Advanced Modeling and Optimization, Volume 10, Number 1, 2008[1]

An Unconstrained Optimization Test Functions Collection

Neculai Andrei
Research Institute for Informatics,
Center for Advanced Modeling and Optimization,
8-10, Averescu Avenue, Bucharest 1, Romania,
and
Academy of Romanian Scientists.
45, Splaiul Independentei, Bucharest 4, Romania
E-mail: nandrei@ici.ro

Abstract. A collection of unconstrained optimization test functions is presented. The purpose of this collection is to give to the optimization community a large number of general test functions to be used in testing the unconstrained optimization algorithms and comparisons studies. For each function we give its algebraic expression and the standard initial point. Some of the test fnctions are from the CUTE collection established by Bongartz, Conn, Gould and Toint, [1995], others are from Moré, Garbow and Hillstrom, [1981], Himmelblau [1972] or from some other papers or technical reports.

1. Introduction

Always theorists working in nonlinear programming area, as well as practical optimizers need to evaluate nonlinear optimization algorithms. Due to the hypothesis introduced in order to prove the convergence and the complexity of algorithms, the theory is not enough to establish the efficiency and the reliability of a method. As a consequence the only way to see the "power" of an algorithm remains its implementation in computer codes and its testing on large classes of test problems of different structures and characteristics. Besides, as George B. Dantzig (1914-2005) said "*the final test of a theory is its capacity to solve the problems which originated it*". This is the main reason we assembled here this collection of large-scale unconstrained optimization problems to test the theoretical developments in mathematical programming.

Nonlinear programming algorithms need to be tested at least in two different senses. Firstly, testing is always profitable into the process of development of an algorithm in order to evaluate the ideas and the corresponding algebraic procedures. Clearly, well designed test problems are very powerful in clarifying the algorithmic ideas and mechanisms. Secondly, a reasonably large set

[1] AMO - Advanced Modeling and Optimization: **ISSN: 1841-4311**

of test problems must be used in order to get an idea about the hypothesis used in proving the quality of the algorithm (local and global convergence, complexity) and to compare algorithms at an experimental level.

Generally, two types of (unconstrained) nonlinear programming problems can be identified: "*artificial problems*" and "*real-life problems*". The artificial nonlinear programming problems are used to see the behavior of the algorithms in different difficult situations like long narrow valleys, functions with significant null-space effects, essentially unimodal functions, functions with a huge number of significant local optima, etc. Figures 1-6 present some types of artificial nonlinear function in unconstrained optimization. All of them are of 2 variables, thus having the possibility for their graphical representation.

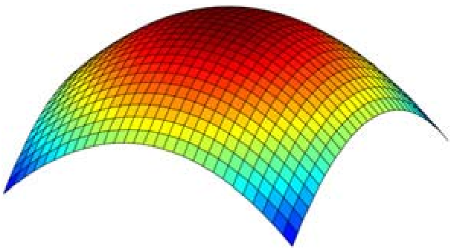

Fig. 1. Unimodal function.

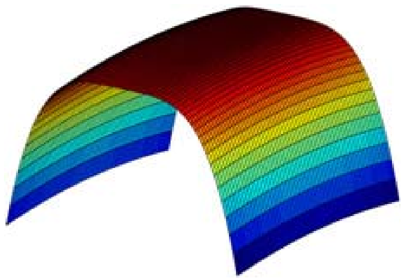

Fig. 2. Functions with significant null-space effects.

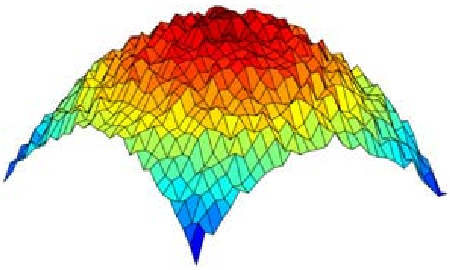

Fig. 3. Essentially unimodal functions.

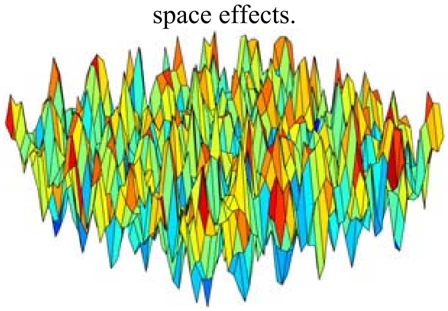

Fig. 4. Functions with a huge number of significant local optima.

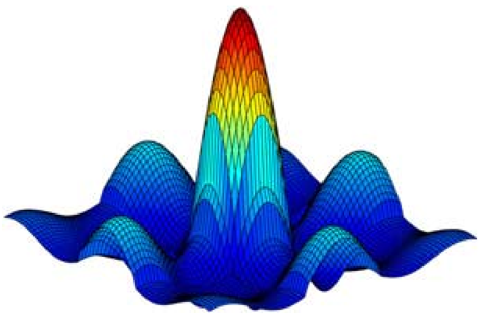

Fig. 5. Functions with a small number of significant local optima.

