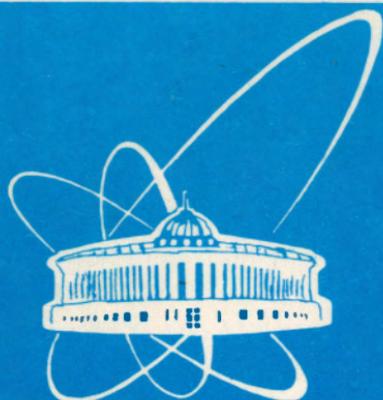


94-120



СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА

P11-94-120

Е.В.Земляная

**SYSINT (SYSINTM) — КОМПЛЕКС
ПРОГРАММ ДЛЯ ЧИСЛЕННОГО РЕШЕНИЯ
ЗАДАЧИ НА СОБСТВЕННЫЕ ЗНАЧЕНИЯ
ДЛЯ СИСТЕМЫ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ**

1994

ВВЕДЕНИЕ

Программный комплекс SYSINT(SYSINTM) предназначен для численного решения системы L интегральных уравнений следующего вида:

$$\vec{\Phi}(z) = \hat{Q}(z)\vec{\phi}(z) - \lambda \hat{R}(z)\vec{\phi}(z) + \int_0^R \hat{K}(z, z')\vec{\phi}(z')dz' = 0, \quad (1)$$

$$z = (\lambda, \vec{\phi}(z)), \quad \vec{\phi} = \{\phi_1(z), \phi_2(z), \dots, \phi_L(z)\}$$

с условием нормирования

$$\Gamma(z) = \int_0^R dz \sum_{i=1}^L \phi_i^2(z) - G = 0, \quad (2)$$

где

$$\hat{Q}(z) = \begin{pmatrix} Q_{11}(z) & \dots & Q_{1L}(z) \\ \dots & \dots & \dots \\ Q_{L1}(z) & \dots & Q_{LL}(z) \end{pmatrix}, \quad \hat{R}(z) = \begin{pmatrix} R_{11}(z) & \dots & R_{1L}(z) \\ \dots & \dots & \dots \\ R_{L1}(z) & \dots & R_{LL}(z) \end{pmatrix},$$
$$\hat{K}(z, z') = \begin{pmatrix} K_{11}(z, z') & \dots & K_{1L}(z, z') \\ \dots & \dots & \dots \\ K_{L1}(z, z') & \dots & K_{LL}(z, z') \end{pmatrix},$$

$Q_{im}(z), R_{im}(z), K_{im}(z, z')$ — заданные функции, обеспечивающие существование нетривиальных решений $z^* = (\lambda^*, \vec{\phi}^*(z))$.

Итерационная схема разработана на основе непрерывного аналога метода Ньютона (НАМН) [1] с использованием предложенной в [2] модификации, обеспечивающей существенный выигрыш во времени при расчетах на векторных вычислительных системах.

Программный комплекс был проверен на ряде тестовых задач и использовался для численного исследования уравнения Бете - Солпитера с потенциалом Гаусса в рамках модели кваркония [3].

В работе дается описание алгоритмов и параметров комплекса, рассматриваются особенности программной реализации и приводятся численные примеры, иллюстрирующие его работу.

1. ОПИСАНИЕ АЛГОРИТМОВ

1.1. Алгоритм программы SYSINT

Согласно подходу, определяемому обобщенным НАМН [1], нелинейное функциональное уравнение в B -пространстве

$$\mathcal{F}(x) \equiv \begin{pmatrix} \bar{\Phi}(x) \\ \Gamma(x) \end{pmatrix} = 0, \quad (3)$$

представляющее исследуемую задачу, заменяется эволюционным уравнением по непрерывному параметру t

$$\frac{d}{dt} \mathcal{F}(t, z(t)) = -\mathcal{F}(t, z(t)), \quad 0 \leq t < \infty \quad (4)$$

с начальным условием $z(0) = z_0$. Обозначив $\hat{A}(t) = \bar{\Phi}'_x(t, z(t))$, $z(t) = (\lambda(t), \bar{\phi}(x))$, получим для уравнения (1):

$$\hat{A}(t) \frac{d\bar{\phi}(x, t)}{dt} = \lambda'_i(t) \hat{R}(x) \bar{\phi}(x, t) - \bar{\Phi}(t, z(t)). \quad (5)$$

