ЧИСЛЕННОЕ РЕШЕНИЕ ИНТЕГРАЛЬНОГО УРАВНЕНИЯ ФРЕДГОЛЬМА МЕТОДОМ АВТОМАТИЧЕСКОГО СГУЩЕНИЯ СЕТКИ

А. Л. Гамарян, А. Б. Нерсесян, Г. А. Оганян

**Известия Академии Наук Армении. Математика,
том 27, №1, 1992**

В работе предложен метод обнаружения быстрой осцилляции решения интегрального уравнения Фредгольма второго года. Модифицирован алгоритм работы [1]. Усовершенствование состоит в применении более чувствительного механизма сгущения сетки в комбинации со сплайн-интерполяцией.

ВВЕДЕНИЕ

В работе [1] предложен следующий метод решения интегрального уравнения Фредгольма второго рода :

$$y(x) = \int_0^1 K(x,t)y(t)dt + f(x), \tag{1}$$

где $f(x) \in C^2([0,1])$ и $K(x,t) \in C^2([0,1] \times [0,1])$.

Интеграл в уравнении (1) заменяется конечной суммой, полученной по формуле трапеций :

$$\int_0^1 K(x,t)y(t)dt = \sum_{i=1}^{N-1} \frac{1}{2} h_i (K_i(x)y_i + K_{i+1}(x)y_{i+1}) - \sum_{i=1}^{N-1} \tau_i h_i^3, \tag{2}$$

где $h_i = x_{i+1} - x_i$ (на первом шаге $h_i = h = \text{const}$), $0 = x_1 < x_2 < \cdots < x_N = 1$ - разбиение $[0,1]$, $K_i(x) = K(x,x_i)$, $y_i = y(x_i)$, $\tau_i \leq \max|(K(x,t)y(t))_{tt}|$.

При подстановке (2) в (1) задача с точностью порядка $O(h^2)(h = \frac{1}{n} \to 0)$ сводится к решению системы

$$(I - KD)Y = F, \tag{3}$$

В наших экспериментах, - на первом этапе применения метода работы [1], на каждом шаге почти все интервалы делились пополам (за исключением тех, на которых функции оставались практически постоянными). В то же время мы ожидали, что окончательное разбиение будет иметь сгущение на интервалах с достаточно быстрой вариацией производных решения $y(x)$ и что будет скорее мало интервалов со "спокойным" поведением. Однако, такая модель релко соблюдалась на практике на начальных этапах.

В настоящей работе мы предлагаем автоматическое разбиение интервалов на два, четыре и восемь частей, зависящее от степени отклонения от оценок (4) и (5). Введем в правой части соотношений (4) и (5) весовые множители 2 и 4.

$$\int_{x_j}^{x_{j+1}} |y'(x)|dx \le 2\delta(\max_{\forall i} |y_i|), \tag{4.1}$$

$$\int_{x_j}^{x_{j+1}} |y'(x)|dx \le 4\delta(\max_{\forall i} |y_i|), \tag{4.2}$$

$$\int_{x_j}^{x_{j+1}} |y''(x)|dx \le 2\gamma(\max_{\forall x} |\frac{dy}{dx}|, x = x_i), \tag{5.1}$$

$$\int_{x_j}^{x_{j+1}} |y''(x)|dx \le 4\gamma(\max_{\forall x} |\frac{dy}{dx}|, x = x_i). \tag{5.2}$$

После проверки условий (4), (4.1), (4.2), (5), (5.1) и (5.2) мы поступаем следующим образом :

– если, по крайней мере, одно из соотношений (4.2) или (5.2) не выполнено, то интервал $[x_j, x_{j+1}]$ делится на восемь частей ;

– если, по крайней мере, одно из соотношений (4.1) или (5.1) не выполнено, то интервал $[x_j, x_{j+1}]$ делится на четыре части ;

– если, по крайней мере, одно из соотношений (4) или (5) не выполнено, то интервал $[x_j, x_{j+1}]$ делится на две части.