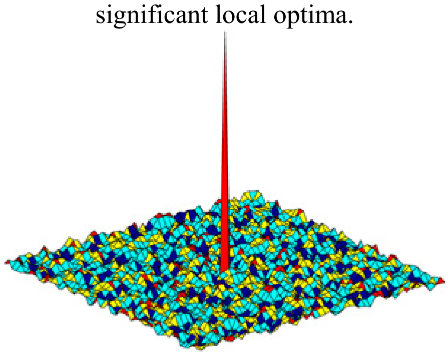

Fig. 6. Functions whose global structure provides no useful information about its optima.

The main characteristic of artificial nonlinear programming problems is that they are relatively easy to manipulate and to use into the process of algorithmic invention. Besides, the algorithmist may rapidly modify the problem in order to place the algorithm in different difficult conditions.

Real-life problems, on the other hand, are coming from different sources of applied optimization problems like physics, chemistry, engineering, biology, economy, oceanography, astronomy, meteorology, etc. Unlike artificial (unconstrained) nonlinear programming problems, real-life problems are not easily available and are difficult to manipulate. They may have complicated algebraic (or differential) expressions, may depend on a huge amount of data, and possible are dependent on some parameters which must be estimated in a specific way. A very nice collection of real-life unconstrained optimization problems is that given by Averick, *et al.* [1991, 1992].

In this collection we consider only artificial unconstrained optimization test problems. All of them are presented in extended or generalized form. The main difference between these forms is that while the problems in generalized form have the Hessian matrix as a block diagonal matrix, the extended forms have the Hessian as a multi-diagonal matrix. Many individuals have contributed, each of them in important ways, to the preparation of this collection. We do not mention them here. An important source of problems was the CUTE collection established by Bongartz, Conn, Gould and Toint, [1995]. Some other problems are from Moré, Garbow and Hillstrom, [1981], Himmelblau [1972] or are extracted from some other papers or technical reports. Generally, the problems in extended forms are slightly more difficult to be solved.

2. Unconstrained Optimization Test Functions

Extended Freudenstein & Roth function:

$$f(x)=\sum_{i=1}^{n/2}\left(-13+x_{2i-1}+((5-x_{2i})x_{2i}-2)x_{2i}\right)^2$$
$$+\left(-29+x_{2i-1}+((x_{2i}+1)x_{2i}-14)x_{2i}\right)^2,$$

$$x_0=[0.5,-2,0.5,-2,\ldots,0.5,-2].$$

Extended Trigonometric function:

$$f(x)=\sum_{i=1}^{n}\left(\left(n-\sum_{j=1}^{n}\cos x_j\right)+i(1-\cos x_i)-\sin x_i\right)^2,$$
$$x_0=[0.2,0.2,\ldots,0.2].$$

Extended Rosenbrock function:

$$f(x)=\sum_{i=1}^{n/2}c\left(x_{2i}-x_{2i-1}^2\right)^2+\left(1-x_{2i-1}\right)^2,\quad x_0=[-1.2,1,\ldots,-1.2,1].\quad \text{c=100.}$$

Generalized Rosenbrock function:

$$f(x)=\sum_{i=1}^{n-1} c\left(x_{i+1}-x_i^2\right)^2+\left(1-x_i\right)^2, \quad x_0=[-1.2,1,\ldots,-1.2,1], \quad c=100.$$

Extended White & Holst function:

$$f(x)=\sum_{i=1}^{n/2} c\left(x_{2i}-x_{2i-1}^3\right)^2+\left(1-x_{2i-1}\right)^2, \quad x_0=[-1.2,1,\ldots,-1.2,1], \quad \text{c=100.}$$

Extended Beale function:

$$f(x)=\sum_{i=1}^{n/2}\left(1.5-x_{2i-1}\left(1-x_{2i}\right)\right)^2+\left(2.25-x_{2i-1}\left(1-x_{2i}^2\right)\right)^2 +\left(2.625-x_{2i-1}\left(1-x_{2i}^3\right)\right)^2,$$

$$x_0=[1,0.8,\ldots,1,0.8].$$

Extended Penalty function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_i-1\right)^2+\left(\sum_{j=1}^{n} x_j^2-0.25\right)^2, \quad x_0=[1,2,\ldots,n].$$

Perturbed Quadratic function:

$$f(x)=\sum_{i=1}^{n} i x_i^2+\frac{1}{100}\left(\sum_{i=1}^{n} x_i\right)^2, \quad x_0=[0.5,0.5,\ldots,0.5]$$

Raydan 1 function:

$$f(x)=\sum_{i=1}^{n}\frac{i}{10}\left(\exp(x_i)-x_i\right), \quad x_0=[1,1,\ldots,1].$$

