ОБ ЭФФЕКТИВНОМ ЧИСЛЕННОМ РЕШЕНИИ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ

А. Б. Нерсесян

**Известия Академии Наук Армении. Математика,
том 27, №2, 1992**

Исследованы экономичные алгоритмы для различного типа интегральных уравнений. В работе предложен общий метод для построения параллельных алгоритмов. Основные результаты относятся к построению новых быстрых, экономящих память алгоритмов для широкого класса ядер. Приведены результаты некоторых численных экспериментов, иллюстрирующих эффективность предложенных методов.

СОДЕРЖАНИЕ

Введение

ГЛАВА 1. *Уравнения общего вида*

§1.1. Метод Нистрема
§1.2. Гладкие ядра на замкнутых многообразиях
§1.3. Принцип продолжения резольвентного ядра
§1.4. Метод дискретного погружения. Одношаговые алгоритмы
§1.5. Метод непрерывного погружения. Многошаговые алгоритмы
§1.6. Автоматический выбор шага и порядка алгоритма
§1.7. Метод конечномерного смещения ядра
§1.8. Об уравнениях первого рода и нелинейных уравнениях
§1.9. Устойчивость алгоритма

ГЛАВА 2. *Специфичные ядра и быстрые алгоритмы*

§2.1. Итерационное уточнение
§2.2. Операторы с ядрами, определяемыми из дифференциальных уравнений
§2.3. Операторы циркулянтного типа
§2.4. Уравнения теплицева и ганкелева типов
§2.5. Композиция ядер и уравнения первого рода

ГЛАВА 3. *Анализ численных результатов*

§3.1. Тестовые уравнения и алгоритмы
§3.2. Метод автоматического сгущения сетки
§3.3. Методы погружения для решения уравнений общего вида
§3.4. Быстрые алгоритмы
§3.5. Заключение

ПРИЛОЖЕНИЕ. *Таблицы расчетов*

Литература

ВВЕДЕНИЕ

Как в теоретических, так и в прикладных дисциплинах, - в математической физике, геофизике, в задачах управления, обработки изображений, автоматического регулирования и во многих других областях,- встречаются интегральные уравнения того или иного типа. Основные положения теории интегральных уравнений разработаны в начале и середине века, однако задача их приближенного решения (особенно при повышенных требованиях к точности и скорости счета) адекватного развития не получила. Во всяком случае, такое впечатление складывается при сравнении с богатым, постоянно модифицируемым арсеналом численного решения дифференциальных уравнений всевозможных типов.

Между тем область приложения интегральных уравнений неуклонно расширяется и разработка новых методов их эффективного решения,- от абстрактных схем до стандартных пакетов программ,- является, пожалуй, одной из важнейших задач, находящихся на стыке теории операторов и вычислительной математики.

На нынешнем этапе развития прикладных наук особенно актуальна разработка алгоритмов решения, ориентированных на современные и перспективные ЭВМ : персональные компьютеры класса IBM PC/AT и системы их транспьютерной поддержки.

К настоящему времени наибольшее развитие получили методы решения линейного интегрального уравнения Фредгольма второго рода

$$(\mathcal{J} - K)y \equiv y(x) - \int_D K(x,t)y(t)dt = f(x), \quad x \in D \tag{0.1}$$

с компактным оператором K. Здесь нас интересует случай, когда $D \subset R^m$ ограниченная область, $dt = dt_1 \cdots dt_m$, K - заданная гладкая комплекснозначная $(l \times l)$-матрица, $(x,t) \in D \times D$, $f(x)$ - заданный, а $y(x)$ - искомый l-векторы (т.е. оператор K может считаться действующим в $C(D)$).

Если область D, ядро K и свободный член f заданы с высокой точностью, а оператор $\mathcal{J} - K$ обратим, то метод квадратур (кубатур) позволяет, в принципе, находить решение y с точностью того же порядка. Однако даже в простейшем одномерном $(D = (a,b) \subset R^1)$ случае, при практически точном задании K и f

(скажем, посредством элементарных функций) далеко не всегда удается согласовать требуемую точность и скорость счета с наличными вычислительными ресурсами (см. ниже, §1.1). Это противоречие резко усиливается при росте размерности области D и усложнении ее конфигурации.

Подобные трудности преодолеваются, в основном, двумя путями : посредством построения экономичных алгоритмов решения данного уравнения конкретно для наличных систем ЭВМ и с помощью новых вычислительных структур, приспособленных к свойствам определенного класса уравнений. В отношении оператора (0.1) наилучшие результаты связаны со специфическими свойствами ядра $K(x,t)$.

Кроме общеизвестных случаев конечномерности оператора $\mathcal{K}$ (т. е. вырожденности ядра $K(x,t)$) или его вольтерровости (см. [1,2]), быстрые алгоритмы, с одновременным сокращением используемой оперативной памяти, были построены в случае теплицевых ядер ($K(x,t) = K(x-t)$) или несколько более общих ядер теплицева типа, для которых

$$(\frac{\partial}{\partial x} + \frac{\partial}{\partial t})K(x,t) = p(x)q(t), \quad x,t \in D = (a,b) \subset R^1, \tag{0.2}$$

где p и q - матрицы размеров $(l \times l_1)$ и $(l_1 \times l)$, соответственно (см. [3,4]).

В работе [5] изучены операторы $\mathcal{K}$ с ядрами, удовлетворяющими более общим дифференциальным уравнениям и на этой основе удалось обнаружить новые классы быстро решаемых уравнений. В частности (см. [7]), это относится к случаю ядер K, для которых

$$(\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial t^2})K + a(x)K - K\,a(t) = p(x)q(t), \quad x,t \in (a,b), \tag{0.3}$$

При $a(x) \equiv 0$ это ядро типа "Теплиц плюс Ганкель" (см. также [8]), а при $a = (\frac{1}{4} - \nu^2)x^{-2}$, $p \equiv 0$ сюда входит известное в теории упругости (см. [9]) ядро типа Вебера-Сонина

$$K(x,t) = \sqrt{xt}\int_0^\infty J_\nu(xs)J_\nu(ts)h(s)ds, \tag{0.4}$$

где J_ν - функция Бесселя, $0 < x,t < +\infty$.

Из-за компактности оператора $\mathcal{K}$ задача решения уравнения Фредгольма первого рода ($D \subset R^m$)

$$\mathcal{K}y = \int_D K(x,t)y(t)dt = f(x), \quad x \in D_1 \subset R^m \tag{0.5}$$

в классических пространствах поставлена некорректно. В то же время к этому уравнению приводят многие прикладные исследования [9] и, к счастью, задача его приближенного решения небезнадежна. Здесь эффективны робастные (регуляризационные) методы (см.[10]), один из которых заключается в приближенном переходе к уравнению второго порядка с симметричным положительно определенным ядром

$$(\alpha\mathcal{J} + Q)y \equiv \alpha y(x) + \int_D Q(x,t)y(t)dt = F(x), \quad \alpha = \text{const} > 0. \tag{0.6}$$

Здесь $Q = \mathcal{K}^*\mathcal{K}$, $F = \mathcal{K}^* f$ ($\mathcal{K}^*$ - сопряженный к $\mathcal{K}$ оператор), а малый параметр α подлежит дополнительному определению.

Ряд задач математической физики, теории упругости и т.д. [9] сводится к решению уравнений (0.1) или (0.5) с сингулярным (неограниченным) ядром. Типичным примером является ядро вида $K(x,t) = |x-t|^{-\alpha}K_1(x,t)$, где K_1 - гладкая функция и $0 < \alpha \leq m (D \subset R^m)$. В случае $0 < \alpha < m$ (интегрируемая сингулярность) эффективна корректировка вышеупомянутых подходов для гладких ядер. Однако при $\alpha = m$ (случай сингулярного интегрального уравнения, оператор $\mathcal{K}$ некомпактен) ситуация кардинально меняется и методы численного решения подобных уравнений разработаны сравнительно мало (см. [11]).

Наименьшие продвижения наблюдаются в случае нелинейных интегральных уравнений. Так, для решения уравнения Урысона

$$y(x) = \int_D F(x,t,y(t))dt + f(x), \quad x \in D \tag{0.7}$$

или даже более простого уравнения Гаммерштейна $F(x,t,y(t)) = K(x,t)F(t,y(t))$ наиболее эффективным остается классический метод Ньютона-Канторовича.

Предлагаемая работа содержит теоретические результаты, приводящие в ряде случаев к новым эффективным и экономичным алгоритмам решения интегральных уравнений. А именно, при заданной гладкости ядра K строятся устойчивые алгоритмы, работающие быстрее известных,- при одинаковой с ними точности,- и использующие, как правило, меньшую память. Все основные алгоритмы изначально распараллелены (векторизованы, см. [12, 13]) и, в принципе, могут быть вычислены на многопроцессорных вычислительных системах

с линейным ускорением. Это означает, что если игнорировать время, затрачиваемое на пересылку данных и синхронизацию счета, то p-процессорная система исполняет данный алгоритм в $k_e p$ раз быстрее однопроцессорной ($0 < k_e \leq 1$, k_e - коэффициент эффективности, $p = p(N) \to \infty, N \to \infty$, где N - число точек дискредизации области D).

Вопрос о технической реализации предлагаемых параллельных алгоритмов здесь не рассматривается. Заметим только, что эти алгоритмы достаточно просты и хорошо вписываются как в схему конвейерных вычислителей, так и в систолические структуры (см. [12]).

Работа имеет обзорный характер лишь в той степени, в которой необходимо провести понятный читателю сравнительный анализ с известными ранее результатами. Часть приводимых результатов ранее депонирована или опубликована в кратких сообщениях [7, 14 – 19, 42].

Содержащиеся в работе новые методы решения интегральных уравнений разных типов могут найти различные применения. Так, уравнения первого рода (0.5), традиционно считающиеся труднорешаемыми, одновременно крайне важны с прикладной точки зрения. В §2.5.2 разработан подход, значительно расширяющий класс известных быстрых алгоритмов для них.

Основная часть предлагаемых алгоритмов и их модификация прошла всестороннюю проверку на тестовых уравнениях (см. анализ в главе III). В Приложении собраны обработки наиболее типичных численных результатов, полученных на ЭВМ ЕС-1046 и персональных компьютерах IBM PC/AT. Эта часть работы, крайне важная для оценки реальных возможностей разработанных методов, вряд ли была бы выполнена столь тщательно и профессионально без помощи сотрудников факультета вычислительной математики и информатики Ереванского государственного университета А. Гарибян, А. Гаспаряна, Ш. Багдасарян, С. Цируняна и К. Дарбиняна. Пользуюсь случаем выразить всем им мою признательность.

ГЛАВА 1. УРАВНЕНИЯ ОБЩЕГО ТИПА

§1.1. МЕТОД НИСТРЕМА

1.1.1. Пусть область D ограничена в R^m, а ядро $K(x,t)$ уравнения (0.1) и свободный член f - гладкие комплекснозначные функции ($K \in C^p(\bar{D} \times \bar{D})$, $f \in C^p(\bar{D})$, $p \geq 1$). Выбрав для интеграла в (0.1) подходящую квадратурную (кубатурную) формулу порядка $O(h^p)$, $h \to 0$ на соответствующей сети точек

$$\{x_p\}_{p=1}^N; \quad x_p \in D_k,\ k = 1, 2, \ldots, N;$$

$$D = \cup D_k,\ D_i \cap D_j = \emptyset,\ i \neq j; \quad h = \max_k \operatorname{diam} D_k \to 0,$$

тем самым приближенно сведем задачу к решению алгебраической системы с $(lN \times lN)$-матрицей относительно $(lN \times l)$-матрицы $[y(x_1), \ldots, y(x_n)]^{tr}$.

Если оператор $\mathcal{J} - \mathcal{K}$ обратим в $C(\bar{D})$, то при достаточно малом h полученная система однозначно разрешима. При этом для ее решения необходимо выполнить $O((Nl)^3)$, $Nl \to \infty$ мультипликативных операций. Точность решения также будет иметь порядок $O(h^p)$, $h \to 0$.

При решении полученной системы одним из наиболее надежных способов - методом Гаусса с выбором ведущего элемента, понадобится выполнить $\frac{1}{3}N^3l^3 + O(N^2l^2)$ операций с использованием $(N+1)l \times lN$ ячеек памяти. В скалярном случае, $(l = 1)$, при наличии $(N+1)N$ процессоров (см., например, [13]) применение параллельного алгоритма Гаусса-Жордана позволяет сократить время счета в $\frac{2}{3}N^2$ раз, т.е. в данном случае (см. Введение) $k_e = \frac{2}{3}$.

Замечание 1.1. Гладкость правой части f в уравнении (0.1) существенного значения не имеет. Например, при $f \in C^q$, $0 \leq q < p$ заменой неизвестной функции $y_1(x) = y(x) - f(x)$ можно свести задачу к случаю правой части $f_1 = \mathcal{K}f$, вычисляемой с нужной точностью, применением квадратурных (кубатурных) формул порядка $O(h_1^q)$, $h_1 << h$. Если $K \in C^p$, то и $f_1 \in C^p$.

1.1.2. Метод Нистрема универсален и тем эффективней, чем выше гладкость ядра K, точнее его задание, и чем меньше размерность области D. Однако при повышенных требованиях к точности решения потребляемые им вычислительные ресурсы могут быть недопустимо велики. Форма области D при $m > 1$

также существенно влияет на точность, ибо может затруднить рациональный выбор сети дискретизации.

Оригинальный подход к задаче рациональной организации вычислений продемонстрирован в работе [18]. При решении одномерного уравнения

$$y(x) = \int_0^1 K(x,t)y(t)dt + f(x), \qquad x \in [0,1], \tag{1.1}$$

где $K, f \in C^2$, авторы предлагают учитывать интервалы наиболее резкого колебания решения следующим образом. За основу принимается метод Нистрема с применением формулы трапеций и последующим решением алгебраической системы итерационным методом Гаусса-Зейделя. На первом шаге выбирается относительно редкая равномерная сеть $x_k = k/r$, $k = 0, 1, ..., r$ и находятся значения $\{y(x_k)\}$. Далее, задавшись числами $0 < \delta_1, \delta_2 < 1$, надо проверить выполнение условий

$$\int_{x_j}^{x_{j+1}} |y^{(i)}(x)|dx \le \delta_i \max_k |y^{(i-1)}(x_k)|, \qquad i = 1,2; \quad j = 0, 1, ..., r-1 \tag{1.2}$$

с использованием квадратурных формул для величин

$$y^{(i)}(x) = \int_0^1 \frac{\partial^i}{\partial x^i} K(x,t)y(t)dt + \frac{\partial^i}{\partial x^i} f(x), \tag{1.2'}$$

На втором этапе интервал $[x_j, x_{j+1}]$ делится пополам, если хотя бы одно из условий (1.2) на нем не выполнятся, а на таким образом сгущенной сети снова находятся значения $y(x_k)$. Далее этот процесс повторяется, обычно 3 – 4 раза.

Однако численный эксперимент показал недостаточную эффективность такой схемы и она была модофицирована в работе [19]. Именно, теперь уже, в зависимости от степени нарушения условий (1.2), допускалось деление интервала $[x_j, x_{j+1}]$ на 2, 4 или 8 частей. Кроме того, на заключительном этапе применялась сплайн-интерполяция третьего порядка вполне конкурентоспособным методом Нистрема-Гаусса (см. ниже, §3.2). Имеется возможность усовершенствовать метод, применением метода конечномерных возмущений (метод Шермана-Моррисона, см. [13]).

§1.2. ГЛАДКИЕ ЯДРА НА ЗАМКНУТЫХ МНОГООБРАЗИЯХ

1.2.1. Особого внимания заслуживает случай, когда область D в уравнении (0.1) может считаться m-мерным замкнутым многообразием без края и $K, f \in C^p(D)$ на нем. В этом случае метод Нистрема резко повышает свою эффективность по следующим соображениям.

Как хорошо известно (см., например, [20]) квадратурные (кубатурные) формулы простейшего типа на равномерной сетке в случае многообразия без края дают максимально возможный порядок приближения.

Остановимся на одномерном случае, когда $\bar{D} = [0,1]$ есть окружность $z = e^{2\pi i x}$, $f \in C^p(\bar{D})$, а $K \in C^p(\bar{D} \times \bar{D})$ на соответствующем торе. Если в этом случае применить к уравнению (0.1) формулу прямоугольников, то в случае равномерной сетки $x_k = k/N$, $k = 0, 1, ..., N-1$ порядок точности будет не меньше $O(h^p)$, $h = \frac{1}{N} \to 0$. Особенно эффективен случай бесконечно-дифференцируемого ядра $K \in C^\infty(\bar{D} \times \bar{D})$, часто встречаемый в прикладных задачах.

Пример 1.1. Хорошо известно, что задача Дирихле для уравнения Лапласа в области, ограниченной эллипсом $\frac{x^2}{a^2} + \frac{y^2}{b^2} \le 1$ $(0 < b < a)$, сводится к решению следующего уравнения на окружности $z = e^{ix}$ $(0 \le x < 2\pi)$:

$$\rho(x) = -\frac{ab}{\pi} \int_0^{2\pi} \frac{\rho(t)dt}{a^2 + b^2 - (a^2 - b^2)\cos(x+t)} + f(x), \qquad (1.3)$$

где ρ - плотность распределения зарядов функции источников (функции Грина). Нетрудно выписать аналогичное уравнение и для задачи Дирихле $\Delta u = 0$, $u|_{\partial D} = f$ в случае, когда $D \subset R^m$ - эллипсоид. И в этом случае $K \in C^\infty$, потому метод Нистрема имеет высокий порядок точности (выше любого степенного). Значения гармонической функции u могут быть восстановлены в любой точке области D также посредством квадратурных формул.

Замечание 1.2. Гладкость и периодичность функции f и здесь (см. также Замечание 1.1) существенной роли не играют, так как переход к новой функции $f_1 = Kf$ очевиден.

§1.3. ПРИНЦИП ПРОДОЛЖЕНИЯ РЕЗОЛЬВЕНТНОГО ЯДРА

1.3.1. Рассмотрим следующую ситуацию. Пусть D - ограниченная область в R^m и $C(\bar{D},\Omega) = C(\bar{D})$ - Банахова алгебра непрерывных отображений D в нормированную алгебру Ω с единицей e. Определим оператор $\mathcal{J} - \mathcal{K}$ по (0.1) ($K \in C(\bar{D}\times\bar{D})$). Если этот оператор обратим в $C(\bar{D})$, то его резольвентное ядро $R(x,t,D)$ удовлетворяет хорошо-известным соотношениям

$$R(x,t,D) = K(x,t) + \int_D K(x,s)R(s,t,D)ds \qquad (1.4)$$

$$R(x,t,D) = K(x,t) + \int_D R(x,s,D)K(s,t)ds, \qquad (1.5)$$

где $(x,t) \in \bar{D}\times\bar{D}$.

1.3.2. Поставим следующий вопрос : если ядро K определено и непрерывно в более широкой области $S \supset \bar{D}\times\bar{D}$, можно ли туда же продолжить R с сохранением соотношений (1.4) и (1.5)? Ответ содержится в следующем результате.

Теорема 1.1. *Пусть ядро $K(x,t)$ определено и непрерывно на множестве $S \supset \supset \bar{D}\times\bar{D}$ и оператор $\mathcal{J}-\mathcal{K}_D$ обратим. Тогда резольвентное ядро $R(x,t,D)$ однозначно и непрерывно, с сохранением соотношений (1.4) и (1.5), продолжается на множество $\tilde{S} = (G_1\times G_2)\cap S$, где*

$$x \in G_1 \Leftrightarrow (x,s) \in S, \quad \forall s \in \bar{D}; \qquad x \in G_2 \Leftrightarrow (s,x) \in S, \quad \forall s \in \bar{D}.$$

Доказательство. Из (1.4) следует, что $R(x,t,D)$ однозначно продолжается на множество $(G_1 \times D)\cap S$. Перейдем теперь к соотношению (1.5). Можем предполагать, что $(x,t) \in (G_1\times G_2)\cap S$. Очевидно, что полученное продолжение непрерывно. Остается доказать единственность.

Пусть R_1 и R_2 - два продолжения, тогда функция $R_{12} = R_1 - R_2$ удовлетворяет одномерному уравнению (1.4) и, поскольку оператор $\mathcal{J} - \mathcal{K}$ обратим, то $R_{12}(x,t) \equiv 0$; $x,t \in \bar{D}$. Но тогда согласно формулам (1.4) и (1.5) имеем $R_{12}(x,t) = 0$ при $(x,t) \in \tilde{S}$.

Замечание 1.3. Нетрудно убедиться, что при гладком ядре $K(x,t)$ (например, при $K \in C^p(S)\cap C^{p-1}(\bar{S})$) резольвентое ядро $R(x,t,D)$ наследует эту гладкость в $\tilde{S}$.

1.3.3. Приведем теперь два характерных примера, используемых ниже (см. §§2.4, 3.5).

Пример 1.2. Ядра Теплица и Ганкеля

Пусть $K(x)$ определено при $|x| \leq 1$, тогда ядро $K_-(x-t) = K_-$ определено в полосе $S = \{|x-t| \leq 1\}$, а резольвентное ядро оператора $\mathcal{J} - \mathcal{K}$ с $D = [0, \tau]$, $0 < \leq \tau \leq 1$, по Теореме 1.1, определено в области $\tilde{S} = \{|x-t| \leq 1\} \cap \{\tau - 1 \leq x, t \leq 1\}$.