На этом пути, мы получаем очень быстрое сгущение сетки, на которой достигается требуемая точность. Сохраненное время можно использовать для решения системы прямым методом, что значительно лучше, чем итерации (как в [1]). Тем самым расширяется область применения алгоритма.

§2. ПРИМЕНЕНИЕ СПЛАЙНОВ

Во многих численных экспериментах экстраполяция решения, использующая кубические сплайны, дает хорошие результаты. Это позволяет найти, в частности, приближения высокой точности даже на интервалах, где производные решения имеют быстрые вариации.

Необходимо отметить, что использование сплайнов вместе с автоматическим разбиением интервалов особенно выгодно, так как это используется при вычислении производных на каждом шаге. Удобно использовать эрмитовы кубические сплайны. После последнего шага достаточно найти значения производных на концах отрезка [0, 1], а потом построить глобальный кубический сплайн. Это приводит к значительному увеличению точности по сравнению со сплайнами со свободными концами, в то время как время практически не увеличивается.

Построение сплайна n-точками нуждается только в $O(n)(n \to \infty)$ дополнительных операциях. Альтернативой является решение системы n-того порядка, когда в итерационных методах сложность имеет порядок $O(n^2)$, тогда как в прямых методах сложность имеет порядок $O(n^3)$, т. е. разница во времени счета существенна.

§3. ЧИСЛЕННЫЕ РЕЗУЛЬТАТЫ

В компьютерных вычислениях мы используем три алгоритма, которые обозначим через A, B и C.

Алгоритм A основан на методе работы [1];

Алгоритм B улучшает метод работы [1], используя автоматическое разбиение разбиение интервалов на два, четыре или восемь частей (см. §1);

Алгоритм C на первых двух этапах использует алгоритм B. На третьем, заключительном этапе, строится кубический сплайн, используя значения y в точках разбиения и значения y' на концах отрезка [0,1]. Результатом является наше решение.

Рассмотрим следующие два ядра :

$$K_1(x,t) = (x-0.5)^2(t-0.5)^2,$$

$$K_2(x,t) = (x-0.5)^2(t-0.5)^2 \sin[\frac{1}{(x-0.75)^2+0.15}].$$

Случай ядра K_1 был изучен в работе [1], с которой сравниваются численные результаты. В отличие от K_1, для ядра K_2 мы добавим сильно осциллирующий в окрестности точки $x = 0.75$ множитель, и поэтому осциляции решения зависят не только от правой части f.

Таблица 1 содержит некоторые тестовые задачи для решения которых проводились численные эксперименты.

Задача	Ядро	$y(x)$	$f(x)$
$P1$	K_1	$\begin{cases} -4x^2+4x, \\ x \in [0;0.25] \cup [0.75;1]; \\ 0.75+8\pi x, \\ x \in (0.25;0.75) \end{cases}$	$y(x) - \frac{29}{640}(x-0.5)^2$
$P2$	K_1	$\begin{cases} 1, \\ x \in [0;0.5] \cup [0.75;1]; \\ (8x-5)^2, \\ x \in [0.5;0.75] \end{cases}$	$y(x) - \frac{31}{480}(x-0.5)^2$
$P3$	K_1	$(x-0.4)\times$ $\times \sin(5\pi(x-0.15))$	$y(x) - (x-0.5)^2 \times$ $\times \frac{\sqrt{2}(-25\pi^3+150\pi^2)}{5000\pi^4} +$ $+ \frac{\sqrt{2}(48\pi-48)}{5000\pi^4}$
$P4$	K_2	$(x-0.4)\times$ $\times \sin(5\pi(x-0.15))$	$y(x) - (x-0.5)^2 \times$ $\times \sin[\frac{1}{(x-0.75)^2+0.15}] \times$ $\times \frac{\sqrt{2}(-25\pi^3+150\pi^2)}{5000\pi^4} +$ $+ \frac{\sqrt{2}(48\pi-48)}{5000\pi^4}$

Таблица 1.

