Computational Mathematics and Mathematical Physics, Vol. 39, No. 5, 1999, pp. 727–737.
From Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, Vol. 39, No. 5, 1999, pp. 759–769.

110

A New Interval Approach to Global Optimization[1]

L. V. Kolev

Sofia Technical University, Faculty of Automation, Sofia, 1756 Bulgaria

Received December 12, 1997; in final form, June 4, 1998

Abstract—Interval methods are iterative methods that can be used to solve the general nonlinear programming problem globally, providing exact bounds both on the optimum (optima) and on the corresponding solution coordinates. However, their computational complexity grows rapidly with the dimension of the problem and the size of the search domain. In this paper, a new interval approach to the global optimization problem is suggested that makes it possible to develop interval optimization methods of improved efficiency. It is based on the following ideas. First, every nonlinear function $f_i(x)$ involved in the solution scheme chosen is transformed into a semiseparable form (sum of terms). Each term of this form is either a function $f_{ij}(x_j)$ of a single variable or the product $x_k x_l$ of two variables. These terms are then enclosed by corresponding linear interval functions. Thus, at each iteration of the computation process, a specific linear interval system is obtained where only the right-hand side involves intervals, whereas the known interval methods are based on a linear system with interval coefficients. The former system is much easier to solve, which accounts for the high numerical efficiency of the new approach.

1. INTRODUCTION

Interval methods (see [1–3] and the references therein cited, as well as [4, Ch. 2], for technical applications) have proved to be reliable as tools for solving globally the general nonlinear programming problem

$$\text{minimize } \varphi_0(x) \qquad (1.1)$$

subject to $\varphi_i(x) \le 0$, $i = 1, 2, \ldots, r_1$, $\varphi_i(x) = 0$, $i = r_1 + 1, \ldots, r_2$, $x \in X^{(0)} \subset \mathbb{R}^{n_0}$, or its variants (when some of the constraints are missing). The methods are iterative, and the initial box $X^{(0)}$ is dynamically subdivided into subboxes during the computation process. At each iteration, an attempt is made to reduce the size of, or to discard altogether, the current box X by applying appropriate reduction–elimination procedures [2], which depend on the problem at hand and the available information on the functions involved. If the convexity test (requiring second-order derivatives of φ_0) is not used, the known reduction–elimination techniques are based on the solution of a corresponding system, which consists (depending on the method chosen for solving the minimization problem considered) of linear interval equalities and/or linear interval inequalities. Each ith row of the system is, typically, of the form

$$\sum_{j=1}^{n_1} A_{ij}(X)(y_j - x_j) \cong b_i(x), \quad i = 1, 2, \ldots, r_3, \qquad (1.2)$$

where $A_{ij}(X)$ is the interval extension of a corresponding first-order interval derivative (or slope) $a_{ij}(x)$ while $b_i(x)$ is (in exact arithmetic) a real number, the symbol $\cong$ stands for either equality or inequality sign, and n_1 is the dimension of x (generally, higher than n_0). System (1.2) is an interval approximation (in X) of the corresponding real system (see [2])

$$\psi_i(x_1, \ldots, x_{n_1}) \cong 0, \quad i = 1, 2, \ldots, r_3 \qquad (1.2)'$$

{having the same number n_1 of variables and the same number r_3 of rows as system (1.2)}.

Thus, the numerical efficiency of a specific first-order interval method for global optimization is essentially determined by the following two factors:

(i) overestimation of $A_{ij}(X)$ with respect to the range $a_{ij}(X)$ of $a_{ij}(x)$ in X;

(ii) overestimation of an approximate solution Y of system (1.2) with respect to its exact (optimal) solution Y^*.

Unfortunately, for larger n_0 and $X^{(0)}$, both overestimations are rather pronounced, especially at early iterations, which accounts for the relatively low efficiency of the known global optimization interval methods.

[1] This article was submitted by the author in English.

In this paper, a new approach to addressing the global optimization problem is suggested, which seems to lead to interval methods of improved numerical efficiency. Starting, once again, from the real system (1.2)', it involves the following major stages.

1. Each nonlinear function $\psi_i(x)$ in equation (1.2)' is transformed into a set of functions of the so-called semiseparable form.

2. Each nonlinear term of each transformed function is enclosed in an optimal manner by a corresponding linear interval expression having a real (non-interval) slope.

3. A system of linear interval equalities and/or linear interval inequalities is thus obtained. Each ith row of the latter system has the form

$$\sum_{j=1}^{n_2} a_{ij}(X)x_j \cong B_i(X), \quad i = 1, 2, \ldots, r_4, \tag{1.3}$$

where, in contrast to system (1.2), all coefficients $a_{ij}(X)$ are now constants while only the right-hand side term $B_i(X)$ is an interval; in the general case, $n_2 > n_1$ and $r_4 > r_3$.

4. System (1.3) is solved by an appropriate method.

Stages 2 to 4 are carried out repeatedly in the iteration process.

Thus, the basic difference between the new approach and the conventional interval methods lies in the fact that system (1.2) is a system with interval coefficients, whereas system (1.3) has constant coefficients. The latter system is much easier to solve [although the size of system (1.3) is larger than that of system (1.2)] and the new approach is expected to have a better numerical efficiency.