Raydan 2 function:

$$f(x)=\sum_{i=1}^{n}\left(\exp(x_i)-x_i\right), \quad x_0=[1,1,\ldots,1].$$

Diagonal 1 function:

$$f(x)=\sum_{i=1}^{n}(\exp(x_i)-i x_i), \quad x_0=[1/n,1/n,\ldots,1/n].$$

Diagonal 2 function:

$$f(x)=\sum_{i=1}^{n}\left(\exp(x_i)-\frac{x_i}{i}\right), \quad x_0=[1/1,1/2,\ldots,1/n].$$

Diagonal 3 function:

$$f(x)=\sum_{i=1}^{n}\left(\exp(x_i)-i\sin(x_i)\right),\qquad x_0=[1,1,\ldots,1].$$

Hager function:

$$f(x)=\sum_{i=1}^{n}\left(\exp(x_i)-\sqrt{i}\,x_i\right),\qquad x_0=[1,1,\ldots,1].$$

Generalized Tridiagonal 1 function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_i+x_{i+1}-3\right)^2+\left(x_i-x_{i+1}+1\right)^4,\qquad x_0=[2,2,\ldots,2].$$

Extended Tridiagonal 1 function:

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}+x_{2i}-3\right)^2+\left(x_{2i-1}-x_{2i}+1\right)^4,\qquad x_0=[2,2,\ldots,2].$$

Extended TET function : (Three exponential terms)

$$f(x)=\sum_{i=1}^{n/2}\left(\exp(x_{2i-1}+3x_{2i}-0.1)+\exp(x_{2i-1}-3x_{2i}-0.1)+\exp(-x_{2i-1}-0.1)\right),$$
$$x_0=[0.1,0.1,\ldots,0.1].$$

Generalized Tridiagonal 2 function:

$$f(x)=\left((5-3x_1-x_1^2)x_1-3x_2+1\right)^2+$$
$$\sum_{i=1}^{n-1}\left((5-3x_i-x_i^2)x_i-x_{i-1}-3x_{i+1}+1\right)^2+\left((5-3x_n-x_n^2)x_n-x_{n-1}+1\right)^2,$$
$$x_0=[-1,-1,\ldots,-1].$$

Diagonal 4 function:

$$f(x)=\sum_{i=1}^{n/2}\frac{1}{2}\left(x_{2i-1}^2+cx_{2i}^2\right),\qquad x_0=[1,1,\ldots,1],\qquad c=100.$$

Diagonal 5 function:

$$f(x)=\sum_{i=1}^{n}\log\left(\exp(x_i)+\exp(-x_i)\right),\qquad x_0=[1.1,1.1,\ldots,1.1].$$

Extended Himmelblau function:

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}^2+x_{2i}-11\right)^2+\left(x_{2i-1}+x_{2i}^2-7\right)^2,\qquad x_0=[1,1,\ldots,1].$$

Generalized White & Holst function:

$$f(x)=\sum_{i=1}^{n-1} c\left(x_{i+1}-x_i^3\right)^2+\left(1-x_i\right)^2, \quad x_0=[-1.2,1\ldots,-1.2,1], \quad c=100.$$

Generalized PSC1 function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_i^2+x_{i+1}^2+x_i x_{i+1}\right)^2+\sin^2(x_i)+\cos^2(x_i), \quad x_0=[3,0.1,\ldots,3,0.1].$$

Extended PSC1 function:

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}^2+x_{2i}^2+x_{2i-1}x_{2i}\right)^2+\sin^2(x_{2i-1})+\cos^2(x_{2i}),$$
$$x_0=[3,0.1,\ldots,3,0.1].$$

Extended Powell function:

$$f(x)=\sum_{i=1}^{n/4}\left(x_{4i-3}+10x_{4i-2}\right)^2+5\left(x_{4i-1}-x_{4i}\right)^2+$$
$$\left(x_{4i-2}-2x_{4i-1}\right)^4+10\left(x_{4i-3}-x_{4i}\right)^4,$$
$$x_0=[3,-1,0,1,\ldots,3,-1,0.1].$$

Full Hessian FH1 function:

$$f(x)=(x_1-3)^2+\sum_{i=2}^{n}\left(x_1-3-2(x_1+x_2+\cdots+x_i)^2\right)^2,$$
$$x_0=[0.01,0.01,\ldots,0.01].$$

Full Hessian FH2 function:

$$f(x)=(x_1-5)^2+\sum_{i=2}^{n}\left(x_1+x_2+\cdots+x_i-1\right)^2, \quad x_0=[0.01,0.01,\ldots,0.01].$$

Extended BD1 function (Block Diagonal):

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}^2+x_{2i}^2-2\right)^2+\left(\exp(x_{2i-1}-1)-x_{2i}\right)^2, \quad x_0=[0.1,0.1,\ldots,0.1].$$

Extended Maratos function:

$$f(x)=\sum_{i=1}^{n/2} x_{2i-1}+c\left(x_{2i-1}^2+x_{2i}^2-1\right)^2, \quad x_0=[1.1,0.1,\ldots,1.1,0.1], \quad c=100.$$