Предположим, что $\delta = 0.5$, $\gamma = 0.5$ (см. [4], [5]).

Результаты вычислений приведены в Таблице 2 и на Рис. 1, 2, 3. В задачах, описанных в Таблице 1 и на Рис. 2, 3, начальное число точек сети было равно 5, на Рис. 1 число точек равно 10. В Таблице 2 мы привели окончательное число точек сети. Вычисления произведены на компьютере IBM PC/AT 286.

Задача	Алгоритм	Число сетей	Ошибка	Время
	A	57	$1.15\,E-3$	$1' - 18"$
*P*1	*B*	67	$1.15\,E-3$	$0' - 52"$
	C	33	$2.17\,E-2$	$0' - 18"$
	A	21	$8.62\,E-3$	$0' - 25"$
*P*2	*B*	34	$8.82\,E-3$	$0' - 23"$
	C	26	$1.35\,E-2$	$0' - 09"$
	A	49	$3.50\,E-3$	$1' - 02"$
*P*3	*B*	52	$1.44\,E-4$	$0' - 35"$
	C	33	$1.64\,E-4$	$0' - 17"$
	A	37	$4.84\,E-3$	$1' - 32"$
*P*4	*B*	39	$2.82\,E-3$	$1' - 08"$
	C	27	$2.83\,E-3$	$0' - 23"$

Таблица 2.

Вычисления, использующие все три метода в Задаче P3 было достаточно проверить, а чистое время, необходимое для построения сплайна по 33 точкам было равно 0'-01.48".

Заметим, что в Задаче P4 решение $y(x)$ значительно отличается от правой части $f(x)$, и в этом случае алгоритм C имеет существенное преимущество в скорости работы. При применении алгоритма C начальное число точек было равно пяти, а после первого этапа число точек стало равно 27 и они были использованы при построении кубического сплайна.

На Рис. 1 показан график зависимости логарифма значения корня от ошибки и затраченное время во всех трех алгоритмах. На Рис. 2 и 3 даны графики сравнения для времени и ошибки соответственно для различных задач.

