УДК 519.6

ЛОКАЛИЗАЦИЯ НУЛЕЙ ФУНКЦИЙ НЕСКОЛЬКИХ ПЕРЕМЕННЫХ НА ОСНОВЕ СОРТИРОВКИ В ПРИЛОЖЕНИИ К АНАЛИЗУ УСТОЙЧИВОСТИ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Веселая А.А., Заика И.В., Тюшнякова И.А.

Таганрогский институт имени А.П. Чехова (филиал) ФГБОУ ВО «РГЭУ (РИНХ)», Таганрог, e-mail: veselaya-a-a@yandex.ru, irin-zaik@yandex.ru, solovyova_irina@mail.ru

Описан распараллеливаемый компьютерный метод численной оптимизации на основе устойчивой сортировки, который позволяет определить с априори заданной границей погрешности нули функций многих переменных по значениям и индексам местоположения. Метод отличается от известных построением на основе сортировки, возможностью распараллеливания и произвольностью задания границ области всех нулей функции. Строится программный метод анализа устойчивости решений систем обыкновенных дифференциальных уравнений на основе нахождения собственных значений матрицы постоянных коэффициентов системы линейных обыкновенных дифференциальных уравнений произвольного порядка при помощи сортировки. Метод применяется для компьютерного анализа устойчивости по Ляпунову системы обыкновенных дифференциальных уравнений в случаях асимптотической и неасимптотической устойчивости. Изложенная методика применяется к анализу устойчивости реальной физической системы.

Ключевые слова: численная оптимизация, сортировка, обыкновенные дифференциальные уравнения, устойчивость

LOCALIZATION OF ZEROS OF FUNCTIONS OF SEVERAL VARIABLES, BASED ON THE SORTING IN THE APPLICATION TO THE STABILITY ANALYSIS OF ORDINARY DIFFERENTIAL EQUATIONS

Veselaya A.A., Zaika I.V., Tyushnyakova I.A.

Taganrog Institute named after A.P. Chekhov (branch) Rostov State University of Economics, Taganrog, e-mail: veselaya-a-a@yandex.ru, irin-zaik@yandex.ru, solovyova_irina@mail.ru

Proposed parallelized computer numerical optimization method on a sustainable basis the sort that allows to identify a priori a given boundary error zeros of functions of several variables on the value and index position. The method differs from the known building on the basis of the sort, the possibility of parallelization and arbitrary boundaries of the field all zeros job. Construct a software method for analyzing the stability of solutions of systems of ordinary differential equations on the basis of a finding by means of sorting the eigenvalues of the constant coefficients of linear ordinary differential equations of arbitrary order. The method is used for computer analysis of Lyapunov stability of a system of ordinary differential equations in the case of asymptotic and non-asymptotic stability. The stated procedure is used to analyze the stability of the real physical system.

Keywords: numerical optimization, sorting, ordinary differential equations, stability

Постановка вопроса. Ставится задача разработать схему программного вычисления всех нулей функций многих переменных в области определения. Схему предполагается перенести на вычисление корней полиномов, а также нулей характеристического полинома матрицы коэффициентов системы линейных обыкновенных дифференциальных уравнений (ОДУ). Рассматриваемая схема используется для компьютерного анализа устойчивости системы ОДУ путем определения собственных значений матрицы коэффициентов системы ОДУ.

Схема сортировки и вычисление нулей функций нескольких переменных по индексам данных

Пусть вначале рассматривается функция одной переменной, для которой необходимо вычислить все минимумы

$$y = f(x). \qquad (1)$$

В области определения функции задается промежуток $[x^{(0)}, x^{(N)}]$. На промежутке $h = \left| x^{(N)} - x^{(0)} \right| / N$, $x_\ell = x^{(0)} + \ell h$, $\ell = 0, 1, ..., N$ считываются значения функции (1)

$$c[i] = f(x_{i-1}),\ i = 1, 2, ..., N. \qquad (2)$$

Значения $c[i]$ функции $f(x)$ сортируются. После осуществляется локализация минимумов функции в наперед заданной окрестности ε среди элементов (2), с помощью условного оператора $\left| e[k-l] - e[k] \right| > \varepsilon$, $l = 1, 2, ..., k-1$. Где $e[k]$ – индексы сортируемых элементов, располагающиеся на выходе по порядку отсортированного массива. Чтобы вычислить нули функции, необходимо на вход сортировки подать величины (2) взятые по модулю $c[i] = \left| f(x_{i-1}) \right|$, $i = 1, 2, ..., n$, и искать нули как минимумы модуля.

