNA. Electronic Transactions on Numerical Analysis [electronic only].* — 2003. — Vol. 16. — Pp. 129–142. http://eudml.org/doc/124803.