The paper is organized as follows. The first three stages of the novel approach are presented in Section 2. The last stage is considered in Section 3 for the case when (1.3) is a system of equations arising from the application of the John conditions. Two constrained optimization problems illustrating the features of the present approach are solved in Section 4. Finally, concluding remarks are given in Section 5.

2. BASIC APPROACH

According to the first stage of the present approach, the functions ψ_i from system (1.2)' are to be transformed into a semiseparable form. This form can be introduced in the most natural manner by first presenting the transformation of a function into separable form.

2.1. Transformation into Separable Form

It is well known that a function f: $\mathbb{R}^n \longrightarrow \mathbb{R}$ is separable (of separable form) if

$$f(x) = \sum_{j=1}^{n} f_j(x_j),$$

where x_j is the jth component of x. Typically, the functions $\psi_i(x)$ in (1.2)' are not of separable form. However, they can be transformed into a set of separable functions

$$f_i(x) = \sum_{j=1}^{n_2'} f_{ij}(x_j), \quad i = 1, 2, \ldots, r_4', \tag{2.1}$$

where $n_2' > n_1$ and $r_4' > r_3$. The functions $\psi_i(x)$ in equation (1.2)' are assumed to be factorable functions [5], i.e., functions that are composed of the arithmetic operations +, −, ×, /, unary operations (sin, exp, log, sqrt, etc.), and powers (∧).

The theoretical basis for such a transformation is a famous theorem formulated by Kolmogorov [6]. However, its proof is not constructive and only recently has a simple algorithm been proposed [7] to convert factorable functions into separable functions automatically on a computer. Several basic facts from [7], needed later on, are briefly presented here.

Let f_L and f_R be subfunctions of ψ_i containing at least one variable. Consider the following three cases:

$$f = f_L \cdot f_R, \tag{2.2}$$

$$f = f_L / f_R, \tag{2.3}$$

$$f = f_L^{f_R}. \tag{2.4}$$

If both f_L and f_R contain only one and the same variable, then f is obviously separable in all of the three cases.

If f_L contains only one variable and f_R contains only another variable, then the functions from equation (2.2) and equation (2.3) can be easily transformed into separable form as follows. The transformation of (2.2) is

$$f = f_L \cdot f_R \longrightarrow \{f = (y_1^2 - f_L^2 - f_R)^2/2,\ y_1 = f_L + f_R\}. \tag{2.5}$$

The second case is reduced to the first one by setting

$$f_R = 1/f_R$$

and applying transformation (2.5).

In the third case, the transformation suggested in [7] is

$$f = f_L^{f_R} \longrightarrow f = \exp(y_1),\ y_1 = f_R \cdot \log(f_L),\ y_1 = [y_2^2 - f_R^2 - \log(f_L^2)]/2,\ y_2 = f_R + \log(f_L). \tag{2.6}$$

It should be mentioned that transformation (2.6) is valid only if $f_L > 0$ for all values of its argument.

If both f_L and f_R contain more than one variable, then we first introduce auxiliary variables and then apply the above approach. To illustrate this possibility, consider formula (2.2). In this case,

$$f = f_L \cdot f_R \longrightarrow f = y_1 \cdot y_2,\ f = (y_3^2 - y_1^2 - y_2^2)/2,\ y_1 = f_L,\ y_2 = f_R,\ y_3 = y_1 + y_2. \tag{2.7}$$

To illustrate the above approach, we consider the following example (see [2]).

Example 1. Minimize

$$\varphi_0(x) = x_1 \tag{2.8}$$

subject to $\varphi_1(x) = x_1^2 + x_2^2 - 1 \le 0$, $\varphi_2(x) = x_1^2 - x_2 \le 0$.

If the normalized Lagrange multipliers method is used, the normalization condition for the multipliers is

$$u_0 + u_1 + u_2 = 1, \tag{2.9}$$

since all constraints in (2.8) are inequalities. After explicitly eliminating u_0, the John conditions corresponding to problem (2.8) can be written as

$$1 - u_1 - u_2 + 2x_1(u_1 + u_2) = 0,\quad 2x_2u_1 - u_2 = 0,\quad u_1\varphi_1(x) = 0,\quad u_2\varphi_2(x) = 0. \tag{2.10}$$

In this instance, the solution scheme chosen (Lagrange multipliers method, John conditions, and explicit elimination of one multiplier u_0) leads to system (2.10), consisting of four equations in four variables. Let $x = (x_1, x_2, u_1, u_2)$ and $\psi = (\psi_1, \ldots, \psi_4)$; then, (2.10) can be recast as

$$\begin{gathered}\psi_1(x) = 1 - x_3 - x_4 + 2x_1(x_3 + x_4) = 0,\quad \psi_2(x) = 2x_2x_3 - x_4 = 0,\\ \psi_3(x) = x_3\varphi_1(x_1, x_2) = 0,\quad \psi_4(x) = x_4\varphi_2(x_1, x_2) = 0.\end{gathered} \tag{2.11}$$

and (2.1) corresponds to system (1.2)' in the special case when (1.2)' is a system for n unknowns,

$$\psi(x) = 0.$$

It is seen from (2.11) that each function $\psi_i(x)$ is not separable, although all functions $\varphi_i(x)$ in the original problem (2.8) are separable.