Соответственно, если функция $K(x)$ определена при $0 \leq x \leq 2$, то ядро $K_+ = K(x+t)$ определено в полосе $S = \{0 \leq x+t \leq 2\}$, а резольвентное ядро $R_+(x,t,\tau)$ в области

$$\tilde{S} = \{0 \leq x+t \leq 2\} \cap \{0 \leq x, t \leq 2-\tau\}.$$

Пример 1.3. Ядро "Теплиц плюс Ганкель".

Рассмотрим теперь ядро $K(x,t) = K_+(x+t) + K_-(x-t)$, заданное в квадрате $S = \{|x-t| \leq 1\} \cap \{0 \leq x+t \leq 2\}$. Нетрудно убедиться, что при $D = (0,\tau)$, $0 < \tau < 1$ резольвентное ядро $R(x,t,\tau)$ не продолжается из квадрата, т. е. $\tilde{S} = \tilde{D} \times \tilde{D}$. Однако, при $D = [\frac{1}{2} - \tau, \frac{1}{2} + \tau]$, $0 < \tau < \frac{1}{2}$ продолжение возможно и $\tilde{S} = S \cap \{\tau - \frac{1}{2} \leq x, t \leq \frac{3}{2} - \tau\}$.

§1.4. МЕТОД ДИСКРЕТНОГО ПОГРУЖЕНИЯ. ОДНОШАГОВЫЕ АЛГОРИТМЫ

1.4.1. Пусть теперь $K(x,t) \in C(\tilde{D} \times \tilde{D})$, $f(x) \in C(\tilde{D})$.

Теорема 1.2. *Если операторы $\mathcal{J} - \mathcal{K}_{D_i}$, $D_i \subset D$; $i = 1,2$ обратимы, то их резольвентные ядра связаны соотношением*

$$\begin{aligned} R(x,t,D_2) + \int_{D_1 \setminus D_2} R(x,s,D_1) R(s,t,D_2) ds = \\ = R(x,t,D_1) + \int_{D_2 \setminus D_1} R(x,s,D_1) R(s,t,D_2) ds; \quad x,t \in D, \end{aligned} \tag{1.6}$$

а решения уравнений $(\mathcal{J} - \mathcal{K}_{D_i}) y(x,D_i) = f(x)$ $(i = 1,2)$ - соотношением

$$\begin{aligned} y(x,D_2) + \int_{D_1 \setminus D_2} R(x,s,D_1) y(s,D_2) ds = \\ = y(x,D_1) + \int_{D_2 \setminus D_1} R(x,s,D_1) y(s,D_2) ds, \quad x \in D. \end{aligned} \tag{1.7}$$

Доказательство. Из формулы (1.4) и Теоремы 1.1 следует, что

$$K(x,t) = R(x,t,D_2) - \int_{D_1} K(x,s)R(s,t,D_2)ds -$$

$$- \int_{D_2 \setminus D_1} K(x,s)R(s,t,D_2)ds + \int_{D_1 \setminus D_2} K(x,s)R(s,t,D_2)ds.$$

Применив к обеим частям оператор $(\mathcal{J} - K_{D_1})^{-1}$, придем к формуле (1.6). Вполне аналогично доказывается и формула (1.7).

Замечание 1.4. Нетрудно убедиться, что формулы (1.6) и (1.7) сохраняют силу, если во всех подынтегральных выражениях D_1 и D_2 поменять местами.

1.4.2. Наметим общую схему метода "дискретного погружения" для решения уравнения (0.1). Пусть Δ_n, $n = 1, 2, ..., N$ - некоторое пронумерованное разбиение области D

$$D = \bigcup_{k=1}^{N} \Delta_k, \qquad \Delta_i \bigcap \Delta_j = \emptyset, \quad i \neq j$$

Согласно (1.7)

$$R(x,t,n+1) = R(x,t,n) + \int_{\Delta_n} R(x,s,n)R(s,t,n+1)ds, \tag{1.8}$$

$$x,t \in D, \quad n = 0, 1, ..., N-1,$$

где $\Delta_o = D_o = \emptyset$, $D_n = \cup_1^n \Delta_j$, $R(x,t,n) = R(x,t,D_n)$.

Очевидно, что

$$R(x,t,0) = K(x,t), \qquad x,t \in D. \tag{1.8'}$$

Соответственно (см. Замечание 1.4), имеем

$$y(x,n+1) = y(x,n) + \int_{\Delta_n} R(x,s,n+1)y(s,n)ds, \quad x \in \bar{D}, \quad n = 0, 1, ..., N-1, \tag{1.9}$$

$$y(x,0) = f(x), \quad x \in \bar{D}. \tag{1.9'}$$

Таким образом, решение основной задачи, - нахождение функции $y(x,N)$ сводится к рекуррентмой серии вычислений $R(x,t,n)$ и $y(x,n)$; $n = 1, 2, ..., N$ по (1.8) и (1.9) при стартовых (начальных) значениях (1.8') и (1.9').

1.4.3. Схема одношагового параллельного алгоритма решения уравнения (0.1) основана на дискретизации упомянутого рекуррентного процесса. Для того, чтобы реализовать каждый шаг численно, необходимо выбрать сеть дискретизации в D и интегралы в (1.8), (1.9) приближенно заменить квадратурной (кубатурной) формулой соответствующего (в зависимости от гладкости K и f) порядка.

Предположим, что на n-том ($n = 0, 1, ..., N-1$) шаге каждый из интегралов (1.8) и (1.9) заменяется комбинацией значений подынтегральных выражений в p_n точках, не обязательно принадлежащих области Δ_n. Тогда общая сеть дискретизации области D будет состоять из q точек, $q \leq p = \sum_1^N p_n$, а для нахождения $R(x, t, n+1)$ на соответствующей сети в $\bar{D} \times \bar{D}$ из q^2-точек потребуется предварительное решение алгебраической системы с $p_n \times p_n$ - матрицей и правой частью в виде $p_n \times q$-матрицы. Для нахождения $R(x, t, n+1)$ потребуется $Nq^2 + q\sum_1^N p_n^2$ мультипликативных операций в алгебре Ω (в случае (0.1) это $l \times l$-матричные операции).

Замечание 1.5. Схема вычисления R неявная, а y - явная. Кроме того, вычисление $y(x, N)$ требует значительно меньшего количества операций порядка Nq. Так как для этого обсчета требуется значение $R(x, s, n+1)$, где x меняется на всей сети, а s - только на ее части, которая соответствует упомянутой выше сети из p_n-точек. Здесь общее количество операций для вычисления R можно сократить. Так, если эти p_n точек принадлежат Δ_n, то достаточно считать в $R(x, t, n)$ $t \in D \setminus D_n$. Что сокращает количество операций для обсчета R примерно вдвое.

Замечание 1.6. Примененный алгоритм – одношаговый, параллельный и может быть реализован с линейным ускорением (см. Введение) на сети из $q(q+1)$ процессоров за N тактов.

Замечание 1.7. Необходимым условием реализации схемы является требование обратимости операторов $\mathcal{J} - K_D$ при $D = D_n = \cup_1^n \Delta_k$, $n = 1, 2, ..., N$.

Замечание 1.8. При высокой размерности $m(D \subset R^m)$ приближенное вычисление интегралов в (1.8) и (1.9) может проводиться методом Монте-Карло или комбинированным методом (см. [25]).

1.4.4. Экономичный алгоритм, предназначенный для счета на однопроцессорных

системах, строится следующим образом.

В формуле (1.8) предполагается, что $(x,t) \in \bar{D}_n \times \bar{D}_n$, а в формуле (1.9) $x \in \bar{D}_n$. Функции $R(x,t,n+1)$ и $y(x,n+1)$ доопределяются соответственно при $x \in (\bar{D}_{n+1} \times \bar{D}_{n+1}) \setminus (\bar{D}_n \times \bar{D}_n)$ и $x \in D_{n+1} \setminus D_n = \Delta_n$ согласно исходным уравнениям (1.4), (1.5) и (0.1). Последние вычисления на сети дискретизации могут привести к решению систем относительно небольшого порядка. Указанная схема приводит к сокращению сложности параллельного варианта алгоритма примерно на 33%.

1.4.5. Рассмотрим случай $m = 1$. Обозначим $D^1 = (0,1)$, $D^2 = (-1,1)$, $D_n^1 = (0,\tau_n)$, а $D_n^2 = (-\tau_n,\tau_n)$; $n = 0,1,...,N$, $0 = \tau_0 < \tau_1 < \cdots < \tau_{N-1} < \tau_N = 1$. Схема (1.8) – (1.9), соответствующая D^1, когда $\Delta_n = (\tau_{n-1},\tau_n]$, имеет (см. Замечание 1.5) вид

$$R^1(x,t,n+1) = R^1(x,t,n) + \int_{\tau_n}^{\tau_{n+1}} R^1(x,s,n)R^1(s,t,n+1)ds, \tag{1.10}$$
$$0 \le x \le 1, \quad \tau_n \le t \le 1, \quad n = 0,1,...,N-1,$$

$$y^1(x,n+1) = y^1(x,n) + \int_{\tau_n}^{\tau_{n+1}} R^1(x,s,n+1)y(s,n)ds; \tag{1.10'}$$
$$0 \le x \le 1, \quad n = 0,1,...,N-1.$$

При $D = D^2$ имеем

$$R^2(x,t,n+1) = R^2(x,t,n) + \int_{\tau_n}^{\tau_{n+1}} [R^2(x,s,n)R^2(s,t,n+1) +$$
$$+ R^2(x,-s,n)R^2(-s,t,n+1)]ds; \ |x| \le 1, \ \tau_n \le |t| \le 1, \quad n = 0,1,...,N-1, \tag{1.11}$$

$$y^2(x,n+1) = y^2(x,n) + \int_{\tau_n}^{\tau_{n+1}} [R^2(x,s,n+1)y^2(s,n) + \tag{1.11'}$$
$$+ R^2(x,-s,n+1)y^2(-s,n)]ds; \quad -1 \le x \le 1, \quad n = 0,1,...,N-1.$$

Выбор сети по x и t зависит от применяемых квадратурных формул. Например, в случае формул закрытого типа сеть $\{\tau_n\}$ является минимально возможной. Указанные схемы (см. Замечание 1.5) - неявного типа.

1.4.6. Пусть теперь $m > 1$. Сначала рассмотрим случай, когда $x = (x_1,...,x_m) \in D$, $0 \le x_m \le 1$ и $\Delta_n = D \cap \{\tau_{n-1} \le x_m \le \tau_n\}$. Тогда схема (1.8) – (1.9) реализуется за N шагов, причем на каждом шаге, для обеспечения максимально возможной точности, решается алгебраическая система отнюдь не малого (при $m \ge 2$) порядка.

Во втором случае выберем Δ_n, $n = 1, 2, ..., N$ так, чтобы

$$\max_n \operatorname{diam} \Delta_n \to 0, \quad n \to \infty.$$

Тогда на каждом шаге будет решаться система небольшого порядка. В типичном случае гиперкуба $D = \{0 \le x_i \le 1\}$, $i = 1, 2, ..., m$ в первом случае

$$\Delta_n = \{\frac{n-1}{N} \le x_m \le \frac{n}{N}\}; \quad n = 1, 2, ..., N,$$

а во втором Δ_n – пронумерованная последовательность кубов

$$\Delta(n_1, n_2, ..., n_m) = \{\frac{n_i - 1}{N} \le x_i \le \frac{n_i}{N}\}; \qquad n_i = 1, 2, ..., N; \quad i = 1, 2, ..., m.$$

§1.5. МЕТОД НЕПРЕРЫВНОГО ПОГРУЖЕНИЯ. МНОГОШАГОВЫЕ АЛГОРИТМЫ

1.5.1. Пусть $D_\tau^1 = (0, \tau)$, $D_\tau^2 = (-\tau, \tau)$, $0 \le \tau \le 1$. Обозначим через y^1 и y^2 решения уравнений

$$(\mathcal{J} - \mathcal{K}_\tau^1)y^1 \equiv y^1(x, \tau) - \int_0^\tau K(x, t)y^1(t, \tau)dt = f(x), \quad 0 \le x \le 1, \tag{1.12}$$

$$(\mathcal{J} - \mathcal{K}_\tau^2)y^2 \equiv y^2(x, \tau) - \int_{-\tau}^\tau K(x, t)y^2(t, \tau)dt = f(x). \quad -1 \le x \le 1 \tag{1.13}$$

Теорема 1.3 *Если $K \in C(\bar{D}_1^i \times \bar{D}_1^i)$ и операторы $\mathcal{J} - \mathcal{K}_{\tau_o}^i$; $i = 1, 2$, $\tau_o \in (0, 1]$ обратимы, то в некоторой окрестности $[\tau_o - \epsilon, \tau_o + \epsilon) \cap [0, 1]$ $(\exists \epsilon > 0)$ для резольвентных ядер $R^i(x, t, \tau)$ и решений уравнений (1.12) и (1.13) справедливы соотношения*

$$\frac{\partial R^1(x, t, \tau)}{\partial \tau} = R^1(x, \tau, \tau)R^1(\tau, t, \tau), \quad 0 \le x, t \le 1, \tag{1.14}$$

$$\frac{\partial y^1(x, \tau)}{\partial \tau} = R^1(x, \tau, \tau)y^1(\tau, \tau), \quad 0 \le x \le 1, \tag{1.14'}$$

$$\frac{\partial R^2(x, t, \tau)}{\partial \tau} = R^2(x, \tau, \tau)R^2(\tau, t, \tau) + R^2(x, -\tau, \tau)R^2(-\tau, t, \tau), \quad -1 \le t, x \le 1, \tag{1.15}$$

$$\frac{\partial y^2(x, \tau)}{\partial \tau} = R^2(x, \tau, \tau)y^2(\tau, \tau) + R^2(x, -\tau, \tau)y^2(-\tau, \tau), \quad -1 \le x \le 1. \tag{1.15'}$$

Доказательство получим, перейдя к пределу в соотношениях (1.10) – (1.11'), предварительно разделив их на h_n

$$h_n = \tau_{n+1} - \tau_n; \qquad \tau_n = \tau, \quad \tau_{n+1} \to \tau, \quad \tau \in [\tau_o - \epsilon, \tau_o + \epsilon) \cap [0, 1], \quad \exists \epsilon > 0.$$

Таким образом, при обратимости $\mathcal{J} - K_\tau^i$, $i = 1, 2$ для всех $\tau \in (0, 1]$, уравнение (1.12) при $\tau = 1$ сводится к системе (1.14) – (1.14′) или (1.15) – (1.15′) с начальными условиями

$$R^i(x, t, 0) = K(x, t), \quad x, t \in D_1^i; \qquad y^i(x, 0) = f(x), \quad x \in D_1^i, \quad i = 1, 2.$$

1.5.2. Рассмотрим случай равномерной по τ сети $\tau_n = nh$, $h = \frac{1}{N}$, $n = 0, 1, ..., N$. (Здесь и далее в случае равномерной сетки nh, $n = 0, 1, ..., N$ множитель h иногда будет опускаться.) Если операторы $\mathcal{J} - K_\tau^i$, $i = 1, 2$, $0 < \tau \leq 1$; обратимы, то, согласно Теореме 1.3

$$R^1(x, t, n+1) = R^1(x, t, n) + \int_{nh}^{(n+1)h} R^1(x, s, s) R^1(s, t, s) ds, \quad 0 \leq x, t \leq 1, \quad (1.16)$$

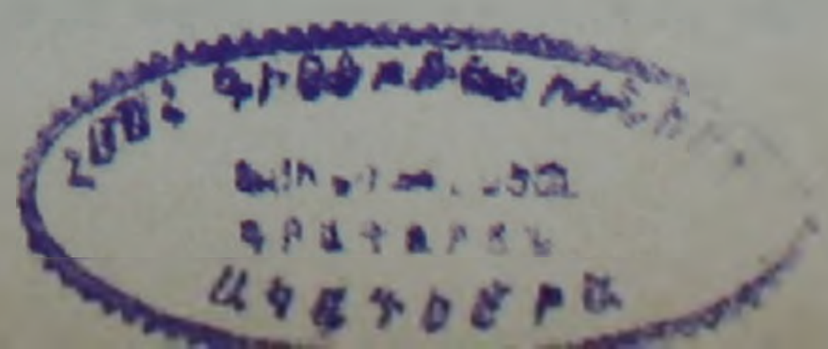
$$y^1(x, n+1) = y^1(x, n) + \int_{nh}^{(n+1)h} R^1(x, s, s) y^1(s, s) ds, \quad 0 \leq x \leq 1, \quad (1.16')$$

$$R^2(x, t, n+1) = R^2(x, t, n) + \int_{nh}^{(n+1)h} [R^2(x, s, s) R^2(s, t, s) + R^2(x, -s, s) R^2(-s, t, s)] ds, \quad -1 \leq t, x \leq 1, \quad (1.17)$$

$$y^2(x, n+1) = y^2(x, n) + \int_{nh}^{(n+1)h} [R^2(x, s, s) y^2(s, s) + R^2(x, -s, s) y^2(-s, s)] ds, \quad -1 \leq x \leq 1. \quad (1.17')$$

В отличие от схемы (1.10) – (1.11′), здесь сети по (x, t) и τ должны совпадать. Используя формулу нижних прямоугольников получим явную схему алгоритма порядка $O(h)$, $h = \frac{1}{N} \to 0$, $K, f \in C^1$ на сети $x, t, \tau = nh$, $n = 0, 1, ..., N$. Для построения аналогичного алгоритма на основе формул (1.17), (1.17′) надо в каждой из них к первому интегралу применить формулу нижних прямоугольников, а ко второму - верхних.

При применении к (1.16), (1.16′) правила верхних прямоугольников (или при $K, f \in C^2$ – трапеций, что приводит к $O(h^2)$-алгоритму) получаем алгоритм неявного типа. Здесь на каждом этапе при $l = 1$ необходимо решить квадратное уравнение. Такой подход, насколько нам известно, впервые был применен в работе [27]. Все попытки получить методы порядка выше $O(h^2)$ приводят к многошаговым алгоритмам по сравнению с [27].



1.5.3. Остановимся подробней на случае $K, f \in C^4$, когда применима формула Симпсона на "двойной" сети $x, t, \tau = \frac{k}{2N}$, $k = 0, 1, ..., 2N$. При этом основное соотношение (1.16) даст (см., также ниже, Замечание 1.5) :

$$R(x,t,n+1) = R(x,t,n) + \frac{h}{6}[R(x,n+1,n+1)R(n+1,t,n+1)+$$
$$+ 4R(x,n+\frac{1}{2},n+\frac{1}{2})R(n+\frac{1}{2},t,n+\frac{1}{2}) + R(x,n,n)R(n,t,n)];$$
$$n = 0, \frac{1}{2}, ..., N-\frac{3}{2}, N-1, x = \frac{k}{2N},\ k = 0,1,...,2N,\ t = \frac{p}{2N},\quad p = n+1,...,2N. \tag{1.18}$$

Это схема двухшаговая и неявная : величина каждого из $2N$ шагов равна $\frac{1}{2N}$ и при этом в скалярном ($l = 1$) случае требуется решить квадратное уравнение (в отношении (1.17) это будет уравнение четвертого порядка). Для того, чтобы избежать излишних трудностей, связанных с выбором нужного корня, можно применить приближенный итерационный метод.

В данном случае необходимо задать два "стартовых" условия при $n = 0$ и $n = 1/2$. Условие при $n = 1/2$ надо найти с точностью $O(h^5)$. Накопление ошибок (см., ниже, Замечание 1.11) можно несколько уменьшить на основе соотношения (1.14).

Замечание 1.9. Схема (1.17) – (1.17′) позволяет получить стартовые значения при $n = 1/2$, применив формулу Грина к (1.11) при $\tau_o = 0$, $\tau = h/2$ и решив систему 3×3.

Замечание 1.10. На основе формул (1.16) – (1.17′) можно построить многошаговые алгоритмы с использованием нескольких точек левее интервала kh, $k < n$. Так как при этом произведения типа $R(x,k,k) \cdot R(k,t,k)$ фигурируют на разных "уровнях", сложность алгоритма увеличивается не так сильно. Существенно увеличивается лишь оперативная память.

1.5.4. В многомерном случае схема непрерывного погружения может быть реализована более разнообразно. Приведем аналог первого способа п. 1.4.6. Пусть $x = (x_1, ..., x_m) \in D \subset R^m$, $0 < x_m < 1$, $D_\tau = D \cap \{x_m < \tau\}$, $L_\tau = D \cap \{x_m = \tau\}$, $0 < \tau \leq 1$.

Теорема 1.4. *Пусть оператор $\mathcal{J} - K$ при $D = D_\tau$, $0 < \tau \leq 1$ обратим, тогда его резольвентное ядро $R(x,t,\tau)$ удовлетворяет интегро-дифференциальному*

уравнению

$$\frac{\partial R(x,t,\tau)}{\partial \tau} = \int_{L_\tau} R(x,s_\tau,\tau)R(s_\tau,t,\tau)ds_1\cdots ds_{m-1}, \qquad x,t\in D, \tag{1.19}$$

где $s_\tau = (s_1, s_2, \ldots, s_{m-1}, \tau)$.

Доказательство может быть проведено как прямым дифференцированием формулы (1.4) по τ (полагаем, что $D = D_\tau$), так и предельным переходом в схеме 1.4.6.