Extended Cliff function:

$$f(x)=\sum_{i=1}^{n/2}\left(\frac{x_{2i-1}-3}{100}\right)^2-\left(x_{2i-1}-x_{2i}\right)+\exp\left(20(x_{2i-1}-x_{2i})\right),$$
$$x_0=[0,-1,\ldots,0,-1].$$

Perturbed quadratic diagonal function:

$$f(x)=\left(\sum_{i=1}^{n}x_i\right)^2+\sum_{i=1}^{n}\frac{i}{100}x_i^2, \qquad x_0=[0.5,0.5,\ldots,0.5].$$

Extended Wood function:

$$f(x)=\sum_{i=1}^{n/4}100\left(x_{4i-3}^2-x_{4i-2}\right)^2+\left(x_{4i-3}-1\right)^2+90\left(x_{4i-1}^2-x_{4i}\right)^2+$$
$$\left(1-x_{4i-1}\right)^2+10.1\left\{\left(x_{4i-2}-1\right)^2+\left(x_{4i}-1\right)^2\right\}+19.8\left(x_{4i-2}-1\right)\left(x_{4i}-1\right),$$
$$x_0=[-3,-1,-3,-1,\ldots,-3,-1,-3,-1].$$

Extended Hiebert function:

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}-10\right)^2+\left(x_{2i-1}x_{2i}-50000\right)^2, \quad x_0=[0,0,\ldots,0].$$

Quadratic QF1 function:

$$f(x)=\frac{1}{2}\sum_{i=1}^{n}ix_i^2-x_n, \qquad x_0=[1,1,\ldots,1].$$

Extended quadratic penalty QP1 function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_i^2-2\right)^2+\left(\sum_{i=1}^{n}x_i^2-0.5\right)^2, \qquad x_0=[1,1,\ldots,1].$$

Extended quadratic penalty QP2 function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_i^2-\sin x_i\right)^2+\left(\sum_{i=1}^{n}x_i^2-100\right)^2, \quad x_0=[1,1,\ldots,1].$$

Quadratic QF2 function:

$$f(x)=\frac{1}{2}\sum_{i=1}^{n}i\left(x_i^2-1\right)^2-x_n, \qquad x_0=[0.5,0.5,\ldots,0.5].$$

Extended quadratic exponential EP1 function:

$$f(x)=\sum_{i=1}^{n/2}\left(\exp(x_{2i-1}-x_{2i})-5\right)^2+\left(x_{2i-1}-x_{2i}\right)^2\left(x_{2i-1}-x_{2i}-11\right)^2,$$
$$x_0=[1.5,1.5,\ldots,1.5].$$

Extended Tridiagonal 2 function:

$$f(x)=\sum_{i=1}^{n-1}\left(x_ix_{i+1}-1\right)^2+c\left(x_i+1\right)\left(x_{i+1}+1\right), \quad x_0=[1.,1.,\ldots,1.]. \qquad c=0.1$$

FLETCBV3 function (CUTE):

$$f(x)=\frac{1}{2}p\left(x_1^2+x_n^2\right)+\sum_{i=1}^{n-1}\frac{p}{2}\left(x_i-x_{i+1}\right)^2-\sum_{i=1}^{n}\left(\frac{p(h^2+2)}{h^2}x_i+\frac{cp}{h^2}\cos(x_i)\right),$$

where:

$$p=1/10^8,\qquad h=1/(n+1),\qquad c=1,\quad x_0=[h,2h,\ldots,nh].$$

FLETCHCR function (CUTE):

$$f(x)=\sum_{i=1}^{n-1}c\left(x_{i+1}-x_i+1-x_i^2\right)^2,\quad x_0=[0.,0.,\ldots,0.],\qquad c=100.$$

BDQRTIC function (CUTE):

$$f(x)=\sum_{i=1}^{n-4}\left(-4x_i+3\right)^2+\left(x_i^2+2x_{i+1}^2+3x_{i+2}^2+4x_{i+3}^2+5x_n^2\right)^2,$$
$$x_0=[1.,1.,\ldots,1.].$$

TRIDIA function (CUTE):

$$f(x)=\gamma\left(\delta x_1-1\right)^2+\sum_{i=2}^{n}i\left(\alpha x_i-\beta x_{i-1}\right)^2,$$
$$\alpha=2,\quad \beta=1,\quad \gamma=1,\quad \delta=1,\quad x_0=[1,1,\ldots,1].$$

ARGLINB function (CUTE):

$$f(x)=\sum_{i=1}^{m}\left(\sum_{j=1}^{n}ijx_j-1\right)^2,\quad x_0=[1,1,\ldots,1].$$

ARWHEAD function (CUTE):

$$f(x)=\sum_{i=1}^{n-1}\left(-4x_i+3\right)+\sum_{i=1}^{n-1}\left(x_i^2+x_n^2\right)^2,\qquad x_0=[1.,1.,\ldots,1.].$$

NONDIA function (CUTE):

$$f(x)=(x_1-1)^2+\sum_{i=2}^{n}100\left(x_1-x_{i-1}^2\right)^2,\quad x_0=[-1.,-1.,\ldots,-1.].$$

