



сообщения
Объединенного
института
ядерных
исследований
Дубна

970 / 2-80

3/3-80
P5 - 12856

В.Ц.Банчев

ПРОГРАММА ДЛЯ ИНТЕГРИРОВАНИЯ
ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ
УРАВНЕНИЙ
С ПОМОЩЬЮ
ОБОБЩЕННОГО МЕТОДА РУНГЕ-КУТТА

1979

P5 - 12856

В.Ц.Банчев

ПРОГРАММА ДЛЯ ИНТЕГРИРОВАНИЯ
ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ
УРАВНЕНИЙ
С ПОМОЩЬЮ
ОБОБЩЕННОГО МЕТОДА РУНГЕ-КУТТА



Банчев В.Ц.

P5 - 12856

Программа для интегрирования обыкновенных дифференциальных уравнений с помощью обобщенного метода Рунге-Кутты

В описываемой программе реализован обобщенный метод Рунге-Кутты, предложенный автором для численного интегрирования задачи Коши для жестких систем обыкновенных дифференциальных уравнений. Особенно удачным является ее использование при нахождении асимптотически устойчивых точек покоя. К этой задаче сводятся многие задачи науки и техники, связанные с исследованием переходных процессов, задача безусловной минимизации функций с помощью непрерывных аналогов различных итерационных процессов и задача аппроксимации некоторых уравнений в частных производных обыкновенными дифференциальными уравнениями.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИАИ.

Сообщение Объединенного института ядерных исследований. Дубна 1979

Banchev V.Ts.

P5 - 12856

A Subroutine for Integration of Ordinary Differential Equations by Means of a Generalized Runge-Kutta Method

The subroutine implements the generalized Runge-Kutta method, suggested by the author for the numerical integration of the Cauchy problem for stiff systems of ordinary differential equations. Its use in finding asymptotically stable equilibrium points is particularly effective. Many scientific and technical problems concerned with the investigation of transient processes, the problem of unconstrained function minimization by means of continuous analogues of different iteration processes, as well as the problem of replacing some partial differential equations by systems of ordinary differential equations reduce to this problem.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1979

В программе GERUN реализованы алгоритмы 1 и 2 обобщенного метода Рунге-Кутты, предложенного автором /1,2/ для численного интегрирования задачи Коши для нелинейной жесткой системы обыкновенных дифференциальных уравнений

$$y' = f(x, y), \quad y(x_0) = y_0, \quad x \in [x_0, x_f], \quad /1/$$

где $y = (y^1, y^2, \dots, y^N)^T$ и $f = (f^1, f^2, \dots, f^N)^T$ - N-мерные векторы /первоначальный вариант программы GERUN, в основу которого положен метод Лоусона /3/, опубликован в /4/ /. Особенно удачным является использование GERUN для нахождения асимптотически устойчивых точек покоя. К этой задаче сводятся многие задачи науки и техники, связанные с исследованием переходных процессов, задача безусловной минимизации функций с помощью непрерывных аналогов различных итерационных процессов и задача аппроксимации некоторых уравнений в частных производных обыкновенными дифференциальными уравнениями.

Для ознакомления с алгоритмом программы GERUN отсылаем читателя к /1,2/ /для сравнения см. /14,17/ /. Здесь приводится только описание программы и способ ее использования для минимизации функций без ограничений с помощью непрерывного аналога метода градиента.

ОПИСАНИЕ ПРОГРАММЫ

Программа GERUN написана на основе той версии языка ФОРТРАН IV, которая реализована в операционных системах ДОС и ОС ЕС ЭВМ /5/, с двойной точностью. Ее можно непосредственно применять на ЭВМ серии ЕС и IBM.

Формальные аргументы GERUN и параметры ее общих блоков GOV1, ..., GOV9 будем называть в дальнейшем параметрами управления. С их помощью можно реализовать различные модификации алгоритма GERUN. Для упрощения процедуры задания модификаций программы GERUN для некоторых параметров управления предусмотрено автоматическое присваивание стандартных значений. Этой цели служат подпрограммы BLOCK DATA

и INIZ. Пользователь должен вызвать подпрограмму INIZ перед первым обращением к GERUN.