To transform (2.11) into a separable system, we first introduce

$$x_5 = \varphi_1(x_1, x_2),\quad x_6 = \varphi_2(x_1, x_2),\quad x_7 = x_3 + x_4 \tag{2.12}$$

to get the system

$$\begin{gathered}1 - x_7 + 2x_1x_7 = 0,\quad 2x_2x_3 - x_4 = 0,\quad x_3x_5 = 0,\quad x_4x_6 = 0,\\ x_1^2 + x_2^2 - 1 - x_5 = 0,\quad x_1^2 - x_2 - x_6 = 0,\quad x_3 + x_4 - x_7 = 0.\end{gathered} \tag{2.13}$$

Systems of the above type, containing only products of at most two arguments as terms that have not yet been transformed into separable form, are henceforth called systems of semiseparable form.

Next, we eliminate the products in the first four equations of system (2.13), using (2.5) to get the final system of separable form

$$1 - x_7 + x_8^2 - x_1^2 - x_7^2 = 0,\quad x_1 + x_7 - x_8 = 0,\quad x_9^2 - x_2^2 - x_3^2 - x_4 = 0,$$

$$x_2 + x_3 - x_9 = 0, \quad x_{10}^2 - x_3^2 - x_5^2 = 0, \quad x_3 + x_5 - x_{10} = 0, \quad x_{11}^2 - x_4^2 - x_6^2 = 0, \tag{2.14a}$$
$$x_4 + x_6 - x_{11} = 0, \quad x_1^2 - x_2^2 - 1 - x_5 = 0, \quad x_1^2 + x_2 - x_6 = 0, \quad x_3 + x_4 - x_7 = 0.$$

Each x_i ($i = 1, ..., 11$) in equations (2.14a) belongs to some interval X_i. Indeed, it follows from the constraints in (2.8) that

$$x_1 \in X_1 = [-1, 1], \quad x_2 \in X_2 = [0, 1]; \tag{2.14b}$$

similarly, on account of (2.9) (see [2]),

$$x_3 \in X_3 = [0, 1], \quad x_4 \in X_4 = [0, 1]. \tag{2.14c}$$

The remaining intervals X_i ($i > 4$) can be easily found by using the corresponding relations (2.12), etc.

It is seen that, following the general scheme from [7], the separable system thus obtained consists of a total of 11 equations, whereas the original system (2.8) contains only four equations.

2.2. *Enclosures by Linear Interval Functions*

Consider a term $f_{ij}(x_j)$ of a function f_i of separable form. No restrictions on these functions are imposed, except that they are assumed to be continuous. In this subsection, a new interval enclosure of $f_{ij}(x_j)$ on a given interval X_j will be suggested. Unlike previous methods, where the functions $f_{ij}(x_j)$ are assumed to be continuously differentiable (CD) and are approximated on X_j by enclosures using interval derivatives [4, 8] or interval slopes [9, 10], the new approximation is chosen in the following form:

$$L_{ij}(X_j) = B_{ij} + a_{ij}x_j, \quad x_j \in X_j, \tag{2.15}$$

where $B_{ij} = [\underline{b}_{ij}, \bar{b}_{ij}]$ is an interval while a_{ij} is a real number. Both B_{ij} and a_{ij} must be defined so that the following inclusion holds:

$$f_{ij}(x_j) \in B_{ij} + a_{ij}x_j, \quad x_j \in X_j. \tag{2.16}$$

A procedure for finding a_{ij}, $\underline{b}_{ij}$, and $\bar{b}_{ij}$ is suggested here for CD functions. It is motivated by elementary geometrical considerations (Fig. 1a) and can be readily adapted for the case of functions that are only continuous (Fig. 1b).

Procedure 1. First, compute

$$\underline{f}_{ij} = f_{ij}(\underline{x}_j), \quad \bar{f}_{ij} = f_{ij}(\bar{x}_j).$$

Then, a_{ij} is defined as the slope

$$a_{ij} = (\bar{f}_{ij} - \underline{f}_{ij})/(\bar{x}_j - \underline{x}_j).$$

Afterwards, the equation

$$\frac{d}{dx_j}(f_{ij}) = f'_{ij}(x_j) = a_{ij} \tag{2.17}$$

is solved on X_j for x_j. In the general case, (2.17) has several solutions. Among them, two solutions, denoted by x'_j and x''_j are chosen so that the following conditions are satisfied. Let

$$l_1(x_j) = \bar{b}_{ij} + a_{ij}x_j$$

be a straight line passing through the point $(x'_j, f_{ij}(x'_j))$ and having a slope a_{ij} such that

$$f_{ij}(x_j) \le l_1(x_j), \quad x_j \in X_j.$$

Similarly, let the straight line

$$l_2(x_j) = \underline{b}_{ij} + a_{ij}x_j$$

passing through the point $(x''_j, f_{ij}(x''_j))$ have the property

$$f_{ij}(x_j) \ge l_2(x_j), \quad x_j \in X_j.$$

Now, it is easily seen that

$$\bar{b}_{ij} = f_{ij}(x'_j) - a_{ij}x'_j, \quad \underline{b}_{ij} = f_{ij}(x''_j) - a_{ij}x''_j. \tag{2.18}$$

1

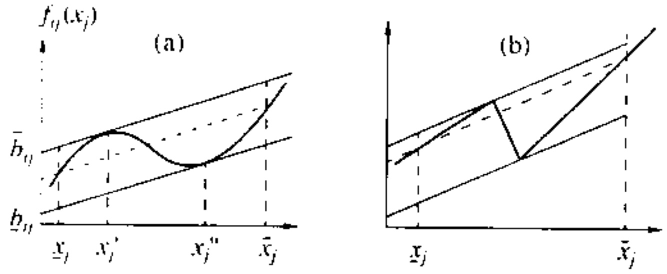

Fig. 1.