1.5.5. Схема непрерывного погружения и на практике работает лишь при обратимости $\mathcal{J} - \mathcal{K}_\tau$ для всех $\tau \in [0,1]$ (см. численные результаты в главе III,§§3.3-3.5). Кроме того, в отличие от схемы дискретного погружения, она неудобна для квадратурных формул с неравномерной сеткой (например, для формул Гаусса). По сравнению с алгоритмами §1.4 здесь ошибки квадратурных формул выше. Причина в том, что в интегралах (1.16) – (1.17′) $R(x,t,\tau)$ и $y(x,\tau)$ интегрирование ведется по всем переменным t, x и τ.

Замечание 1.11. В некоторых случаях этот отрицательный эффект можно нейтрализовать. Например, при $K, f \in C^1$ можно, используя соотношение (1.41) и формулу нижних прямоугольников, (1.16) дискретизировать с точностью $O(h^2)$ в виде

$$R(x,t,n+1) = R(x,t,n) + hR(x,n,n)\alpha_n R(n,t,n), \tag{1.20}$$

где $\alpha_n = E_l - \frac{h}{2}R(n,n,n)$ и E_l - единичная $l\times l$-матрица.

1.5.6. Остановимся вкратце на истории развития методов, являющихся, по сути, континуальным аналогом метода гауссовой треугольной факторизации матриц.

Схема непрерывного погружения впервые была применена в работах С. Л. Соболева, Н. П. Сергеева и М. Г. Крейна [29 – 31]. В серии работ А. Макнабба и А. Шумицки (см., например, [27]) этот подход был систематически развит в разных направлениях, в том числе и в вычислительном плане. Н. Б. Енгибарян и другие (см. [33]) применили его в случае $K(x,t) \geq 0$, $\|K\| = 1$, не останавливаясь на вычислительном аспекте.

Основные соотношения схемы дискретного погружения, аналогия которой с методом блочной факторизации матриц очевидна можно найти в работах [29,30].

Однако, насколько нам известно, эта схема ранее в вычислительном плане не обсуждалась и не использовалась (см. 1.5.5, §§3.3–3.5 и [7, 14 – 17, 42]). Схемы п.п. 1.4.5 и 1.5.1 являются континуальными аналогами "метода смыкающих квадрантов", эффективного при решении алгебраических систем с матрицами ленточного типа.

Абстрактные аспекты погружения вписываются в схему теории вольтерровых операторов (см. [34]).

Замечание 1.12. Схемы погружения §§1.4, 1.5 работают не только при "стартовой" точке $\tau_o = 0$, но и с любого значения τ_o, для которого известны $R(x, t, \tau_o)$ и $y(x, \tau_o)$, (см. [26], где $\tau_o = 2$, $\tau \downarrow 1$).

§1.6. АВТОМАТИЧЕСКИЙ ВЫБОР ШАГА И ПОРЯДКА АЛГОРИТМА

1.6.1. Схема дискретного погружения, в принципе, аналогична известному методу Рунге-Кутты, применяемому для решения задачи Коши для обыкновенного дифференциального уравнения. В свою очередь, схема непрерывного погружения имеет определенную аналогию с методами типа Адамса-Мултона [21].

В нашем случае ситуация сложнее из-за увеличения размерности. Однако, здесь мы имеем дело лишь с нелинейностью квадратичного типа, аналогичной нелинейности в уравнении Рикатти. Это частично снижает трудности.

1.6.2. Пусть $m = 1$. Рассмотрим метод автоматического выбора "переменного шага", т.е. выбора разбиения $D = \cup\Delta_n$ в схеме (1.10) – (1.10′). Предположим, что $K, f \in C^p$ и применим квадратурные формулы p-ого порядка. Тогда ошибка в (1.10) и (1.10′) зависит от величин

$$\alpha_n = \frac{\partial^p}{\partial s^p}[R(x, s, \tau_n)R(s, t, \tau_{n+1}], \quad 0 \le x, t \le 1, \quad \tau_n \le s \le \tau_{n+1},$$

$$\alpha'_n = \frac{\partial^p}{\partial s^p}[R(x, s, \tau_{n+1})y(s, \tau_n)], \quad 0 \le x \le 1, \quad \tau_n \le s \le \tau_{n+1}.$$

Будем стремиться к тому, чтобы на каждом шаге $n = 0, 1, ..., N-1$ величины $\alpha_n(\alpha'_n)$ были примерно одинаковы. Этого можно достичь, в частности, регулируя величину шага.

Время счета можно сократить, не теряя в точности, также изменением порядка применяемых квадратурных формул, учитывая, что формулы более низкого порядка, как правило, требуют меньшего количества вычислений.

Наиболее простой подход заключается в игнорировании величины α_n и учете на каждом шаге усредненного по (x, s)-сети значения α'_n.

Замечание 1.13. Методы типа Рунге-Кутты (в частности, метод переменного шага) для интегральных уравнений были реализованы ранее, по-видимому, лишь в вольтерровском случае (см. [1, 35]).

§1.7. МЕТОД КОНЕЧНОМЕРНОГО СМЕЩЕНИЯ ЯДРА

1.7.1. Рассмотрим следующее семейство интегральных уравнений :

$$(\mathcal{J} - \mathcal{K}_\tau)y \equiv y(x,\tau) - \int_0^\tau K(x,t,\tau)y(t,\tau)dt = f(x), \quad 0 \le x \le 1, \tag{1.21}$$

$$K(x,t,\tau) = K(x,t) - p(x)^{\mathrm{tr}}d(\tau)q(t), \quad 0 \le x,t \le 1, \tag{1.21'}$$

где ядро K то же, что и в §§1.4, 1.5 ; p, q – $r \times l$-матрицы, а d – $rl \times rl$-матрица.

Эта схема несколько белее общая, чем вышеприведенные схемы погружения. Если при данном τ оператор $\mathcal{J} - \mathcal{K}_\tau$ обратим, то через $R(x,t,\tau)$ обозначим его резольвентное ядро ($(\mathcal{J} - \mathcal{K}_\tau)^{-1} = J + \mathcal{R}_\tau$).

Введем в рассмотрение оператор

$$(\mathcal{J} - \mathcal{K}_\tau^{\#})z \equiv z(x) - \int_0^\tau z(t)K(t,x)dt, \quad 0 \le x \le 1. \tag{1.22}$$

Обозначим через $P(x,\tau)$ и $Q(t,\tau)$ решения уравнения $(\mathcal{J} - \mathcal{K}_\tau)P^{\mathrm{tr}}(x,\tau) = p^{\mathrm{tr}}(x)$, $(\mathcal{J} - \mathcal{K}_\tau^{\#})Q(x,\tau) = q(x)$. Пусть наконец

$$\omega(\tau) = \int_0^\tau q(s)P(s,\tau)ds, \qquad \sigma(s) = \int_0^\tau q(s)y(s,\tau)ds, \tag{1.22'}$$

$$(\mathcal{J} - \mathcal{K})Y = Y(x) - \int_0^1 K(x,t)Y(t)dt = f(x), \quad 0 \le x \le 1. \tag{1.23}$$

Теорема 1.5. *Если оператор $\mathcal{J} - \mathcal{K}_\tau$ обратим при $\tau = \tau_n \in [0,1]$, $n = 0, 1, ..., N$, то*

$$\begin{aligned} R(x,t,\tau_{n+1}) = R(x,t,\tau_n) + \int_{\tau_n}^{\tau_{n+1}} R(x,s,\tau_n)K(s,t,\tau_{n+1})ds - \\ - P^{\mathrm{tr}}(x,\tau_n)[d(\tau_{n+1}) - d(\tau_n)]Q(t,\tau_{n+1}), \quad 0 \le x,t \le 1, \end{aligned} \tag{1.24}$$

$$P^{\mathrm{tr}}(x,\tau_{n+1}) = P^{\mathrm{tr}}(x,\tau_n)[E_l - (d(\tau_{n+1}) - d(\tau_n))\omega(\tau_{n+1})] + \int_{\tau_n}^{\tau_{n+1}} R(x,s,\tau_n)P^{\mathrm{tr}}(s,\tau_{n+1})ds, \quad 0 \le x \le 1, \tag{1.25}$$

$$Q(t,\tau_{n+1}) = [E_l - \omega(\tau_{n+1})(d(\tau_{n+1}) - d(\tau_n))]Q(t,\tau_n) + \int_{\tau_n}^{\tau_{n+1}} Q(s,\tau_{n+1})R(s,t,\tau_n)ds, \quad 0 \le t \le 1, \tag{1.25'}$$

$$y(x,\tau_{n+1}) = y(x,\tau_n) + \int_{\tau_n}^{\tau_{n+1}} R(x,s,\tau_{n+1})y(s,\tau_n) - P^{\mathrm{tr}}(x,\tau_n)[d(\tau_{n+1}) - d(\tau_n)]\sigma(\tau_{n+1}), \quad 0 \le x \le 1, \tag{1.26}$$

где E_l – единичная $l \times l$-матрица.

Если и оператор $\mathcal{J} - \mathcal{K}$ обратим (см. (1.23)), то $\det[E_l - \omega(1)d(1)] \neq 0$ и решение уравнения (1.23) восстанавливается по формуле

$$Y(x) = y(x,N) + P^{\mathrm{tr}}(x,N)d(1)[E_l - \omega(1)d(1)]^{-1}\sigma(1). \tag{1.27}$$

Доказательство. Из соотношений (1.4) при $D = [0,\tau_n)$ и $D = [0,\tau_{n+1})$ имеем (см. (1.20′))

$$R(x,t,\tau_{n+1}) - R(x,t,\tau_n) = \int_{\tau_n}^{\tau_{n+1}} K(x,s,\tau_{n+1})R(s,t,\tau_{n+1})ds +$$
$$+ \int_0^{\tau_n} K(x,s,\tau_n)[R(s,t,\tau_{n+1}) - R(s,t,\tau_n)]ds +$$
$$+ p^{\mathrm{tr}}(x)[d(\tau_{n+1}) - d(\tau_n)][q(t) + \int_0^{\tau_n} q(s)R(s,t,\tau_{n+1})ds].$$

Применив к обеим частям этого равенства оператор $(\mathcal{J} - \mathcal{K}_\tau)^{-1} = J + \mathcal{R}_\tau$, $\tau = \tau_n$, получим соотношение (1.24). Формулы (1.25), (1.25′) и (1.26) выводятся аналогично. Что же касается формулы (1.27), то она известна как следствие формул для конечномерного возмущения оператора $\mathcal{J} - \mathcal{K}$ (см. формулу Шермана-Моррисона в [13]).

1.7.2. Схема непрерывного погружения обобщается следующим образом.

Теорема 1.6. *Пусть $K, f, p, q \in C$, функция $d(\tau)$ непрерывно дифференцируема и операторы $\mathcal{J} - \mathcal{K}_\tau$ обратимы при $\tau \in [0,1]$. Тогда справедливы соотношения*

$$\frac{\partial R(x,t,\tau)}{\partial \tau} = R(x,\tau,\tau)R(\tau,t,\tau) - P^{\mathrm{tr}}(x,\tau)d'(\tau)Q(t,\tau), \quad 0 \le x,t \le 1, \tag{1.28}$$

$$\frac{\partial P^{\mathrm{tr}}(x,\tau)}{\partial \tau} = R(x,\tau,\tau)P^{\mathrm{tr}}(\tau,\tau) - P^{\mathrm{tr}}(x,\tau)d'(\tau)\omega(\tau), \quad 0 \le x \le 1, \tag{1.29}$$

$$\frac{\partial Q(t,\tau)}{\partial \tau} = Q(\tau,\tau)R(\tau,t,\tau) - \omega(\tau)d'(\tau)Q(t,\tau), \qquad 0 \le t \le 1, \tag{1.29'}$$

$$\frac{\partial y(x,\tau)}{\partial \tau} = R(x,\tau,\tau)y(\tau,\tau) - P^{\mathrm{tr}}(x,\tau)d'(\tau)\sigma(\tau), \quad 0 \le x \le 1, \tag{1.30}$$

$$\omega'(\tau) = Q(\tau,\tau)P^{\mathrm{tr}}(\tau,\tau) - \omega(\tau)d'(\tau)\omega(\tau), \tag{1.31}$$

$$\sigma'(\tau) = Q(\tau,\tau)y(\tau,\tau) - \omega(\tau)d'(\tau)\sigma(\tau). \tag{1.32}$$

Доказательство можно провести как непосредственным дифференцированием исходных формул п. 1.7.1 по τ, так и переходом к пределу в Теореме 1.5 при $\tau_n = \tau$, $\tau_{n+1} \to \tau$.

1.7.3. Приведенные обобщения схем погружения предназначены для рекуррентного решения основного уравнения (1.2.3) при стартовых условиях

$$\begin{aligned} &R(x,t,0) = K(x,t) - p^{\mathrm{tr}}(x)d(0)q(t), \qquad y(x,0) = f(x), \\ &P(x,0) = p(x), \qquad Q(t,0) = q(t), \qquad \omega(0) = \sigma(0) = 0. \end{aligned} \tag{1.33}$$

Схемы эти призваны уменьшить накопление ошибок в методах §§1.4, 1.5, соответствующих случаю $d(\tau) \equiv 0$. Предполагается, например, подобрать p, q и d так, чтобы уменьшить норму $K(x,t,\tau)$ и его производных в квадрате $0 \le x, t \le \tau$. По сути дела, этот подход является комбинацией метода погружения с проекционным методом (см. [1, 2]).

Пример 1.4. Предположим, что $l = 1$ и $p^{\mathrm{tr}}(x) = \big(p_1(x), \ldots, p_r(x)\big)$, $q^{\mathrm{tr}}(t) = \big(q_1(t), \ldots, q_r(t)\big)$. Поставим задачу минимизации нормы функционала

$$I(d(\tau)) = \int_0^\tau \int_0^\tau ||K(x,t,\tau)||^2 dx dt, \qquad d(\tau) = ||d_{ij}(\tau)||; \quad i,j = 1,\ldots,r \tag{1.34}$$

посредством выбора матрицы $d(\tau)$. Элементы d_{ij} матрицы d определяются из системы (невырожденность которой предполагается)

$$\sum_{\alpha,\beta=1}^{r} \int_0^\tau p_\alpha(x)p_i(x)dx \int_0^\tau q_\beta(x)q_j(x)dx \cdot d_{\alpha\beta} = \int_0^\tau \int_0^\tau K(x,t)p_i(x)q_j(t)dx dt, \quad i,j = 1,2,\ldots,r. \tag{1.35}$$

Величина $I_o = \min I(d)$ имеет вид

$$I_o = \int_0^\tau \int_0^\tau |K(x,t)|^2 dx dt - \int_0^\tau \int_0^\tau K(x,t)p(x)dq(t)dx dt. \tag{1.35'}$$

1.7.4. Усложненная смещением ядра схема погружения приводит к увеличению высоты соответствующих алгоритмов. Основное увеличение происходит за счет дополнительных мультипликативных операций в формулах (1.24) и (1.28). В результате в некоторых случаях достигается высокая точность.

Сокращение числа операций можно получить специальным выбором функций $p(x)$ и $q(t)$. Например

$$p^{\mathrm{tr}}(x) = \big(K(x,t_1), K(x,t_2), ..., K(x,t_r)\big), \quad 0 \le x \le 1,$$
$$q^{\mathrm{tr}}(t) = \big(K(x_1,t), K(x_2,t), ..., K(x_r,t)\big), \quad 0 \le t \le 1, \tag{1.36}$$

где $\{x_k\}, \{t_k\}, k = 1,2,...,r$ – некоторые точки из выбранной $N \times N$-сети, $r << N$. В этом случае

$$P^{\mathrm{tr}}(x,\tau) = \big(R(x,t_1,\tau), ..., R(x,t_r,\tau)\big),$$
$$Q^{\mathrm{tr}}(t,\tau) = \big(R(x_1,t,\tau), ..., R(x_r,t,\tau)\big). \tag{1.37}$$

Вследствие этого на сети $\{\tau_n\}$ вычисляются только $R(x,t,\tau)$ и $y(x,\tau)$.

Пример 1.5. Пусть $K(x,t)$ – скалярное ядро, $p(x) = K(x,t_o)$, $q(t) = K(x_o,t)$. Если $x_o, t_o \in [0.1]$, то упрощенная схема уже не содержит $P(x,\tau)$, $Q(t,\tau)$, $\sigma(\tau)$, $\omega(\tau)$ и имеет вид (см. Теорему 1.5, $R(x,t,n) = R(x,t,\tau_n)$)

$$R(x,t,n+1) = R(x,t,n) + \int_{\tau_n}^{\tau_{n+1}} R(x,s,n)R(s,t,n+1)ds -$$
$$-[d(n+1) - d(n)]R(x,t_o,n)R(x_o,t,n+1), \tag{1.38}$$

$$y(x,n+1) = y(x,n) + \int_{\tau_n}^{\tau_{n+1}} R(x,s,n+1)y(s,n)ds -$$
$$-[d(n+1) - d(n)][y(x_o,n+1) - f(x_o)]. \tag{1.38'}$$

Последовательность $d(n)$ может быть выбрана (см. п. 1.7.3) по формуле (1.3.5). Норма $||K||_\tau$, $0 \le x,t \le \tau$ получается как следствие текущих вычислений.

§1.8. ОБ УРАВНЕНИЯХ ПЕРВОГО РОДА И НЕЛИНЕЙНЫХ УРАВНЕНИЯХ

1.8.1. Как было отмечено во введении, решение уравнения (0.5) сводится к решению (0.6). Поскольку оператор в (0.6) положительно определен, методы §§1.4, 1.5 проходят при любой дискретизации $\{\tau_n\} \subset [0,1]$, $0 = \tau_o < \tau_1 < \cdots < \tau_{N-1} < \tau_N = 1$.

Метод обобщенного погружения §1.7 также будет безусловно эффективен, если только оператор (на $[0,\tau]$) с ядром $p^{\mathrm{tr}}(x)d(\tau)q(t)$ будет неотрицателен ($\tau \in [0,1]$).

Замечание 1.14. Уравнение первого рода с ядром $K(x,t)$ можно свести к уравнению с ядром вида $A(x)K(x,t)B(t)$, где $A(x)$ и $B(t)$ – любые невырожденные матрицы. Это так же очевидно, как и то, что в случае уравнения второго рода необходимо, чтобы было $A(x) = B^{-1}(t)$. Другое важное свойство, присущее уравнению первого рода, заключается в возможности независимой замены переменных в (0.5) по x и t.

1.8.2. Покажем теперь, что методы погружения работают и в случае нелинейных уравнений. Рассмотрим уравнение Урысона (0.7). Формально наметим схему реализации дискретного погружения по формуле

$$y(x,n) = \int_0^{\tau_n} F(x,t,y(t,n))dt + f(x), \quad 0 \le x \le 1, \quad n = 0,1,...,N, \qquad (1.39)$$

где $\{\tau_n\}$ – некоторая сеть $(0 = \tau_o < \tau_1 < \cdots < \tau_N = 1)$.

Очевидно, стартовым значением будет $y(x,0) = f(x)$. Далее, имеем

$$z_n(x) \equiv y(x,n+1) - y(x,n) =$$

$$= \int_0^{\tau_{n+1}} \big[F(x,t,y(t,n+1)) - F(x,t,y(t,n))\big]dt + \int_{\tau_n}^{\tau_{n+1}} F(x,t,y(t,n))dt =$$

$$= \int_0^{\tau_{n+1}} F'_y(x,t,y(t,n))z_n(t)dt + \int_{\tau_n}^{\tau_{n+1}} F(x,t,y(t,n))dt + \qquad (1.40)$$

$$+\frac{1}{2}\int_0^{\tau_{n+1}} F''_{yy}(x,t,\theta(t,n))z_n^2(t)dt, \quad 0 \le x \le 1.$$

где $\min\big(y(x,n), y(x,n+1)\big) \le \theta(x,n) \le \max\big(y(x,n), y(x,n+1)\big)$.

Если F и f – гладкие функции, то $z_n(x)$ имеет порядок $O(h_n)$ $h_n = \tau_{n+1} - \tau_n$, а последний член в (1.40) – порядок h_n^3 $((h_n \to 0,\ N \to \infty)$.

Таким образом, с точностью $O(h^3)$ z_n является решением линейного интегрального уравнения второго рода с ядром $K(x,t) = F'_y(x,t,y(t,n))$, что позволяет численно реализовать рекуррентный процесс решения уравнения (0.7), точнее выделения одного из его решений. Даже если ограничиться малым N, полученное приближение позволит в дальнейшем эффективней использовать метод Ньютона-Канторовича [1].

Предложенный подход нетрудно распространить на многомерный случай.

§1.9. УСТОЙЧИВОСТЬ АЛГОРИТМОВ

1.9.1. Сами по себе методы погружения для уравнения второго рода не выдерживают сравнения с методом Нистрема. Так схема дискретного погружения фактически является методом Нистрема, в котором линейная система решается блочным (вообще говоря) методом Гаусса, но без выбора ведущего элемента.

Схемы погружения выгодны лишь тогда, когда по ходу вычислений применяются те или иные средства уменьшения накопления ошибок (см. §§1.6, 1.7, 1.8) или сокращения сложности алгоритма (см. Главу III, §§3.3, 3.5 и 2.1.2).

Замечание 1.15. Хороший эффект дает применение к вычислению $y(x,n)$ формулы более высокого порядка, чем к $K(x,t,n)$. Но при этом необходимо учесть, что на уточнение по Ричардсону при вычислениях на двух разных сетках нельзя рассчитывать.