Для вычисления экспоненты матрицы A с помощью алгоритма 1 обобщенного метода Рунге-Кутты в GERUN используются следующие три способа /подробнее о вычислении матричной экспоненты см. /6./ /:

1. В общем случае матричная экспонента аппроксимируется с помощью разложения показательной функции в непрерывную дробь /7,8./

2. Если матрица A имеет полный набор линейно-независимых собственных векторов, матричная экспонента вычисляется с помощью собственных значений и собственных векторов A . Для этого служат подпрограммы BALANC, ELMHES /либо ORTHES/, ELTRAN /либо ORTRAN/, HQR2 и BALBAK /9./;

3. Экспонента симметрической матрицы A вычисляется с помощью подпрограмм TRED2 и TQL2 /9./.

Явные формулы Рунге-Кутты, которые положены в основу алгоритма GERUN, - это так называемые формулы-пары /10-13/.

Они состоят из двух формул, которые на каждом шагу используют один и тот же набор значений правой части $f(x, y)$ в /1/, при этом одна из формул имеет порядок точности p , а другая - $p+q$ / $q \geq 1$ / /говорят еще, что первая из формул вложена во вторую/. Разность между значениями, полученными с помощью этих формул, дает оценку локальной ошибки усечения. Основное достоинство методов вложения /т.е. тех методов, которые используют формулы-пары/ состоит в том, что они уменьшают необходимое количество вычислений по сравнению с обычными методами Рунге-Кутты и при этом получаемая оценка локальной ошибки усечения достаточно эффективна. В настоящем варианте программы GERUN используются четыре формулы-пары 1-го, 2-го, 3-го /12, с.81/ и 5-го порядков точности /11/. Их параметры задаются в подпрограмме INIZ /INIZ может быть заменена, если пользователь хочет применить другие формулы-пары/.

Остановимся подробнее на оценке локальной ошибки усечения. Программа GERUN оценивает абсолютную либо относительную локальную ошибку усечения вышеупомянутыми методами вложения. Однако в случае, когда в алгоритме 1 обобщенного метода Рунге-Кутты матричная экспонента аппроксимируется с помощью разложения показательной функции в непрерывную дробь, полученная таким образом оценка верна только тогда, когда ошибкой аппроксимации матричной экспоненты можно пренебречь. В противном случае локальная ошибка усечения оценивается следующим образом /в GERUN не учитываются ошибки при вычислении матричной экспоненты остальными двумя спосо-

бами, так как предполагается, что они достаточно малы/. Запишем численное решение задачи Коши /1/ в виде

$$y_{n+1}^i = P_h^{m,i}(y_n), \quad i = 1, 2, \dots, N,$$

где m - порядок аппроксимации матричной экспоненты, а h и n - величина и номер шага. Пусть $y(t)$ - точное решение /1/. Тогда относительная локальная ошибка усечения r_n на n -ом шаге задается равенством

$$r_n^i = [y^i(x_{n+1}) - P_h^{m,i}(y(x_n))] / [|y^i(x_{n+1})| + \epsilon], \quad i = 1, 2, \dots, N, \quad /2/$$

где ϵ - некоторое малое положительное число / ϵ вводится для того, чтобы избежать деления на нуль/. Имеем

$$\|r_n\| = \max_{1 \leq i \leq N} |r_n^i| \leq r_n^h + r_n^m, \quad /3/$$

где

$$r_n^h = \max_{1 \leq i \leq N} |y^i(x_{n+1}) - P_h^{\infty,i}(y(x_n))| / [|y^i(x_{n+1})| + \epsilon]$$

можно оценить приближенно методом вложения, а

$$r_n^m = \max_{1 \leq i \leq N} |P_h^{\infty,i}(y(x_n)) - P_h^{m,i}(y(x_n))| / [|y^i(x_{n+1})| + \epsilon]$$

оценивается величиной

$$\max_{1 \leq i \leq N} |P_h^{m+1,i}(y(x_n)) - P_h^{m,i}(y(x_n))| / [|y^i(x_{n+1})| + \epsilon]$$

/ $m = \infty$ означает, что матричная экспонента задается точно/. В силу свойств используемых аппроксимаций матричной экспоненты /7/ эта оценка относительной локальной ошибки усечения является эффективной при достаточно малом h , что доказано вычислительной практикой на ЭВМ /8/. Аналогичным образом оценивается и абсолютная локальная ошибка усечения.