Для нахождения минимумов для функции двух переменных $z = f(x, y)$ вну-

три области определения $[x^{(0)}, x^{(N)}]$ и $[y^{(0)}, y^{(M)}]$ задается прямоугольная сетка: $h = \left| x^{(N)} - x^{(0)} \right| / N$, $x_\ell = x^{(0)} + \ell h$, $\ell = 0, 1, ..., N$, $y_\ell = y^{(0)} + \ell h$, $\ell = 0, 1, ..., M$. Далее осуществляется проход вдоль оси OY, j столбца прямоугольной сетки, в результате которого находится наименьшее значение $c[j] = \min f(x_j, y_i)$, $i = 1, 2, ..., M$, полученный минимум поступает на вход сортировки как j элемент одномерного массива. После чего оператор локализации минимума вычисляет все индексы минимумов функции двух переменных.

Описанная схема переносится для вычисления всех нулей функции z. На вход программы достаточно подать значения: $\tilde{z} = \left| f(x, y) \right|$. Нули функции локализуются как минимумы модуля [1–7, 17]. Таким образом, вычисляются все корни полиномов с комплексными коэффициентами [9, 10, 12, 13].

Локализация и вычисление собственных значений матриц

Собственные значения квадратной матрицы А размерности $n \times n$ определяются по описанной выше схеме как нули характеристического полинома:

$$|A - \lambda E| = (-1)^n (\lambda^n + p_1 \lambda^{n-1} + ... + p_n).$$

Коэффициенты данного полинома вычисляются с помощью метода Леверье. Для этого необходимо решить систему уравнений Ньютона: $S_k + S_{k-1} p_1 + S_{k-2} p_2 + ... + S_1 p_{k-1} = -k p_k$. Здесь $k = \overline{1..n}$, $S_k = Sp(A^k)$ – след матрицы A^k. Полином $P_n(\lambda) = (-1)^n (\lambda^n + p_1 \lambda^{n-1} + ... + p_n)$ ($\lambda = x + Iy$) задается своими коэффициентами. Для того чтобы на вход метода подать соответственную полиному функцию $f(x, y)$, применяется биномиальное разложение [3, 6, 8]; иначе значение полинома умножается на комплексно-сопряженное. В результате получим функцию $f(x, y) = (\mathrm{Re}\,(P_n(x + Iy)))^2 + (\mathrm{Im}(P_n(x + Iy)))^2$. В случае если коэффициенты характеристического полинома вычислены с достаточной точностью, изложенный метод определения собственных значений матриц является устойчивым ввиду фактически верного вычисления нулей полиномов с учетом кратности [14–16].

Программное определение нулей характеристического полинома матрицы постоянных коэффициентов системы линейных ОДУ произвольного порядка

Пусть дана система линейных ОДУ в матричной форме с постоянными коэффициентами [8, 11, 22]

$$\frac{dY}{dt} = AY, \qquad (3)$$

где Y и A – квадратные матрицы постоянных коэффициентов размерности $n \times n$. Общее решение системы ОДУ (3) с постоянной матрицей A:

$$Y = e^{At} c. \qquad (4)$$

Начальные условия для (4):

$$Y(t_0) = Y_0, \qquad (5)$$

тогда решение $\overline{Y}$ удовлетворяет задаче Коши $\overline{Y}(t_0) = \overline{Y}_0$. Ставится цель: провести исследование решения (4) системы ОДУ (3) на устойчивость в смысле Ляпунова [11, 12].

Рассмотрим подход к решению задачи анализа устойчивости, который основан на оценке устойчивости в соответствии с характеристическими нулями матрицы коэффициентов системы, то есть по виду собственных значений матрицы коэффициентов.

Из I теоремы Ляпунова об устойчивости известно, что нули характеристического полинома матрицы A из (3) имеют связь с характером устойчивости (3): в случае если все собственные значения имеют отрицательные действительные части, то система асимптотически устойчива; в противном случае система неустойчива [19, 20].