Вводя дискретную сетку по непрерывному параметру t : $\{0 = t_0 < t_1 < \dots < t_k < \dots\}$ и обозначая $z_k = z(t_k)$, $\lambda_k = \lambda(t_k)$, $\bar{\phi}_k(x) = \bar{\phi}(x, t_k)$, $\hat{A}_k = \hat{A}(t_k)$, $\bar{\Phi}_k = \bar{\Phi}(t_k, z(t_k))$, $\bar{v}_k(x) = \bar{\phi}'_i(x, t_k)$, $\mu_k = \lambda'_i(t_k)$, после аппроксимации задачи (5) по схеме Эйлера получаем формулы, определяющие итерационный процесс:

$$\begin{cases} \bar{\phi}_{k+1}(x) = \bar{\phi}_k(x) + \tau_k \bar{v}_k(x), \\ \lambda_{k+1} = \lambda_k + \tau_k \mu_k, \end{cases} \quad (6)$$

где

$$\bar{v}_k(x) = -\bar{v}_k^{(1)}(x) + \mu_k \bar{v}_k^{(2)}(x), \quad (7)$$

$$\hat{B}_k = \hat{A}_k^{-1}, \quad (8)$$

$$\bar{v}_k^{(1)}(x) = \hat{B}_k \bar{\Phi}_k, \quad (9)$$

$$\bar{v}_k^{(2)}(x) = \hat{B}_k \hat{R}(x) \bar{\phi}_k(x), \quad (10)$$

μ_k вычисляется из условия нормировки (2):

$$\mu_k = \frac{G - \int_0^R dx \sum_{i=1}^L (\phi_i)_k^2(x) + 2 \int_0^R dx \sum_{i=1}^L (v_i^{(1)})_k(x) (\phi_i)_k(x)}{2 \int_0^R dx \sum_{i=1}^L (v_i^{(2)})_k(x) (\phi_i)_k(x)}. \quad (11)$$

Вычисляя для каждого значения t_k итерационные поправки $\bar{v}_k(x)$, μ_k и шаг τ_k , получаем новое приближение x_{k+1} к решению x^* .

Итерационный процесс должен продолжаться до тех пор, пока не будет выполнено соотношение

$$\delta_k = \|\bar{\Phi}_k\| \leq \epsilon, \quad (12)$$

где $\epsilon > 0$ — заранее заданное малое число, а невязка δ_k может вычисляться по одной из формул:

$$\delta_k = \max_{i=1, L} \max_x |(\Phi_i)_k| \quad (13)$$

или

$$\delta_k = \int_0^R dx \sum_{i=1}^L (\Phi_i)_k^2. \quad (14)$$

1.2. Алгоритм программы SYSINTM

Перейдем к описанию модифицированного алгоритма [2]. Вместо уравнения (3) рассматривается система следующего вида:

$$\begin{cases} \mathcal{F}(x) = 0, \\ \mathcal{B}\mathcal{A} - \mathcal{I} = 0, \end{cases} \quad (15)$$

где $\mathcal{A} = \mathcal{F}'_x(x)$, $\mathcal{B} = \mathcal{A}^{-1}$, \mathcal{I} — единичный оператор.

Вводя непрерывный параметр t ($0 \leq t < \infty$) и переходя к системе эволюционных уравнений, получаем:

$$\begin{cases} \frac{d}{dt} \mathcal{F}(t, x(t)) = -\mathcal{F}(t, x(t)), \\ \frac{d}{dt} [\mathcal{B}(t)\mathcal{A}(t) - \mathcal{I}] = \mathcal{I} - \mathcal{B}(t)\mathcal{A}(t). \end{cases} \quad (16)$$

В соответствии с (16) для уравнения (1) имеем:

$$\begin{cases} \dot{\lambda} \frac{d\bar{\phi}(x, t)}{dt} = -\lambda'_i(t) \dot{R}(x) \bar{\phi}(x, t) - \bar{\Phi}(t, x(t)), \\ \frac{d}{dt} [\hat{B}(t)\hat{A}(t) - \hat{I}] = \hat{I} - \hat{B}(t)\hat{A}(t). \end{cases} \quad (17)$$

После дискретизации непрерывного параметра t получаем на основе схемы Эйлера следующую итерационную схему:

$$\begin{cases} \bar{\phi}_{k+1}(x) = \bar{\phi}_k(x) + \tau_k \bar{v}_k(x), \\ \lambda_{k+1} = \lambda_k + \tau_k \mu_k, \\ \hat{B}_{k+1} = \hat{B}_k + \tau_k \hat{W}_k, \end{cases} \quad (18)$$

где $\bar{v}_k(x)$ вычисляется с помощью формул (7),(9),(10), а μ_k определяется по формуле (11).

