```
        end;
  x := (a + b)/2;
  effect := (b - a) / L0
end; {MetodZolot}
```

Тұжырым. Барлық қолданылған әдістердің ерекшелігі және артықшылығы болып олардың әрбір итерация қадамында функцияның туындыларын есептеуді қажет етпей, тек қана сәйкес нүктелердегі функция мәнін есептеумен шектелуінде.

Әдебиеттер тізімі

1. *Пантелеев А.В., Летова Т.А.* Методы оптимизации в примерах и задачах. — М.: Высш. шк., 2002. — 542 с.
2. Сборник задач по математике для втузов. Специальные курсы / Под ред. А.В.Ефимова. — М.: Наука, 1984. — 301 с.
3. *Аттетков А.В., Галкин С.В., Зарубин В.С.* Методы оптимизации. — М.: Изд-во МГТУ им. Н.Э.Баумана, 2001. — 436 с.
4. *Вирт Н.* Алгоритмы и структуры данных / Пер. с англ. — М.: Мир, 1989. — 360 с.
5. *Культин Н.Б.* Программирование в Turbo Pascal 7.0 и Delphi. — 2-е изд., перераб. и доп. — СПб.: БХВ-Санкт-Петербург, 1999. — 416 с.

ӘОЖ 510.67

Л.С.Фазылова, А.Е.Сланбекова

Е.А.Бөкетов атындағы Қарағанды мемлекеттік университеті

АҚЫРЛЫ ӨЛШЕМДІ КЕҢІСТІКТЕГІ МИНИМУМДАУДЫҢ САНДЫҚ ӘДІСТЕРІН ПРОГРАММАЛАУ

В настоящей работе рассмотрены две стратегии поиска минимума функции одной переменной, приведена программа вычисления точек минимума одномерной функции различными методами, проводится сравнительный анализ рассмотренных методов.

Present work explains two strategies of searching minimums in function with one variable, contains program for calculation points of minimum in one dimensional function using different methods and benchmark analysis of these methods.

Есептің қойылымы [1]. $J(u)$ функциясына минималды мән беретін, яғни

$$\min_{u \in R_1} J(u) = J(u_*)$$

шартын қанағаттандыратын, $u_* \in U$ нүктесін табу керек.

Мақалада минимум нүктелерін есептеудің екі стратегиясы қарастырылады:

– пассивтік (асыра артық алу әдісі);

– тізбектік (дихотомия, «алтын қима», Фибоначчи әдістері).

Егер барлық нүктелер алдын ала, яғни есептеуге дейін, берілсе — бұл *пассивтік стратегия* болады. *Асыра артық алу әдісі* іздеудің пассивтік стратегиясына жатады. Бірөлшемді минимизациялы есепті шешудің алгоритмін қарастырайық:

a) Айталық $[a_0,\ b_0]$ анықталмағандықтың бастапқы интервалы болсын, $u^i = a_0 + i\dfrac{(b_0 - a_0)}{N+1}$, $(i = 0,\ 1,\ \ldots,\ N+1)$ формуласы арқылы бір-бірінен бірдей қашықтықтағы нүктелерді есептейміз;

b) табылған нүктелерде функцияның мәнін есептеу

$$J(u^i)\,,\ (i = 0,\ 1,\ \ldots,\ N+1)\,;$$

c) u^i $(i = 0,\ 1,\ \ldots,\ N+1)$ нүктелері арасынан функцияның ең кіші мән қабылдайтын нүктені таңдап алу:

$$J(u_*) = \min_{0 \le i \le N+1} J(u^i).$$

Асыра артық алу әдісімен минимум нүктесін табудың қателігі $\varepsilon \le \dfrac{(b_0 - a_0)}{N+1}$ -ден аспайды.