In the special case when $f_{ij}(x_j)$ is either convex or concave on X_j, equation (2.17) has a unique solution x_j^*. In this cases formulas (2.18) should be modified as follows. If $f_{ij}(x_j)$ is convex, then set $x_j' = \underline{x}_j$ and $x_j'' = x_j^*$; similarly, in the case of a concave function $f_{ij}(x_j)$, set $x_j' = x_j^*$ and $x_j'' = \underline{x}_j$.

The efficiency of the above procedure primarily depends on how much computational effort is needed to locate all the real solutions to equation (2.17) within the interval X_j. If this problem is easy to solve, then Procedure 1 provides a simple and efficient way of determining the linear interval approximation (2.15).

Adding up all terms $L_{ij}(X_j)$ for a given index i, we obtain a linear enclosure for $f_i(x)$ in (2.1)

$$L_i(X) = \sum_{j=1}^{n_2'} L_{ij}(X_j) = \sum_{j=1}^{n_2'} a_{ij}(X_j)x_j - B_i(X), \tag{2.19}$$

where

$$B_i(X) = -\sum_{j=1}^{n_2'} B_{ij}(X_j). \tag{2.20}$$

From (2.1), (2.15), (2.16), (2.19), and (2.20), it is clear that

$$f_i(x) \in L_i(X). \tag{2.21}$$

It is seen from (2.19) and (2.21) that, for the current box $X = (X_1, \ldots, X_{n_2'})$, the nonlinear system (1.2)' can be represented by the linear interval separable system

$$\sum_{j=1}^{n_2'} a_{ij}(X_j)x_j \cong B_i(X), \quad i = 1, 2, \ldots, r_4'. \tag{2.22}$$

The above algorithm for transforming system (1.2)' into system (2.22), which leads to the full separable form (2.1) and the linear interval expressions (2.15), is referred to as Algorithm 1. Applying this algorithm to system (2.17a), we get 11 linear interval equations of the form (2.22).

Remark 1. An alternative algorithm for constructing enclosing functions of the type (2.22) is suggested in the next section. It is based on the transformation of the original system (1.2)' into a system of semiseparable form.

2.3. Alternative Algorithm

Consider the product

$$xy, \quad x \in X, \quad y \in Y, \tag{2.23}$$

where X and Y are intervals. If x_0 and y_0 are the centers of X and Y, respectively, then

$$xy = (x_0 + u)(y_0 + v) = x_0y_0 + y_0u + x_0v + uv = -x_0y_0 + y_0x + x_0y + uv. \tag{2.24}$$

When $x \in X$ and $y \in Y$, the centered variable $u \in R_x$ and the centered variable $v \in R_y$, where R_x and R_y are the radii of X and Y, respectively. Define $R = R_x R_y$; it follows from (2.24) that

$$xy \in -x_0y_0 + y_0x + x_0y + [-R, R], \quad x \in X, \ y \in Y. \tag{2.25}$$

Thus, the product xy is enclosed by an interval expression of separable form; i.e.,

$$xy \in \alpha x + \beta y + B_{xy}, \tag{2.26}$$

where $B_{xy} = -x_0y_0 + [-R, R]$ is an interval.

The above technique is readily extended to products of the form

$$x\left(\alpha_0 + \sum_j \alpha_j x_j\right). \tag{2.27}$$

The procedure involving formulas (2.23) to (2.27) for enclosing the product in (2.23) or (2.27) by corresponding linear interval expressions is referred to as Procedure 2.

As mentioned in Remark 1, an alternative approach to linear interval separability is possible. It is based on the semiseparable form introduced in subsection 2.1 and the linear interval enclosures (determined by Procedure 2) of the products of arguments involved in the semiseparable form. Thus, the following algorithm (Algorithm 2) for transforming the original nonlinear functions $\psi_i(x)$ from system (1.2)' into linear enclosures of the type (2.22) is suggested. First, the original system is transformed into a semiseparable form using transformations (2.2) to (2.7). The terms $f_{ij}(x_j)$ are then approximated by linear interval expressions (2.15), and the products (2.23) or (2.27) are enclosed by expressions of the form (2.26).

An appealing feature of the second algorithm is the fact that it converts semiseparable functions into linear interval enclosures of the type (2.22) without introducing new auxiliary variables and equations. Thus, we finally obtain the system

$$\sum_{j=1}^{n_2'} a_{ij}(X_j)x_j \cong B_i(X), \quad i = 1, 2, \ldots, r_4', \tag{2.28}$$

which has the same structure as system (2.22). However, the size of system (2.28) is smaller than that of (2.22) since, typically, both $n_2 > n_2'$ and $r_4 > r_4'$.