Формула для итерационной поправки \hat{W}_k имеет вид:

$$\hat{W}_k = [\hat{I} - \hat{B}_k(\hat{A}_k + \hat{A}'_k)]\hat{B}_k. \quad (19)$$

\hat{W}_k может также вычисляться по более простой формуле, полученной на основе работы [4]:

$$\hat{W}_k = [\hat{I} - \hat{B}_k\hat{A}_k]\hat{B}_k. \quad (20)$$

Однако при этом, как показали вычисления, сужается область сходимости итерационной схемы.

Таким образом, имея начальное приближение z_0 и \hat{B}_0 , можно последовательно найти все приближения z_k и \hat{B}_k . Итерационный процесс продолжается до выполнения неравенства (12). Расчеты показывают, что в качестве начального приближения \hat{B}_0 лучше всего использовать $\hat{B}_0 = \hat{A}^{-1}(z_0)$.

1.3. Алгоритмы вычисления параметра τ_k

Шаг τ_k по непрерывному параметру l может вычисляться по одному из следующих алгоритмов [5]:

1.

$$\tau_k = \tau_0 = \text{const}, \quad 0 < \tau_0 \leq 1,$$

2.

$$\tau_k = \begin{cases} \min(1, 2\tau_{k-1}), & \delta_k \leq \delta_{k-1}, \\ \max(\tau_0, \tau_{k-1}/2), & \delta_k > \delta_{k-1}, \end{cases}$$

3.

$$\tau_k = \begin{cases} \min(1, \tau_{k-1} \frac{\delta_{k-1}}{\delta_k}), & \delta_k \leq \delta_{k-1}, \\ \max(\tau_0, \tau_{k-1} \frac{\delta_{k-1}}{\delta_k}), & \delta_k > \delta_{k-1}, \end{cases}$$

4.

$$\tau_k = \max(\tau_0, \frac{\delta_{k-1}}{\delta_k(1) + \delta_{k-1}}),$$

где $\delta_k(1)$ — невязка на k -й итерации для $\tau_k = 1$.

5. На равномерной сетке ω_τ отрезка $[0, 1]$ с шагом $\Delta\tau$ вычисляется последовательность невязок δ_k и выбирается значение τ_k , которому соответствует минимальное значение невязки.

В алгоритмах 2,3,5 для вычисления δ_k используется формула (13), в алгоритме 4 невязка δ_k вычисляется по формуле (14).

2. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

2.1. Дискретное представление

Аппроксимация системы (1) на дискретной сетке по аргументу x , ($0 = x_1, x_2, \dots, x_N = R$) с шагом $h_i = x_{i+1} - x_i$ имеет в точке x_i ($i = 1, \dots, N$) следующий вид:

$$\begin{aligned} \Phi_i(\lambda, \psi(x_i)) &= \sum_{m=1}^L (Q_{lm}(x_i) - \lambda R_{lm}(x_i)) \phi_m(x_i) + \\ &+ \sum_{m=1}^L \sum_{j=1}^N K_{lm}(x_i, x_j) \phi_m(x_j) \xi_j = 0, \quad l = 1, \dots, L. \end{aligned}$$

Коэффициенты ξ_j зависят от способа численного интегрирования. При этом порядок сходимости численного решения x_k к точному решению x^* соответствует порядку точности выбранной квадратурной формулы, что подтверждается расчетами на последовательности сходящихся сеток [2].

Программа написана на языке FORTRAN, реализована на ЭВМ VAX-8350, PC AT/XT-386, CONVEX C120 и SUN SPARK STATION.