1-мысал. [1] Асыра артық алу әдісімен келесі есепті шешу керек:

$$J(u) = 2u^2 - 12u \to \min .$$

Шешуі. 1. Анықталмағандықтың бастапқы интервалын Свенн әдісі бойынша табамыз:

a) $u^0 = 5$ бастапқы нүктені және $t = 5$ береміз, $k = 0$ болсын.

b) Функцияның мәнін келесі нүктелерде есептейміз:

$$u^0 - t = 0,\ u^0 = 5,\ u^0 + t = 10,$$

онда $J(0) = 0$, $J(5) = -10$, $J(10) = 80$.

c) $J(u^0 - t) > J(u^0) < J(u^0 + t)$ болғандықтан, онда бастапқы интервал табылды: $L_0 = [0,\ 10]$. $N = 9$-ды L_0 аралықта $N + 1 = 10$ тең интервалдар болатындай мән береміз.

2. $u^i = 0 + i\dfrac{(10-0)}{10}$, $(i = 1,\ \ldots,\ 9)$ нүктелерін анықтаймыз.

3. 9-нүктеде функцияның мәнін есептейміз

$$J(1) = -10;\ J(2) = -16;\ J(3) = -18;\ J(4) = -16;$$

$$J(5) = -10;\ J(6) = 0;\ J(7) = 14;\ J(8) = 32;\ J(1) = -10.$$

4. $u^3 = 3$ нүктесінде функция ең кіші мән қабылдайды $J(u^3) = -18$.

5. Ізделініп отырған минимум нүктесі 9 есептеуден кейін $u_* \in [2,\ 4] = L_9$ аралығында жатады. $u_* \approx u^3 = 3$ нүктесі таңдалады.

Анықталмаған бастапқы интервалдың азаюы келесі сипаттамамен анықталады:

$$R(N) = \frac{|L_9|}{|L_0|} = \frac{4-2}{10-0} = 0{,}2 .$$

Егер ізделінді нүкте алдынғы есептеулер нәтижесін ескеріліп, таңдап алынса, онда *тізбектік стратегия* деп аталады.

Тізбектік стратегия үш кезеңді қамтиды:

1. Бастапқы интервалды таңдау. Интервалдың a_0, b_0 шекаралары $J(u)$ функциясы унимодалды болатындай болу керек.

2. Анықталмағандық интервалының азаюы.

3. Аяқталғандықтың шартын тексеру. Егер ағымдағы анықталмағандық интервалдың $[a_k,\ b_k]$ ұзындығы орнатылған шамадан кіші болса, іздеу тоқтатылады.

Унимодалды функцияның анықтамасын келтіреміз.

1-анықтама. [2] $J(u)$ функциясы $[a,\ b] = U$ аралығында *унимодалды функция* деп аталады, егер $[a,\ b]$ кесіндісінде үзіліссіз болса және $\alpha,\ \beta \in [a,\ b]$ сандары табылып төмендегі шарттар орындалса:

1) $J(u)$ функциясы $a \le u \le \alpha$ $(a < \alpha)$ қатаң бірсарынды кемісе,

2) $J(u)$ функциясы $\beta \le u \le b$ $(\beta < b)$ қатаң бірсарынды өссе,

3) $\alpha \le u \le \beta$ кезінде $J(u) = J_* = \inf J(u)$, сондықтан $U_* = [\alpha,\ \beta]$.

Дербес жағдайда, егер $\alpha = \beta$, онда $J(u)$ функциясы $[a,\ b]$ аралығында *қатаң унимодалды функция* деп аталады.

Соңғы анықталмағандық интервалында жататын нүктелер жиыны есептің шешімі болып табылады. Осы нүктелерден қандай да бір тәсілмен есептің шешімі u_* таңдап алынады.