1.9.2. Важнейшим свойством алгоритмов, построенных на основе методов погружения, является их устойчивость. Поскольку в вычислительном плане (см. 1.5.1 и Главу III) метод дискретного погружения обладает явным преимуществом, остановимся для доказательства устойчивости только на нем.

Приняв за основу общую схему п. 1.4.2 и исходя из сети $\{x_k\} \subset D$, обозначим

$$h = \max_i \min_{j(\neq i)} |x_i - x_j|.$$

Далее, введем следующие обозначения :

$$\delta(x,t,n) = |\tilde{K}(x,t,n) - K(x,t,n)|, \qquad x,t \in D,$$

$$\gamma(x,n) = |\tilde{y}(x,n) - y(x,n)|, \quad n = 0,1,...,N, \quad x \in D, \tag{1.41}$$

$$r = \max_{x,t,n} |\tilde{K}(x,t,n)|, \quad n = 0,1,...,N, \qquad x,t \in D, \tag{1.42}$$

где $\tilde{K}$ и $\tilde{y}$ точные значения фупкций $K(x,t,n)$ и $y(x,n)$.

Теорема 1.7. *Пусть операторы $\mathcal{J} - K_D$ ($D = D_n = \cup_1^n \Delta_k$, mes $\Delta_n \le c/N \le c_1 h$) обратимы, а $K(x,t,n)$ и $y(x,n)$ - соответствующие им приближенные резольвентные ядра и решения уравнений $(\mathcal{J} - K_D)y = f$, $D = D_n$, $n = 1,2,...,N$. Пусть, далее, $K, f \in C^p$ и в интегралах формул (1.8) и (1.9) используются квадратурные формулы порядка $O(h^p)$. Тогда при $h \le h_o$ справедливы оценки*

$$\delta_n = \max_{x,t} \delta(x,t,n) \le M_1 \left(1 + \frac{3cr}{N}\right)^n nh^{p+1}, \tag{1.43}$$

$$\gamma_n = \max_x \gamma(x,n) \le M_2\left(1+\frac{3cr}{N}\right)^n nh^{p+1}, \qquad (1.43')$$

где M_1 *и* M_2 *– не зависящие от* n, N *постоянные.*

В частности

$$\delta_N \le M_1 e^{3cr}h^p, \qquad \gamma_N \le M_2 e^{3cr}h^p. \qquad (1.44)$$

Доказательство. Из (1.8) имеем

$$\bar{R}(x,t,n+1) = \bar{R}(x,t,n) + \sum_{k=1}^{p_n} w_k \bar{R}(x,s_k,n)\bar{R}(s_k,t,n+1) + A_p h^p \operatorname{mes}\Delta_n, \qquad (1.45)$$

где w_k – весовые множители соответствующей квадратурной (кубатурной) формулы, а $A_p(x,t,n)$ – соответствующая ей функция, зависящая от производных $\bar{R}(x,s,n)$ и $R(s,t,n+1)$ по s.

Поскольку $R(x,t,n+1)$ находится из (1.45), то

$$\delta(x,t,n+1) \le \delta(x,t,n) + \sum w_k \delta(x,s_k,n)|\bar{R}(s_k,t,n+1)| +$$

$$+ \sum w_k |R(x,s_k,n)|\delta(s_k,t,n+1) + A_p\frac{c}{N}h^p,$$

откуда

$$\delta_{n+1} \le \delta_n + \frac{c}{N}r(\delta_n+\delta_{n+1}) + \delta_n\delta_{n+1} + A\frac{h^p}{N}, \quad A = c\max|A_p|.$$

Для достаточно малого h

$$\delta_{n+1} \le \left(1+\frac{3cr}{N}\right)\delta_n + 2A\frac{h^p}{N}. \qquad (1.46)$$

Далее доказательство проведем по индукции. Предположим, что оценка (1.43) верна для данного $n \ge 0$ (при $n = 0$ она очевидна). Тогда из (1.46) имеем

$$\delta_{n+1} \le M_1\left(1+\frac{3cr}{N}\right)^{n+1}\frac{nh^p}{N} + 2A\frac{h^p}{N}.$$

Поскольку ,очевидно, $\exists N_0$, $N > N_0 \Rightarrow 2A/N < M_1$, то отсюда следуют оценка (1.43) и первая из оценок (1.44). Оценки (1.45), как и вторая из оценок (1.44), теперь очевидны.

ГЛАВА II. СПЕЦИФИЧНЫЕ ЯДРА И БЫСТРЫЕ АЛГОРИТМЫ

§2.1. ИТЕРАЦИОННОЕ УТОЧНЕНИЕ

2.1.1. Итерационные методы для уравнений второго порядка эффективны лишь в случае специальных ядер. Прежде всего это относится к ядрам сжимающих операторов K, ($\|K\| < 1$) и операторов вольтерровского типа (см. [1, 2] и [22, 23]).

Сложность итерационного алгоритма для интегральных уравнений не может считаться линейной функцией количества итераций. Дело в том, что одновременно с наращиванием количества итераций необходимо сгущать сеть так, чтобы не влияли ошибки квадратурных формул. В то же время, в течение первых итераций, пока еще точность решения невелика, использовать густую сеть невыгодно.

Другое замечание о тактике итерационного счета касается вычисления на универсальных (однопроцессорных) ЭВМ. Пусть $\|K\| < 1$ и итерационный процесс решения уравнения (0.1) задан формулой

$$y_2(x) = \sum_{k=1}^{N} w_k K(x, x_k) y_1(x_k) + f(x), \quad x \in D, \tag{2.1}$$

где $\{w_k\}$ – коэффициенты применяемой квадратурной формулы. Допустим, что на первом шаге вычисляется значение $y_2(x_1)$. Далее рациональнее в сумме (2.1) значение $y_1(x_1)$ заменить на уже вычисленное $y_2(x_1)$. Итак, "циклическая" схема итераций выглядит следующим образом : $y_1(x_1)$ вычисляется по (2.1) и далее на p-том шаге ($p = 2, 3, ..., N$)

$$y_2(x_p) = \sum_{k=0}^{p-1} w_k K(x_p, x_k) y_2(x_k) + \sum_{k=p}^{N} w_k K(x_p, x_k) y_1(x_k). \tag{2.2}$$

Такой подход, очевидно, эффективней обычного, но при параллельном счете эта возможность уменьшается.

2.1.2. Как уже отмечалось в п. 1.9.1, недостатки методов погружения §§1.6, 1.7 можно компенсировать в процессе счета. Рассмотрим схему дискретного погружения §1.4. Наша цель – нахождение решения $y(x, n)$, $n = 1, 2, ..., N$, уравнения (0.1). Предположим, что L_2-нормы ядра K растут с увеличением n.

Однако, пока $||K||_n < 1$, имеется возможность уточнить $y(x,n)$ итерационным способом, уменьшив накопившиеся ошибки. Если даже норма $||K||_N$ меньше единицы, но близка к ней (например, $||K||_N \approx 0.9$), то выгодней проводить итерации не на последнем шаге, ка это принято, а несколько ранее (см. ниже 3.3.2).

Поскольку метод смещения ядра позволяет уменьшить его норму, возможности указанного итерационного уточнения возрастают. Итерационные методы для симметричных положительно определенных, но необязательно сжимающих, ядер (см. [23]) расширяют рамки применимости обсуждаемого подхода.

§2.2. ОПЕРАТОРЫ С ЯДРАМИ, ОПРЕДЕЛЯЕМЫМИ ИЗ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

2.2.1. В работах [5], [6] изучались уравнения второго рода (0.1) с ядрами, удовлетворяющими дифференциальным уравнениям вида (индекс при L снизу указывает переменную, по которой действует этот оператор) :

$$\mathcal{L}K \stackrel{def}{=\!=} L_x K(x,t) - L_t^{\#} K(x,t) = \int_\omega p(x,s)q(t,s)ds, \quad x,t \in D \subset R^m, \tag{2.3}$$

где в случае $(l \times l)$-матрицы $f(x) \in C^r(D)$, $r \geq 1$ принято

$$Lf \stackrel{def}{=\!=} \sum_{|\alpha| \leq r} A_\alpha(x) D^\alpha f(x), \qquad L^{\#} f \stackrel{def}{=\!=} \sum_{|\alpha| \leq r} (-1)^{|\alpha|} D^\alpha \left(f(x) A_\alpha(x)\right), \tag{2.3'}$$

$$D^\alpha = \partial^{|\alpha|} / \partial x_1^{\alpha_1} \cdots \partial x_m^{\alpha_m}, \quad \alpha = (\alpha_1, ..., \alpha_m), \quad \alpha = \alpha_1 + \alpha_2 + \cdots + \alpha_m,$$

а $p(x,s)$ и $q(t,s)$ кусочно-гладкие матрицы размеров $l \times l_1$ и $l_1 \times l$, соответственно, $x \in D$, $s \in \omega$, ω - κ-мерное компактное многообразие с мерой ds ($\kappa \leq m-1$, $m \geq 1$, а если $\kappa = 0$, то ω состоит из конечного числа точек).

Операторы L, $L^{\#}$, действующие в пространстве (l,l)-матриц, выбраны формально сопряженными и поэтому, при естественных ограничениях на границу ∂D, для $f, g \in C^r(D) \cap C^{r-1}(D)$ справедлива формула Грина-Стокса :

$$\int_D [gLf - (L^{\#} g) f] dx = \int_{\partial D} F_\sigma(f,g) d\sigma, \tag{2.4}$$

где $F_x(f,g)$ – соответствующая билинейная форма от $\xi = \{D^\alpha f\}$, $\eta = \{D^\beta g\}$, $|\alpha|, |\beta| \leq r-1$, F_σ – ее значения на границе ∂D с ориентированной мерой $d\sigma$.

Теорема 2.1. [5, 6] *Пусть D – ограниченная область с кусочно-гладкой границей и $K \in C^r(D \times D) \cap C^{r-1}(\bar{D} \times \bar{D})$ удовлетворяет уравнению (2.3). Если оператор $\mathcal{J} - \mathcal{K}$ обратим и R – его резольвентное ядро, то R имеет ту же гладкость и удовлетворяет уравнению*

$$\mathcal{L}_1 R \stackrel{def}{=\!=} \mathcal{L}R - \int_{\partial D} F_\sigma\{R(x,\sigma), R(\sigma,t)\}d\sigma = \int_\omega P(x,s)Q(t,s)ds, \qquad (2.5)$$

где $P(x,s)$ и $Q(x,s)$ определяются из уравнений $(\mathcal{J} - \mathcal{K})P = p(x,s)$ и $(\mathcal{J} - \mathcal{K}^\#)Q = q(x,s)$, см. (1.22).

Этот результат можно истолковать следующим образом : если известны значения $R(x,t)$ и его нормальных производных до $r-1$-го порядка при $x \in \partial D$ или $t \in \partial D$, а также $P(x,s)$ и $Q(t,s)$ при $s \in \omega$, то ядро резольвенты подлежит восстановлению во всей области $D \times D$ как решение дифференциального уравнения (2.5) с указанными граничными условиями.

Замечание 2.1. Если ядро R задано на многообразии без края (см. §1.2), то операторы $\mathcal{L}$ и $\mathcal{L}_1$ совпадают. В частности, при $p(x,s) \stackrel{def}{=\!=} 0$ ядро оператора $\mathcal{J} - \mathcal{K}$ и ядро его резольвенты $\mathcal{J} + \mathcal{R}$ являются решениями одного и того же дифференциального уравнения.

2.2.2. Рассмотрим одномерный случай, когда $K \in C^1\big((0,1) \times (0,1)\big) \cap C([0,1] \times [0,1])$, $D = (0,1)$. Предположим, что

$$\mathcal{L}K \stackrel{def}{=\!=} A(x)\frac{\partial K}{\partial x} + \frac{\partial K A(t)}{\partial t} + B(x)K - KB(t) = p(x)q(t), \quad 0 \le x,t \le 1, \qquad (2.6)$$

где $A(x) \in C^1$, $B(x), p(x), q(x) \in C$, K, A, B – $l \times l$-матрицы, p и q матрицы размеров $l \times l_1$ и $l_1 \times l$ соответственно.

Исходя из схемы непрерывного погружения п. 1.5.1, при $D = D' = (0,1)$, $D_\tau = (0,\tau)$ придем к следующему следствию из Теоремы 2.1 (см. также Теорему 1.1 и Замечание 1.3).

Теорема 2.2. [5, 6] *Если оператор $\mathcal{J} - \mathcal{K}_\tau$ обратим, а его ядро K удовлетворяет уравнению (2.6), то резольвентное ядро $R(x,t,\tau)$ является решением уравнения*

$$\begin{aligned}\mathcal{L}_1 R \stackrel{def}{=\!=}\ & \mathcal{L}R + R(x,\tau,\tau)A(\tau)R(\tau,t,\tau) - R(x,0,\tau)A(0)R(0,t,\tau) = \\ & = P(x,\tau)Q(t,\tau), \quad 0 \le x,t \le 1,\end{aligned} \qquad (2.7)$$

где $P(x,\tau)$ *и* $Q(x,\tau)$ *являются решениями уравнений* $(\mathcal{J}-K_\tau)P=p(x)$, $(\mathcal{J}-K_\tau^{\#})Q=q(x)$.

Введем теперь обозначения :

$$\varphi_1(x,\tau)=R(x,0,\tau),\qquad \varphi_3(x,\tau)=R(x,\tau,\tau),$$
$$\varphi_2(x,\tau)=R(0,x,\tau),\qquad \varphi_4(x,\tau)=R(\tau,x,\tau). \tag{2.8}$$

Теорема 2.3. *Пусть ядро* K *удовлетворяет уравнению (2.6). Тогда функции* $\varphi_k(x,\tau)$, $P(x,\tau)$, $Q(x,\tau)$ *и* $y(x,\tau)$ *удовлетворяют системе уравнений*

$$\frac{\partial y(x,\tau)}{\partial\tau}=\varphi_3(x,\tau)y(\tau,\tau),\quad \frac{\partial\varphi_1(x,\tau)}{\partial\tau}=\varphi_3(x,\tau)\varphi_1(\tau,\tau),$$
$$\frac{\partial\varphi_2(x,\tau)}{\partial\tau}=\varphi_2(\tau,\tau)\varphi_4(x,\tau),\quad \frac{\partial P(x,\tau)}{\partial\tau}=\varphi_3(x,\tau)P(\tau,\tau),$$
$$\frac{\partial Q(x,\tau)}{\partial\tau}=Q(\tau,\tau)\varphi_4(x,\tau),$$
$$\frac{\partial\varphi_3(x,\tau)A(\tau)}{\partial\tau}+A(x)\frac{\partial\varphi_3(x,\tau)}{\partial x}=\varphi_3(x,\tau)B(\tau)-B(x)\varphi_3(x,\tau)+$$
$$+\varphi_1(x,\tau)A(0)\varphi_2(\tau,\tau)+\varphi_3(x,\tau)[\varphi_2(\tau,\tau)A(\tau)-A(\tau)\varphi_3(\tau,\tau)]+P(x,\tau)Q(\tau,\tau),$$
$$A(\tau)\frac{\partial\varphi_4(x,\tau)}{\partial\tau}+\frac{\partial\varphi_4(x,\tau)A(x)}{\partial x}=\varphi_4(x,\tau)B(x)-B(\tau)\varphi_4(x,\tau)+$$
$$+\varphi_1(0,\tau)A(0)\varphi_2(x,\tau)+[A(\tau)\varphi_4(\tau,\tau)-\varphi_4(\tau,\tau)A(\tau)]\varphi_4(x,\tau)+P(\tau,\tau)Q(x,\tau). \tag{2.9}$$

Доказательство. Первые пять уравнений системы являются прямым следствием соотношений (1.14) и (1.14′). Поскольку доказательства шестого и седьмого уравнений вполне аналогичны, остановимся на последнем.

Согласно (2.8) имеем

$$A(\tau)\frac{\partial\varphi_4(x,\tau)}{\partial\tau}=A(\tau)\frac{\partial K(s,x,\tau)}{\partial s}\bigg|_{s=\tau}+A(\tau)\frac{\partial R(s,x,\tau)}{\partial\tau}\bigg|_{s=\tau},\qquad 0\le x,\tau\le 1.$$

Применив к первому слагаемому правой части формулу (1.14), а ко второму - (2.7) , получим последнее из уравнений (2.9).

2.2.3. Теорема 2.3 позволяет наметить формальную схему быстрого (по сравнению с общим методом Нистрема, см. §1.1) решения уравнения (0.1) (при $y(x)=y(x,N)$). Для этого надо решить систему (2.9) с начальными условиями

$$\varphi_1(x,0)=\varphi_3(x,0)=K(x,0),\quad \varphi_2(x,0)=\varphi_4(x,0)=K(0,x),$$
$$Q(x,0)=q(x),\quad P(x,0)=p(x),\quad y(x,0)=f(x). \tag{2.9′}$$

Основная проблема заключается в построении корректной разностной схемы для приближенного решения этой системы.

Прежде всего отметим, что любая формальная разностная схема, основанная на дискретизации $0 = \tau_0 < \tau_1 < \cdots < \tau_{N-1} < \tau_N = 1$, $0 \leq x_0 < x_1 < \cdots < x_{N'}$ будет иметь сложность $O(NN')$, $N, N' \to \infty$. Поэтому алгоритм, построенный на ее основе, может считаться быстрым (при $N = N'$ имеем сложность $O(N^2)$, в то время, как метод Нистрема требует выполнения $O(N^3)$ операций, см. §1.1).

Однако и при сомнениях в устойчивости схемы ее реализация не лишена смысла. Дело в том, что нашей конечной целью является нахождение не всех решений системы (2.9), а только функции $y(x, N)$, процесс нахождения которой можно попытаться стабилизировать. Например, при $||K|| < 1$ значение $y(x, N)$, полученное ценой малых затрат порядка $O(NN')$, может быть использовано как первое приближение в итерационном процессе (см. 2.1.1). Сама неустойчивая вычислительная схема также может быть частично стабилизирована (см. 3.4.2).

2.2.4. Рассмотрим теперь вопрос о корректной постановке задачи (2.9) – (2.9′). Нетрудно заметить, что устойчивость схемы 2.2.2 зависит от свойств оператора $\mathcal{L}_o$, действующего в пространстве $(l \times l)$-матриц-функций по формуле :

$$\mathcal{L}_o u \stackrel{def}{=\!=} A(x)\frac{\partial u}{\partial x} + \frac{\partial u}{\partial \tau}A(\tau), \quad 0 < x, \tau < 1. \tag{2.10}$$

Исходя из известных результатов теории уравнений в частных производных (см., например, [36]), корректность постановки начальной задачи (2.9′) для системы (2.9) может быть обеспечена лишь в случае строгой τ-гиперболичности оператора (2.10). Оператор (2.10), определен в пространстве l^2-векторов. Если элементы вектора $v^{tr} = (v_1, v_2, ..., v_{l^2})$ считать построчной разверткой элементов матрицы $u = ||u_{ij}||$; $i, j = 1, ..., l$, то оператор $\mathcal{L}_o$ примет вид

$$\mathcal{L}'_o v = G(x)\frac{\partial v}{\partial x} + H(\tau)\frac{\partial v}{\partial \tau}, \tag{2.10′}$$

где $(l^2 \times l^2)$-матрица G имеет $l \times l$ - блочно-диагональный вид : $G(x) = \\ = \mathrm{diag}(A(x), ..., A(x))$, а $(l^2 \times l^2)$-матрица $H = ||H_{ij}||$ состоит из $l \times l$-блоков :

$H_{ij}(\tau) = \mathrm{diag}(A_{ji}(\tau), ..., A_{ji}(\tau))$, $A = ||A_{ij}||$, $i, j = 1, 2, ..., l$.

Теорема 2.4. *Пусть выполнены условия Теоремы 2.3 и корни уравнения*

$$\det\big(H(\tau)+\xi G(\tau)\big)=0 \tag{2.11}$$

действительны, различны и отличны от нуля, тогда задача (2.9) – (2.9′) поставлена корректно.

Доказательство. Гладким неособым преобразованием $v(x,\tau)=\alpha(x,\tau)w(x,\tau)\times$ $\times\beta(x,\tau)$ главная часть оператора $\mathcal{L}_o'$ преобразуется к виду

$$\mathcal{L}_o''=\frac{\partial w(x,\tau)}{\partial\tau}+\lambda_k(x,\tau)\frac{\partial w(x,\tau)}{\partial x},$$

где $\lambda_k\neq 0$, $\lambda_k\in C$, $x,\tau\in[0,1]$ (см., например, [36]), что обеспечивает корректность постановки задачи (2.9) – (2.9′).

Замечание 2.2. Условие несовпадения корней уравнения (2.11) обеспечивает нужную гладкость α,β и λ_k и не является необходимым, в отличие от требований действительности и отличия от нуля для $\{\lambda_k\}$.