To illustrate the second algorithm, we consider the following example.

Example 2. Consider the minimization problem considered in Example 1. Applying the new transformation scheme to this problem, we need to introduce only two auxiliary variables, x_5 and x_6, to get the system

$$\begin{gathered} 1 - x_3 - x_2 + 2x_1(x_3 + x_4) = 0, \quad 2x_2x_3 - x_4 = 0, \quad x_3x_5 = 0, \\ x_4x_6 = 0, \quad x_1^2 + x_2^2 - 1 - x_5 = 0, \quad x_1^2 - x_2 - x_6 = 0, \end{gathered} \tag{2.29}$$

where the variables $x_1, x_2, \ldots$ belong to the intervals in (2.14b), (2.14c), etc. Now, enclosing the products in the first four equations and the nonlinear functions x_1^2 and x_2^2 in the last two equations of the system by appropriate approximations, the following system of linear interval equations is finally obtained from (2.29):

$$\begin{gathered} 1 - x_3 - x_4 + a_{11}x_1 + a_{13}x_3 + a_{14}x_4 = B_1, \quad a_{22}x_2 + a_{23}x_3 - x_4 = B_2, \quad a_{33}x_3 + a_{35}x_5 = B_3, \\ a_{44}x_4 + a_{46}x_6 = B_4, \quad a_{51}x_1 + a_{52}x_2 - 1 - x_5 = B_5, \quad a_{61}x_1 - x_2 - x_6 = B_6, \end{gathered} \tag{2.30}$$

where only the terms B_i on the right-hand side are intervals. [In the example considered, system (2.30) corresponds to the general system (2.28). To simplify notation, the dependence of a_{ij} and B_i on the current box X is not explicitly indicated in (2.30). It should however be stressed that system (2.30) must be solved at each iteration of the computation process (for each box X).]

Enclosing in the known manner the nonlinear functions in system (2.14a) by linear interval expressions, we get a system of the same kind as system (2.30). However, system (2.30) consists of only six equations, while the system corresponding to (2.14a) involves 11 equations. Thus, as illustrated by the present example, the application of Algronithm 2 leads to a smaller system of equations and therefore seems to be preferable as compared to Algorithm 1.

3. SOLVING SYSTEMS OF EQUATIONS

In this section, the last stage of the new approach is considered in the special case when the system (1.2)' employed to solve the corresponding optimization problem (1.1) is a system of n_1 nonlinear equations in n_1

variables. It will be pointed out at the end of the section that the present approach can be easily extended to systems of the general type consisting of both equalities and inequalities.

Let the system of semiseparable form corresponding to the original system (1.2)' be

$$f_i(x) = 0, \quad i = 1, 2, \ldots, n, \quad x \in X \subset \mathbb{R}^n, \tag{3.1}$$

where n stands for n_2 and X is any subbox of the initial box $X^{(0)}$. Applying Algorithm 2, we obtain the inclusions

$$f_i(x) \in \sum_{j=1}^{n} a_{ij}x_j + B_i, \quad x \in X, \quad i = 1, 2, \ldots, n,$$

or, in vector form,

$$f(x) \in -Ax + B, \quad x \in X, \tag{3.2a}$$

where the matrix A is chosen, for convenience, in the form

$$A = \{-a_{ij}\}. \tag{3.2b}$$

If y is a solution to system (3.1) in X, then $f(y) = 0$ and, by (3.2),

$$0 \in -Ay + B, \quad y \in X.$$

Now, the following results can be easily proved.

Theorem 1. *All solutions y to*

$$f(x) = 0, \tag{3.3}$$

contained in X are also contained in the solution set $S(X)$ of the system

$$Ax = b, \quad b \in B, \tag{3.4}$$

where b is any real vector contained in B.

Since B is an interval vector, the set $S(X)$ is a convex polyhedron. Indeed, (3.4) is in fact a system of n linear equalities and $2n$ linear inequalities.

Theorem 2. *All solutions y to (3.3) in X are also contained in the intersection*

$$P(X) = S(X) \cap X. \tag{3.5}$$

Since $S(X)$ and X are convex polyhedra, it is seen from (3.5) that $P(X)$ is also a convex polyhedron.

Corollary 1. If $P(X)$ is empty, i.e., if

$$S(X) \cap X = \varnothing,$$

then system (3.3) has no solution in X.

Let $H(P, X)$ denote the interval hull of $S(X)$, that is, the smallest interval vector (box) containing $P(X)$. By applying Theorem 2 and Corollary 1, an iterative method for finding all real solutions to (3.3) in an initial box $X^{(0)}$ could be designed. Such an approach would, however, require $2n$ linear programming problems to be solved at each iteration to determine the corresponding hull $H(P, X)$. Unless implemented by parallel computation, this would be a rather time-consuming iteration. Therefore, a simpler method is suggested here. It is based on the determination of the interval hull $H(S, X)$ of $S(X)$, Theorem 1 and the following Corollary 2.

Corollary 2. If

$$H(S, X) \cap X = \varnothing,$$

then system (3.3) has no solution in X.