Анықталмағандықтың бастапқы аралығын эвристикалық таңдау үшін *Свенн алгоритмін* қолдануға болады [1]:

1) келесі параметрлерді өз еркімен беру: u^0 — қандай да бір бастапқы нүкте; $t > 0$ — қадамның шамасы. $k = 0$ болсын;

2) үш нүктеде $J(u)$ функциясының мәнін есептеу: $u^0 - t$, u^0, $u^0 + t$;

3) шарттың аяқталғанын тексеру:

a) егер $J(u^0-t)\geq J(u^0)\leq J(u^0+t)$ болса, онда анықталмаған бастапқы аралық табылады

$$[a_0,\ b_0]=[u^0-t,\ u^0+t];$$

b) егер $J(u^0-t)\leq J(u^0)\geq J(u^0+t)$ болса, онда функция унимодалды болып табылмайды, ал керекті анықталмаған интервалды табу мүмкін емес. Есептеу бұл жағдайда тоқтатылады (бастапқы u^0 нүктесін басқаша алу ұсынылады);

c) егер аяқталу шарты орындалмаса, онда 4-ші қадамға көшеміз;

4) Δ шамасын анықтау:

a) егер $J(u^0-t)\geq J(u^0)\geq J(u^0+t)$ болса, онда

$$\Delta=t,\ a_0=u^0,\ u^1=u^0+t,\ k=1;$$

b) егер $J(u^0-t)\leq J(u^0)\leq J(u^0+t)$ болса, онда

$$\Delta=-t,\ b_0=u^0,\ u^1=u^0-t,\ k=1;$$

5) келесі нүктені табу $u^{k+1}=u^k+2^k\Delta$;

6) функцияның кему шартын тексеру:

a) егер $J(u^{k+1})<J(u^k)$ және $\Delta=t$ болса, онда $a_0=u^k$; егер $J(u^{k+1})<J(u^k)$ және $\Delta=-t$ болса, онда $b_0=u^k$; екі жағдайда да $k=k+1$ және қадам 5-ке көшу;

b) егер $J(u^{k+1})\geq J(u^k)$ болса, онда процедура аяқталады.

$\Delta=t$ болғанда $b_0=u^{k+1}$ болсын, ал $\Delta=-t$ болғанда $a_0=u^{k+1}$ болсын. Нәтижесінде $[a_0,\ b_0]$ — ізделініп отырған анықталмаған бастапқы интервалды аламыз.

2-мысал [1]. $J(u)=(u-5)^2$ функцияның минимумын табу үшін анықталмаған бастапқы аралықты анықтау керек.

Шешуі. Свенн алгоритмін қолданамыз.

1. $u^0=1,\ t=1,\ k=0$ болсын.

2^0. Функцияның мәнін келесі нүктелерде есептейміз:

$$u^0-t=0,\ u^0=1,\ u^0+t=2,$$

онда $J(0)=25$, $J(1)=16$, $J(2)=9$.

3^0. Аяқталу шарты орындалмайды.

4^0. $J(0)>J(1)>J(2)$ болғандықтан, онда

$$\Delta=1,\ a_0=1,\ u^1=u^0+t=2,\ k=1.$$

5^0. Келесі нүктені табамыз: $u^2=u^1+2\Delta=2+2=4$.

6^0. $J(u^2)=1<J(u^1)$ және $\Delta=1$ болғандықтан, $a_0=u^1=2$. $k=2$ болсын және 5-ші қадамға көшеміз.

5^1. Келесі нүктені табамыз: $u^3=u^2+4\Delta=4+4=8$.

6^1. $J(u^3)=9>J(u^2)=1$ және $\Delta=t=1$ болғандықтан, іздеу тоқталады және $b_0=u^3=8$.

Сондықтан анықталмаған бастапқы аралық: $[a_0,\ b_0]=[2,\ 8]$.

Әрі қарай тізбектік стратегияны қолданғанда анықталмаған интервалдың азаюы байқалады, яғни функцияның есептелуі ағымдағы интервалдың екі нүктесінде орындалады. Унимодалдық қасиетінің көмегімен қандай аралықта минимум нүктесі бар болуын анықтауға болады. *Дихотомия әдісі* тізбектік стратегияға жатады. Оның мағынасы [2].