Теорема 2.5. *Пусть $A(x)$ – симметричная (эрмитова) невырожденная в $\bar{D}$. матрица. Тогда задача (2.9) – (2.9′) корректна.*

Доказательство. Имеет место следующее представление :

$$A(x)=U(x)\Lambda(x)U^*(x),\qquad x\in[0,1]. \tag{2.12}$$

где матрица $U(x)$ ортогональная ($U^*=U^{-1}$), а $\Lambda(x)=\mathrm{diag}(\lambda_1(x),\ldots,\lambda_l(x))$, $\mathrm{Im}\,\lambda_k=0$, $\lambda_k\neq 0$, $x\in[0,1]$, $k=1,2,\ldots,l$. Главная часть оператора (2.10) после невырожденной замены $v(x,\tau)=U^*(x)u(x,\tau)U(\tau)$ покомпонентно принимает вид

$$(\mathcal{L}_o u)_{ij}=\lambda_i(x)\frac{\partial v}{\partial x}+\lambda_j(\tau)\frac{\partial v}{\partial\tau}.$$

Существование решения этой системы обеспечивается условием обратимости операторов $\mathcal{I}-\mathcal{K}_\tau$, при $\tau\in[0,1]$, единственность и устойчивость относительно начальных данных выводятся стандартно.

Хотя последний результат уступает по общности Теореме 2.4, его условия гораздо более обозримы (см. Замечание 2.2).

Пример 2.1. В случае $A = \mathrm{diag}(\lambda_1, ..., \lambda_l)$, $\lambda_k = \mathrm{const} \neq 0$, $k = 1, 2, ..., l$ матрица K имеет вид

$$K(x,t) = ||K_{ij}(\lambda_j x - \lambda_i t)||; \qquad i,j = 1,2,...,l, \quad 0 \leq x,t \leq 1 \tag{2.13}$$

и все условия корректности Теоремы 2.5 выполнены. Такие ядра были изучены в [5, 6], см. также [39].

2.2.5. Наше доказательство Теоремы 2.4 не указывает практический путь построения устойчивого алгоритма (2.9) – (2.9′). Отнюдь не просто численно осуществить переход от V к W, даже имея явный вид корней уравнения (2.11).

В условиях Теоремы 2.5 устойчивую конечно-разностную схему для задачи (2.9) – (2.9′) в случае, когда оператор (2.10) гиперболический, можно построить, имея информацию об асимптотическом поведении наименьшего и наибольшего собственных значений $\{\lambda_k\}$ матрицы $G(x)H^{-1}(\tau)$. Этот подход состоит из правильного выбора отношения h_τ/h_x, где h_τ и h_x - шаги равномерной сети по τ и x, соответственно. При этом, как правило, $h_\tau \ll h_x$. Например, в работе [37] содержится семейство устойчивых конечно-разностных схем с точностью четвертого порядка по h_x и второго – по h_τ, при условии

$$h_\tau \leq 0.73 \max_{x,\tau,i,j} |\frac{\lambda_i}{\lambda_j}| h_x, \quad 0 \leq x,t \leq 1; \quad i,j = 1,2,...,l. \tag{2.14}$$

Замечание 2.3. Даже в более простых случаях необходимо объяснить теоретическую устойчивость схем и проверить оценки типа (2.14). Например, в случае ядра (2.13), когда по крайней мере одна из дробей λ_i/λ_j иррациональна, непросто построить сеть дискретизации, согласованную с характеристиками $\lambda_j x - \lambda_i t = \mathrm{const}$, $\mathrm{Im}\ \lambda_k = 0$, $i,j,k = 1,2,...,l$. Очевидно, что при $l = 2$, $\lambda_1 = 1$, $\lambda_2 = m$, ($m \geq +2$ - целое) минимальная сеть должна удовлетворять соотношению $h_\tau = \frac{1}{m} h_x$, что близко к (2.14), см. также ниже 3.4.2.

§2.3. ОПЕРАТОРЫ ЦИРКУЛЯНТНОГО ТИПА

2.3.1. Рассмотрим теплицево скалярное ядро K, $K(x,t) = K(x-t)$ такое, что $K(x+1) = K(x)$, $|x| \leq 1$. Это означает, что в уравнении (0.1) отрезок $D = [0,1]$ можно рассматривать, как единичную окружность. В этом случае

метод Нистрема для гладких ядер K даже в общей форме является быстрым, см. §§1.1, 1.2.

С другой стороны, при $K(x) \in C^p([0,1])$, $p \geq 2$ коэффициенты $\{k_n\}$ ряда Фурье

$$K(x) = \sum_{n=-\infty}^{+\infty} k_n e^{2i\pi nx}, \qquad x \in [0,1] \tag{2.15}$$

имеют порядок не менее $O(|n|^{-p})$, $|n| \to \infty$. Если теперь в (0.1) $f(x) \in C^p$, то ряд Фурье ($D = [0,1]$, $l = 1$, $f(x) = \sum f_n \exp[\theta]$)

$$y(x) = \sum_{n=-\infty}^{+\infty} y_n e^{2i\pi nx}, \qquad y_n = \frac{f_n}{1 - k_n}, \quad x \in [0,1] \tag{2.15'}$$

является решением уравнения, а условие обратимости оператора $\mathcal{J} - \mathcal{K}$ имеет вид $1 - k_n \neq 0$, $n = 0, \pm 1, \pm 2, \ldots$. Очевидно, что при этом $y_n = O(|n|^{-p})$, $|n| \to \infty$.

Соотношение (2.15') является прямым аналогом формулы обращения невырожденной циркулянтной матрицы, (см. [38]). Поэтому оператор $\mathcal{J} - \mathcal{K}$ с периодическим ядром (2.15) естественно назвать оператором циркулянтного типа.

Схема (2.15) – (2.15') допускает применение быстрого преобразования Фурье (БПФ, см. [38]). Весьма эффективны также приводимые ниже быстрые алгоритмы (см. 2.4.1, 3.4.1).

2.3.2. Покажем теперь, что понятие оператора циркулянтного типа естественным образом обобщается и на ядра вида (2.3).

Пусть гладкая функция $K(x,t)$, заданная на многообразии без края $D \times D$, $D \subset R^m$ удовлетворяет уравнению (2.3), при $p, q \equiv 0$. Рассмотрим следующие взаимно-сопряженные задачи на собственные значения :

$$\begin{aligned} L\Phi(x) &= \lambda\Phi(x), \\ L^{\#}\Phi^{\#}(x) &= \bar{\lambda}\Phi^{\#}(x), \end{aligned} \qquad x \in \bar{D}. \tag{2.16}$$

Предположим, что спектр оператора L дискретен и обозначим через $\{\lambda_k\}$, $k = 1, 2, \ldots$ собственные значения, а через $\Phi_k(x), \Phi_k^{\#}(x)$ – соответствующие собственные функции. Очевидно, что при соответствующей нормировке система $\{\Phi_k(x), \Phi_k^{\#}(x)\}$ биортогональна в D. Если теперь ядро $K(x,t)$ и свободный член $f(x)$ заданы в виде разложений

$$K(x,t) = \sum_{p=1}^{\infty} k_p \Phi_p(x) \Phi_p^{\#}(t), \qquad f(x) = \sum_{p=1}^{\infty} f_p \Phi_p(x), \qquad x \in \bar{D}. \tag{2.17}$$

то нетрудно убедиться, что решение уравнения (0.1) формально записывается в виде

$$y(x) = \sum_{p=1}^{\infty} y_p \Phi_p(x), \qquad y_p = (E_l - K_p)^{-1} f_p, \tag{2.17'}$$

а условием обратимости оператора $\mathcal{J} - \mathcal{K}$ является невырожденность матриц $E_l - K_p$, $p = 1, 2, \ldots$.

Если система $\{\Phi_p(x)\}$, $p = 1, 2, \ldots$ полна в $C(\bar{D})$, то вопросы сходимости разложений (2.17) и (2.17′) решаются стандартно.

2.3.3. Пусть теперь гладкое ядро K удовлетворяет на многообразии без края $\bar{D} \times \bar{D}$ уравнению (2.3), при $p, q \neq 0$. Поставим себе целью обобщить и на этот случай понятие оператора циркулянтного типа так, чтобы решение уравнения (0.1) снова записывалось в виде явного разложения по системе $\{\phi_k(x)\}$.

Предположим, что

$$p(x, s) = \sum p_k(s)\Phi_k(x); \qquad q(t, s) = \sum \Phi_k^{\#}(t) q_k(s), \quad s \in \omega. \tag{2.18}$$

Для того, чтобы выполнялось соотношение (2.3) необходимо потребовать, чтобы выполнялись соотношения

$$\int_{\omega} p_k(s) q_k(s) ds = 0, \qquad k = 1, 2, \ldots \tag{2.18'}$$

и

$$K(x, t) = \sum_{n=-\infty}^{+\infty} k_n \Phi_n(x) \Phi_n^{\#}(t) + \sum_{\alpha \neq \beta} (\lambda_\alpha - \lambda_\beta)^{-1} \Phi_\alpha(x) \int_{\omega} p_\alpha(s) q_\beta(s) ds \Phi_\beta^{\#}(t). \tag{2.19}$$

Предположим теперь, что множество индексов $\{n\}_1^{\infty}$ разбито на две части $\{n\} = \{n'\} \cup \{n''\}$, $\{n'\} \cap \{n''\} = \emptyset$ так, что

$$p_\alpha(s) \overset{def}{=\!=} 0, \quad \alpha \in \{n'\}; \qquad q_\beta(s) \overset{def}{=\!=} 0, \quad \beta \in \{n''\}. \tag{2.20}$$

Прямая проверка показывает, что уравнения (0.1) выписываются в виде

$$y(x) = \sum_{n=1}^{\infty} y_n \Phi_n(x), \tag{2.20'}$$

$$y_n = (E_l - k_n)^{-1} \Big[E_l + \sum_{\alpha \neq n} (\lambda_n - \lambda_\alpha)^{-1} (E_l - k_n)^{-1} \int_{\omega} p_n(s) q_\alpha(s) ds \Big] f_n.$$

Пример 2.2. Рассмотрим простейший одномерный случай $m = 1$, когда скалярное ($l = 1$) ядро K уловлетворяет на торе $[0,1] \times [0,1]$ уравнению (0.2) с гладкими на окружности $\exp(2\pi i x)$, $x \in [0,1]$ скалярными функциями $p(x)$ и $q(x)$. Например, если

$$p(x) = \sum_{0}^{\infty} p_n e^{\theta}, \qquad q(x) = \sum_{-\infty}^{-1} q_n e^{\theta},$$

где

$$\sum_{0}^{\infty} |p_n| < \infty, \qquad \sum_{-\infty}^{-1} |q_n| < \infty,$$

то все вышеприведенные условия выполнены. В этом случае ядро (0.2) представляется в виде

$$K(x,t) = \sum_{n=-\infty}^{+\infty} k_n e^{2\pi i n(x-t)} + \frac{1}{2\pi i} \sum_{\alpha=0}^{\infty} \sum_{\beta=-\infty}^{-1} \frac{p_\alpha q_\beta}{\alpha - \beta} e^{2\pi i(\alpha x - \beta t)}$$

Если $1 - k_n \neq 0$, $n = 1, 2, ...$, то оператор $\mathcal{J} - \mathcal{K}$ обратим, а решение $y(x)$ выписывается в явном виде (2.20) при $E_l = 1$, $\Phi_n(x) = \Phi_n^{\#}(-x) = \exp[2\pi i n x]$, $\lambda_n = 2\pi i n$ $(n = 0, \pm 1, \pm 2, ...)$.

Замечание 2.5. Дискретность спектра оператора L (2.16) несущественна, как и ограниченность области D. В качестве примера можно привести оператор свертки $K = K(x-t)$, $K(x) \in L_{1,2}(D)$, $D = R^1$. Здесь спектр оператора $L = i\frac{\partial}{\partial x}$ с условиями ограниченности на бесконечности заполняет всю ось R^1, а роль рядов Фурье выполняет преобразование Фурье. Нетрудно на этот случай обобщить и пример (2.2), предположив, что $p(x)$ и $q(t)$ – граничные значения аналитических, соответственно, в верхней и нижней полуплоскости, функций.

§2.4. УРАВНЕНИЯ ТЕПЛИЦЕВА И ГАНКЕЛЕВА ТИПОВ

2.4.1. Оператор с ядром (0.2) является простейшим обобщением теплицева оператора с ядром $K = K(x-t)$. Наметим для него схему быстрых одношаговых алгоритмов, основанных на методе дискретного погружения.

Теорема 2.6. *Пусть* $D = D^1 = (0,1)$ *и операторы* $\mathcal{J} - \mathcal{K}_\tau$ *обратимы при* $\tau = \tau_k \in [0,1]$, $k = 0, 1, ..., N$ *с резольвентными ядрами* $R(x,t,k) = R(x,t,\tau_k)$. *Тогда*

$$R(x,t,n+1) = R(x-h, t-h, n) +$$

$$+\int_0^h [R(x,s,n+1)R(s-h,t-h,n)+P(x,s,n+1)Q(s-h,t-h,n)]ds; \quad (2.21)$$

$$\tau_{n+1}-1 \le x,t \le 1, \quad |x-t| \le 1,$$

где $h=\tau_{n+1}-\tau_n$, *а* $P(x,t,\tau), Q(x,t,\tau)$ *являются решениями уравнений*

$$P(x,t,\tau)=p(x-t)+\int_0^\tau K(x,s)P(s,t,\tau)ds,$$
$$Q(x,t,\tau)=q(t-x)+\int_0^\tau Q(x,s,\tau)K(s,t)ds. \quad (2.21')$$

Доказательство. Из уравнения (0.2) вытекает представление

$$K(x,t)=K_o(x-t)+\int_0^{\min(x,t)} p(x-s)q(t-s)ds, \quad (2.22)$$

откуда имеем

$$K(x,t)-K(x-h,t-h)=\int_0^h p(x-s)q(t-s)ds. \quad (2.22')$$

Применив эту формулу в известном представлении для резольвентного ядра $R(x,t,\tau)$ (см. (1.4)), получим

$$R(x,t,n+1)-R(x-h,t-h,n)=$$

$$=\int_0^h p(x-s)q(t-s)ds+\int_0^h K(x-h,u-h)R(u-h,t-h,n)du+$$
$$+\int_0^{\tau_{n+1}}\int_0^h p(x-u)q(s-u)duR(s-h,t-h,n)ds+$$
$$+\int_0^{\tau_{n+1}} K(x,s)|R(s,t,n+1)-R(s-h,t-h,n)]ds.$$

Формула (2.21) при $h \le x,t \le 1$, $|x-t| \le 1$ получается посредством применения к обеим частям оператора $\mathcal{J}-K_\tau$ при $\tau=\tau_{n+1}$.

Покажем теперь, что в (2.21) можно считать $\tau_{n+1}-1 \le x,t \le 1$. Продолжив $p(x)$ и $q(t)$ на всю ось R^1 (хотя бы нулем) и используя формулу (2.21)(см. (2.22)), можно вычислить ядро $K(x,t)$ с высокой точностью во всей полосе $|x-t| \le 1$. Остается применить Теорему 1.2 (см. также Пример 1.2).

2.4.2. Формула (2.21) позволяет построить быстрый алгоритм любого порядка точности для решения уравнения с ядром (0.2). Для этого надо иметь в виду и формулы (1.10), (1.10′).

Остановимся на случае постоянного шага $h = 1/N$, $\tau_n = nh$, $n = 0, 1, ..., N$. Применим к вычислению $y(x, n)$ по (1.10′) квадратурную формулу с узлами в точках s_k, $k = 1, 2, ..., p$ (обычно $s_k \in [nh, (n+1)h]$). Достаточно вычислять лишь значения $R(x, s_k, n+1)$, где s_k зависят от n. Эти значения находятся из формулы (2.21) при $t = s_k$, но для этого надо знать величины, вычисляемые посредством применения квадратурных формул с узлами в точках $s = t_i$, $i = 1, 2, ..., q$ к интегралу в (2.21) (как правило, $t_i \in [0, h]$). Если величины $s_k - nh$, $t_i - h$ при всех k, i не зависят от n, то достаточно использовать формулы типа (1.10) и (1.10′).

Пример 2.3. Приведем один из лучших быстрых параллельных алгоритмов порядка $O(h^4)$ ($K, f \in C^4$), основанный на квадратурных формулах Гаусса. Для этого в случае теплицева ядра $K = K(x - t)$ введем пять матриц

$$y(x, \tau), \quad \Phi_1(x, \tau) = R(-\theta, x, \tau), \quad \Phi_2(x, \tau) = R(x, \theta, \tau),$$
$$\Psi_1(x, \tau) = R(\theta - 1, x, \tau), \quad \Psi_2(x, \tau) = R(x, 1 - \theta, \tau), \quad . \tag{2.23}$$

где сеть по τ равномерна ($\tau_n = nh$, $h = 1/N$, $n = 0, 1, ..., N$), а по x используются точки $x = \pm\theta, \pm(h - \theta), ..., \pm(Nh - \theta)$, $\theta = \frac{3-\sqrt{3}}{6}h$.

Если $l = 1$ (см. ниже Замечание 2.6), то на каждом шаге вычисляется y при $x = \{\theta, h - \theta, ..., 1 - \theta\}$, а векторы Φ_1, Φ_2, Ψ_1 и Ψ_2 имеют переменную длину : $x = \{-1 + nh + \theta, -1 + nh + h - \theta, ..., -\theta, \theta, h - \theta, ..., 1 - \theta\}$.

Общая сложность алгоритма равна $14N^2 + 15N$ скалярных мультипликативных операций. Необходимая оперативная память – $34N$ ячейки. Напомним, что этот алгоритм – неявного типа (см. п. 1.4.5).

Применение метода непрерывного погружения без труда приводит к следующему результату.

Теорема 2.7. [3, 28] *Пусть $D_\tau = [0, \tau)$ и операторы $\mathcal{J} - \mathcal{K}_\tau$ обратимы при всех $\tau \in (0, 1]$. Тогда (см. (2.21′)) $R(x, t, \tau)$ удовлетворяет уравнению*

$$(\frac{\partial}{\partial x} + \frac{\partial}{\partial t} + \frac{\partial}{\partial \tau})R(x, t, \tau) = R(x, 0, \tau)K(0, t, \tau) + P(x, 0, \tau)Q(0, t, \tau), \tag{2.24}$$
$$\tau - 1 \le x, t \le 1, \qquad |x - t| \le 1.$$

Несмотря на то, что на основе этой теоремы можно строить многошаговые алгоритмы, их характеристики, как правило, уступают приведенным выше одношаговым.

Замечание 2.6. В случае скалярного теплицева ядра $K = K(x-t)$ вместе с соотношениями Теорем 2.6 и 2.7 можно использовать свойство персимметричности $R(x,t,\tau) = R(\tau-t,\tau-x,\tau)$, упрощающее структуру алгоритмов [16, 28].

2.4.3. Обсуждаемые выше подходы применимы и в случае ядер, удовлетворяющих уравнениям второго порядка. В качестве примера рассмотрим ядро $K(x,t)$, удовлетворяющее однородному уравнению (0.3), при $p,q \equiv 0$, $K \in C^p$, $A \in C^{p-2}$, $p \geq 2$. В этом случае резольвентное ядро оператора $\mathcal{J} - \mathcal{K}_\tau$ (см. 2.2.1 и [5]) удовлетворяет соотношению

$$\begin{aligned}&(\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial t^2})R(x,t,\tau) + A(x)R(x,t,\tau) - R(x,t,\tau)A(t) = \\ &= \left\{\frac{\partial R(x,s,\tau)}{\partial s}R(s,t,\tau) - R(x,s,\tau)\frac{\partial R(s,t,\tau)}{\partial s}\right\}_{s=0}^{s=\tau} \overset{def}{=\!=} F(x,t,\tau).\end{aligned} \tag{2.25}$$

Пусть теперь операторы $\mathcal{J} - \mathcal{K}_\tau$ обратимы на равномерной τ-сети ($\tau = \tau_n = nh$, $h = 1/N$, $n = 0,1,...,N$). Применив известную формулу Даламбера для уравнения колебания струны, получим

$$\begin{aligned}&2R(x,t,n) = R(x-t+nh,nh,n) + R(x+t-nh,nh,n) + \\ &+\int_{x-t+nh}^{x+t-nh}\left.\frac{\partial R(u,s,n)}{\partial s}\right|_{s=nh}du + \int_{nh}^{t}\int_{x-t+v}^{x+t-v}F(u,v,n)dudv,\end{aligned} \tag{2.25'}$$

где $R(x,t,n) = R(x,t,\tau_n)$, $\quad nh \leq t \leq (n+1)h$.

Схема построения устойчивого быстрого алгоритма порядка $O(h)$ в случае симметричного ядра $K^{tr}(x,t) = K(t,x) \in C^2$ выглядит следующим образом :

1. Применением формулы нижних прямоугольников к (1.10), (1.10′) получим, с погрешностью порядка $O(h^2)$

$$\frac{\partial^i R(x,t,n+1)}{\partial t^i} = \frac{\partial^i R(x,t,n)}{\partial t^i} + hR(x,n,n)\frac{\partial^i K(n,t,n+1)}{\partial t^i}, \quad 0 \leq x,t \leq 1, \quad i = 0,1 \tag{2.26}$$

$$y(x,n+1) = y(x,n) + hR(x,n,n)y(n+1,n). \tag{2.26'}$$

2. Первое слагаемое справа в формуле (2.26) при $t = (n+1)h$ заменим по формуле (2.25′) при том же t, применив также формулу прямоугольников.

3. Значения $\frac{\partial^i}{\partial t^i}R(x,t,n+1)$, $t = (n+1)h$ будем вычислять по (2.26).