It follows from (3.4) that if matrix A is not singular, then

$$H(S, X) = A^{-1}B.$$

Let $C = A^{-1}$ and $Y = H(S, X)$. It is seen from (3.4) that the components $Y_i = [\underline{y}_i, \bar{y}_i]$ of Y are given by the formulas

$$\underline{Y}_i = \sum_{j=1}^{n} \underline{y}_{ij}, \tag{3.6}$$

where

$$\underline{y}_{ij} = \begin{cases} c_{ij}\underline{b}_j, & c_{ij} \geq 0, \\ c_{ij}\bar{b}_j, & c_{ij} < 0, \end{cases} \tag{3.7}$$

$$\bar{Y}_i = \sum_{j=1}^{n} \bar{y}_{ij} \tag{3.8}$$

with

$$\bar{y}_{ij} = \begin{cases} c_{ij}\bar{b}_j, & c_{ij} \geq 0, \\ c_{ij}\underline{b}_j, & c_{ij} < 0. \end{cases} \tag{3.9}$$

In the case when A is singular in the current box X, X is split along its widest side into two boxes, and each new box is processed separately by using Procedures 1 and 2.

Now, a new interval method for locating all real solutions of the system

$$f(x) = 0 \tag{3.10a}$$

contained in $X^{(0)}$, i.e., when

$$x \in X^{(0)} \tag{3.10b}$$

is suggested. It is based on the following procedure.

Procedure 3. Let $X^{(k)}$ be a current box. Using Procedures 1 and 2, determine $C^{(k)}$ and $B^{(k)}$ corresponding to $X^{(k)}$. By formulas (3.6) to (3.9), compute $Y^{(k)}$. The iterative procedure is then defined as follows:

$$X^{(k+1)} = Y^{(k)} \cap X^{(k)}, \quad k \geq 0. \tag{3.11}$$

The procedure may result in three outcomes.

A. The sequence $X^{(k+1)}$ converges to a solution $x^{(s)}$ as k increases. Actually, the iterations are stopped whenever the width of $X^{(k+1)}$ becomes smaller than a constant ε_1 (which prescribes the accuracy with respect to x). Now $x^{(s)}$ is approximated by the center x^c of $X^{(k+1)}$ and x^c is substituted in (3.10a). If

$$tol = \max[|f_i(x^c)|, i = 1, 2, \ldots, n] > \varepsilon_2$$

(ε_2 defines the accuracy of x^c with respect to the system of equations), then the iterations are resumed; otherwise, x^c is accepted as a solution to (54).

B. At some k,

$$Y^{(k)} \cap X^{(k)} = \varnothing.$$

It follows from Corollary 2 that system (3.10a) has no solution in $X^{(k)}$. In this case, $X^{(k)}$ is discarded from further consideration.

C. The sequence $X^{(k+1)}$ converges to a fixed interval (box) X^*. In practice, the procedure is stopped whenever the reduction in the volume of the current box $X^{(k+1)}$, as compared to that of the preceding box $X^{(k)}$, is smaller than a constant ε_3. In this case, $X^{(k+1)}$ is split along its widest side into two (left and right) boxes, X^L and X^R. The right box is stored in a list L for further processing. The left box is renamed $X^{(0)}$, and the iterative procedure (3.11) is resumed.

Remark 2. For large-scale problems, A is a rather sparse matrix, and the inversion of A is then, obviously, numerically inefficient. In this case, the vector Y should be determined in an equivalent componentwise manner, which leads to finding successively the solution of n systems

$$Ac^{(i)} = e^{(i)}, \tag{3.12}$$

where $e^{(i)}$ is the ith column of the $n \times n$ identity matrix. Sparse-matrix methods should be employed for solving system (3.12) in this case.

Remark 3. It should be pointed out that the approach suggested in this paper can be used to construct methods designed to solve (in the sense of [2]) systems of nonlinear inequalities or mixed systems consisting of both equalities and inequalities. Indeed, solving such a system reduces to solving simultaneously a system of linear interval equalities

$$A_1 x = B_1 \tag{3.13a}$$

and a system of linear interval inequalities

$$A_2 x \le B_2, \tag{3.13b}$$

where only B_1 and B_2 are interval vectors.

Several methods for tackling problem (3.13) have been suggested in [2] for the general case when A_1 and A_2 are interval matrices. The fact that A_1 and A_2 are now real matrices can be advantageously exploited to simplify the known methods from [2] or to design new methods for solving (3.13).

4. NUMERICAL EXAMPLES

The numerical performance of the method suggested in Section 3 was tested on several systems of equations whose size n ranged from 2 to 20. It is illustrated here by three examples, solved on a Pentium 120-MHz computer.

Example 3. In this example, we consider the following system of ten nonlinear equations of separable form

$$f(x) = \varphi(x) - Hx - s = 0 \tag{4.1a}$$

with

$$\varphi_i(x_i) = 2.5x_i^3 - 10.5x_i^2 + 11.8x_i, \quad i = 1, \dots, 10, \quad H = \{h_{ij}\} \text{ with } h_{ij} = -1, \quad s = (-1, \dots, -10). \tag{4.1b}$$

The initial box $X^{(0)}$ is defined by

$$x_i \in [-1, 4], \quad i = 1, 2, \dots, n. \tag{4.2}$$

The value of ε_1 was set equal to 10^{-4}. We seek all real solutions to system (4.1) contained in the initial box (4.2).