Отметим, что предложенный метод компьютерного анализа устойчивости содержит все аспекты анализа устойчивости по критериям Гурвица, Михайлова и Найквиста. Помимо этого, метод позволяет анализировать устойчивость и в случаях, не подпадающих под классические критерии (случай варьирования параметров системы ОДУ). Рассматриваемый метод применим для оценки устойчивости и асимптотической устойчивости во всех случаях без исключения. Классические критерии не всегда подходят для компьютеризации, а информацию об устойчивости, как правило, дают лишь косвенно, в виде, который характерен для качественной теории диффуравнений. К тому же метод основывается на алгоритме, который влечет однозначное указание на характер устойчивости, неустойчивости, асимптотической устойчивости в случае варьирования параметров решения систем линейных ОДУ с постоянными коэффициентами. Метод допускает использование в случаях, когда классические критерии невозможно применить, например при наличии трансцендентностей и нелинейностей в правой части.

Предложенный метод дает полную автоматизацию компьютерного анализа устойчивости. Отметим, что для определения знака действительной части нет необходи-

мости точно вычислять нули полинома. Достаточно только произвести локализацию нулей, при этом сохраняется гарантия верных знаков их действительных частей. Таким образом произведена рационализация схемы компьютерного анализа устойчивости с точки зрения ее временной сложности.

Отметим, что в случае когда число нулей меньше степени многочлена, требуется локализовать действительные и мнимые части всех нулей с учетом их кратности.

Пример, приведенный ниже, показывает достаточность локализации действительных частей нулей характеристического полинома матрицы.

Пример 1. Пусть система (4) имеет матрицу коэффициентов:

$$A = ((1,2,3,4,5),(1,1,1,1,4),$$
$$(1,1,1,4,1),(1,1,4,1,1),(5,4,3,2,1)).$$

Необходимо определить знаки действительной части собственных значений матрицы. Ниже представлен алгоритм, который вычисляет действительные части собственных значений с точностью $eps0 = 0.1$.

```
program korni;
LABEL 2; const eps0=0.1;eps=1e-10*1E-33; nn1=5; n00=2000; mm=128; h=eps0/10;
 nn=n00+round(n00/2);y00=-10; y11=10; x00=-10; x11=10;
type vvec=array[0..nn1] of extended; mmatrix = array[1..nn1,1..nn1] of double;
vvec1=array[0..2*nn+nn] of double; vect2= array[0..2* nn+nn] of longint;
vvec3= array[0..nn1] of double;
const aa:mmatrix =((1,2,3,4,5),  (1,1,1,1,4), (1,1,1,4,1),(1,1,4,1,1),(5,4,3,2,1));
var  cc1,cc: mmatrix; ep, pp: vect; ss,ss1: double; ii, jj, kk, ll: integer; i,j,k,k1,r,ee,ee1,tty,nn0 : longint;
c,a1: vvec1; e, e33,e3: vvec2; x0, x1, min, eps1,x, xk, xk1,xk0, hx, hy: double;
y,y0,y1,yk,yk0,yk1, ykk0,aak, hh, bbk: double; bdv, bmv, bd, bm: vect3;
var db, da: vvec3;b, ca, b11, a , a11, cb: double ;
procedure uum1(var a,b,a11,b11: double; var ca,cb: double);
Begin ca:=a*a11-b*b11; cb:=a*b11+b*a11 end;
procedure  ssum1 (var a,b,a11,b11: double; VAR ca, cb: double);
Begin ca:=a+a11; cb:=b+b11 end;
function ffunc1 (var x, y: double; var: bdv, bmv: vect3): double;
var i1: 0..nn1; p2, pp2, p1,d1, pp1, d4,d3, d2: double; begin p1:=bdv[nn1];p2:=bmv[nn1];
 for i1:=1 to nn1 do begin uum1(p1,p2,x,y,d1,d2); pp1:=bdv[nn1-i1];
pp2:=bmv[nn1-i1];ssum1(pp1,pp2,d1,d2,d3,d4); p2:=d4; p1:=d3; end; ffunc1:=p1*p1+p2*p2; end;
procedure  mminx1 (var x,y,min: double;var ee:integer);
begin ee:=0;  min:=ffunc1(x,y,bdv,bmv); for i:=1 to mm do begin x:=xk0+i*hx;
if min>ffunc1 (x,y,bdv,bmv) then begin ee:=i; min:=ffunc1(x,y,bdv,bmv); end; end; end;
procedure mminy1 (var x,y,min: double;var ee1:integer);
begin ee1:=0; min:=ffunc1(x,y,bdv,bmv); for i:=1 to tty do begin y:=ykk0+i*hy;
if min > ffunc1(x,y,bdv,bmv) then begin ee1:=i ;min:=ffunc1(x,y,bdv,bmv); end; end; end;
//Описание процедуры сортировки sorting
begin
//Реализация метода  Леверье
i:=0; while i<=n1 do begin bd[i]:=pp[n1-i];i:=i+1;end; i:=0; while i<= n1 do deign  bm[i]:=0; i:=i+1;
end; bbk:=1e32; aak:=1e32; nn0:=n00; hh:=nn0*h;
 x1:=x11; y0:=y00; y1:=y11; x0:=x00; i:=0; while i<=n1 do  begin bdv[i]:=bd[i]; i:=i+1;
i:=0; while i<=n1 do begin bmv[i]:=bm[i]; i:=i+1;end; while  x0<=x11+hh do begin
while y0<=y11+hh do begin  r:=1; while r<=nn0 do begin
hy:=h; tty:=n00; y:=y0; ykk0:=y0; x:=x0+r*h; mminy1 (x,y,min,ee1);
a1[r]:=min; r:=r+1; end; sorting ( nn0, a1, e3); k:=1;  while k<= nn0 do begin  r := 1; while r<= k-1 do begin
if abs(e3[k-r] -e3[k])<=eps0/h then goto 2;
r:=r+1; end; xk:= x0+e3[k]*h; write ('   ', xk:3:3,'  ');
2: k:=k+1;  end; end; x0:=x0+hh; end; end.
```