В результате (см. [7]) придем к рекуррентному алгоритму вычисления функций $y(x,n)$ и векторов

$$\Phi_1(x,n) = R(x,n,n), \quad \Phi_2(x,n) = R(x,0,n),$$
$$\Phi_3(x,n) = R(N,x,n), \quad \Phi_4(x,n) = \frac{\partial}{\partial t}R(x,t,n)\Big|_{t=n},$$
$$\Phi_5(x,n) = \frac{\partial}{\partial t}R(x,t,n)\Big|_{t=0}, \quad \Phi_6(x,n) = \frac{\partial}{\partial x}R(0,x,n), \qquad (2.27)$$
$$\Phi_7(x,n) = \frac{\partial}{\partial x}R(N,x,n), \quad \Phi_8(x,n) = \frac{\partial}{\partial x}R(x,n,n),$$
$$\Phi_9(x,n) = \frac{\partial}{\partial t}R(t,x,n)\Big|_{t=N},$$

при очевидных стартовых значениях (n=0). Сложность этого алгоритма - - $20N^2 + O(N)$ мультипликативных $l \times l$-матричных операций, оперативная память - $20(N+l)l$ ячеек.

Замечание 2.7. При $a(x) \equiv 0$, K является ядром "Теплиц плюс Ганкель" и алгоритм упрощается. Оператор $\mathcal{L}$ высокого порядка, к сожалению, гиперболическим не может быть, однако и в этом случае. (см. ниже Пример 2.4) быстрые устойчивые алгоритмы иногда могут быть построены.

2.4.4. Пусть теперь $l \times l$-ядро K зависит от параметра $\kappa \in (\alpha,\beta)$ $(K = K(\kappa,x,t))$ и удовлетворяет уравнению

$$\epsilon(\kappa)\frac{\partial K}{\partial \kappa} = A(\kappa,x)\frac{\partial K}{\partial x} + \frac{\partial K\, A(\kappa,t)}{\partial t} + B(\kappa,x)K - K\,B(\kappa,t) \stackrel{def}{=\!=} \mathcal{L}K, \qquad (2.28)$$

$$x,t \in (a,b) \subset R^1, \qquad \kappa \in (\alpha,\beta) \subset R^1,$$

где $\epsilon(\kappa)$ - комплекснозначная скалярная функция.

Теорема **2.8.**(Об эволюции резольвентного ядра). *Пусть $K \in C^1$ и удовлетворяется уравнение (2.28). Если оператор (0.1) при $D = (a,b) \subset R^1$ обратим для $\kappa \in (\alpha,\beta)$, то резольвентное ядро $R(\kappa,x,t)$ удовлетворяет уравнению*

$$\epsilon(\kappa)\frac{\partial R}{\partial \kappa} = \mathcal{L}R + \Big[R(\kappa,x,s)A(\kappa,s)R(\kappa,s,t)\Big]_{s=a}^{s=b}. \qquad (2.28')$$

(При $a = -\infty$ или $b = +\infty$ в формуле (2.28') соответствующие значения $R(x,t)$ считаются равными нулю.)

Доказательство аналогично доказательству Теоремы 2.2.

Этот результат позволяет при $\epsilon(\kappa) \neq 0$, с учетом уравнений типа (1.10), (1.10′), строить рекуррентный экономичный алгоритм обращения операторов $\mathcal{J} - \mathcal{K}_\tau$ $(K = K_\kappa)$ при $\kappa \in (a, b)$, начиная со стартовых значений $\kappa = \kappa_o \in (a, b)$.

Замечание 2.8. Нетрудно доказать аналогичный результат в случае оператора высокого порядка $\mathcal{L}$ (при $m = 1$).

Замечательно, что изучаемые подходы включают в себя известную схему Захарова-Шабада, устанавливающую связь между интегральными, ядра которых удовлетворяют уравнениям типа (2.28), и *солитонными* решениями нелинейных дифференциальных уравнений в частных производных (см. [40]).

2.4.5. Остановимся теперь на простейших уравнениях теплицева типа в многомерном случае. Предположим, что $K(x, t)$ имеет вид

$$K(x, t) = K(x', t', x_m - t_m); \qquad (x, t) \in \bar{D} \times \bar{D}, \quad D \subset R^m, \qquad (2.29)$$

где $x' = (x_1, x_2, ..., x_{m-1})$, $t' = (t_1, t_2, ..., t_{m-1})$. Пусть, далее $x \in D$, т.е. $0 < x_m < 1$. Рассмотрим (см. п. 1.5.4) $D = D_\tau = D \cap \{0 < x_m < \tau\}$.

Теорема 2.9. *Если* $K \in C(\bar{D} \times \bar{D})$ *и операторы (0.1)* $\mathcal{J} - \mathcal{K}_\tau$ $(D = D_\tau)$ *обратимы при всех* $\tau \in [0, 1]$, *то резольвентные ядра* $R(x, t, \tau) = R(x, t, D_\tau)$ *удовлетворяют уравнению*

$$\left(\frac{\partial}{\partial \tau} + \frac{\partial}{\partial x_m} + \frac{\partial}{\partial t_m}\right) R(x, t, \tau) = \int_{\Gamma_\tau} R(x, s, \tau) R(s, t, \tau) ds_1, \qquad (2.30)$$

$$x = (x', x_m), \quad t = (t', t_m), \quad \tau - 1 \leq x_m, t_m \leq 1, \quad |x_m - t_m| \leq 1,$$

где $s \in \Gamma_\tau = \partial D_\tau \setminus \{x_m = \tau\}$ *и* $ds = ds_1 \times ds_2 \times \cdots \times ds_{m-1}$ *ориентированная мера на поверхности* Γ_τ.

Доказательство. Дифференцируя формулу (1.4), где $D = D_\tau$, получим, по аналогии с доказательством Теоремы 2.1

$$\left(\frac{\partial}{\partial x_m} + \frac{\partial}{\partial t_m}\right) R(x, t, \tau) = \int_{\partial D_\tau} R(x, s, \tau) R(s, t, \tau) ds.$$

Учитывая теперь формулу (1.19), придем к (2.30).

Теорема 2.9 позволяет строить быстрые (по сравнению с методом Нистрема) многошаговые, вообще говоря, алгоритмы решения многомерного уравнения (0.1), экономящие одновременно память ЭВМ. Как и в одномерном случае, предпочтение необходимо отдать схеме дискретного погружения. Чтобы избежать некоторой геометрической громоздкости остановимся на области D в виде гиперкуба $\{0 < x_i < 1\}$, $i = 1, 2, ..., m$, т. е. примем $D_\tau = \{0 < x_i < 1,\ i = 1, 2, ..., m-1;\ 0 < x_m < \tau \leq 1\}$, $\Gamma_\tau = \bar{D}_o = \bar{D} \cap \{x_m = 0\}$.

Теорема 2.10. *Пусть операторы $\mathcal{J} - K_\tau$ $(D = D_\tau)$ обратимы на сети $\{\tau_n\}_0^N \subset$ $\subset [0, 1]$, $\tau_o = 0$, $\tau_N = 1$. Тогда для резольвентных ядер $R(x, t, \tau_n) = R(x', t', x_m, t_m, n)$ справедливы соотношения*

$$R(x', t', x_m, t_m, n+1) = R(x', t', x_m - h, t_m - h, n) + \int_0^h \int_{D'} R(x', s', x_m - h, u - h, n) R(s', t', u, t_m, n+1) ds' du, \tag{2.31}$$

$$x', t' \in D', \quad \tau_{n+1} - 1 \leq x_m, t_m \leq 1, \qquad |x_m - t_m| \leq 1,$$

где $D' = \{0 < x_i < 1,\ i = 1, 2, ..., m-1\}$, $ds' = ds_1 ds_2 \cdots ds_{m-1}$.

Доказательство проводится по схеме доказательства Теоремы 2.6.

2.4.6. Первый результат типа формулы (2.7) получил Г. Плачек в случае ядра $K = Ei(|x - t|)$ (1945). В случае общих теплицевых ядер $K = K(x - t)$ соответствующую формулу вывели В. В. Соболев (1958, $K = K(|x - t|)$), И.Ц. Гохберг и Г. Хайнинг [41] (1975, см. подробности в [5]).

Соотношение (2.23) было впервые получено Т. Кайлатом, Л. Льюнгом и М. Морфом ([3], 1978). В работах И. Ц. Гохберга, И. Колтрахта и П. Ланкастера [4, 48] на этой основе строились одношаговые алгоритмы первого и второго порядков точности.

Схема дискретного погружения, насколько нам известно, впервые в вычислительных целях использовалась в работах [14 – 17] (см. ниже п. 3.4.1).

Быстрые алгоритмы для многомерных уравнений ранее строились лишь в случае $m = 2$, $K = K(x_1, t_1, x_2 - t_2)$, $D = (a, b) \times (c, d)$ сведением уравнения (0.1) с ядром (2.29) к алгебраической системе с блочно-теплицевой матрицей $A = \|a_{i-j}\|$, $i, j = 1, 2, ..., N$; $a_k = \|a_k^{pq}\|$, $p, q = 1, 2, ..., N$, $k = 0, \pm 1, ..., \pm N$ (см. [40]).

§2.5. КОМПОЗИЦИЯ ЯДЕР И УРАВНЕНИЯ ПЕРВОГО РОДА

2.5.1 Следуя п. 2.2.1, обозначим

$$L_i f \stackrel{def}{=} \sum_{k-0}^{m_i} A_k^i(x) \frac{\partial^k f(x)}{\partial x^k},$$

$$L_i^{\#} f \stackrel{def}{=} \sum_{k=0}^{m_i} (-1)^k \frac{\partial^k}{\partial x^k}[f(x) A_k^i(x)], \tag{2.32}$$

$$\mathcal{L}_{ij} \stackrel{def}{=} L_{ix} - L_{jt}^{\#}, \qquad i,j = 1,2,$$

где $A_k^i, f \in C^{m_i}((a,b))$ – $(l \times l)$-матрицы, $i = 1,2$, $m_i \geq 1$.

Для двух интегральных операторов $\mathcal{K}_1 : C(c,d) \to C(a,b)$, $\mathcal{K}_2 : C(a,b) \to C(c,d)$ с ядрами $K_1(x,t)$ $(a \leq x \leq b, \quad c \leq t \leq d)$ и $K_2(x,t)$ $(c \leq x \leq d, \ a \leq t \leq b)$, соответственно, через $\mathcal{K}_{12} = \mathcal{K}_1 * \mathcal{K}_2$ обозначим оператор с композицией ядер

$$K_{12}(x,t) = \int_c^d K_1(x,s) K_2(s,t) ds, \quad a \leq x,t \leq b. \tag{2.33}$$

Теорема 2.11. *Пусть функции*

$$\frac{\partial^p K_i(x,t)}{\partial x^p}, \ \frac{\partial^q K_i(x,t)}{\partial t^q}; \ i = 1,2, \ p = 0,1,\ldots,m_1, \ q = 0,1,\ldots,m_2, \ m_1, m_2 \geq 1$$

непрерывны и

$$\mathcal{L}_{12} K_1(x,t) = p_1(x) q_1(t), \quad a \leq x \leq b, \quad c \leq t \leq d,$$
$$\mathcal{L}_{21} K_2(x,t) = p_2(x) q_2(t), \quad c \leq x \leq d, \quad a \leq t \leq b, \tag{2.34}$$

где p_i и q_i – кусочно-гладкие матрицы размеров $(l \times l_i)$ и $(l_i \times l)$, соответственно. Тогда ядра операторов $\mathcal{K}_{ij} = \mathcal{K}_i * \mathcal{K}_j$ удовлетворяют уравнениям

$$\mathcal{L}_{ii} K_{ij}(x,t) = p_{ij}(x) q_{ij}(t), \quad i,j - 1,2, \tag{2.34'}$$

где p_{ij}, q_{ij} – матрицы размеров $(l \times l_{ij})$ и $(l_{ij} \times l)$, соответственно $(l_{ij} \leq \leq l_i + l_j + 2l)$.

Доказательство проводится прямой проверкой на основе формулы интегрирования по частям. Отметим, что при $i = j$ формула (2.34′) совпадает с формулой (2.5) Теоремы 2.1 (при $m = 1$, $D = (\alpha, \beta)$).

Пример 2.4. (см. также Замечание 2.7 и Пример 4 из [5]). Рассмотрим гармоническое ядро $H(x,t)$, $b \le x,t \le a$ ($\Delta H = H_{xx} + H_{tt} = 0$). Это ядро удовлетворяет уравнению (2.1) четвертого порядка ($m = 1$)

$$\frac{\partial^4 H}{\partial x^4} - \frac{\partial^4 H}{\partial t^4} = 0, \quad a \le x,t \le b \tag{2.35}$$

и, вследствие нарушения условия гиперболичности у этого уравнения, не вписывается в устойчивые схемы §2.4.

Однако, если оператора $\mathcal{J} \pm K$ ($K = H$) обратимы, то уравнение $(\mathcal{J} - K_\tau)y = f$ эквивалентно уравнению $(\mathcal{J} - K_\tau * K)y = (\mathcal{J} + K)f$ и для оператора с ядром K_{12}, $K_1 = K_2 = H$ (см. (2.33)) имеем

$$(\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial t^2})K_{12} = p(x)q(t),$$

$$p(x) = \left\{ H(x,b), -H(x,a), -\left.\frac{\partial H(x,s)}{\partial s}\right|_{s=b}, \left.\frac{\partial H(x,s)}{\partial s}\right|_{s=a} \right\},$$

$$q(t)^{\mathrm{tr}} = \left\{ \left.\frac{\partial H(s,t)}{\partial s}\right|_{s=b}, \left.\frac{\partial H(s,t)}{\partial s}\right|_{s=a}, H(b,t), H(a,t) \right\}. \tag{2.35'}$$

Применив теперь уже к оператору $\mathcal{J} - K_{12}$ схему п. 2.4.3 придем к быстрому устойчивому алгоритму решения уравнения с ядром H (см. также Примеры 2 и 3 из [5]).

2.5.2. Теорема 2.11 позволяет применить методы §§2.3, 2.4 при решении широкого класса уравнений первого рода с ядрами более общего вида, чем описанные в §2.2. Рассмотрим уравнение

$$\int_0^1 K(x,t)y(t)dt = f(x), \quad a \le x \le b, \tag{2.36}$$

Соответствующее регуляризованное уравнение (см. Введение)

имеет вид

$$\alpha y(x) + \int_0^1 Q(x,t)y(t)dt = F(x), \quad 0 \le x \le 1, \tag{2.36'}$$

где $\alpha > 0$ и

$$Q(x,t) = \int_a^b \overline{K(s,x)}K(s,t)ds, \qquad F(x) = \int_a^b \overline{K(s,x)}f(x)ds, \tag{2.36''}$$

Из Теоремы 2.11 вытекает следующая

Теорема 2.12. *Если ядро K уравнения (2.36) удовлетворяет (см. (2.32)) соотношению*

$$\mathcal{L}_{ij}K(x,t) = p(x)q(t), \quad a \le x \le b, \quad 0 \le t \le 1, \tag{2.37}$$

то для ядра $Q(x,t)$ уравнения (2.36′) имеем

$$\mathcal{L}_{ii}Q(x,t) = p_i(x)q_i(t). \tag{2.37′}$$

Таким образом, в случае ядер вида (2.37) регуляризованное уравнение (2.36′) принадлежит классу, изученному в §2.4. Ядро $Q(x,t)$ – симметрично и неотрицательно определено. Поэтому оператор (2.36′) положительно определен и для него применимы методы как дискретного, так и непрерывного погружения. Сеть по $\{\tau_n\}$ можно выбрать любой, поскольку все операторы $\mathcal{J} + Q_\tau$, $0 < \tau \le 1$ обратимы.

Останавливаться на случае многомерных уравнений первого рода (0.5) мы не будем из-за громоздкости соответствующих формул. Отметим, тем не менее, что Теорема 2.12 может быть обобщена и на этот случай.

Пример 2.5. Пусть ядро K в (2.36) удовлетворяет уравнению теплопроводности

$$\frac{\partial K(x,t)}{\partial t} = \frac{\partial^2 K(x,t)}{\partial x^2}, \quad a \le x \le b, \quad 0 \le t \le 1. \tag{2.38}$$

В частности,(см., например, [36]) функция $K(x,t)$ может иметь вид

$$K(x,t) = \frac{1}{\sqrt{x}} \int_{-\infty}^{\infty} \exp[-\frac{(t-s)^2}{x}] d\rho(s), \qquad \int_{-\infty}^{\infty} |d\rho(s)| < +\infty, \tag{2.38}$$

$$0 < a \le x \le b \le +\infty, \quad 0 \le t \le 1.$$

В этом случае, согласно Теореме 2.12, ядро $Q(x,t)$ является ядром ганкелева типа, поскольку

$$\frac{\partial Q(x,t)}{\partial x} - \frac{\partial Q(x,t)}{\partial t} = p(x)q(t),$$

$$p(x) = \left\{ \overline{K(x,a)}, \overline{K(x,b)}, \left.\frac{\partial \overline{K(x,s)}}{\partial s}\right|_{s=a}, \left.\frac{\partial \overline{K(x,s)}}{\partial s}\right|_{s=b} \right\}, \tag{2.39}$$

$$q(t)^{\mathrm{tr}} = \left\{ \left.\frac{\partial K(s,t)}{\partial s}\right|_{s=a}, -\left.\frac{\partial K(s,t)}{\partial s}\right|_{s=b}, -K(a,t), K(b,t) \right\}.$$

Фактически Q – ядро типа (2.35′) (см. также Примеры 2 и 3 из [5]).

Замечание 2.9. Насколько нам известно, быстрые алгоритмы решения уравнений первого рода применялись лишь для уравнений в полных свертках ($K = K(x-t)$, $-\infty \le x, t \le +\infty$, см. [10]). Результаты данного пункта значительно расширяют эти возможности.

ГЛАВА III. АНАЛИЗ ЧИСЛЕННЫХ РЕЗУЛЬТАТОВ

§3.1. ТЕСТОВЫЕ УРАВНЕНИЯ И АЛГОРИТМЫ

3.1.1. На примерах некоторых тестовых уравнений мы продемонстрируем сравнительные возможности предложенных выше алгоритмов. Ядра интегральных уравнений и их решения подобраны так, чтобы свободный член явно выражался через элементарные функции.

В качестве меры ошибки вычислений на сети $\{x_1, x_2, ..., x_s\} \subset D$ выбрана среднеквадратичная погрешность

$$\delta(s) = \sqrt{\frac{1}{s}\sum_{k=1}^{s} |Y(x_k) - y(x_k)|^2}, \tag{3.1}$$

где $Y(x)$ – точное решение, а $y(x)(x = x_k)$ – решение, полученное в ходе численного эксперимента.

Одним из объективных критериев соответствия свойств применяемого алгоритма теоретическим выводам является эффективность экстраполяции по Ричардсону. С этой целью ряд алгоритмов просчитывался на равномерных , последовательно удваиваемых τ-сетях ($\tau_n = nh$, $h = 1/N$, $N = 2^k$, $k = 1, 2, ...$), и, если теоретическая точность была равна $O(h^p)$, $h \to 0$, то уточненное решение $\tilde{y}_{2N}$ строилось по формуле

$$\tilde{y}_{2N}(x) = \frac{2^p}{2^p - 1} y_{2N}(x) - \frac{1}{2^p - 1} y_N(x). \tag{3.2}$$

В случае доброкачественного алгоритма, при $K, f \in C^{p+1}$, погрешность $\delta_{2N} = \delta(N, 2N)$ решения $\tilde{y}_{2N}$ должна иметь порядок не менее $O(h^{p+1})$. По той же схеме иногда проводилась и вторая экстраполяция $\tilde{\tilde{y}}_{2N}$, основанная на результатах первой в общих точках соответствующих x-сетей (при $K, f \in C^{p+2}$).

Замечание 3.1. Сети для y_N и y_{2N} не всегда пересекаются. Например, в случае применения квадратурных формул Гаусса общих точек у них нет (см. п. 2.4.2, Пример 2.3). Добавление точек τ–сети вида $x_k = kh$ $(k = 0, 1, 2, ..., N)$ сильно усложнит алгоритм. Гораздо эффективней построить сплайны соответствующего порядка.

В Таблице 1 приведены параметры тестовых линейных уравнений второго рода с гладкими ядрами. Задачи P1 – P5 относятся к скалярным $(l = 1)$ одномерным $(D = [0, 1])$ уравнениям,задача P6 – к системе двух одномерных уравнений $(l = 2,\ D = [0, 1])$.

Кроме того, решается следующее нелинейное уравнение типа Гаммерштейна (Задача P8)

$$y(x) = \lambda \int_0^1 e^{x-t} y^3(t)dt + e^x, \qquad \lambda = 0.25(e^2 - 1)^{-1}, \quad 0 \le x \le 1. \tag{3.3}$$

Это уравнение имеет лишь три решения

$$y_1(x) = (\sqrt{5} - 1)e^x, \quad y_2(x) = (1 - \sqrt{5})e^x, \quad y_3(x) = 2e^x. \tag{3.3'}$$

3.1.2. Для обозначений применяемых в экспериментах вводится следующая аббревиатура.

Через (h^p) обозначим порядок применяемых квадратурных формул (h - максимальный шаг сетки). Заглавные буквы перед (h^p) обозначают :

NG – классический алгоритм на основе метода Нистрема с применением метода Гаусса с построчным выбором ведущего элемента (1.1.1).

A (или B) – алгоритмы для линейных уравнений с ядрами общего вида с применением метода дискретного (или непрерывного) погружения (§§1.4, 1.5).

B' – алгоритм типа B с коррекцией ошибки (см. Замечание 1.11 и 1.5.5).