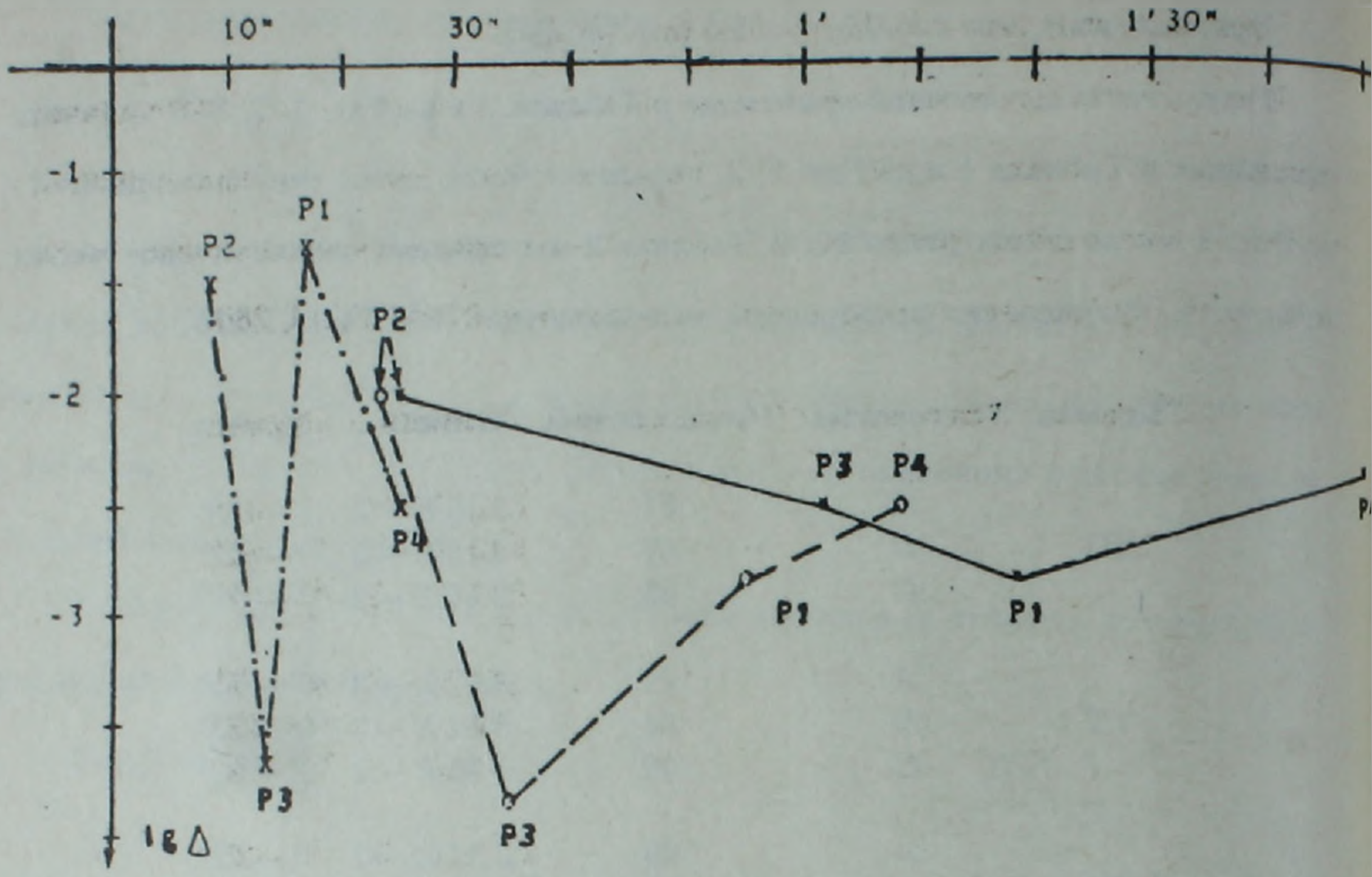

Рис. 1.