Кейбір жеткілікті аз оң δ-ді шамасы арқылы, c және d нүктелерін келесі теңдіктермен есептейміз:

$$c=\frac{a+b-\delta}{2},\ d=\frac{a+b+\delta}{2}. \qquad (1)$$

Нүктелер алынғаннан кейін $J(c)$ және $J(d)$ функцияларының мәнін есептемейміз, өзара салыстырамыз. Егер

1) $J(c)\leq J(d)$ болса, онда $a_1=a$, $b_1=d$;

2) $J(c)>J(d)$ болса, онда $a_1=c$, $b_1=b$.

$J(u)$ функциясы $[a,\ b]$ кесіндісінде унимодалды болғандықтан, $[a,\ b]$ кесіндісінде u_* нүктесі функцияның минимум нүктесі болады. (1) теңдікте a-ны a_1-ге, b-ны b_1-ге ауыстырсақ, онда c_1 және d_1 есептеп аламыз және т.с.с.

Егер операцияны n-рет қайталасақ, u_* нүктесі орналасқан $[a_n,\ b_n]$ кесіндісіндегі $b_n - a_n$ ұзындығын табамыз. $u_* \approx \dfrac{a_n + b_n}{2}$ деп алып, минимум нүктені анықтаймыз. Қателігі $h = \dfrac{b_n - a_n}{2}$ санынан аспайтындығын көреміз. $J\left(\dfrac{a_n + b_n}{2}\right)$ саны $[a,\ b]$ кесіндісіндегі $J(u)$ функциясының минималды мәні болып есептеледі.

Функцияны есептеу кезінде N саны берілген жағдайда анықталмағандық интервалының азаю алгоритмінің тиімділігін бағалау үшін келесі критерийді енгіземіз.

2-анықтама. [1] $R(N) = \dfrac{|L_n|}{|L_0|}$ қатынасы анықталмағандықтың бастапқы интервалын азаюының сипаттамасы деп аталады, мұнда $|L_n|$ — функцияның N есептеу нәтижесінде алынған интервалдың ұзындығы; $|L_0|$ — анықталмаған бастапқы интервалдың ұзындығы.

Бір айнымалы функциялардың (унимодалды функциялар) минимум нүктелерін есептеудің программасы келесі әдістермен орындалған:

– асыра артық алу әдісі [1];
– дихотомия әдісі [2];
– «алтын қима» әдісі [1];
– Фибоначчи әдісі [1].

Программаның алгоритмін құрғанда [3, 4] пайдаландық.

Программаның Turbo Pascal тілінде құрылған коды және нәтижесі.

Программаның коды

Бірінші модуль

```
{Унимодалды функцияның минимумын берілген әдістермен табу.
Аздыратын модулі}
{$M 4096, 0, 0}
uses CRT, DOS;
var
  i: Integer;
{------}
procedure MakeFile;
{Клавиатурадан функцияны кіргізу және оны файлға жаздыру}
var
  f: Text;
  s: String [80];
begin {MakeFile}
  Assign (f,'function.pas');
  ReWrite (f);
  WriteLn (' Pascal тілінің ережесімен функцияны жазыңыз:');
  Write ('f (x) = ');
  ReadLn (s);
  WriteLn (f, ' begin F := '+s+' end;');
  Close (f)
end; {MakeFile}
{----------}
begin {main}
  ClrScr;
  WriteLn (Унимодалды функцияның минимумы);
  WriteLn (' берілген әдістерімен табу ');
  MakeFile;
  {Негізгі программаны компиляция жасаймыз...}
```

```
  Exec ('tpc.exe','optimiz.pas');
  GotoXY (1,WhereY-4);
  for i:=1 to 4 do DelLine;
  {...және оны орындауға аздыртамыз }
  Exec ('optimiz.exe','')
end. {main}
```