Результаты вычислений: – 3.000; 2.510; – 0.430; 4.725; 9.985.

С помощью изложенной программы удается вычислить действительные части нулей с учетом знака за относительно малый период времен. Точность вычислений проигрывает по сравнению с результатами полной программы, однако это оказывается достаточным для того, чтобы сделать вывод о неустойчивости точки покоя системы.

Предложенный метод используется для быстрого анализа устойчивости. В случае если действительные части положительны, то система неустойчива. Для достоверности оценки устойчивости можно осуществить локализацию действительных и мнимых частей нулей характеристического многочлена. Результат вычисления действительных и мнимых частей (полный код программы приводится в [6]):

Действительная часть нулей	Мнимая часть нулей	Значение функции
– 3.00000000000000000E + 0000	0.00000000000000000E + 0000	0.00000000000000E + 0000
2.51052877410466521E + 0000	– 2.71473540786538243E-0029	1.23259516440783E-0032
– 4.28061588346591394E-0001	– 6.52520446799852453E-0056	5.49509021016461E-0106
– 4.72255545982224341E + 0000	– 4.44724119213053305E-0029	3.25482160601443E-0032
1.06400882740641695E + 0001	– 4.38707603928584072E-0034	2.30564629222262E-0029

С помощью изложенной схемы удается вычислить действительные и мнимые части нулей относительно точно. Вывод о состоянии равновесия точки покоя системы аналогичен предыдущему примеру.

Пример 2. Оценить устойчивость системы. Система имеет следующий вид:

$$A = \begin{pmatrix} 0 & 0.6793 & 0.6099 & 0 & 0.4948 & 0.5463 & 0 & -0.952 & -0.7494 \\ 0 & -13.7658 & 1.4409 & 0 & 3.6163 & 1.1781 & 0 & 8.5472 & -3.3161 \\ 0 & -15.5076 & -150.1554 & 0 & -12.6793 & 38.9205 & 0 & 42.4023 & -21.4333 \\ 0 & -6.5352 & -1.1714 & 0 & 0.9552 & 2.2156 & 0 & 5.4592 & -2.3385 \\ 0 & 5.6334 & 0.4076 & 0 & -16.5675 & 1.1141 & 0 & -4.2309 & 10.117 \\ 0 & -3.8073 & 52.627 & 0 & -13.1829 & -156.9117 & 0 & -38.8349 & 68.5987 \\ 0 & 2.9781 & 3.9766 & 0 & -10.6238 & -4.7247 & 0 & -5.201 & 10.7116 \\ 10000 & 0 & 0 & -10000 & 0 & 0 & 0 & 0 & 0 \\ 10000 & 0 & 0 & 0 & 0 & 0 & -10000 & 0 & 0 \end{pmatrix}$$

Листинг программы полностью приводится в [6]. Результат работы программы:

Действительная часть нулей	Мнимая часть нулей	Значение функции
– 4.573	0.000	3.76070674693958763E + 0035
– 2.805	0.000	2.67668758647364437E + 0033
– 103.668	0.000	5.88014178051377293E + 0037
– 1.66.591	0.000	1.54394867607685689E + 0033
0.000	0.000	3.68630606796407673E + 0032
– 5.272	229.567	4.35465580557896453E + 0036
– 5.272	– 229.837	3.67676705867643764E + 0038
– 24.590	346.355	4.14596930357129645E + 0039
– 24.593	– 346.355	2.94264390362970963E + 0034

В списке нулей есть значение с нулевой действительной частью. Для того чтобы убедиться в том, что нуль не результат погрешности вычислений, корень возможно вычислить с помощью алгоритма поиска действительных нулей, так как мнимая часть является нулевой. Если действительная часть равна нулю, то система устойчива относительно состояния равновесия.

Заключение

Представлена схема программного вычисления нулей функций нескольких переменных в области определения функции. Данная схема применяется для вычисления корней полиномов с учетом кратности, а также для идентификации нулей характеристического полинома матрицы. Эта же модифицированная схема используется для компьютерного анализа устойчивости системы ОДУ на базе метода Леверье. За счет того, что изложенный алгоритм работает только с индексами данных входных элементов сортируемых последовательностей, исключено накопление погрешности, в результате чего достигается высокая точность вычислений. Чтобы определить характер устойчивости системы, необходимо просто идентифицировать нули характеристического полинома, и знак их действительной части, что ускоряет компьютерный анализ.

Список литературы

1. Андреева Е.А. Вариационное исчисление и методы оптимизации. – М.: Высшая школа, 2010. – 586 с.

2. Антонов В.И. Линейная алгебра и аналитическая геометрия / Проспект. – 2011. – 139 с.

3. Бугров Я.С. Высшая математика. – М: Дрофа, 2004. – 284 с.

4. Васильев Ф.П. Методы оптимизации. – М.: МЦНМО, 2011. – 433 с.

5. Васильева А.Б., Тихонов Н.А. Интегральные уравнения. – М.: Физматлит, 2004.

6. Веселая А.А. Вычисление нулей и экстремумов функций при вариации параметров на основе сортировки с приложением к моделированию устойчивости систем линейных дифференциальных уравнений. Автореферат диссертации на соискание ученой степени канд. техн. наук. – Таганрог, 2009. – 19 с.

7. Галисеев Г.В. Программирование в Delphi. – М.: Диалектика, 2003.

8. Заика И.В., Тюшнякова И.А. Программное определение нулей функций и разностных решений систем обыкновенных дифференциальных уравнений на основе сортировки // Научный вестник. – 2015. – № 4 (6). – С. 39–52.

9. Ильин В.А. Высшая математика / «ТК Велби». – 2002.

10. Макконнелл Д. Анализ алгоритмов. – М.: Техносфера, 2002. – 304 с.

11. Романко В.К. Курс дифференциальных уравнений и вариационного исчисления. – М.: ЛБЗ. – 2001.

12. Ромм Я.Е., Заика И.В., Лабинцева А.А. Безусловная численная оптимизация при вариации параметров. I / Депонированная рукопись № 193-В2008 04.03.2008.

13. Ромм Я.Е., Заика И.В. Программная локализация экстремумов функций и разностных приближений решений дифференциальных уравнений. Известия высших учебных заведений // Известия высших учебных заведений. Северо-Кавказский регион. Серия: Технические науки. – 2005. – № М. – С. 55.

14. Ромм Я.Е., Гуревич М.Ю., Белоконова С.С., Соловьева И.А. Вычисление нулей и полюсов функций на основе устойчивой адресной сортировки с приложением к поиску и распознаванию // Проблеми програмування. – 2004. – № 2–3. – С. 462.

15. Ромм Я.Е., Тюшнякова И.А. Применение сортировки для поиска нулей и особенностей функций с приложением к идентификации плоских изображений // Учебное пособие. – Таганрог, 2009.

16. Треногин В.А. Функциональный анализ. – М.: Физматлит, 2002.

17. Тюшнякова И.А. Разработка и исследование схем применения сортировки для поиска нулей и особенностей функций с приложением к идентификации плоских изображений // Диссертация на соискание ученой степени кандидата технических наук. – Таганрог, 2006.

18. Флёнов М.Е. Delphi. Секреты программирования. – СПб.: Питер, 2006.