Two interval methods were applied in [8] to solve the problem considered. The first method, here referred to as M1, is based on the use of interval derivatives, while the second method, M2, employs interval slopes (see [9, 10]). The present method, denoted by M3, was also used to solve the problem considered and find $\varepsilon_1 = 10^{-4}$ all of the nine solutions contained in $X^{(0)}$ up to the same 1. Data concerning the numerical efficiency of the three methods are shown in Table 1, where N_i stands for the number of iterations required to globally solve system (4.1), (4.2); t is the execution time (in seconds); and n_m is the maximum number of boxes stored during the computation in the list L:

Method	M1	M2	M3
N_i	524143	116522	146
t, s	482	121	0.503
n_m	525	93	3

These data reveal that the present method is vastly superior to the other two (rather sophisticated) interval methods as regards computing time. On account of its fast convergence rate, the new method also has improved characteristics in terms of memory requirements. It should also be stressed that no cluster effect [11] has been observed in solving the present example by M3. In contrast, among the two previous methods, even the better method M2 generated 24 solution boxes instead of 9 as in method M3.

Example 4. In this example, we consider optimization problem (2.8). To solve it globally, we invoke Algorithm 2; i.e., method M3 is applied to semiseparable system (2.29) to find all of its solutions contained in the initial box $X^{(0)}$ with components

$$X_1 = [-1, 1], \quad X_2 = [0, 1], \quad X_3 = [0, 1], \quad X_4 = [0, 1], \quad X_5 = [-1, 0], \quad X_6 = [-1, 0]. \tag{4.3}$$

The value of ε_1 was set equal to 10^{-5}. The following unique solution has thus been found:

$$x^* = (-0.7861, 0.6180, 0.1739, 0.2149, 0, 0), \tag{4.4}$$

where each component x_i^* of x^* recorded to four decimal places is the midpoint of a corresponding interval X_i^*. The global minimum φ_0^* in minimization problem (2.8) is approximated by x_1^* and is guaranteed to be enclosed by the interval X_1^*; i.e.,

$$\varphi_0^* \in [-0.7861513783, -0.7861513769].$$

System (2.29), (4.3) has also been solved by Krawczyk's method [2], denoted by M4 (in fact, a more efficient componentwise version of the method was used). Data on the number of iterations and computer time required by methods M3 and M4 are presented here.

Method	M4	M3
N_i	993	20
t, s	0.914	0.018

Remark 4. The above example has also been solved by Algorithm 1, i.e., by globally solving the fully separable system (2.14). Applying method M3 to (2.14), one finds, once again, a unique solution x^* (having 11 entries), whose first six components are the same as in (4.4). It is, however, worth mentioning that the number N_i' of iterations required to locate x^* is now larger than N_i by approximately an order of magnitude ($N_i' = 193$). The computing time corresponding to N_i' is $t' = 0.151$ s, which confirms the superiority of Algorithm 2 over Algorithm 1.

Example 5. We consider the following optimization problem (see [2, p. 177]): minimize

$$\varphi_0(x) = x_1^6 - 6.3x_1^4 + 12x_1^2 + 6x_1x_2 + 6x_2$$

subject to $\varphi_1(x) = 1 - 16x_1^2 - 25x_2^2 \le 0$, $\varphi_2(x) = 13x_1^3 - 145x_1 + 85x_2 - 400 \le 0$, $\varphi_3(x) = x_1x_2 - 400 \le 0$.

Application of the John conditions leads to the following system of six nonlinear equations:

$$x_3(6x_1^5 - 25\cdot 2x_1^3 + 24x_1 + 6x_2) - 32x_1x_4 + x_5(39x_1^2 - 145) + x_2x_6 = 0,$$

$$x_3(6x_1 + 12x_2) - 50x_2x_4 + 85x_5 + x_1x_6 = 0, \quad x_4(16x_1^2 + 25x_2^2 - 1) = 0, \tag{4.5a}$$

$$x_5(13x_1^3 - 145x_1 + 85x_2 - 400) = 0, \quad x_6(x_1x_2 - 4) = 0, \quad x_3 + x_4 + x_5 + x_6 - 1 = 0,$$

where the variables x_3, x_4, x_5, and x_6 correspond to the normalized Lagrange multipliers u_0, u_1, u_2, and u_3, respectively.

In this example, the problem is to globally solve system (4.5a) in the following initial region $X^{(0)}$:

$$X_1^{(0)} = X_2^{(0)} = [-2, 4], \quad X_i^{(0)} = [0, 1], \quad i = 3, \ldots, 6. \tag{4.5b}$$

Problem (4.5) was solved by methods M3 and M4. It has 9 solutions (recorded in the same way as in Example 4):

$x^{(1)} = (-1.7475, 0.8738, 1.000, 0.0000, 0.0000, 0.0000)$,
$x^{(2)} = (-1.0705, 0.5353, 1.000, 0.0000, 0.0000, 0.0000)$,
$x^{(3)} = (-0.2398, -0.05648, 0.5716, 0.4284, 0.000, 0.0000)$,
$x^{(4)} = (-0.06604, -0.1929, 0.8341, 0.1659, 0.000, 0.0000)$,
$x^{(5)} = (-0.2398, 0.05648, 0.5716, 0.4284, 0.0000, 0.0000)$,
$x^{(6)} = (-0.0000, 0.0000, 1.000, 0.0000, 0.0000, 0.0000)$,
$x^{(7)} = (-0.06604, 0.1929, 0.8341, 0.1659, 0.0000, 0.0000)$,
$x^{(8)} = (1.0705, -0.5353, 1.0000, 0.0000, 0.0000, 0.0000)$,
$x^{(9)} = (1.7475, -0.8738, 1.0000, 0.0000, 0.0000, 0.0000)$.