Для вычисления $\hat{B}_k = \hat{A}_k^{-1}$ (формула (8)) в программе SYSINT используется подпрограмма MATIN2 из библиотеки CERNLIB [6].

2.2. Подпрограммы пользователя

Подпрограмма SUBROUTINE KSI(N,X,TKSI), предназначенная для вычисления коэффициентов ξ_j , должна быть составлена пользователем. Здесь N и X — соответственно число точек и массив узлов дискретной сетки по аргументу x , TKSI — массив коэффициентов ξ_j . Массивы X и TKSI имеют размерность N.

Пользователь должен также составить и включить в комплекс следующие подпрограммы - функции:

QQ(XI,L,M) для вычисления $Q_{lm}(x_i)$,

RR(XI,L,M) для вычисления $R_{lm}(x_i)$,

YK(XI,XJ,L,M) для вычисления $K_{lm}(x_i, x_j)$.

2.3. Описание параметров программ SYSINT и SYSINTM

Обращение к комплексам SYSINT и SYSINTM осуществляется операторами

CALL SYSINT (N, LM, LMN, X, EV, FI, TAU0, NTAU, GNORM, EPS, NMAX, F, V, V1, R1, R2, R3, B),

CALL SYSINTM (N, LM, LMN, X, EV, FI, TAU0, NTAU, GNORM, EPS, NMAX, F, V, V1, R1, R2, R3, B, W, A, NMOD).

N — число узлов сетки по аргументу λ .

LM — число уравнений системы (1).

LMN — размерность рабочих массивов, $LMN = N \cdot LM$.

X — массив узлов сетки по x размерности N.

EV — собственное значение λ . При обращении к программе этому параметру присваивается начальное приближение для собственного значения λ_0 , после окончания работы программы здесь находится k -е приближение λ_k .

FI — двумерный массив размерности $LM \times N$. При обращении к программе в нем задается начальное приближение $(\phi_l)_0(x_i)$, ($l = 1, \dots, L$, $i = 1, \dots, N$), на выходе здесь находится полученное решение $(\phi_l)_k(x_i)$.

TAU0 — начальное значение шага τ_0 .

NTAU — номер алгоритма вычисления τ_k по п.1.3.

GNORM — значение G из условия нормировки (2).

EPS — заданное малое число $\epsilon > 0$ из соотношения (12).

NMAX — максимально допустимое число итераций k . При его превышении происходит выход из программы.

F, V, V1, R1, R2, R3, B — рабочие массивы. Массивы F, V, V1, R3 имеют размерность LMN. Массивы R1, R2 имеют размерность N. Двумерный массив B имеет размерность $LM \times N$.

W, A — дополнительные двумерные рабочие массивы программы SYSINTM, имеющие размерность $LM \times N$.

NMOD — параметр модифицированного алгоритма. При NMOD=1 итерационная поправка \hat{W}_k должна вычисляться по формуле (19). Если NMOD=2, то \hat{W}_k вычисляется по формуле (20).

3. ЧИСЛЕННЫЕ РЕЗУЛЬТАТЫ

3.1. Случай одного уравнения

Уравнение Шредингера в импульсном представлении для кулоновского взаимодействия [7] представляет собой задачу на собственные

значения для одного интегрального уравнения

$$\Phi(z) \equiv (p^2 - \lambda)y(p) - 2\bar{Z} \int_0^{\infty} dp' \ln \left| \frac{p+p'}{p-p'} \right| y(p') = 0 \quad (21)$$

с условием нормировки $\int_0^{\infty} y^2(p) dp = 1$,

где $z = (y(p), \lambda)$, $\lambda = 2E$, λ — собственные значения, E — уровни энергии. Одно из аналитических решений при $\bar{Z} = 1$ имеет вид:

$$\lambda_1^* = -1, \quad y_1^*(p) = \sqrt{\frac{2}{\pi}} \frac{p}{(p^2 + 1)^2}.$$

Решение этого уравнения подробно рассматривается в работе [2].

В таблице 1 представлены значения λ_k и δ_k , полученные в результате работы программ SYSINT и SYSINTM. Вычисления проводились на равномерной сетке по аргументу p при $R=30$, $N=301$, $EPS=0.0001$, $N\tau U=3$, $\tau U=0.2$, $\lambda_0 = -0.9$, $\phi_0(x) = x \cdot \exp(-x)$. Коэффициенты $\{\xi_j\}$ выбраны в соответствии с формулой Грегори [8]. Начальное значение невязки $\delta_0 \approx 0.1$.