Заметим, что если в алгоритме 1 матричная экспонента вычисляется с помощью первого из перечисленных выше способов, оценка локальной ошибки усечения служит для автоматического выбора не только величины шага h , но и порядка m аппроксимации матричной экспоненты /8/. Это происходит следующим образом. Задаются некоторые начальные значения параметров h и m . Для достижения заданной величины оценки равномерной нормы локальной ошибки усечения r_n сначала увеличивается порядок аппроксимации m до некоторого m_{\max} или до тех пор, пока

$$\gamma_n^m \leq \delta \gamma_n^h \quad (0 \leq \delta \leq 1), \quad /4/$$

после чего, в случае необходимости, шаг уменьшается, а m принимает свое начальное значение. Этот цикл повторяется, пока оценка нормы локальной ошибки усечения не будет меньше заданной.

Кроме обычного контроля локальной ошибки усечения в GERUN предусмотрена еще возможность следующего контроля /подробнее см. /15/ /:

$$\|e_n\| \leq \beta \exp(-\gamma h_n) \|y_{n+1} - y_n\|, \quad /5/$$

где $\|e_n\|$ - равномерная норма абсолютной локальной ошибки усечения, $0 \leq \beta < 1/2$, $\gamma \geq 0$. Этот способ контроля локальной ошибки усечения является особенно эффективным для нахождения асимптотически устойчивых точек покоя.

Алгоритм 2 обобщенного метода Рунге-Кутты рекомендуется применять на участках быстрого изменения решения задачи Коши /1/ при условии, что вычисление соответствующих производных правых частей в /1/ не требует больших затрат машинного времени и памяти. Можно показать, что этот алгоритм увеличивает порядок точности используемой явной формулы Рунге-Кутты.

Различные модификации алгоритма GERUN позволяют: а/ находить численное решение задачи Коши /1/ с помощью алгоритмов 1 или 2 обобщенного метода Рунге-Кутты с оценкой локальной ошибки усечения методом вложения и с автоматическим выбором шага /без учета или с учетом ошибки аппроксимации матричной экспоненты и с автоматическим выбором порядка аппроксимации матричной экспоненты в случае, когда в алгоритме 1 используется первый способ вычисления экспоненты матрицы A/; б/ решать /1/ для жестких систем, используя только явные формулы-пары Рунге-Кутты с оценкой локальной ошибки усечения и с автоматическим выбором шага.

Кроме того, с целью уменьшения затрат машинного времени и памяти для алгоритма 1 в GERUN предусмотрены возможности использования:

- 1/ диагональной матрицы A;
- 2/ квазидиагональной матрицы A;

$$A = \text{diag} \{ A_1, A_2, \dots, A_s \}, \quad /6/$$

где A_i - квадратные матрицы порядка $m_i, i=1, \dots, s$ ($\sum_{i=1}^s m_i = N$);

3/ матрицы A вида /6/ с $A_i=0, i=2, \dots, s$ /матрицу такого вида естественно применять в случае разделения системы /1/ на жесткую и нежесткую части/.

GERUN использует 43 подпрограммы. Подпрограмма EXCON вычисляет аппроксимацию матричной экспоненты с помощью разложения показательной функции в непрерывную дробь /8/. Для этого она вызывает подпрограмму INVITR, которая с помощью подпрограмм DECOM, ITERIN, FBSUBM и VIPDA обращает матрицы F_{2m+1} в /5/ из /8/. Точность обращения матрицы F_{2m+1} оценивается следующим образом. Пусть

$$p_{ik} = \sum_{\ell=1}^N (F_{2m+1})_{i\ell} (F_{2m+1})_{\ell k} - \delta_{ik},$$

где δ_{ik} - символ Кронекера. Если $\max_{i,k} |p_{ik}| \leq 10^{-12}$, считается, что точность обращения удовлетворительная. Выдача информации осуществляется в OUTF. Подпрограммы VIPD, FMMX, FMTMX, FMVX, FMTVX, FMC, FMMA, FMEQ, SMAXA, VADD, VLINE, VSUB, VEQ, VMOD, VMAXA, FNORM выполняют различные действия с матрицами и векторами. SYMVX, EIMVX, PAMVX и DISEIG реализуются в алгоритме 1, а TS - в алгоритме 2. Описание подпрограмм TRED2, TQL2, BALANC, ELMHES, ORTHES, ELTRAN, ORTRAN, HQR2 и BALBAK дано в /9/. Кроме того, GERUN использует подпрограммы FUN и AMAT, которые должен написать пользователь /см. ниже/. Подпрограммы GLESON и FMTMX включены в GERUN с целью облегчения задачи составления подпрограммы AMAT.