NONDQUAR function (CUTE):

$$f(x)=(x_1-x_2)^2+\sum_{i=1}^{n-2}\left(x_i+x_{i+1}+x_n\right)^4+(x_{n-1}+x_n)^2,$$
$$x_0=[1.,-1.,\ldots,1.,-1.,].$$

DQDRTIC function (CUTE):

$$f(x)=\sum_{i=1}^{n-2}\left(x_i^2+cx_{i+1}^2+dx_{i+2}^2\right),\quad c=100.,\quad d=100.,\quad x_0=[3.,3.,\ldots,3.].$$

EG2 function (CUTE):

$$f(x)=\sum_{i=1}^{n-1}\sin(x_1+x_i^2-1)+\frac{1}{2}\sin(x_n^2),\qquad x_0=[1.,1.,\ldots,1.].$$

CURLY20 function (CUTE):

$$f(x)=\sum_{i=1}^{n}q_i^4-20q_i^2-0.1q_i,$$

where:

$$q_i=\begin{cases}x_i+x_{i+1}+\cdots+x_{i+k}, & i\le n-k,\\ x_i+x_{i+1}+\cdots+x_n, & i>n-k,\end{cases}\qquad k=20,$$

$$x_0=[0.001/(n+1),\ldots,0.001/(n+1)].$$

DIXMAANA - DIXMAANL functions:

$$f(x)=1+\sum_{i=1}^{n}\alpha\,x_i^2\left(\frac{i}{n}\right)^{k1}+\sum_{i=1}^{n-1}\beta\,x_i^2\left(x_{i+1}+x_{i+1}^2\right)^2\left(\frac{i}{n}\right)^{k2}+$$
$$\sum_{i=1}^{2m}\gamma\,x_i^2x_{i+m}^4\left(\frac{i}{n}\right)^{k3}+\sum_{i=1}^{m}\delta\,x_ix_{i+2m}\left(\frac{i}{n}\right)^{k4},$$
$$m=n/3,$$

	α	β	γ	δ	k1	k2	k3	k4
A	1	0	0.125	0.125	0	0	0	0
B	1	0.0625	0.0625	0.0625	0	0	0	1
C	1	0.125	0.125	0.125	0	0	0	0
D	1	0.26	0.26	0.26	0	0	0	0
E	1	0	0.125	0.125	1	0	0	1
F	1	0.0625	0.0625	0.0625	1	0	0	1
G	1	0.125	0.125	0.125	1	0	0	1
H	1	0.26	0.26	0.26	1	0	0	1
I	1	0	0.125	0.125	2	0	0	2
J	1	0.0625	0.0625	0.0625	2	0	0	2
K	1	0.125	0.125	0.125	2	0	0	2
L	1	0.26	0.26	0.26	2	0	0	2

$$x_0=[2.,2.,\ldots,2.].$$

Partial Perturbed Quadratic function:

$$f(x)=x_1^2+\sum_{i=1}^{n}\left(ix_i^2+\frac{1}{100}\left(x_1+x_2+\cdots+x_i\right)^2\right),\qquad x_0=[0.5,0.5,\ldots,0.5].$$

Broyden Tridiagonal function:

$$f(x)=(3x_1-2x_1^2)^2+\sum_{i=2}^{n-1}(3x_i-2x_i^2-x_{i-1}-2x_{i+1}+1)^2+(3x_n-2x_n^2-x_{n-1}+1)^2,$$
$$x_0=[-1,-1,\ldots,-1].$$

Almost Perturbed Quadratic function:

$$f(x)=\sum_{i=1}^{n}ix_i^2+\frac{1}{100}(x_1+x_n)^2,\qquad x_0=[0.5,0.5,\ldots,0.5].$$

Perturbed Tridiagonal Quadratic function:

$$f(x)=x_1^2+\sum_{i=2}^{n-1}ix_i^2+(x_{i-1}+x_i+x_{i+1})^2,\qquad x_0=[0.5,0.5,\ldots,0.5].$$

Staircase 1 function:

$$f(x)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i}x_j\right)^2,\qquad x_0=[1,1,\ldots,1].$$

Staircase 2 function:

$$f(x)=\sum_{i=1}^{n}\left[\left(\sum_{j=1}^{i}x_j\right)-i\right]^2,\qquad x_0=[0,0,\ldots,0].$$

LIARWHD function (CUTE):

$$f(x)=\sum_{i=1}^{n}4(-x_1+x_i^2)^2+\sum_{i=1}^{n}(x_i-1)^2,\qquad x_0=[4,4,\ldots,4].$$

POWER function (CUTE):

$$f(x)=\sum_{i=1}^{n}(ix_i)^2,\qquad x_0=[1,1,\ldots,1].$$

ENGVAL1 function (CUTE):

$$f(x)=\sum_{i=1}^{n-1}(x_i^2+x_{i+1}^2)^2+\sum_{i=1}^{n-1}(-4x_i+3),\qquad x_0=[2,2,\ldots,2].$$