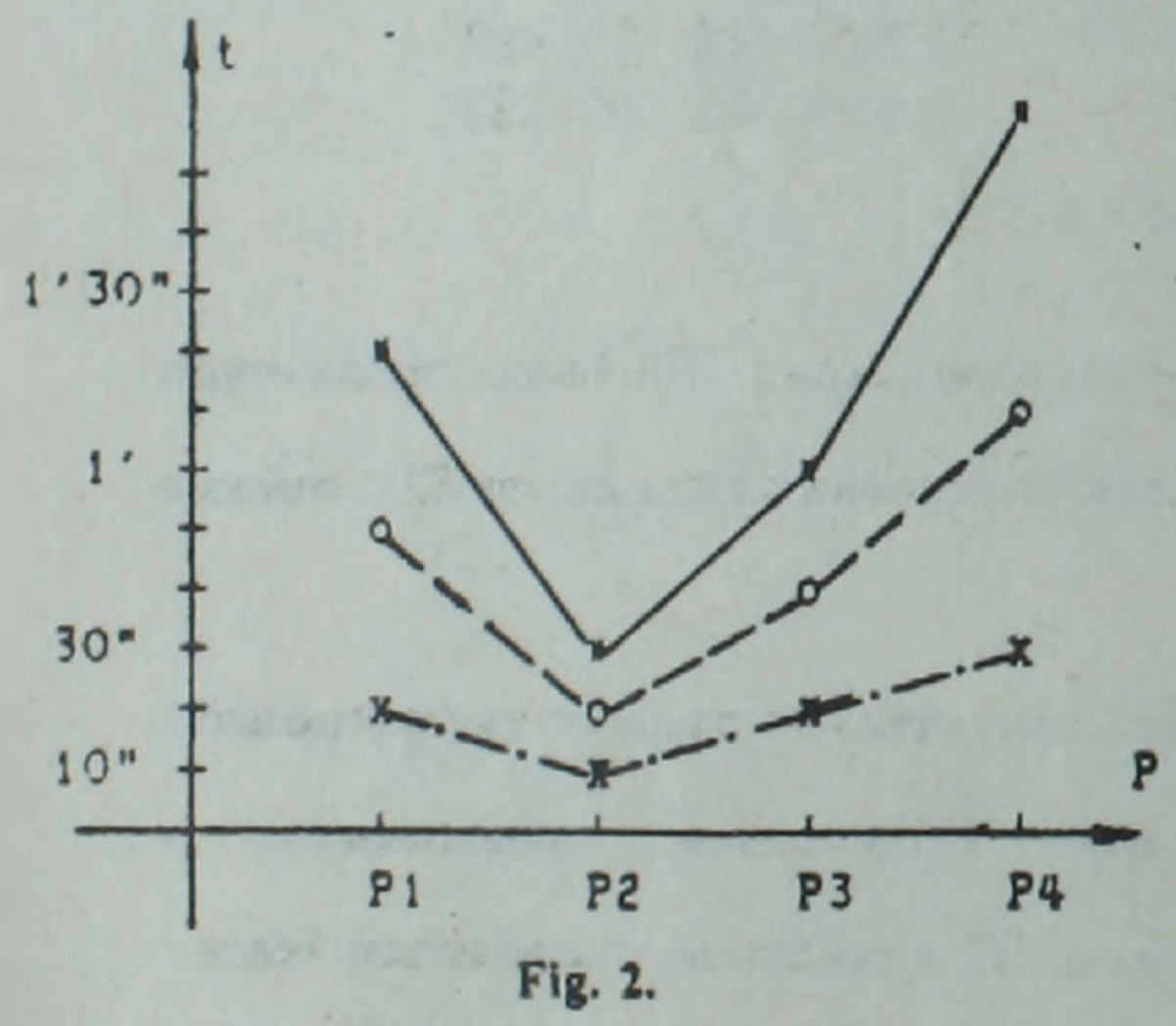

Fig. 2.

Рис. 2.