19. Харбор Р. Системы управления с обратной связью. – М.: ЛБЗ, 2001.

20. Эльсгольц Л.Э. Дифференциальные уравнения и вариационное исчисление. – М.: Наука, 1969.

21. Чиртик А.А. Программирование в Delphi. – СПб: Питер, 2011.

22. Romm Y.E., Zaika I.V. Numerical sorting-based optimization as applied to general differential and nonlinear equations // Cybernetics and Systems Analysis. – 2011. – Т. 47. № 2. – P. 316–329.

References

1. Andreeva E.A. Variacionnoe ischislenie i metody optimizacii. M.: Vysshaja shkola, 2010. 586 p.

2. Antonov V.I. Linejnaja algebra i analiticheskaja geometrija / Prospekt. 2011. 139 p.

3. Bugrov Ja.S. Vysshaja matematika. M: Drofa, 2004. 284 p.

4. Vasil'ev F.P. Metody optimizacii. M.: MCNMO, 2011. 433 p.

5. Vasil'eva A.B., Tihonov N.A. Integralnye uravnenija. M.: Fizmatlit, 2004.

6. Veselaja A.A. Vychislenie nulej i jekstremumov funkcij pri variacii parametrov na osnove sortirovki s prilozheniem k modelirovaniju ustojchivosti sistem linejnyh differencialnyh uravnenij. Avtoreferat dissertacii na soiskanie uchenoj stepeni kand. tehn. nauk. Taganrog, 2009. 19 p.

7. Galiseev G.V. Programmirovanie v Delphi. M.: Dialektika, 2003.

8. Zaika I.V., Tjushnjakova I.A. Programmnoe opredelenie nulej funkcij i raznostnyh reshenij sistem obyknovennyh differencialnyh uravnenij na osnove sortirovki // Nauchnyj vestnik. 2015. no. 4 (6). pp. 39–52.

9. Ilin V.A. Vysshaja matematika / «TK Velbi». 2002.

10. Makkonnell D. Analiz algoritmov. M.: Tehnosfera, 2002. 304 p.

11. Romanko V.K. Kurs differencialnyh uravnenij i variacionnogo ischislenija. M.: LBZ. 2001.

12. Romm Ja.E., Zaika I.V., Labinceva A.A. Bezuslovnaja chislennaja optimizacija pri variacii parametrov. I / Deponirovannaja rukopis no. 193-V2008 04.03.2008.

13. Romm Ja.E., Zaika I.V. Programmnaja lokalizacija jekstremumov funkcij i raznostnyh priblizhenij reshenij differencialnyh uravnenij. Izvestija vysshih uchebnyh zavedenij // Izvestija vysshih uchebnyh zavedenij. Severo-Kavkazskij region. Serija: Tehnicheskie nauki. 2005. no. M. pp. 55.

14. Romm Ja.E., Gurevich M.Ju., Belokonova S.S., Soloveva I.A. Vychislenie nulej i poljusov funkcij na osnove ustojchivoj adresnoj sortirovki s prilozheniem k poisku i raspoznavaniju // Problemi programuvannja. 2004. no. 2–3. pp. 462.

15. Romm Ja.E., Tjushnjakova I.A. Primenenie sortirovki dlja poiska nulej i osobennostej funkcij s prilozheniem k identifikacii ploskih izobrazhenij // Uchebnoe posobie. Taganrog, 2009.

16. Trenogin V.A. Funkcionalnyj analiz. M.: Fizmatlit, 2002.

17. Tjushnjakova I.A. Razrabotka i issledovanie shem primenenija sortirovki dlja poiska nulej i osobennostej funkcij s prilozheniem k identifikacii ploskih izobrazhenij // Dissertacija na soiskanie uchenoj stepeni kandidata tehnicheskih nauk. Taganrog, 2006.

18. Fljonov M.E. Delphi. Sekrety programmirovanija. SPb.: Piter, 2006.

19. Harbor R. Sistemy upravlenija s obratnoj svjazju. M.: LBZ, 2001.

20. Jelsgolc L.Je. Differencialnye uravnenija i variacionnoe ischislenie. M.: Nauka, 1969.

21. Chirtik A.A. Programmirovanie v Delphi. SPb: Piter, 2011.

22. Romm Y.E., Zaika I.V. Numerical sorting-based optimization as applied to general differential and nonlinear equations // Cybernetics and Systems Analysis. 2011. T. 47. no. 2. pp. 316–329.