CRAGGLVY function (CUTE):

$$f(x)=\sum_{i=1}^{m}\left(\exp(x_{2i-1})-x_{2i}\right)^4+100(x_{2i}-x_{2i+1})^6+$$
$$(\tan(x_{2i+1}-x_{2i+2})+x_{2i+1}-x_{2i+2})^4+x_{2i-1}^8+(x_{2i+2}-1)^2,$$
$$x_0=[1,2,\ldots,2].$$

EDENSCH function (CUTE):

$$f(x) = 16 + \sum_{i=1}^{n-1}\left[(x_i - 2)^4 + (x_i x_{i+1} - 2x_{i+1})^2 + (x_{i+1} + 1)^2\right], \quad x_0 = [0,0,\ldots,0].$$

INDEF function (CUTE):

$$f(x) = \sum_{i=1}^{n} x_i + \sum_{i=2}^{n-1} \frac{1}{2}\cos(2x_i - x_n - x_1), \qquad x_0 = \left[\frac{1}{n+1}, \frac{2}{n+1}, \ldots, \frac{n}{n+1}\right].$$

CUBE function (CUTE):

$$f(x) = (x_1 - 1)^2 + \sum_{i=2}^{n} 100(x_i - x_{i-1}^3)^2, \qquad x_0 = [-1.2,1,-1.2,1,\ldots,-1.2,1].$$

EXPLIN1 function (CUTE):

$$f(x) = \exp(0.1 x_i x_{i+1}) - 10\sum_{i=1}^{n}(i x_i), \qquad x_0 = [0,0,\ldots,0].$$

EXPLIN2 function (CUTE):

$$f(x) = \sum_{i=1}^{m} \exp\left(\frac{i x_i x_{i+1}}{10m}\right) - 10\sum_{i=1}^{n}(i x_i), \qquad x_0 = [0,0,\ldots,0].$$

ARGLINC function (CUTE):

$$f(x) = 2 + \sum_{i=2}^{m-1}\left(\sum_{j=2}^{n-1} j x_j (i-1) - 1\right)^2, \qquad x_0 = [1,1,\ldots,1].$$

BDEXP function (CUTE):

$$f(x) = \sum_{i=1}^{n-2}\left(x_i + x_{i+1}\right)\exp\left(-x_{i+2}(x_i + x_{i+1})\right), \qquad x_0 = [1,1,\ldots,1].$$

HARKERP2 function (CUTE):

$$f(x) = \left(\sum_{i=1}^{n} x_i\right)^2 - \sum_{i=1}^{n}\left(x_i + \frac{1}{2}x_i^2\right) + 2\sum_{j=2}^{n}\left(\sum_{i=j}^{n} x_i\right)^2, \qquad x_0 = [1,2,\ldots,n].$$

GENHUMPS function (CUTE):

$$f(x) = \sum_{i=1}^{n-1} \sin(2x_i)^2 \sin(2x_{i+1})^2 + 0.05(x_i^2 + x_{i+1}^2),$$
$$x_0 = [-506.,506.2,\ldots,506.2].$$

MCCORMCK function (CUTE):

$$f(x) = \sum_{i=1}^{n-1}\left(-1.5x_i + 2.5x_{i+1} + 1 + (x_i - x_{i+1})^2 + \sin(x_i + x_{i+1})\right),$$
$$x_0 = [1,1,\ldots,1].$$

NONSCOMP function (CUTE):

$$f(x)=(x_1-1)^2+\sum_{i=2}^{n}4(x_i-x_{i-1}^2)^2, \qquad x_0=[3,3,\ldots,3].$$

VARDIM function (CUTE):

$$f(x)=\sum_{i=1}^{n}(x_i-1)^2+\left(\sum_{i=1}^{n}ix_i-\frac{n(n+1)}{2}\right)^2+\left(\sum_{i=1}^{n}ix_i-\frac{n(n+1)}{2}\right)^4,$$

$$x_0=\left[1-\frac{1}{n},1-\frac{2}{n},\ldots,1-\frac{n}{n}\right].$$

QUARTC function (CUTE):

$$f(x)=\sum_{i=1}^{n}\left(x_i-1\right)^4, \qquad x_0=[2.,2.,\ldots,2.].$$

Diagonal 6 function:

$$f(x)=\sum_{i=1}^{n}e^{x_i}-(1-x_i), \qquad x_0=[1,1,\ldots,1].$$

SINQUAD function (CUTE):

$$f(x)=(x_1-1)^4+\sum_{i=2}^{n-1}\left(\sin(x_i-x_n)-x_1^2+x_i^2\right)^2+(x_n^2-x_1^2)^2,$$

$$x_0=[0.1,0.1,\ldots,0.1].$$

Extended DENSCHNB function (CUTE):

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}-2\right)^2+\left(x_{2i-1}-2\right)^2x_{2i}^2+\left(x_{2i}+1\right)^2, \quad x_0=[1,1,\ldots,1].$$