ТАБЛИЦА 1

PROGRAM	k	λ_k	δ_k
SYSINT	5	-1.0001	$1.06E^{-5}$
SYSINTM	5	-1.000098	$1.4E^{-5}$

Оба алгоритма, как видно из таблицы 1, при одних и тех же значениях входных параметров имеют после одинакового числа итераций k близкие значения λ_k и δ_k .

3.2. Тестовая задача для случая двух уравнений

Тестовая задача для системы (1) при $L = 2$ имеет следующий вид:

$$\hat{Q}(x) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \hat{R}(x) = \begin{pmatrix} 1 & 0 \\ 0 & 1 + x - \frac{1}{x} \end{pmatrix}, \quad \hat{K}(x, x') = \begin{pmatrix} \frac{x'}{x^3} & -1 \\ \frac{x}{x'} & \frac{-x}{x'^2} \end{pmatrix}. \quad (22)$$

Условие нормировки $\int_0^R dx (\phi_1^2(x) + \phi_2^2(x)) - G = 0$, $G = 1/3 + 1/5$, $R = 1$.

Аналитическое решение задачи (22) $\lambda^* = 1$, $\phi_1^*(x) = x$, $\phi_2^*(x) = x^2$. Вычисления проводились на равномерной сетке по x с шагом $h = 0.05$.

Коэффициенты $\{\epsilon_j\}$ вычислялись в соответствии с формулой Симпсона [8]. Результаты счета при начальном приближении $\lambda_0 = 1.01$, $\phi_{10}(x) = \phi_1^*(x \cdot 0.01)$, NTAU=3, TAU0=0.1, EPS=0.0001 приведены в таблице 2. Начальное значение поправки $\delta_0 \approx 24.2$.

ТАБЛИЦА 2

PROGRAM	k	λ_k	δ_k
SYSINT	13	1.00009	$6.6E^{-5}$
SYSINTM	13	1.00003	$8.7E^{-5}$

3.3. Задача Бете - Солпитера

Система уравнений Бете - Солпитера для псевдоскалярных мезонов с гауссовским потенциалом имеет следующий вид [3]:

$$MU_{(\frac{1}{2})}(\rho) = E_i(\rho)U_{(\frac{1}{2})}(\rho) - \quad (23)$$

$$-2 \int_0^{\infty} dq [C_p^{(+)} C_q^{(-)} V_1(\rho, q) + S_p^{(-)} S_q^{(+)} V_2(\rho, q)] U_{(\frac{1}{2})}(q),$$

где

$$V_1(\rho, q) = \beta [\exp(-\beta^2(\rho^2 + q^2)) \operatorname{sh}(2\beta^2 \rho q)],$$

$$V_2(\rho, q) = \frac{1}{2\beta} [\exp(-\beta^2(\rho^2 + q^2))(2\beta^2 \rho q \operatorname{ch}(2\beta^2 \rho q) - \operatorname{sh}(2\beta^2 \rho q)],$$

$$C_p^{(\pm)} = \cos(v_1(\rho) \pm v_2(\rho)), \quad S_p^{(\pm)} = \sin(v_1(\rho) \pm v_2(\rho)),$$

$v_1(\rho), v_2(\rho)$ и $E_1(\rho), E_2(\rho)$ - решения уравнения Швингера - Дайсона [3] для кварка и антикварка с массами m_{01} и m_{02} , $E_i(\rho) = E_1(\rho) + E_2(\rho)$ - полная энергия мезона, M - собственное значение (масса связанного состояния), $U_{(\frac{1}{2})}$ - волновые функции мезона. Условие нормировки имеет вид:

$$\frac{4N_C}{M} \frac{1}{(2\pi)^3} \int dq U_1(q) U_2(q) = 1, \quad N_C = 12.$$

В таблице 3 приведены результаты работы программ SYSINT и SYSINTM при следующих значениях параметров задачи: $m_{01} = m_{02} = 0.1$, $\beta = 3$. Вычисления выполнены на интервале $[0, R]$, $R = 5$ с равномерным шагом дискретной сетки $h = 0.1$ при NTAU=3, TAU0=0.1, EPS=0.0001. В качестве начального приближения используется $M_0 =$

0.5, $U_{0(\frac{1}{2})}(p) = p \cdot \exp(-p)$. Начальное значение невязки при этом $\delta_0 \approx 1.2$.