Ниже приводится описание параметров управления GERUN. Везде положено

$$NW = \begin{cases} MQ, & \text{если } MBLD(1) \geq N, \\ \max_i MBLD(i), & \text{если } 0 < MBLD(1) < N. \end{cases}$$

N - число уравнений в /1/.

H - величина шага численного интегрирования. На выход в H засылается величина шага, которую рекомендуется изменить при следующем обращении к программе.

X - независимая переменная, изменяемая программой после каждого шага; задает x_0 в /1/ при первом обращении.

Y - одномерный массив размерности N, содержащий значения зависимых переменных в точке X.

P, G, GN - одномерные рабочие массивы размерности N.

S - двумерный рабочий массив размерности (N, NSTAG). На выходе первый столбец S(1, I) (I=1, ..., N) содержит приращения зависимых переменных, вычисленные на последнем шаге.

NC: NC=NW в случае, когда используется алгоритм 1 обобщенного метода Рунге-Кутты и первый способ вычисления матричной экспоненты. В остальных случаях можно положить NC=1.

CC,PC,F,B - одномерные рабочие массивы размерности N, которые нужны в случае применения обобщенного метода Рунге-Кутты, при этом CC и PC используются только в алгоритме 1. Кроме того, CC и PC не используются, если матричная экспонента вычисляется первым способом и KMAX=KIN. PC не применяется также в случае, когда матричная экспонента вычисляется третьим способом.

NA. NA > NW.
NEX. NEX > NW.

A,A2,F1,F2,G1,G2,FINV,EXC,R - двумерные рабочие массивы размерности A(NA,NW) /если NTS>0 - размерности A(NA,NTS) /, A2(NA,NW), F1(NC,NC), G1(NC,NC), G2(NC,NC), FINV(NEX,NW), EXC(NEX,NW), R(NEX,NW), F2(NEX,NW).

В случае:

а/NTS>0 либо NTS≤0, KDIAG=0, KSYM≠0 применяется только массив A;

б/NTS≤0, KDIAG=0, KSYN=0, KEIGV≠0 - массивы F2,FINV, EXC,R, если KEIGV<0, KSUP<0 - только FINV и R;

в/NTS≤0, KDIAG=0, KSYM=0, KEIGV=0, KIN>0 используются все массивы;

г/ если задача Коши /1/ решается с помощью обычного явного метода Рунге-Кутты /NTS≤0, KDIAG=KSYM=KEIGV=KIN=0/ либо если NTS≤0 и KDIAG≠0, двумерные рабочие массивы не используются.

FUN - подпрограмма, составляемая пользователем для вычисления правых частей f(x,y) в /1/. Способ составления следующий:

SUBROUTINE FUN(X, Y, F, N),

IMPLICIT REAL*8 (A-H, O-Z),

DIMENSION Y(N), F(N),

где параметры X,Y задают значения x,y, а F содержит вычисленные значения f(x,y). FUN должна быть объявлена EXTERNAL в программе, вызывающей GERUN.

AMAT - подпрограмма, составляемая пользователем, с помощью которой задаются матрица A_n и вектор d_n для алгоритма 1 и векторы f_k для алгоритма 2 обобщенного метода Рунге-Кутты /подробнее см. /1,2/.

```
SUBROUTINE AMAT(X,Y,A,B,A2,CC,PC,G,P,F,N,NA,NC,IBD,IND,KBLD),
IMPLICIT REAL*8(A-H, O-Z),
DIMENSION Y(N),P(N),G(N),B(NC),CC(NC),PC(NC),F(NC),
DIMENSION A(NA,NC),A2(NA,NC),
COMMON/GOV8/KSAV,MQ,MBLD(100).
```

В качестве A_n /массив A / и d_n /массив B / для алгоритма 1 рекомендуется брать соответственно матрицу B(x_n,y_n) системы /1/ и решение системы линейных уравнений A_nd_n=b(x_n,y_n),

где x_n и y_n задаются с помощью X и Y, а B(x_n,y_n) и b(x_n,y_n) определены в /1,2/. В частности,

$$b(x_n, y_n) = f(x_n, y_n) - \partial f / \partial y(x_n, y_n) y_n,$$

$$B(x_n, y_n) = \partial f / \partial y(x_n, y_n),$$

где ∂f/∂y - матрица Якоби. Если матрица A_n вырожденная, в качестве d_n рекомендуется брать нормальное решение системы A_nd_n=b(x_n,y_n), которое можно найти с помощью метода регуляризации Тихонова /16/, т.е. путем решения системы A_n^TA_nd_n+αd_n=A_n^Tb(x_n,y_n), A_n^T - транспонированная матрица и α - параметр регуляризации.