Екінші модуль

```
{Унимодалды функцияның минимумын берілген әдістерімен табу.
 Негізгі модулі}
uses Metod, CRT;
var
  u0,t,a,b,eps,x,effect: Real;
  c: Char;
  e: Boolean;
  i: Integer;
{----------}
function F (x:Real) : Real; far;
{Берілген  функциясы}
{$I function.pas}
{---------}
begin {main}
  Write ('Дәлдігі: eps = ');
  ReadLn (eps);
  {Свенн әдісімен [a,b] бастапқы кесіндіні табамыз}
  WriteLn ('Унимодалды кесіндіні табу');
  repeat
    Write ('Бастапқы нүкте: u0 = ');
    ReadLn (u0);
    Write ('Қадам: t = ');
    ReadLn (t);
    e := MetodSvenna (f,u0,t,a,b);
    if not e then
      begin
        WriteLn ('Берілген функция унимодалды функция емес');
        WriteLn ('Жаңа бастапқы нүктені алу керек');
        WriteLn ('Жалғастыру керек пе? (y/n)');
        if UpCase (c) = 'N' then Halt (0)
        else
        {Экраннан артық жолдарды алып тастаймыз}
          begin
            GotoXY (1,WhereY-3);
            for i:=1 to 3 do DelLine;
          end;
      end {if}
  until e;
  WriteLn (Бастапқы кесінді [a,b] = [',a:4:2,',',b:4:2,']');
  WriteLn;
  {Әр түрлі әдістерді қолдану}
  WriteLn ('Асыра артық алу әдісі:');
  MetodPerebora (f,a,b,eps,x,effect);
  Resultat (f,x,effect);
  WriteLn;
  WriteLn ('Дихотомия әдісі:');
  MetodDihotomii (f,a,b,eps,eps/10,x,effect);
  Resultat (f,x,effect);
  WriteLn;
```

```
  WriteLn ('Фибоначчи әдісі:');
  MetodFibonacci (f,a,b,eps,eps/10,x,effect);
  Resultat (f,x,effect);
  WriteLn;
  WriteLn (' «Алтын қима» әдісі:');
  MetodZolot (f,a,b,eps,x,effect);
  Resultat (f,x,effect);
  ReadLn
end. {main}
```

Үшінші модуль

```
{ Унимодалды функцияның минимумын берілген әдістерімен табу.
 Көмекші модулі}
unit Metod;
interface
type
  TFunc = function (x:Real) : Real;
function MetodSvenna (f:TFunc; u0,t:Real; var a,b:Real) : Boolean;
procedure MetodPerebora (f:TFunc; a,b,eps:Real; var x,effect:Real);
procedure MetodDihotomii (f:TFunc; a,b,eps,delta:Real; var x,effect:Real);
procedure MetodFibonacci (f:TFunc; a,b,l,e:Real; var x,effect:Real);
procedure MetodZolot (f:TFunc; a,b,eps:Real; var x,effect:Real);
procedure Resultat (f:TFunc; x,effect:Real);
implementation
{Бес әдістің орындалу процедуралары жазылады. Программаның жалғасын келесі мақалада келтіреміз}
{-----------}
procedure Resultat (f:TFunc; x,effect:Real);
{ Нәтижені  шығару }
begin {Resultat}
WriteLn ('x* = ',x:8:6,', f (x*) = ',f(x):8:6,', R = ',effect:6:4)
end; {Resultat}
{-----------}
end.
```

Программаның нәтижесі

(пайдаланушымен клавиатурадан кіргізетін мәліметтері бөлектелген)

1) Унимодалды функцияның минимумын берілген әдістерімен табу.
Pascal тілінің ережесімен функцияны жазыңыз:
f(x) = 2*x*-12*x
Дәлдігі: eps = **0.01**
Кесіндіні табу
Бастапқы нүкте: u0 = **5**
Қадам: t = **0.01**
Бастапқы кесінді [a,b] = [-0.11,3.73]
Асыра артық алу әдісі:
x* = 3.000000, f (x*) = -18.000000, R = 0.0026
Дихотомия әдісі:
x* = 3.002189, f (x*) = -17.999990, R = 0.0042
Фибоначчи әдісі:
x* = 2.999770, f (x*) = -18.000000, R = 0.0033
«Алтын қима» әдісі:
x* = 2.996625, f (x*) = -17.999977, R = 0.0050
2) Унимодалды функцияның минимумын берілген әдістерімен табу.
Pascal тілінің ережесімен функцияны жазыңыз:
f(x) = x*sin(x)+2*cos(x)
Дәлдігі: eps = **0.05**
Кесіндіні табу