AD (или BD) – быстрый алгоритм для уравнений с разностными ядрами $K = K(x - t)$ на основе п. 2.4.1 (или 2.4.2).

ADM – алгоритм типа AD в двухмерном случае (ядро (2.29)).

ABT – гибридный быстрый алгоритм с применением алгоритма типа A к вычислению y и типа B - к вычислению R [42], для ядер теплицева типа (0.2) (см. п. 2.4.2).

ATH – быстрый алгоритм на основе метода п. 2.4.3 для ядер (0.3).

AL – быстрый (вообще говоря, неустойчивый см. п. 2.2.3) алгоритм для 2×2-матричных ядер типа (2.13).

ALSt – устойчивый быстрый алгоритм для задач с такими ядрами (см. п. 2.2.5 и Замечание 2.3).

Буква C после (h^p) означает применение метода смещения (п. 2.1.2), а буква I – применение итерационного уточнения (п. 2.1.2).

В случае нелинейных уравнений обозначено :

Nt – метод Ньютона,

AN(h) – метод дискретного погружения п. 1.8.2 с применением на каждом этапе формулы (1.40) точности $O(h^2)$ и NG(h).

Замечание 3.2. В численных экспериментах испытывались полностью распараллеленые (векторизованные) алгоритмы. Исключение составляет лишь метод Гауссовой факторизации NG. Время вычисления в параллельном варианте (методом Гаусса-Жордана) приводится в скобках (Таблицы 5,7,10).

§3.2. МЕТОД АВТОМАТИЧЕСКОГО СГУЩЕНИЯ СЕТКИ

3.2.1. Назовем Алгоритмом NGI модифицированный подход работы [18] (см. п. 1.1.2), разрешающий алгебраическую систему универсальным методом NG(h^2).

Алгоритм NGII (см. п. 1.1.2) отличается от NGI следующей процедурой.

Фиксируются два числа a и b, $0 < a, b < 1$ и затем

1) если при $\delta_1 = 4a$, $\delta_2 = 4b$ оба неравенства (1.2) не выполняются, то j-тый интервал делится на 8 равных частей. В противном случае

2) если при $\delta_1 = 2a$, $\delta_2 = 2b$ оба неравенства (1.2) не выполняются, то интерпал $[x_j, x_{j+1}]$ делится на равные части,

3) если условия 1), 2) не выполнены и при $\delta_1 = a$, $\delta_2 = b$ нарушено хотя бы одно из условий (1.2), то интервал $[x_j, x_{j+1}]$ делится на две равные,

4) если условие (1.2) для всех j в случае 3, алгоритм заканчивает работу и выдает результат решения последней системы уравнений.

Обозначим, наконец, через NGIII алгоритм, отличающийся от NGII лишь тем, что на предпоследнем этапе вместо упомянутого перехода к последнему

применяется кубический сплайн по значениям $y(x)$ на уже сгущенной сети с использованием граничных значений $y'(0)$ і $y'(1)$ из (1.2'). Критерием завершения работы Алгоритма NGIII здесь выбрано априорное задание числа этапов.

Характер работы Алгоритмов NGII и NGIII иллюстрируется на Рисунке 1. Здесь мы видим, каким образом сгущение сети на первом (общем для NGII и III) этапе зависит от поведения графика точного решения на соответствующем отрезке.

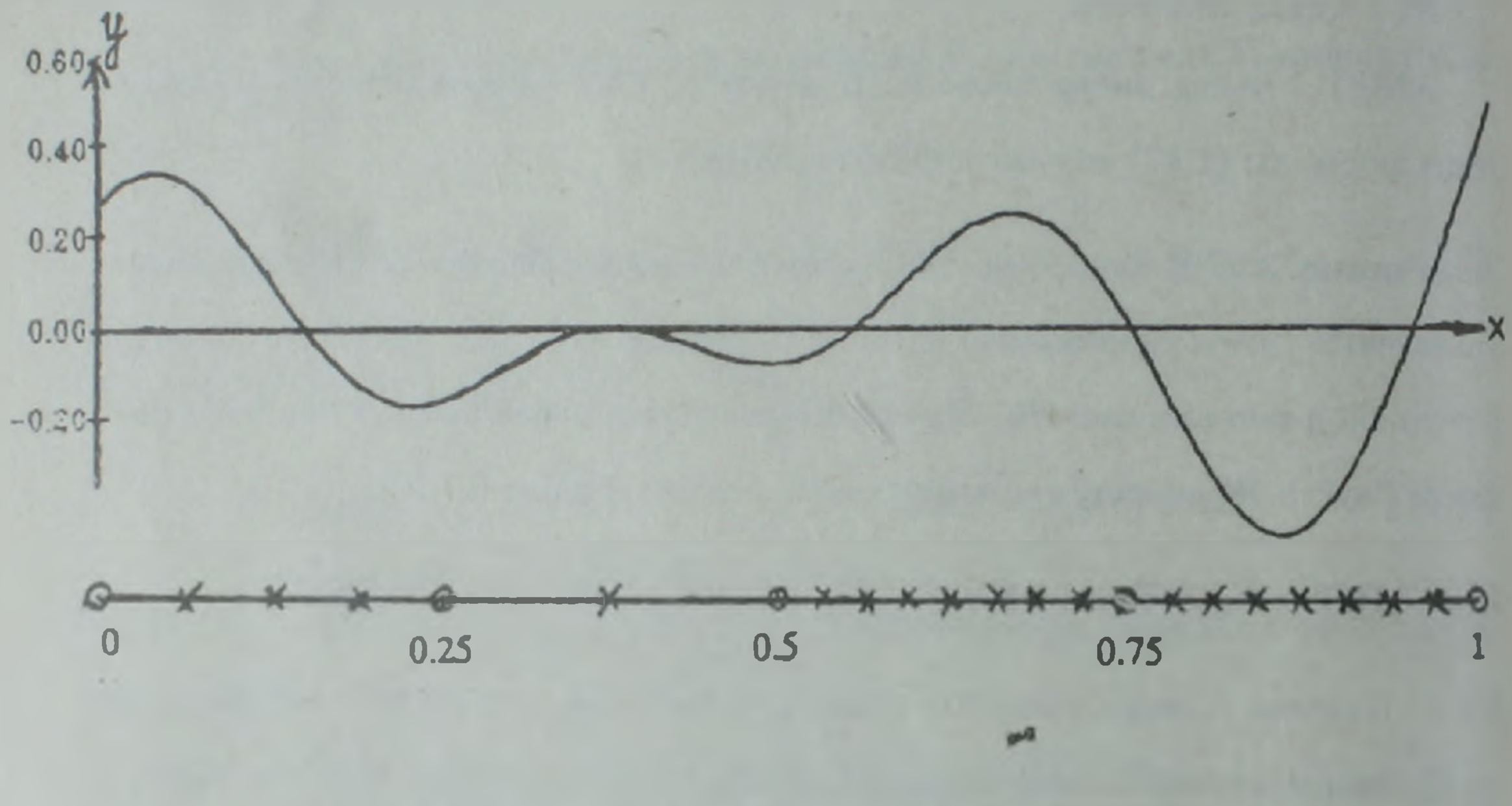


Рис. 1. Задача P1, $a = b = 0.5$. График точного решения и (внизу) схема результата автоматического сгущения сети по Алгоритмам NGII и NGIII на первом этапе. Кружками отмечена стартовая сеть на [0; 1], крестами – точки сгущения.

Такое реагирование на поведение $y(x)$ приводит к результатам, отраженным в Таблице 2 Приложения. Как видим, Алгоритм NGIII в применении к задаче P1 имеет преимущества перед NGII, а последний заметно лучше Алгоритма NGI.

Замечание 3.3. Значение сплайна определяется в любой точке отрезка [0, 1], а время его построения ничтожно по сравнению с временем решения алгебраических систем в NGI и NGII. В Таблице 2 ошибка при применении NGII и NGIII вычислялась в одних и тех же точках.

§3.3. МЕТОДЫ ПОГРУЖЕНИЯ ДЛЯ РЕШЕНИЯ УРАВНЕНИЙ ОБЩЕГО ВИДА

3.3.1. Свойства методов погружения §§1.4, 1.5, являющихся основой быстрых алгоритмов, были проверены на задачах P2, P3 и P9 (см. Приложение, Таблицы 3, 5, 10, 11).

Результаты подтверждают выводы теории о полном преимуществе алгоритмов типа A. Самым большим недостатком алгоритмов типа B является.его большая чувствительность к условию обратимости оператора $\mathcal{J} - \mathcal{K}_\tau$ при всех $\tau \in [0, 1]$. Знак ∞ =overflow в таблицах свидетельствует о том, что при нарушении этого условия переполнение происходит практически во всех случаях, когда N велико - даже если на выбранной τ-сети $\{\tau_n\}_1^N$ операторы $\mathcal{J} - \mathcal{K}_\tau$ обратимы (в этом случае алгоритм типа A почти не уступает алгоритму NG). Но даже при $\|K\| < 1$ накопление ошибок в алгоритмах типа B больше (Таблицы 3, 4). Скорректированный алгоритм типа B' несколько смягчает эти недостатки (Таблица 3).

3.3.2. Метод смещения (см. §1.7) представлен лишь в Таблице 5, Задача P2. Именно, в случае ядра $K = 2e^{-x^t}$ ($\|K\| = 1.63$) применена самая элементарная схема дискретного погружения (см. 1.7.4) с $p, q \equiv 1$, $D(n) = 0$ $(n = 0, 1, ..., N-1)$, $D(N) = 1.6$.

В результате фактически решается уравнение с ядром $\tilde{K} = K - 1.6$ в случае правых частей $f(x)$ и $p(x) \equiv 1$. Алгоритм A(h)C уменьшает накопление ошибок по сравнению с A(h). Увеличения времени счета практически нет. Так как $\|\tilde{K}\| = 0.31$, то дополнительная итерация A(h)GI была эффективной .

Диаграмма 1 подтверждает полезность итерационного уточнения. В задаче P3 ($\lambda = 1$) гораздо выгодней проводить итерационное уточнение при $\tau = 0.875$, чем традиционным способом, при $\tau = 1$. Отметим, что при малых нормах $\|K\|$ (например, при $\|K\| \leq 0.5$) положение как правило, меняется.

3.3.3. Таблица 11 подтверждает эффективность идеи дискретного погружения и при решении нелинейных уравнений. Пока что речь идет лишь о выделении одного (какого-либо) решения Задачи P8, в то время как классический метод Ньютона-Канторовича позволяет найти все решения про постоянной начальной

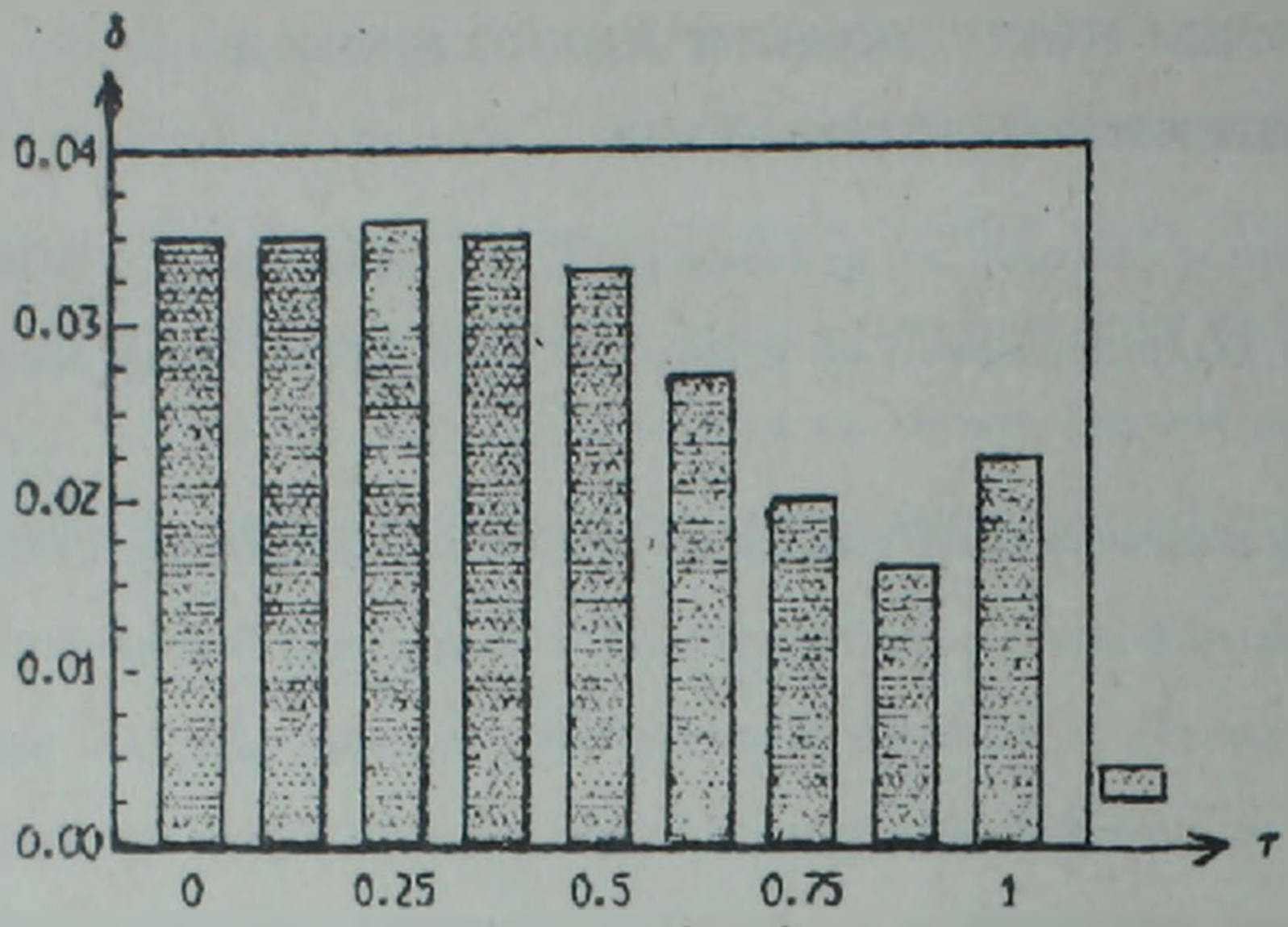


Диаграмма 1. Задача P3, $\lambda = 1$. Ошибка $\delta = \delta(\tau)$ Алгоритма A(h) при $N = 64$ с двойной итерацией на n-том этапе ($\tau = n/8$, итерации не производилось при $\tau = 0$).

функции $y_o(x) = C$ =const. Таким образом, при C, принадлежащем отрезкам $[0, 2.5]$, $(-\infty, -3]$, $[-2, -2.5] \cup [3, +\infty)$ метод Nt(h) сводится к решению $y_1 = (\sqrt{5} - 1)e^x$, $y_2 = (1 - \sqrt{5})e^x$, $y_3 = 2e^x$, соответственно.

Тем не менее результаты предлагаемого подхода, особенно отличная работа уточнений по Ричардсону, позволяют надеяться на перспективность исследований в данном направлении.

§3.4. БЫСТРЫЕ АЛГОРИТМЫ

3.4.1. Основным результатом данной работы является разработка методов построения быстрых алгоритмов для определенных классов интегральных уравнений. Таблицы 3 – 10 содержат результаты работы таких алгоритмов. Алгоритмы, предназначенные для ядер теплицева типа, представлены в Таблицах 3 – 7. Не вызывает сомнений доминирование алгоритмов A–серии (AD и ADM).

Результаты Таблицы 4 свидетельствуют о “сверхбыстроте” (или “сверхточности”) работы Алгоритма AD(h) в случае ядра периодического (циркулянтного, см. п. 2.3.1) типа. Заметим также, что в экспериментах, не нашедших здесь отражения, был подтвержден также теоретический вывод с повышенной точностью работы Алгоритма ADM(h), когда ядро (2.29) периодично хотя бы по одной паре

переменных (x_k, t_k).

Результаты экстраполяций по Ричардсону (Таблицы 6,7) также свидетельствуют о доброкачественности Алгоритмов ADM и ABT. В Задаче P5 последний алгоритм работает устойчиво в широком диапазоне изменения λ, поскольку операторы $\mathcal{J} - \mathcal{K}_\tau$ обратимы при всех $\tau \in [0, 1]$ (так как ядро K5 антисимметрично.)

Алгоритм $\mathrm{AD}(h^4)$, построенный на основе двухточечной формулы Гаусса (см. Пример 2.3) , не только работает быстро и устойчиво (Таблица 3), но и позволяет проводить исключительно эффективную экстраполяцию по Ричардсону, поскольку ошибка его применении разлагается в асимптотический ряд по четным степеням h $(h = 1/N)$.

Замечание 3.5. Ошибки алгоритмов, построенных на основе формул трапеций, средней точки и Симпсона (в (1.10') и (2.21)) также облалают таким свойством. Любой другой алгоритм можно преобразовать к "симметрическому" виду по схеме, известной в методах Рунге-Кутты.

Пример 3.1. Эффективность экстраполяции в случае Алгоритма $\mathrm{A}(h^2)$, построенного на основе формулы (1.10), (1.10'), продемонстрируем на примере Задачи P2. Так, при $\lambda = 1$, $\delta(16) = 2.1 \times 10^{-3}$ первая экстраполяция $\tilde{y}_{16}$ (с использованием значений y_8 и y_{16}) приводит к ошибке $\tilde{\delta} = 1.5 \times 10^{-5}$, а вторая (на основе y_1, y_8 и y_{16}) дает результат $\tilde{\delta} = 1.5 \times 10^{-6}$. Как видим, в случае гладких ядер общего вида этот алгоритм может считаться универсальным.

3.4.2. Алгоритм AL(h) для широкого класса ядер K, удовлетворяющих уравнению (2.6), представлен в Таблицах 8 и 9. Он построен на основе аналога Теоремы 2.3, соответствующего методу дискретного погружения. Вычисление производных по τ (см. формулы (2.6), (2.9)) проведено с шагом $h = 1/N$ на равномерной сети $\tau_n = nh$ $(n = 0, 1, ..., N)$, а производной по x, при $m = 1, 2, 3$ по формулам

$$\frac{\partial f}{\partial x} \approx (mh)^{-1}\big(f(x) - f(x - mh)\big); \quad x = nh, \quad n = m, m+1, ..., N-m. \qquad (3.4)$$

При $x = nh$ $(n = 0, 1, ..., m-1, N-m+1, ..., N)$ использованы формулы (1.10) и (1.10'). Фактически здесь шаг по τ в m раз меньше шага по x $(h_x = mh_\tau)$.

Этот алгоритм , вообще говоря, неустойчив, но результаты Таблиц 8 и 9 свидетельствуют об эффективности его стабилизации за счет выбора m. При

росте $|\sigma|$ (Задача P6, $\mathrm{Im}\ \sigma = 0$), как и рекомендуется в работе [37] (см. п. 2.5.5) надо выбрать больше m.

Неожиданными являются результаты Таблицы 8, относящиеся к комплексным σ. Несмотря на то, что в этом случае оператор $\mathcal{L}_o$ (см. 2.2.4) эллиптический, стабилизация проходит весьма эффективно при $\mathrm{Re}\ \sigma \neq 0$. Роль величины $\mathrm{Im}\ \sigma$ при этом проявляется гораздо слабее, чем можно было ожидать.

3.4.3. Работа новых устойчивых быстрых Алгоритмов ATH и ALSt представлена в Таблицах 9 и 10. Экстраполяции и здесь проходят устойчиво. Интересно отметить, что (Таблица 9) результаты стабилизации неустойчивого Алгоритма AL(h) при $m = 2$ не хуже, а иногда и лучше, чем у ALSt(h).

При $N = 64$ время работы Алгоритма ATH(h) несколько меньше, чем у параллельного варианта NG(h) (см. время в скобках), а при $N = 128$ – вдвое меньше, чем даже у последовательного варианта.

§3.5. ЗАКЛЮЧЕНИЕ

3.5.1. В предыдущем параграфе были проанализированы численные эксперименты с применением лишь части разработанных в Главах I и II алгоритмов. Причина заключается в том, что реализация некоторых алгоритмов пока нас не удовлетворяет, а иные из них вообще не испытаны на ЭВМ. Так, хотя метод автоматического выбора шага и порядка алгоритма в процессе счета (§1.6) дает обнадеживающие результаты (см. [15]), стабильно действующего и экономичного механизма выбора шага еще не найдено. Не завершен цикл испытаний семейства многошаговых алгоритмов на базе метода непрерывного погружения (см. 1.5.3, 1.5.4).

С другой стороны, в случае многомерных областей $D \subset R^m$ более или менее общего вида не найдено пока программной реализации быстрого алгоритма для ядер разностного (теплицева) вида (см. п. 2.4.4). Здесь трудности связаны с дискретной интерполяцией на границе области, привносящей дополнительные ошибки в расчеты.

3.5.2. Однако результаты §§3.3, 3.4 фактически имеют отношение не только к Задачам типа P1 – P8. Например, испытанные быстрые алгоритмы можно смело ре-

комендовать для решения уравнений первого рода с соответствующими ядрами. Действительно, регуляризованное уравнение (2.36') является уравнением второго рода с (возможно) очень большой нормой ядра. Но при этом ядро симметрично и положительно определено и поэтому метод дискретного погружения отлично работает при выборе любой сети $\{\tau_n\}$ (см., например, Таблицы 4, 6). Остается применить хорошо разработанный механизм оптимального выбора параметра α (см. [10]).