Для решения системы уравнений, определяющей d_n, можно использовать подпрограммы GLESON, FM1MX и FM1VX, описание которых дано ниже. Рабочие массивы A2, CC, PC, F, P /за исключением MBLD(1)<N и KBLD≠0 / могут быть использованы для промежуточных вычислений в AMAT, например тогда, когда ∂f/∂y находится численным дифференцированием, а также при нахождении d_n. Если матрица A_n диагональна, ее элементы задаются в массиве CC /массив A не используется/. Параметры IBD, IND, KBLD вместе с параметром MQ общего блока COV8 применяются, если A_n - квази-диагональная матрица вида /6/. IBD содержит номер i блока A_iⁿ, элементы которого необходимо задать в массиве A, MQ содержит порядок этого блока /в этом случае MQ=MBLD(IBD)/, а IND содержит номер строки матрицы A_n, в которой находится первая строка блока A_iⁿ. Отметим, что одновременно с заданием блока A_iⁿ в массиве A в массиве B нужно задать только элементы вектора d_n с номерами IND, IND+1, ..., IND+MQ-1 /они определяются из соответствующей подсистемы системы A_nd_n=b(x_n,y_n) /. Параметр KBLD служит для уменьшения затрат машинного времени. Это происходит следующим образом.

При $KBLD \leq 0$ сначала нужно переписать массивы A, B, A2, CC, PC, в которых задана информация /получаемая в GERUN /, связанная с блоком порядка MBLD(i) с номером $i=IBD-1$ или $i=s$, если $IBD-1=0$, в соответствующих дополнительных массивах, вводимых самим пользователем в AMAT /если $KBLD > 0$, этого делать не надо/. После этого: а/ если $KBLD=0$, необходимо задать элементы блока A_i^n с $i=IBD$ в массиве A и элементы вектора d_n с номерами IND, IND+1, ..., IND+MQ-1 в массиве B; б/ если $KBLD \neq 0$, нужно переписать соответствующие дополнительные массивы в массивах A, B, A2, CC, PC.

В случае квазидиагональной матрицы A_n с $A_i^n=0, i=2, \dots, s$ параметры IBD, IND и KBLD не используются. В массиве A задается только первый блок порядка MQ матрицы A_n , а в массиве B - только элементы $d_i^n, i=1, 2, \dots, MQ$ /остальные считаются равными нулю/.

Если используется алгоритм 2, в массиве A задаются векторы $f_k, k=0, 1, \dots, m^{1,2}$. Элемент $A(i, k+1)$ определяет значение i -ой компоненты вектора f_k .

AMAT должна быть объявлена EXTERNAL в вызывающей программе.

- HESS - имя подпрограммы приведения действительной матрицы общего вида к матрице Хоссенберга /ELMHES либо ORTHES/. HESS должна быть объявлена в вызывающей программе.
- TRANS - имя подпрограммы, вычисляющей матрицу преобразования при приведении действительной матрицы общего вида к матрице Хоссенберга /ELTRAN либо ORTRAN/. TRANS должна быть объявлена EXTERNAL в вызывающей программе.
- XEND - задает x_f в /1/.
- MST - максимальное число шагов.
- KPR - число шагов, через которое будет печататься решение /1/.
- KPD. - При $KPD \neq 0$ печатаются и оценки локальных ошибок усечения для каждой компоненты решения.
- EMAX - задает верхнюю границу для локальной ошибки усечения. Если $EMAX \leq 0$ и $KAR \leq 0$, оценка локальной ошибки усечения не вычисляется /величина шага не контролируется/. Если $KAR > 0$, значение EMAX определяется автоматически в GERUN.
- EMIN - нижняя граница для локальной ошибки усечения.
- HMAX - максимальная абсолютная величина шага. Если $|H| > HMAX$, то $|H|$ больше не увеличивается.
- HMIN - минимальная абсолютная величина шага.