Extended DENSCHNF function (CUTE):

$$f(x)=\sum_{i=1}^{n/2}\left(2(x_{2i-1}+x_{2i})^2+(x_{2i-1}-x_{2i})^2-8\right)^2+\left(5x_{2i-1}^2+(x_{2i}-3)^2-9\right)^2,$$

$$x_0=[2.,0.,2.,0.,\ldots,2.,0.].$$

LIARWHD function (CUTE):

$$f(x)=\sum_{i=1}^{n}4(x_i^2-x_1)^2+\sum_{i=1}^{n}(x_i-1)^2, \qquad x_0=[4.,4.,\ldots,4.].$$

DIXON3DQ function (CUTE):

$$f(x)=(x_1-1)^2+\sum_{j=1}^{n-1}(x_j-x_{j+1})^2+(x_n-1)^2, \qquad x_0=[-1.,-1.,\ldots,-1].$$

COSINE function (CUTE):

$$f(x)=\sum_{i=1}^{n-1}\cos\left(-0.5x_{i+1}+x_i^2\right), \qquad x_0=[1,1,\ldots,1].$$

SINE function:

$$f(x)=\sum_{i=1}^{n-1}\sin\left(-0.5x_{i+1}+x_i^2\right), \qquad x_0=[1,1,\ldots,1].$$

BIGGSB1 function (CUTE):

$$f(x)=(x_1-1)^2+\sum_{i=1}^{n-1}\left(x_{i+1}-x_i\right)^2+(1-x_n)^2, \quad x_0=[0,0,\ldots,0].$$

Generalized Quartic function:

$$f(x)=\sum_{i=1}^{n-1}x_i^2+\left(x_{i+1}+x_i^2\right)^2, \qquad x_0=[1,1,\ldots,1].$$

Diagonal 7 function:

$$f(x)=\sum_{i=1}^{n}\exp(x_i)-2x_i-x_i^2, \qquad x_0=[1,1,\ldots,1].$$

Diagonal 8 function:

$$f(x)=\sum_{i=1}^{n}x_i\exp(x_i)-2x_i-x_i^2, \qquad x_0=[1,1,\ldots,1].$$

Full Hessian FH3 function:

$$f(x)=\left(\sum_{i=1}^{n}x_i\right)^2+\sum_{i=1}^{n}(x_i\exp(x_i)-2x_i-x_i^2), \qquad x_0=[1,1,\ldots,1].$$

SINCOS function:

$$f(x)=\sum_{i=1}^{n/2}\left(x_{2i-1}^2+x_{2i}^2+x_{2i-1}x_{2i}\right)^2+\sin^2 x_{2i-1}+\cos^2 x_{2i},$$
$$x_0=[3,0.1,3,0.1,\ldots,3,0.1].$$

Diagonal 9 function:

$$f(x)=\sum_{i=1}^{n-1}(\exp(x_i)-ix_i)+10000x_n^2, \quad x_0=[1,1,\ldots,1].$$

HIMMELBG function (CUTE):

$$f(x)=\sum_{i=1}^{n/2}\left(2x_{2i-1}^2+3x_{2i}^2\right)\exp\left(-x_{2i-1}-x_{2i}\right), \quad x_0=[1.5,1.5,\ldots,1.5].$$

HIMMELH function (CUTE):

$$f(x)=\sum_{i=1}^{n/2}(-3x_{2i-1}-2x_{2i}+2+x_{2i-1}^3+x_{2i}^2), \quad x_0=[1.5,1.5,\ldots,1.5].$$

References

Andrei, N., (1995) *Computational experience with conjugate gradient algorithms for large-scale unconstrained optimization.* Technical Report, Research Institute for Informatics, ICI, Bucharest, July 21, 1995, pg.1-14.

Andrei, N., (1999a) *Programarea Matematică Avansată. Teorie, Metode Computaţionale, Aplicaţii.* Editura Tehnică, Bucureşti, 1999.

Andrei, N., (2003) *Modele, Probleme de Test şi Aplicaţii de Programare Matematică.* Editura Tehnică, Bucureşti, 2003.

Andrei, N., "*Test functions for unconstrained optimization*". http://www.ici.ro/camo/neculai/HYBRID/evalfg.for

Andrei, N., (2006) *An acceleration of gradient descent algorithm with backtracking for unconstrained optimization*, Numerical Algorithms, Vol. 42, pp. 63-73, 2006.

Andrei, N., (2007) *Scaled conjugate gradient algorithms for unconstrained optimization*. Computational Optimization and Applications, 38 (2007), pp. 401-416.

Andrei, N., (2007) *Scaled memoryless BFGS preconditioned conjugate gradient algorithm for unconstrained optimization.* Optimization Methods and Software, 22 (2007), 561-571.

Andrei, N., (2007) *A scaled BFGS preconditioned conjugate gradient algorithm for unconstrained optimization*. Applied Mathematics Letters, 20 (2007), 645-650.

Andrei, N., (2007) *Numerical comparison of conjugate gradient algorithms for unconstrained optimization.* Studies in Informatics and Control, 16 (2007), pp.333-352.