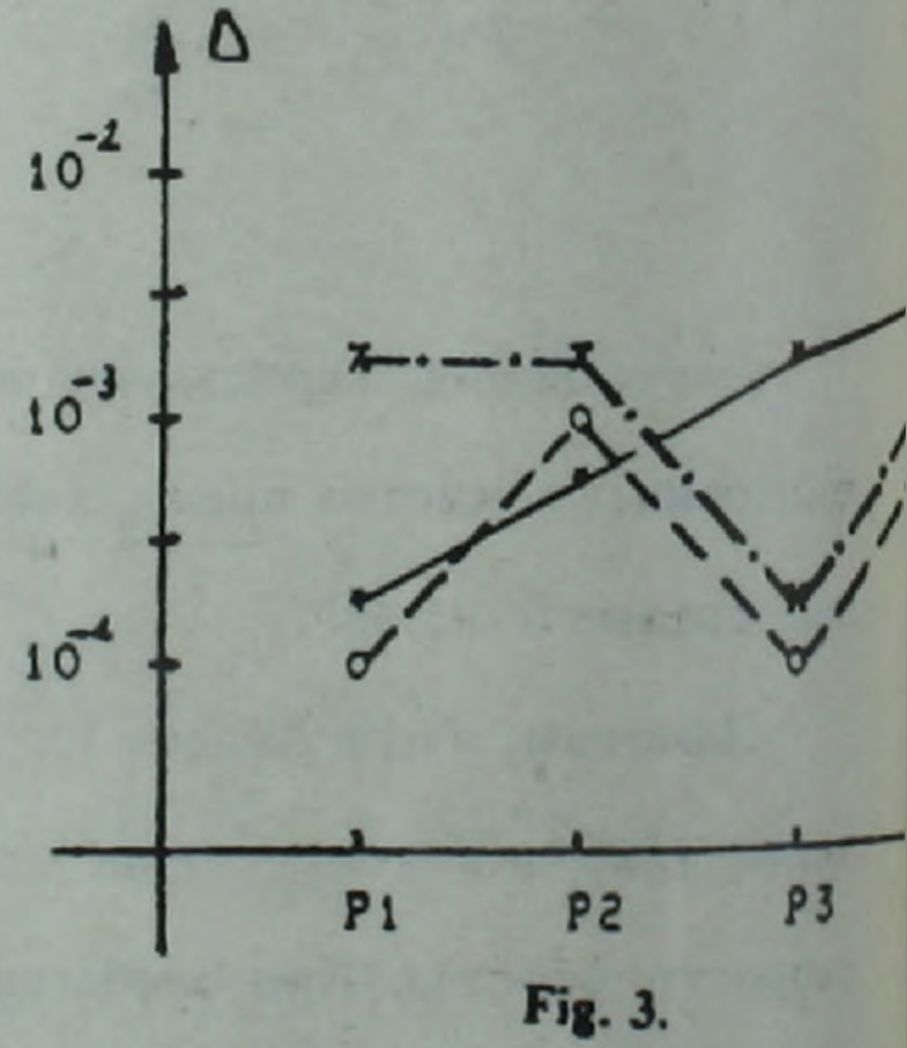

Fig. 3.

Рис. 3.

§4. ВЫВОДЫ

Результаты численных экспериментов дают возможность сделать следующие выводы :

а) автоматическое разбиение интервалов, используемое в алгоритме B, позволяет получить в областях сильной вариации y, y', y'' более густых сетей, чем

это получается в алгоритме A ;

b) алгоритм B дает вполне удовлетворительные результаты в трех из четырех шагов, тогда как алгоритм A обычно требует большего числа шагов ;

c) алгоритм C превосходит оба алгоритма (A и B) во всех основных характеристиках. Сплайн допускает нахождение приближенного решения на всем отрезке [0,1] и эффективен для выявления колебаний решения ;

d) при плохих (быстро осциллирующих) ядрах преимущество алгоритмов B и C над алгоритмом A особенно заметно (получается более высокая точность и сокращается время применения методов).

Для дальнейшего исследования промежуточное сгущение сетей можно сделать очень устойчивым, если применить эрмитовы сплайны (а также сплайны пятого порядка на заключительных этапах, так как можно вычислить значения $y''(x_i)$). Применение этой схемы внушает надежду для решения изменяющихся в процессе счета уравнений, для которых экономия времени вычисления особенно важна.

В заключение заметим, что наш подход можно применить для решения уравнений первого рода, используя метод регуляризации (см., например, [3]). Отметим, что при этом самосопряженность и положительная определенность интегрального оператора в (1), вероятно, повысит эффективность метода.

ABSTRACT. A numerical method for detection of sharp oscillations of solutions of Fredholm integral equations is suggested. The algorithm is a modified approach of the paper [1]. The improvement consists in an application of a more sensitive mechanism of grid condensation combined with the spline interpolation.

ЛИТЕРАТУРА

1. B. Neta, P. Nelson, " An adaptive method for numerical solution of Fredholm integral equation of the second kind. I. Regular kernels", Applied Math. and Computation, vol. 21, pp. 171–184, 1987.
2. С. Б. Стечкин, Ю. Н. Субботин, Сплайны в Вычислительной Математике, М., Наука, 1976.
3. А. Н. Тихонов, А. Б. Гончарский, Б. Б. Степанов, А. Г. Ягола, Численные Методы Решения Некорректных Залач, М., Наука, 1990.

2 Февраля 1992

Институт математики НАН Армении,
Ереванский государственный университет