- DB - Если найденная оценка локальной ошибки усечения больше EMAX, шаг уменьшается с помощью соотношения $H=H*DB / DB < 1$ /. Стандартное значение $DB = 0,5$.
- DF - Если найденная оценка локальной ошибки усечения меньше EMIN, шаг увеличивается с помощью соотношения $H=H*DF / DF > 1$ /. Стандартное значение $DF = 2$.
- KMAX - максимальный порядок аппроксимации матричной экспоненты в случае, когда используется первый способ вычисления экспоненты матрицы A_n в алгоритме 1.
- KIN - содержит начальное значение порядка аппроксимации матричной экспоненты / $0 < KIN \leq KMAX$ /. Если $KIN > 0$ и $NTS \leq 0, KDIAG=KSYM=KEIGV=0$, в GERUN используется алгоритм 1 обобщенного метода Рунге-Кутты и первый способ вычисления экспоненты матрицы A_n /если и $KIN=0$, для решения /1/ используются обычные явные методы Рунге-Кутты/. При $KIN=KMAX$ ошибка аппроксимации матричной экспоненты /второй член в правой части /3// не учитывается при оценке локальной ошибки усечения, что приводит к уменьшению времени счета.
- KEKX - число шагов, через которое перевычисляются матрица A_n и вектор d_n или векторы f_k в обобщенном методе Рунге-Кутты. Если $KEKX \leq MST$, вся необходимая информация, связанная с A_n, d_n или f_k , получается в GERUN с использованием AMAT. Если $KEKX > MST$, эта информация предполагается предварительно /до вызова GERUN / заданной в соответствующих массивах. Значения $KEKX > MST$ позволяют использовать GERUN более гибким образом, например в случае, когда ее необходимо вызывать на каждом шагу интегрирования, не вычисляя всегда заново A_n, d_n или f_k . Отметим также, что если $KEKX > 1$, матрица A_n и вектор d_n или векторы f_k перевычисляются каждый раз, когда найденная оценка локальной ошибки усечения больше EMAX независимо от значения KEKX. Стандартное значение $KEKX=1$.
- MRUP, MRIN, MRED - используются только в алгоритме 1, когда экспонента матрицы A_n вычисляется первым способом. Если из-за плохой обусловленности матриц F_{2m+1} в /5/ из /8/ точность их обращения недостаточна, значение MRED увеличивается / $MRED=MRED+MRIN$ / до тех пор, пока не будет достигнута необходимая точность или MRED не станет

- больше MRUP / в этом случае печатается соответствующее сообщение/. Кроме того, большее значение MRED позволяет использовать меньшее значение параметра KIN^{4,6}. Стандартные значения MRUP=20, MRIN=1, MRED=0.
- REL - EMIN=EMAX*REL в случае KAR>0. Стандартное значение REL = 0,1.
- BET - задает значение β в /5/. Стандартное значение BET = 0,1.
- GAM - задает значение γ в /5/. Стандартное значение GAM = 0.
- FM,EFN. - Если равномерная норма FM правой части $f(x,y)$ в /1/ становится меньше EFN, в GERUN полагается X=XEND и процесс интегрирования прекращается /считается, что найдена асимптотически устойчивая точка покоя/. Стандартное значение EFN=10⁻¹⁶.
- EL - задает наименьшее допустимое значение параметра EMAX. Стандартное значение $EL = 10^{-15} \max_i |Y(i)|$, если KAR≠0, либо $EL = 10^{-15}$ если KAR = 0.
- ENT - используется в случае NTS>0, т.е. в случае интегрирования /1/ с помощью алгоритма 2 обобщенного метода Рунге-Кутты. Если FM становится меньше ENT, GERUN полагает NTS=0 и интегрирование осуществляется с помощью соответствующей модификации алгоритма GERUN. Стандартное значение ENT=0.
- NLP - Стандартное значение NLP = N.
- EPS - содержит оценку равномерной нормы локальной ошибки усечения. Вычисляется программой.
- ES - масштабный множитель для оценки локальной ошибки усечения $EPS/EPS=ES*EPS$. Стандартное значение ES=1.
- RS - задает значение δ в /4/. Стандартное значение RS=0,001.
- OM - задает значение ϵ в /2/. Стандартное значение $OM = 10^{-10}$.
- KAR - Если KAR=0, в GERUN оценивается относительная локальная ошибка усечения. В противном случае оценивается абсолютная локальная ошибка усечения. При этом, если KAR>0, контроль локальной ошибки усечения происходит согласно /5/.
- NUMS - содержит число сделанных шагов в GERUN.
- KIS - определяется в GERUN.
- KINL - содержит значение использованного на последнем шаге порядка m аппроксимации матричной экспоненты в случае, когда экспонента матрицы A_n вычисляется первым способом.