Andrei, N., (2008) *Acceleration of conjugate gradient algorithms for unconstrained optimization.* Submitted.

Andrei, N., (2008) *Accelerated conjugate gradient algorithm with finite difference Hessian / vector product approximation for unconstrained optimization.* ICI Technical Report, March 4, 2008.

Andrei, N., (2008) *Performance profiles of conjugate gradient algorithms for unconstrained optimization.* Encyclopedia of Optimization, 2nd Edition, C.A. Floudas and P. Pardalos (Eds.), Spriger Science + Business Media, New York, 2008, Entry 762.

Averick, B.M., Carter, R.G., Moré, J.J., (1991) *The MINPACK-2 test problem collection (Preliminary version),* Mathematics and Computer Science Division, Argonne National Laboratory, Thechnical Memorandum No. 150, May 1991.

Averick, B.M., Carter, R.G., Moré, J.J., Xue, G.L. (1992) *The MINPACK-2 test problem collection.* Mathematics and Computer Science Division, Argonne National Laboratory, Preprint MCS-P153-0692, June 1992.

Bazaraa, M.S., Sheraly, H.D., Shety, C.M., (1993) *Nonlinear Programming. Theory and algorithms.* John Wiley, New York, 1993.

Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, P.L., (1995) *CUTE: constrained and unconstrained testing environments*, ACM TOMS, 21, (1995) 123-160.

Brown, A.A., Bartholomew-Biggs, M.C., (1987a) *Some effective methods for unconstrained optimization based on the solution of systems of ordinary differential equations*. Technical Report No. 178, The Hatfield Polytechnic, Numerical Optimisation Centre, April 1987.

Brown, A.A., Bartholomew-Biggs, M.C., (1987b) *ODE vs SQP methods for constrained optimization.* Technical Report No. 179, The Hatfield Polytechnic, Numerical Optimisation Centre, June 1987.

Brown, A.A., Bartholomew-Biggs, M.C., (1989) *Some effective methods for unconstrained optimization based on the solution of systems of ordinary differential equations*. J. Optim. Theory Appl., vol.62, (1989), pp. 211-224.

Conn, A.R., Nick Gould, Ph. L. Toint, (1995) *Intensive numerical tests with LANCELOT (Release A). The complete results.* Report 92/15 (updated), September 11, 1995.

Dai, Y.H., Liao, L.Z., (2001) *New conjugacy conditions and related nonlinear conjugate gradient methods.* Appl. Math. Optim., 43, (2001), pp.87-101.

Dai, Y.H., Yuan, Y., (2001) *An efficient hybrid conjugate gradient method for unconstrained optimization.* Annals of Operations Research, 103 (2001) pp.33-47.

Dumitru, V., (1975) *Programare Neliniară. Algoritmi, programe, rezultate numerice.* [Apendix P în colaborare cu Fl. Luban, S. Moga, R. Şerban]. Editura Academiei Române, Bucureşti, 1975.

Grippo, L., Lampariello, F., Lucidi, S., (1986) A nonmonotone line search technique for Newton's method. SIAM J. Numer. Anal., 23 (1986), pp.707-716

Hager, W.W., Zhang, H., (2005) *A new conjugate gradient method with guaranteed descent and an efficient line search*. SIAM Journal on Optimization, 16 (2005) 170-192.

Himmelblau, D., (1972) *Applied nonlinear programming*. McGraw-Hill, New York, 1972.

Lee, D., *A fast and robust unconstrained optimization method requiring minimum storage*. Mathematical programming, 32 (1985), pp.41-68.

Martínez, J.M., (2008) *Geometric nonlinear programming test problems.* SIAG/OPT Views and News. A forum for the SIAM Activity Group on Optimization, Vol. 19, No. 1, March 2008, pp.1-10.

Mittelmann, H.D., (2007) *DTOS – A service for the optimization community.* SIAG/OPT Views and News. A Forum for the SIAM Activity Group on Optimization, Vol. 18, No. 1, March 2007, pp. 17-20.

Moré, J.J., Garbow, B.S., Hillstrom, K.E., (1981) *Testing unconstrained optimization software*. ACM Trans. Math. Soft. 7, (1981), pp.17-41.

Osborne, M., and **Saunders, M.A.,** (1972) *Descent methods for minimisation.* In R.S. Andersen, L.S. Jennings and D.M. Ryan (Eds.) University of Queensland Press, Brisbane, 1972, pp.221-237.

Raydan, M., (1997) *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7 (1997), pp. 26-33.

Resende, M.G.C., (2007) *An optimizer in the telecommunications industry.* SIAG/OPT Views and News. A Forum for the SIAM Activity Group on Optimization, Vol. 18, No. 2, October 2007, pp. 8-19.

Shanno, D.F., (1978) *Conjugate gradient methods with inexact searches*. Mathematics of Operations Research, 3 (1978), pp.244-256.

Shanno, D.F. and **Phua, K.H.,** (1978) *Matrix conditioning and nonlinear optimisation*. Mathematical Programming, 14 (1978), pp.149-160.