ТАБЛИЦА 3

PROGRAM	k	λ_k	δ_k
SYSINT	10	0.5776	$2.04E^{-5}$
SYSINTM	10	0.5776	$2.53E^{-5}$

3.4. Сравнительный анализ временных характеристик

Из сравнения описанных в п.1.1 и в п.1.2 итерационных схем видно, что в модифицированном алгоритме одно обращение матрицы \hat{A} заменяется на два матричных умножения. Так как в скалярном варианте реализации время одного обращения матрицы близко ко времени умножения двух матриц [8], то при одинаковых входных данных время работы программы SYSINTM должно быть примерно в два раза больше времени программы SYSINT. Однако с точки зрения векторизации операций умножение матриц более предпочтительно, чем обращение матрицы. Поэтому при работе на векторных вычислительных системах можно ожидать, что программа SYSINTM будет работать быстрее. Следует отметить, что обе программы используют соответствующие стандартные программы линейной алгебры без каких-либо специальных оптимизаций применительно к операционным системам и особенностям различных ЭВМ.

В таблице 4 для задач (21), (22) и (23) приведены времена счета на ЭВМ VAX-8350 и CONVEX C120 с использованием и без использования векторизации, а также отношение T/Твект (для CONVEX C120), характеризующее свойство "ускорения вычислений" этих схем.

ТАБЛИЦА 4

Задача	Program	T(vax)	T(convex)	Твект(convex)	T/Твект
(21)	SYSINT	1'17"	25.33"	11.68"	≈ 2.1
(21)	SYSINTM	2'19"	72.47"	8.02"	≈ 9.1
(22)	SYSINT	1'31"	38.51"	15.75"	≈ 2.4
(22)	SYSINTM	3'01"	100.47"	9.07"	≈ 11.2
(23)	SYSINT	5'06"	101.60"	69.84"	≈ 1.4
(23)	SYSINTM	9'22"	232.01"	47.06"	≈ 5.0

Из данных таблицы 4 можно сделать вывод, что по временным характеристикам модифицированный алгоритм программы SYSINTM является более эффективным для векторной вычислительной системы.

Эффективность программы может быть значительно повышена при использовании специальных приемов векторизации и, особенно, распараллеливания вычислений на многопроцессорных системах.

Автор благодарит Пузынина И.В. и Пузынину Т.П. за поддержку, внимание к работе, полезные советы и замечания.

ЛИТЕРАТУРА

1. Жанлаев Т., Пузынин И.В. — ЖВММФ, 1992,32, 1, с.3;
Жанлаев Т., Пузынин И.В. — ЖВММФ, 1992,32, 6, с.846.
2. Puzynin L.V., Amirkhanov I.V., Puzynina T.P., Zemlyanaya E.V.
— JINR Rapid Comm. 5[62]-93, Dubna, 1993, с.63.
3. Амирханов И.В., Земляная Е.В., Первушин В.Н., Пузынин И.В.,
Пузынина Т.П., Сариев Н.А., Стриж Т.А. — ОИЯИ, P11-94-74,
Дубна, 1994.
4. Давиденко Д.Ф. — ИАЭ им.И.В.Курчатова, ИАЭ-1963, Москва,
1970.
5. Пузынин И.В., Пузынина Т.П., Стриж Т.А. — ОИЯИ, P11-87-332,
Дубна, 1987; Пузынина Т.П. — ОИЯИ, P11-89-728, Дубна, 1989;
Амирханов И.В., Земляная Е.В., Пузынина Т.П. — ОИЯИ, P11-
91-87, Дубна, 1991.
6. CERN Computer Centre, Program Library, Geneva, 1988.
7. Бете Г., Солпитер Э.; Квантовая механика атомов с одним и двумя
электронами. — М.: Физматгиз, 1960.
8. Каниткин Н.Н. Численные методы. М.: Наука, 1978.

Рукопись поступила в издательский отдел
6 апреля 1994 года.