- NTS - Если NTS>0, в GERUN используется алгоритм 2 обобщенного метода Рунге-Кутты. При этом, если применяются первые k+1 члены разложения
- $$f(x,y) = \sum_{i=0}^k f_i(x-x_n)^i + o(|x-x_n|^k),$$
- NTS=k+1.
- KDIAG - Если NTS<0, KDIAG≠0, в GERUN используется алгоритм 1 обобщенного метода Рунге-Кутты с диагональной матрицей A_n .
- KSYM - Если NTS<0, KDIAG=0, KSYM≠0, используется алгоритм 1 с симметрической матрицей A_n .
- KEIGV - Если NTS<0, KDIAG=KSYM=0, KEIGV≠0, в GERUN применяется алгоритм 1 и второй способ вычисления экспоненты матрицы A_n . При KEIGV>0 матрица A_n предварительно масштабируется /для этого вызываются подпрограммы BALANC и BALBAK /, при KEIGV<0 это не делается. Если окажется, что матрица A_n не имеет полного набора линейно-независимых собственных векторов /в этом случае печатается соответствующее сообщение/, то при KIN>0 дальнейший счет ведется с использованием первого способа вычисления матричной экспоненты, а при KIN=0 - с помощью обычного метода Рунге-Кутты.
- KSUP - Если NTS≤0, KDIAG=KSYM=0, KEIGV<0 и KSUP<0, в алгоритме 1 используется нормальная матрица T_n /вычисляется в GERUN /, такая, что матрица $A_n - T_n$ имеет нулевые собственные значения /в этом случае нужно использовать ORTHES и ORTRAN /. Если NTS≤0, KDIAG=KSYM=0, KEIGV≠0 и KSUP>0, в алгоритме 1 применяется матрица Q_n с различными собственными значениями, такая, что $\|A_n - Q_n\| \leq \epsilon$, где ϵ - достаточно малое положительное число /зависит от значения параметра DIS общего блока DIS /. Для этой цели в подпрограмме HQR2 вызывается подпрограмма DISEIG /при необходимости пользователь может составить свою программу DISEIG /. Если KSUP=0/KEIGV≠0/, в алгоритме 1 используется матрица A_n .
- KSAV - Если KSAV≠0, в GERUN используется такой метод вложения, который позволяет уменьшить число вычислений правой части $f(x,y)$ на единицу /подробнее см. /12, с.80/ и /13, с.10/ /. При этом, если KSAV>0, в массиве G необходимо задать значения правых частей $f^i(x_0, y_0)$ перед вызовом GERUN.
- MQ - В случае квазидиагональной матрицы A_n вида /6/ с $A_i^n = 0, i=2, \dots, s$, параметр MQ задает порядок первого блока.

- MBLD - одномерный массив размерности 100. В случае квазидиагональной матрицы A_n вида /6/MBLD(i) задает порядок i -го блока A_i^n , $i=1,2,\dots,s$.
- CRK, ARK, B1RK, B2RK - одномерные массивы размерности CRK(20), ARK(190), B1RK(20), B2RK(20). Определяются в подпрограмме INIZ и содержат параметры используемого обычного метода Рунге-Кутты. В GERUN положено CRK(1) и CRK(NSTAG)=1.
- CRED, MCR - используются в алгоритме 1 в случае, когда экспонента матрицы A_n вычисляется первым способом. Определяются в подпрограмме INIZ. Стандартное значение MCR=1.
- NSTAG - задает число вычислений правой части $f(x,y)$ в используемом явном методе Рунге-Кутты /NSTAG \geq 2/.

Приведем и параметры подпрограмм GLESON, FMTVX и FMTMX. С помощью GLESON можно решать систему линейных уравнений. В массиве A размерности (NR,N) задаются элементы матрицы коэффициентов системы, в массиве B размерности N - правая часть системы, массивы IPR и V размерности N - рабочие, N - число уравнений и неизвестных. Значение параметра D1=0 при выходе из GLESON указывает, что матрица коэффициентов близка к вырожденной и решение нельзя получить с помощью GLESON. Подпрограмма FMTVX находит произведение транспонированной матрицы A^T на вектор x. В массиве A размерности (MA,N) задается матрица A, в массиве X размерности M,N - вектор x, в массиве Y размерности N - результат $A^T x$, M,N - размеры прямоугольной матрицы A. Подпрограмма FMTMX находит произведение транспонированной матрицы A^T на матрицу B. В массиве A размерности (MA,N) задается матрица A, в массиве B размерности (MB,K) - матрица B, в массиве C размерности (MC,K) - результат $A^T B$, M,N - размеры матрицы A, M, K - размеры матрицы B, N, K - размеры матрицы C.

