Journal of Global Optimization 9: 23–40, 1996.

Rigorous Convex Underestimators for General Twice–Differentiable Problems

Claire S. Adjiman and Christodoulos A. Floudas *
Department of Chemical Engineering, Princeton University, Princeton, N.J. 08544-5263

Abstract. In order to generate valid convex lower bounding problems for nonconvex twice–differentiable optimization problems, a method that is based on second–order information of general twice–differentiable functions is presented. Using interval Hessian matrices, valid lower bounds on the eigenvalues of such functions are obtained and used in constructing convex underestimators. By solving several nonlinear example problems, it is shown that the lower bounds are sufficiently tight to ensure satisfactory convergence of the αBB, a branch and bound algorithm which relies on this underestimation procedure [3].

Key words: convex underestimators; twice–differentiable; interval anlysis; eigenvalues

1. Introduction

The mathematical description of many physical phenomena, such as phase equilibrium, or of chemical processes generally requires the introduction of nonconvex functions. As the number of local solutions to a nonconvex optimization problem cannot be predicted *a priori*, the identification of the global optimum solution of the majority of problems of practical importance poses a stimulating challenge. A brute–force approach such as complete enumeration of all solutions is not viable. Indeed, it has been shown that such problems are NP–hard [21]. Despite these difficulties, techniques have been developed that successfully address some classes of nonconvex optimization problems. Deterministic optimization methods usually rely on a bounding procedure which generates converging sequences of upper and lower bounds. It is supplemented by a branching scheme which allows the elimination of certain regions of the solution space through successive refinements. Complete exploration of the space is thereby avoided, and the global solution can be attained in reasonable computational times. Numerous alternatives are available both for the branching and the bounding steps. For the latter, the tools of interval analysis may be used [9], [18], or a valid convex underestimating problem can be constructed by exploiting special mathematical features of the original problem. Thus, if the nonconvexities arise from bilinearities only, the convex

* Author to whom all correspondence should be addressed

envelope described in [2] provides the desired underestimator. For concave functions, linearization generates the tightest lower bound. Convex underestimators for several special classes of functions, such as biconvex, fractional, trilinear and signomial terms have also been developed [7],[8], [16], [22], [23]. In the case of an arbitrary nonconvex function, the use of a convex underestimator based on second–order information was suggested in [3]. It was incorporated in a branch–and–bound algorithm, the αBB, which is based on a difference of convex functions transformation and is applicable to twice–differentiable continuous optimization problems. However, the rigorous construction of the lower bounding problem was only possible in cases where explicit analytical expressions could be derived for the eigenvalues or for bounds on the eigenvalues of the functions present in the problem. In this paper, a new technique is discussed, which provides theoretical guarantees of validity for the construction of quadratic underestimators for general twice–differentiable functions.

2. Rigorous Convex Underestimators

2.1. The αBB algorithm

The αBB algorithm [3] is designed to handle problems of the following form :

$$\begin{aligned} \min_{\mathbf{x}} \ & f(\mathbf{x}) \\ s.t. \ & \mathbf{g}(\mathbf{x}) \leq 0 \\ & \mathbf{h}(\mathbf{x}) = 0 \\ & \mathbf{x} \in X \subseteq \Re^n \end{aligned} \tag{1}$$

where f, $\mathbf{g}$ and $\mathbf{h}$ belong to C^2, the set of twice-differentiable functions, and $\mathbf{x}$ is a vector of size n.

The algorithm is based on the construction of a branch–and–bound tree and a difference of convex functions transformation for the lower bounding problem. Although many alternatives are available in selecting a branching strategy, all of the schemes advocated for this algorithm correspond to a partition of the solution space into rectangular subdomains. More specific decisions such as which variable(s) to branch on or whether to apply bisection or a higher order section affect the convergence rate of the algorithm. However, these considerations do not modify in any way the theoretical guarantees of global optimality of the solution.

Similarly, the upper bounding step does not present any major difficulties. An evaluation of the optimization problem at a feasible point,

or a local solution obtained through a nonlinear optimization package such as MINOS 5.4 provide the required value.

The lower bounding step, on the other hand, must be very carefully designed so that a *valid* lower bound on the problem is obtained at all nodes of the tree. For this purpose, the original problem must be transformed into an underestimating convex problem, which can then be solved to global optimality. In special instances where the functions possess known mathematical properties, such as bilinearity or concavity for example, efficient convex lower bounding can be performed using the techniques referred to in the introduction. For general nonconvex functions, it is not possible to resort to a simple scheme such as linearization. Moreover, the only condition imposed on the participating functions is second–order differentiability, so that the underestimating procedure must be applicable to any kind of nonconvexity. As noted in [15], a twice–differentiable function $f(\mathbf{x})$ defined over a region X can be underestimated by the function $\mathcal{L}(\mathbf{x})$:

$$\mathcal{L}(\mathbf{x}) = f(\mathbf{x}) + \alpha \sum_{i=1}^{n} (x_i^L - x_i)(x_i^U - x_i) \tag{2}$$

where α is a positive scalar.