Бастапқы нүкте: u0 = **1**
Қадам: t = **0.001**
Нәтижесі:
Бастапқы кесінді [a,b] = [3.05,9.19]
Асыра артық алу әдісі:
x* = 4.497000, f (x*) = -4.820544, R = 0.0081
Дихотомия әдісі:
x* = 4.512309, f (x*) = -4.819787, R = 0.0086
Фибоначчи әдісі:
x* = 4.497667, f (x*) = -4.820533, R = 0.0139
«Алтын қима» әдісі:
x* = 4.506942, f (x*) = -4.820170, R = 0.0132

Қорытынды. Тиімділіктің критерийі болып анықталмағандық интервалының азаюы сипаттамасы, яғни $R(N)$, табылады. Тиімді деп «алтын қима» мен Фибоначчи әдістерін санауға негіз бар. Асыра артық алу әдісі баяу жинақталады.

Әдебиеттер тізімі

1. *Пантелеев А.В., Летова Т.А.* Методы оптимизации в примерах и задачах. — М.: Высш. шк., 2002. — 542 с.
2. Сборник задач по математике для втузов. Специальные курсы / Под ред. А.В.Ефимова. — М.: Наука, 1984. — 301 с.
3. *Вирт Н.* Алгоритмы и структуры данных. — М.: Мир, 1989. — 360 с.
4. *Культин Н.Б.* Программирование в Turbo Pascal 7.0 и Delphi. — 2-е изд., перераб. и доп. — СПб.: БХВ–Санкт-Петербург, 1999. — 416 с.

УДК 517.43

А.Б.Шыракбаев

Таразский институт Международного казахско-турецкого университета им. А.Ясауи

АППРОКСИМАТИВНЫЕ СВОЙСТВА РЕШЕНИИ ЗАДАЧИ ДИРИХЛЕ ДЛЯ НЕЛИНЕЙНЫХ ВЫРОЖДАЮЩИХСЯ ЭЛЛИПТИЧЕСКИХ УРАВНЕНИЙ В СЛУЧАЕ ПРОИЗВОЛЬНОГО ВЫРОЖДЕНИЯ

Мақалада нұқсанды эллипстік түрдегі теңдеудің бір класы үшін жартылайпериодты Дирихле (ЖПД) есебінің бар болу теоремасы және жатықтығы дәлелденген. ЖПД теорема шешімімен байланысты жиынның Колмогоров көлденеңдерінің екі жақты бағасы туралы теорема алынған. Қорытынды нәтижелер механикада, газдинамикада және математика, физиканың басқа салаларында қолданыс табады.

The theorems are demonstrated about existence of smooth decisions of Dirihle's (DSP) semi-periodical problem for for one class innating elliptical equation in this work. With the help of this theorem, double-sided estimations of diameter are got according to Kolmogorov's multitudes tired bu DSP problems. The given results of problem are used in mechanics, gaure dynamics and in other the fields of mathematics and physics.

Введение

В области $\Omega = \{(x,y): -\pi < x < \pi, 0 < y < 1\}$ рассмотрим задачу

$$Lu = -k(y)u_{xx} - u_{yy} + a\left(y, \int_{-\pi}^{\pi} u(x,y)dx\right)u_x + c\left(y, \int_{-\pi}^{\pi} u(x,y)dx\right)u = f(x,y), \quad (0.1)$$

$$u(-\pi, y) = u(\pi, y), \ \ u_x(-\pi, y) = u_x(\pi, y), \quad (0.2)$$