О БЕЗУСЛОВНОЙ МИНИМИЗАЦИИ ФУНКЦИЙ С ПОМОЩЬЮ ПРОГРАММЫ GERUN

В работе /15/ показано, что использование алгоритма 1 обобщенного метода Рунге-Кутты и непрерывного аналога метода градиента позволяет получить новый метод минимизации функций без ограничений со сверхлинейной скоростью сходимости, который не использует процедуры одномерного поиска, минимизирует квадратичные функции за один шаг и удачно объединяет преимущества метода наискорейшего спуска и метода Ньютона. При этом величина шага контролируется путем

оценки локальной ошибки усечения согласно /5/. В связи с этим приведем один из возможных вариантов использования GERUN для безусловной минимизации функций непрерывным аналогом метода градиента /аналогичный вариант рекомендуется использовать и для нахождения асимптотически устойчивых точек покоя автономных систем/: XEND \gg 1 /например, XEND=10³⁰/, MST = число итераций, HMAX \gg 1 /например, HMAX=10¹⁰/, HMIN=10⁻¹⁰, REL \leq 1, BET \leq 0.5, GAM= ϵ >0 /например, ϵ = 10⁻⁵/, EFN=10⁻¹², KAR>0, KSYM \neq 0, X=0 и стандартные значения остальных параметров управления /в массиве Y задаются начальные приближения/. При этом в GERUN используется только массив A из двумерных рабочих массивов, что приводит к уменьшению необходимой памяти, а значение GAM определяется автоматически подпрограммой GADET. Если GAM=0, пользователь сам должен задать подходящее значение HMAX, которое в этом случае зависит от минимизируемой функции. Подпрограмма FUN должна вычислять правые части системы обыкновенных дифференциальных уравнений /3/ в /15/. В качестве матрицы A_n естественно брать матрицу Гессе $H(y_n)$ минимизируемой функции со знаком минус /она является симметрической/, а в качестве вектора d_n - решение системы линейных уравнений

$$-H(y_n)d_n = -g(y_n) + H(y_n)y_n,$$

где g - градиент минимизируемой функции.

ЛИТЕРАТУРА

1. Банчев В.Ц. ОИЯИ, P5-12174, Дубна, 1979.
2. Банчев В.Ц. Докл. БАН, 1979, т.32, №5.
3. Lawson J.D. SIAM J.Numer.Anal., 1967, v.4, p.372-380.
4. Банчев В.Ц. ОИЯИ, P11-9677, Дубна, 1976.
5. Брич З.С. и др. ФОРТРАН ЕС ЭВМ. "Статистика", М., 1978.
6. Moler C., Van Loan Ch. SIAM Review, 1978, v.20, No. 4, p.801-836.
7. Mori M. Publ. RIMS, Kyoto Univ.
8. Банчев В.Ц. ОИЯИ, P11-9678, Дубна, 1976.
9. Smith B.T. et al. Matrix Eigensystem Routines - EISPACK Guide. 2nd ed. Lecture Notes in Computer Science, v.6. Springer-Verlag, New York, 1976.
10. Jackson K.R., Enright W.H., Hull T.E. SIAM J.Numer.Anal., 1978, v.15, No.3, p.618-641.
11. Verner J.H. SIAM J.Numer.Anal., 1978, v.15, No.4, p.772-790.

12. Hall G., Watt J.M. (eds.) Modern Numerical Methods for Ordinary Differential Equations. Clarendon Press, Oxford, 1976.
13. Bettis D.G. Lect. Notes Math., 1978, v.631, p.9-18.
14. Friedli A. Lect. Notes Math., 1978, v.631, p.35-50.
15. Банчев В.Ц. ОИЯИ, Р5-12167, Дубна, 1979.
16. Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. "Наука", М., 1974.
17. Ehle B.L., Lawson J.D. J.Inst.Math. and Appl., 1975, v.16, No.1, p.11-21.

Рукопись поступила в издательский отдел
12 октября 1979 года.