Многие алгоритмы серий A и B можно, с уверенностью в успехе применить в многомерном ($m \geq 3$) случае. Проблема здесь лишь в возможности использования компьютеров с большой оперативной памятью (такой возможности у нас не было).

3.5.3. По аналогичной причине (отсутствие многопроцессорных ЭВМ) эффективность использования параллельной (векторной) структуры алгоритмов пока не проверена. Однако нетрудно убедиться, что графы предлагаемых быстрых алгоритмов обладают достаточно простой структурой, аналогичной известным графам алгоритмов обращения матриц теплицева типа (см. [43, 44]. Это обстоятельство позволяет утверждать, что разработанные здесь алгоритмы удобны для использования как на многопроцессорных ЭВМ с относительно небольшим числом мощных вычислителей (см. [12]), так и на специально сконструированных систолических массивах из большего числа однотипных элементов [43].

3.5.4. Устойчивость всех предлагаемых в работе алгоритмов, за исключением Алгоритма AL (см. пп. 2.2.3 и 3.4.2), можно строго доказать по схеме доказательства Теоремы 1.7 (см. также [4, 28]). Теретическая оценка накопления ошибок (1.44) на самом деле довольно грубая. Во всяком случае, в многочисленных экспериментах уровень ошибки вычисления решения уравнения второго рода был значительно ниже.

В заключение отметим, что на основе предлагаемых подходов можно разработать алгоритмы, приспособленные для случая интегральных уравнений с ядрами, обладающими слабой (интегрируемой) сингулярностью в отдельных точках, на линиях и т. п. (см. [1.2]).

ПРИЛОЖЕНИЕ. ТАБЛИЦЫ РАСЧЕТОВ

В приводимых таблицах N означает число шагов по τ на равномерной сетке $\tau_n = nh$ $(h = 1/N,\ n = 0, 1, ..., N)$. Ошибка $\delta(N)$ приведена с плавающей точкой (например, $\delta = 0.03127 = 3.13, -2$). Принято ∞ = overflow. Ввиду использования компьютеров с разной тактовой частотой время работы алгоритмов сравнимо лишь в пределах той же таблицы.

Ориентировочное время экстраполяции по Ричардсону можно определить из таблиц, учитывая способ экстраполяций (см. п. 3.1.1). Например (см. Таблицу 10) на счет по Алгоритму ATH(h) при $N = 128$ $(\delta(128) = 7.34, -2)$ затрачено 88 секунд, на счет по ATH(h)R $(\bar{\delta}(128) - 1.90, -4)$ - $88 + 22 = 110$ секунд , а на счет по ATH(h)RR $(\bar{\delta}(128) = 1.21, -4)$ - $88 + 22 + 6 = 116$ секунд.

Таблица 1

Задача	*Ядро*	*Решение*	*Примечание*
$P1$	$K1 = (x - 0.5)^2 \times$ $\times (t - 0.5)^2 \times$ $\times \sin[(x - 0.75)^2 +$ $+ 0.15]^{-1}$	$y = (x - 0.4) \times$ $\times \sin 5\pi(x - 0.15)$	$K1$ отличается от ядра работы [18] осциллирующим сомножителем $\|\|K2\|\| = 0.81\|\lambda\|$.
$P2$	$K2 = \lambda e^{-xt}$	$y \equiv 1$	Задача P2 при $\lambda = 0.5$ изучалась в[27] K2 удовлетворяет уравнению (0.3) при $a(x) = x^2$, $p \equiv 0$ $\|\|K3\|\| = 0.87\|\lambda\|$
$P3$	$K3 = \lambda \exp[-(x - t)^2]$	$y = e^{x^2}$	$K3$ – симметричное теплицево ядро
$P4$	$K4 = \lambda \cos 2\pi(x - t)$	$y \equiv 1$	$\|\|K4\|\| = 0.71\|\lambda\|$ $K4$ – симметричное теплицево ядро циркулянтного типа (см. п. 2.3.1)

Задача	*Ядро*	*Решение*	*Примечание*
P5	$K5 = \lambda(x-1)^{-2} \times$ $\times \left(\frac{x-1}{x+1} - \ln\frac{x+1}{x-1}\right)$, $x \neq 1$ $K5 = -0.5(x+1)^{-2}$, $x = 1$	$y \equiv 1$	$K5$ – антисимметричное ядро теплицева типа (0.2) $p(x) = q(x) = \frac{1}{x+1}$ (см. п.п. 2.4.1)
P6	$K6 = [K_{ij}]$, $i,j = 1,2$ $K_{11} = \lambda_{11} \times$ $\times \exp[-0.1(x-1)^2]$ $K_{12} = \lambda_{12} \times$ $\times \exp[0.1(\sigma x - t)^2]$ $K_{21} = \lambda_{21} \times$ $\times \exp[-0.1(x-\sigma t)^2]$ $K_{22} = \lambda_{22} \times$ $\times \exp[0.1(x-t)^2]$	$y = \begin{pmatrix} \exp(0.1x^2) \\ \exp(-0.1x^2) \end{pmatrix}$	$K6$ – матричное ядро вида (2.6), где $B, p, q \equiv 0$; $\sigma = \text{const} \in \mathbf{C}$ $A = \begin{pmatrix} 1 & 0 \\ 0 & \sigma \end{pmatrix}$ (см. п. 2.4.3)
P7	$K7 = \frac{(x_2 - t_2)\sin(x_1^2 + t_1)}{1 + x_1^2 + (x_2 - t_2)^2}$	$y \equiv 1$	Двумерное уравнение с ядром теплицева вида (2.29) $D = [0,1] \times [0,1]$

Тестовые задачи. $||K||$ – норма ядра в L_2, в Задачах P1 – P6 принято $D = [0,1]$.

Таблица 2

Алгоритм	*Число узлов*	*δ*	*Время*
$NG(h^2)I$	5, 9, 17, 29, 35, 37	4.84, –3	1′57″
$NG(h^2)II$	5, 27, 39	2.82, –3	1′10″
$NG(h^2)III$	5, 27, *Spline*	2.83, –3	0′24″

Задача P1. $a = b = 0.5$ (см. п. 3.2.1). Число узлов по этапам, среднеквадратическая ошибка по 37 точкам (NG I) и 39 точкам (NG II, III). Счет на компьютере IBM PC/AT 286.

Таблица 3

Алг. λ	$B(h)$	$B'(h)$	$A(h)$	$AD(h)$	$BD(h)$	$NG(h)$	$AD(h^4)$	$A(h^6)$
1	5.15	1.89	1.44	3.28	4.61	1.44	7.1	2.1,
	−1	−1	−1	−2	−1	−1	−9	−10
2	∞	∞	6.04,	3.44,	∞	6.13,	3.8,	1.1,
			−2	−2		−2	−9	−10

Задача P3. Ошибка $\delta(16)$ в Алгоритме A(h^6) и $\delta(32)$ – в остальных алгоритмах.

Таблица 4

λ	1	1.5	1.9	$2-10^{-6}$	$2+10^{-6}$	20	10^8
Алг.							
$AD(h)$	$<10^{-17}$	$<10^{-17}$	$<10^{-17}$	1.2, −14	1.3, −12	3.0, −16	6.6, −15
$BD(h)$	8.0, −4	8.1, −3	1.0, −1	6.9, −1	6.9, −1	∞	∞

Задача P4. Ошибка $\delta(256)$. Расчеты выполнены с высокой точностью на IBM PC/AT 486.

Таблица 5

Алг. N	$A(h)C$	$A(h)CI$	$NG(h)$	t (*сек.*)	$ATH(h)$	t (*сек.*)
8	7.1, −2	4.2, −4	6.06, −2	0(0)	6.6, −2	0
16	3.6, −2	3.1, −4	3.33, −2	1(1)	3.1, −2	2
32	1.8, −2	2.0, −4	1.75, −2	4(6)	6.1, −2	6
64	9.2, −3	1.1, −4	8.97, −3	24(36)	1.2, −2	22
128	4.6, −3	6.0, −5	4.54, −3	160(240)	4.6, −3	88

Задача P2, $\lambda = 2$. Ошибка $\delta(N)$ и время счета для NG и ATH. В A(h)CI проведены 3 итерации .

Таблица 6

Alg.	$\lambda = -0.5$		$\lambda = 1$		$\lambda = 2$		$\lambda = 4$	
N	$ABT(h)$	$ABT(h)R$	$ABT(h)$	$ABT(h)R$	$ABT(h)$	$ABT(h)R$	$ABT(h)$	$ABT(h)R$
16	3.7, −3		1.0, −2		2.8, −2		1.0, −1	
32	1.9, −3	5.8, −5	5.4, −3	1.3, −4	1.4, −2	3.2, −4	5.0, −2	1.2, −3
64	9.6, −4	1.3, −5	2.7, −3	3.2, −5	7.2, −3	8.0, −5	2.5, −2	2.1, −4
128	4.8, −4	3.3, −6	1.3, −3	7.9, −6	3.6, −3	2.0, −5	1.2, −2	4.5, −5

Задача P5. Ошибка $\delta(N)$ и экстраполяция по Ричардсону (см. 2.4.2).

Таблица 7

$N_1 \times N_2$	4 × 4	4 × 8	4 × 16	4 × 32	8 × 8	8 × 16	16 × 8	16 × 16
$NG(h)$	4.79, −2	2.85, −2	2.11, −2	1.88, −2	2.44, −2	1.42, −2	2.35, −2	1.32, −2
$t(sec)$	0(0)	3(4)	14(21)	89(138)	15(22)	98(148)	99(149)	703(1050)
$ADM(h)$	4.58, −2	2.86, −2	2.08, −2	1.78, −2	2.46, −2	1.48, −2	2.36, −2	1.28, −2
$t(sec)$	1	5	16	62	26	97	176	670

Двухмерная Задача P7. Ошибка $\delta(N_1, N_2)$ и время счета (N_1, N_2 - количество точек дискретизации по x_1, x_2, см. п. п. 1.4.6, 2.4.5.)

Таблица 8

σ / m	3	$\sqrt{2}$	$\sqrt{2} + i/4$	$2^{1/8}$	$2^{1/8} + i/16$	$1 + i/2$	$1 + i/4$	$i/4$
1	∞	∞	∞	8.28, −3	1.02, −1	∞	8.8, +2	1.11, +11
2	∞	7.03, −3	7.0, −3	1.87, −3	4.81, −3	4.29, −2	3.2, −3	1.01, +10
3	5.3, −2	7.66, −3	7.7, −3	5.95, −3	5.89, −3	2.28, −3	4.6	3

Задача P6. $\lambda_{11} = \lambda_{22} = 0.5$, $\lambda_{12} = \lambda_{21} = 0.25$. Алгоритм AL(h) (см. п.п. 2.2.3 и 3.4.2). Ошибка $\delta(256)$ (при $\sigma = 3$ $\delta = \delta(64)$).

Таблица 9

N	$m=1$	$m=2$	$m=3$	$ALSt(h)$	$ALSt(h)R$	$ALSt(h)RR$
32	1.67, +6	8.59, −4	1.12, −3	1.80, −3		
64	∞	4.77, −4	6.01, −4	9.15, −4	6.20, −5	
128	∞	3.49, −4	3.95, −4	4.60, −4	1.60, −5	2.10, −6
256	∞	3.35, −4	3.44, −4	2.30, −4	2.30, −6	4.50, −7

Задача P6. $\sigma = 2$, $\lambda_{11} = \lambda_{22} = -0.5$, $\lambda_{12} = \lambda_{21} = -0.25$. Ошибка $\delta(N)$. Слева – результат применения Алгоритма AL(h) при $m = 1, 2, 3$ (см. п. 3.4.2). Справа – Алгоритм ALSt(h) и две экстраполяции по Ричардсону (см. п.п. 3.1.1 и 3.4.3).

Таблица 10

N	$NG(h)$	t(сек.)	$ATH(h)$	$ATH(h)R$	$ATH(h)RR$	t(сек.)
16	6.07, −2	1(1)	5.94, −2			2
32	2.93, −2	4(6)	2.95, −2	1.28, −3		6
64	1.44, −2	24(36)	1.47, −2	4.68, −4	2.59, −4	22
128	7.15, −3	161(240)	7.34, −3	1.90, −4	1.21, −4	88
256			3.65, −3	8.04, −5	5.27, −5	349

Задача P2. $\lambda = 1$. Ошибка $\delta(N)$, время счета. Для АТН – две экстраполяции по Ричардсону.

Таблица 11

N	$Nl(h)$	l(сек.)	$AN(h)$	$AN(h)R$	$AN(h)RR$	l(сек.)
4	1.43, −1	0	1.84, −1			0
8	9.16, −2	2	1.14, −1	4.83, −2		1
16	5.26, −2	5	6.67, −2	2.00, −2	1.04, −2	5
32	2.84, −2	24	3.68, −2	7.20, −3	2.86, −3	47

Нелинейная задача P8 (см. п. 1.8.2 и (3.3), (3.3')). Сходимость к решению $y = (\sqrt{5}-1)e^x$. Ошибка $\delta(N)$ и время счета (в Nt(h) при $y_0 = f$ – пять итераций). Для AN(h) – две экстраполяции по Ричардсону.

ABSTRACT. Economic algorithms for various types of integral equations are investigated. The paper presents a general method for construction of parallel algorithms. The main results refer to the construction of new fast, memory saving algorithms for a wide class of kernels. Results of some numerical experiments illustrating the efficiency of the proposed methods are attached.

ЛИТЕРАТУРА

1. А. Ф. Верлань, В. С. Сизиков, Интегральные Уравнения : методы, алгоритмы, программы, Киев, Наукова Думка, 1986.
2. L. M. Delves, J. L. Mohamed, Computational Methods for Integral Equations, Cambridge Univ. Press, 1985.
3. T. Kailath, L. Ljung, M. Morf,“ Generalized Krein–Levinson equations for efficient calculation of Fredholm resolvents of nondisplasement kernels”, Topics in Functional Anal. Advances in Math. Suppl. Studies, v. 3, pp. 169 – 183, 1978.
4. I. Gohberg, I. Koltracht,“Numerical solution of integral equations. Fast algorithms and Krein–Sobolev equation”, Numer. Math., v. 47, pp. 233 – 288, 1985.
5. А. Б. Нерсесян, “ Структура резольвент некоторых интегральных операторов”, Изв. АН Арм. ССР, Математика, т. 17, №6, стр. 442 – 463, 1982.
6. А. Б. Нерсесян, “ О структуре резольвент некоторых интегральных операторов с ядрами, определяемыми дифференциальными уравнениями”, ДАН СССР, т. 30, №3, стр. 727 – 732, 1984.
7. А. Б. Нерсесян, М. Абду, Ш. А. Багдасарян, “Новое быстро решаемое интегральное уравнение”, ДАН Арм. ССР, т. 93, №2, стр. 51 – 57, 1992.
8. G. Heinig, K. Rost, “Matrix representations of Toeplitz-plus-Hankel matrix inverses”, Linear Algebra and its Appl., v. 113, pp. 65 – 78, 1989.
9. И. Я. Штаерман, Контактная Задача Теории Упругости, М.– Л. Гостехиздат, 1949.
10. А. Н. Тихонов, А. В. Гончарский, В. В. Степанов, Я. Г. Ягола, Численные Методы Решения Некорректных Задач, М., Наука, 1990.
11. И. К. Лифанов, Е. Е. Тыртышников, Теплицевы Матрицы и Сингулярные Интегральные Уравнения, Вычислительные процессы и системы, выпуск 7, М., Наука, стр. 94 – 279, 1990.
12. В. В. Воеводин, Математические Модели и Методы в Параллельных Процессах, М., Наука, 1986.
13. Е. Валях, Последовательно-Параллельные Вычисления, М., Мир, 1985.
14. А. Б. Нерсесян, “ Новые алгоритмы численного решения интегральных уравнений второго рода”, ДАН Арм. ССР, т. 89, №4, стр. 171-176, 1989.
15. А. В. Гаспарян, А. Б. Нерсесян,“ Численное решение интегральных уравнений с автоматическим выбором шага”, деп. в АрмНИИНТИ, №68-Ар.89, 1989.
16. А. Б. Гарибян, А. Б. Нерсесян, “Численное решение интегральных уравнений второго рода с разностным ядром”, деп. в АрмНИИНТИ, №1-Ар.90, 1990.
17. А. Б. Нерсесян, Ш. А. Багдасарян, “О повышении устойчивости алгоритма быстрого решения одного класса интегральных уравнений”, ДАН Армении, т. 91, №3, 1990.

18. R. Neta, P. Nelson," An adaptive method for numerical solution of Fregholm integral equation of the second kind I. Regular kernels", Applied Mathematics and Computation, v. 21, pp. 171 – 184, 1987.
19. А. А. Гамарян, А. Б. Нерсесян, О. А. Оганян, " Решение интегральных уравнений методом автоматического сгущения сетки", Изв. НАН Армении, Математика, т. 27, №1, 1992.
20. С. М. Никольский, Квадратурные формулы, М., Наука, 1974.
21. Э. Хайрер, С. Нерсетт, Г. Ваннер, " Решение Обыкновенных Дифференциальных Уравнений. Нежесткие Задачи, М., Мир, 1990.
22. C. Wagner, "On the solution of Fredholm integral equation of the second kind by iteration", J. Math. Phys., v. 30, pp. 23 – 30, 1951.
23. L. B. Rall, "Error bounds for interative solutions of Fredholm integral equations", Pacific Journ. of Math., v. 5, pp. 977 – 986, 1955.
24. D. J. Evans, A. Hadjidimos, " A modification of the quadrant interlocking factorization parallel method", Intern. J. Computer Math., Section B., v. 8, pp. 149 – 166, 1980.
25. С. М. Ермаков, Г. А. Михайлов, Курс Статистического Моделирования, М., Наука, 1976.
26. А. Б. Нерсесян, Х. М. Бешликян, "Численная реализация нового метода решения уравнения теплицева типа", ДАН Армении, т. 19, №4, стр. 152 - - 158, 1990.
27. A. M. McNabb, A. Shumitzky, "Factorization of operators II : a nonlinear Volterra method for numerical solution of linear Fredholm equations", Journ. Comp. Syst. Sciences, v. 4, pp. 103 – 128, 1970.
28. I. Gohberg, I. Koltracht, P. Lancaster, "Second order parallel algorithms for Fredholm integral equations with continuous displacement kernels", Integral Equations and Operator Theory, v. 10, pp. 577 – 594, 1987.
29. С. Л. Соболев, " Некоторые замечания о численном решении интегральных уравнений", Успехи мат. наук, 9, вып. 3(61), 1954.
30. Н. П. Сергеев, "Об одном новом методе решения линейных интегральных уравнений", Записки Ленингр. Горного ин-та, т.33, 1956.
31. М. Г. Крейн, " Об одном методе решения линейных интегральных уравнений первого и второго рода", ДАН СССР, т. 100, №4, 1955.
32. R. E. Bellman, "Functional equations in the theory of dynamic program VIII : a partial differential equation for Fredholm resolvent", Proc. Amer. Math. Soc., v. 8, 1957.
33. Н. Б. Енгибарян, М. А. Мнацаканян,"О факторизации интегральных операторов", ДАН СССР, т. 206, №4, 1972.
34. И. Ц. Гохберг, М.Г. Крейн, Теория Вольтерровых Операторов в Гильбертовом Пространстве и ее Приложения, М., Наука, 1967.
35. J. G. Blom, H. Brunner,"The numerical solution of nonlinear Volterra integral equations of the second kind by collocation and iterated collocation methods", SIAM J. Sci. Stat. Comput., v. 8, no.5, pp. 806 – 830, 1987.
36. И. Г. Петровский, Лекции об Уравнениях с Частными Производных, М., Физматгиз, 1961.
37. D.Gottlieb, E.Turkel," Dissipative two-four methods for time-dependent problems", Math. of Comput., v. 30, no. 136, pp. 703 – 723, 1976.
38. В. В. Воеводин, Е. Е. Тыртышников, Вычислительные Процессы с Теплицевыми Матрицами", М., Наука, 1987.
39. Л. А. Сахнович, " Уравнения с разностным ядром на системе отрезков", Теория функций, функц. анализ и их приложения, вып. 45, Харьков, Вища школа, стр. 111 – 122, 1986.
40. С. Н. Воеводин ., "Клеточно-теплицевы матрицы и интегральные уравнения",

Вычислительные методы и программирование, М., Изд-во, МГУ, стр. 19 - - 94, 1975.

41. И. Ц. Гохберг, Г. Хайниг, О матричных интегральных операторах на конечном интервале с ядрами, зависящими от разности аргументов", Rev. Roumanie Math. Pures at Appl., v. 20, no. 1, pp. 55 – 74, 1975.
42. А. Б. Гарибян, "Численное решение интегрального уравнения с ядром теплицева типа", Прикладная математика, №6, Изд. ЕрГУ, стр. 22 – 29, 1990.
43. S. Y. Kung, "A highly concurrent algorithm and pipelined architecture for olving Toeplitz systems", IEEE Trans. Acoust. Speech, Sign. Processing, v.ASSP-31, no. 1, pp. 66 –76, 1983.
44. С. А. Краснов, Е. Е. Тыртышников, "Векторизованные алгоритмы и систолические массивы для теплицевых систем", Архитектура ЭВМ и численные методы, ОВМ АН СССР, стр. 33 – 56, 1985.

20 Марта 1992

Институт математики
АН Армении