Data on N_i and t related to M4 and M3 are presented

Method	M4	M3
N_i	33089	3233
t, s	88.381	9.816

Once again, the new method is much faster than M4.

5. CONCLUSIONS

In this paper, the general nonlinear programming problem (or its variants) is addressed. A new interval approach to the global solution of the problem considered has been suggested. It is based on the following underlying ideas. The nonlinear system (1.2)' (used to solve the original optimization problem) is first transformed into a larger system of semiseparable form. Each term of the latter system is either a nonlinear function $f_{ij}(x_j)$ of a single variable or a product $x_k x_l$ of two variables. The semiseparable system is then enclosed by a corresponding linear interval system (2.28), using enclosures (2.15) and (2.26) for $f_{ij}(x_j)$ and $x_k x_l$, respectively. System (2.28), involving in the general case both equalities and inequalities, is then solved repeatedly (at each iteration) by an appropriate method.

An appealing feature of the new approach is the fact that only the right-hand side of system (2.28) is an interval vector, while the previous interval methods lead to linear interval systems where only the right-hand side is a real vector. The former type of system is much easier to solve, and, therefore, the new approach is expected to improve numerical efficiency. Another advantage is the fact that it can be directly applied to problems involving nondifferentiable functions in the objective function and/or constraints while the known interval first-order methods require a preliminary transformation of the initial problem into an equivalent problem of larger size, so that the new objective function and constraints have the desired smoothness [2, Ch. 14].

In the special case when (1.2)' is a system of n nonlinear equations in n variables, the novel approach has been applied in Section 3 to design a new interval method for solving such systems. The numerical results obtained so far and illustrated by Examples 3 to 5 show that the new method is considerably superior to other known interval methods for problems of larger size n and wider interval region $X^{(0)}$ in the case when equations (2.17) are not difficult to solve.

The novel approach to global optimization has been illustrated by Examples 4 and 5 in the case when the John conditions are used to obtain a corresponding nonlinear system of equations. The system has been solved by the method developed in Section 3. It should however be borne in mind that many alternative solution schemes (other than that of the John conditions) are possible. The application of the new interval approach to these alternative schemes and the development of corresponding interval methods for global optimization seems to be a promising area for future research.

ACKNOWLEDGMENTS

This work was supported by the Bulgarian Research Foundation under contract no. TN-554/97.

The author is grateful to one of the referees for his valuable comments and suggestions.

REFERENCES

1. Ratscek, H. and Rokne, J., *New Computer Methods for Global Optimization*, Chichester: Horwood, 1988.
2. Hansen, E., *Global Optimization Using Interval Analysis*, New York: Marcel Dekker, 1992.
3. Eriksson, J. and Lindstrom, P., A Parallel Interval Method Implementation for Global Optimization Using Dynamic Load Balancing, *Reliable Comput.*, 1995, vol. 1, no. 6, pp. 77–91.
4. Kolev, L., *Interval Methods for Circuit Analysis*, Singapore: World Scientific, 1993.
5. Zuhe, S. and Wolfe, M., On Interval Enclosures Using Slope Arithmetic, *Appl. Math. Comput.*, 1990, vol. 39, pp. 89–105.
6. Kolmogorov, A.N., On the Representation of Continuous Functions of Many Variables by Superposition of Continuous Functions of One Variable and Addition, *Dokl. Akad. Nauk SSSR*, 1957, vol. 14, pp. 953–956. (English transl.: in *Am. Math. Biophys.*, 1963, vol. 5, pp. 55–59).
7. Yamamura, K., An Algorithm for Representing Functions of Many Variables by Superpositions of Functions of One Variable and Addition, *IEEE Trans. Circuits Systems I*, 1996, vol. 43, no. 3, pp. 338–340.
8. Mladenov, V. and Kolev, L., Interval Methods for Solving Cellular Neural Networks (CNNs) Equations, *Third Int. Conf. on Electronics, Circuits and Systems (ICECS'96)*, Rodos (Greece), October 13–16, 1996.
9. Kolev, L. and Mladenov, V., Use of Interval Slopes in Implementing an Interval Method for Global Nonlinear DC Circuits Analysis, *Int. J. Circuit Theory Appl.*, 1997, vol. 25, pp. 37–42.
10. Kolev, L., Use of Interval Slopes for the Irrational Part of Factorable Functions, *Reliable Comput.*, 1997, vol. 3, pp. 83–93.
11. Morgan, A.P. and Shapiro, V., Box Bisection for Solving Second-Degree Systems and the Problem of Clustering, *ACM Trans. Math. Software*, 1987, vol. 13, pp. 152–167.