Furthermore, it was showed that this underestimator is convex if and only if the following condition is imposed on α :

$$\alpha \geq \max\{0, -\frac{1}{2} \min_{k, \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U} \lambda_k(\mathbf{x})\} \tag{3}$$

where the $\lambda_k(\mathbf{x})$'s are the eigenvalues of the function $f(\mathbf{x})$.

The α parameter, which governs the convexity of the underestimator, therefore depends on the second–order characteristics of the original function. α may thus be viewed as a measure of the degree of nonconvexity of the function $f(\mathbf{x})$: if $f(\mathbf{x})$ is convex over X, all its eigenvalues are positive and α is therefore zero. α increases as the eigenvalues become more and more negative or, equivalently, as $f(\mathbf{x})$ becomes more and more nonconvex.

In certain cases, an analytical expression may be obtained for the minimum eigenvalue of the function $f(\mathbf{x})$ over X [13], [14]. However, for an arbitrary $f(\mathbf{x})$, the identification of the smallest eigenvalue requires the solution of a nonconvex optimization problem of the form given in (4).

$$\lambda_{min} = \begin{cases} \min\limits_{\lambda, \mathbf{x}} \ \lambda \\ s.t. \ \ \mathcal{P}_f(\mathbf{x}, \lambda) = 0 \\ \qquad \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U \end{cases} \tag{4}$$

where $\mathcal{P}_f(\mathbf{x}, \lambda)$ is the characteristic polynomial of $f(\mathbf{x})$.

Since problem (4) cannot be solved to global optimality using currently available techniques, an *exact value* for λ_{min} cannot be obtained. An alternative approach to calculating α is therefore required. According to equation (3), a valid *lower bound* on λ_{min} suffices for the generation of a convex underestimator. This allows a relaxation of problem (4) to a more tractable form, through interval analysis. It is first noted that the coefficients of the characteristic polynomial $\mathcal{P}_f(\mathbf{x}, \lambda)$ depend on the x variables so that $\mathcal{P}_f(\mathbf{x}, \lambda)$ represents a family of polynomials with real coefficients. If the natural interval extensions of the coefficients over the domain X are evaluated, the polynomial family $\mathcal{P}_f(\mathbf{x}, \lambda)$ can then be transformed into a larger family $\mathcal{P}_f(X, \lambda)$ where all the coefficients of $\mathcal{P}_f(X, \lambda)$ are intervals. Because the initial family is included into the interval family, the minimum root of the latter provides a valid lower bound on the smallest eigenvalue of $f(\mathbf{x})$.

The family $\mathcal{P}_f(X, \lambda)$ contains an infinite number of polynomials, and the identification of the spectrum of its roots appears as a challenging problem. The seminal work of Kharitonov [12] stands as a powerful starting point in tackling it, for it allows the determination of the stability characteristics of any interval polynomial family. A description of how Kharitonov's theorem can be incorporated into a method for the calculation of lower bound on λ_{min} is given in [1]. Considering that the derivation of the interval polynomial family from the Hessian matrix of $f(\mathbf{x})$ is a computationally expensive task, advantage can be taken of Kharitonov–like techniques for the computation of extreme eigenvalues of interval matrices which have been developed in recent years [10], [11]. By working in the matrix space, construction of the characteristic polynomial is avoided.

2.2. Interval Hessian Matrices

The Hessian matrix $H_f(\mathbf{x})$ of a twice–differentiable function $f(\mathbf{x})$ defined over X can be transformed into an interval matrix $H_{f,X}$. Just as is the case for the characteristic polynomials, the set of matrices described by $H_{f,X}$ contains those defined by $H_f(\mathbf{x})$.

$$H_f(\mathbf{x}) = \begin{pmatrix} a_{11}(\mathbf{x}) & \cdots & a_{1n}(\mathbf{x}) \\ & \vdots & \\ a_{n1}(\mathbf{x}) & \cdots & a_{nn}(\mathbf{x}) \end{pmatrix}$$

$$\subseteq \quad H_{f,X} = \begin{pmatrix} [a_{11}^L, a_{11}^U] & \cdots & [a_{1n}^L, a_{1n}^U] \\ & \vdots & \\ [a_{n1}^L, a_{n1}^U] & \cdots & [a_{nn}^L, a_{nn}^U] \end{pmatrix} \tag{5}$$

Hessian matrices possess interesting properties in that they are symmetric and their eigenvalues are real. Some valuable results that exploit these characteristics were presented in [10]. It was shown that the eigenvalues of all the matrices in the family $H_{f,X}$ are bounded by the extreme eigenvalues of a subset of these matrices. In order to determine a lower bound on the smallest eigenvalue λ_{min}, it suffices to compute the minimal eigenvalue of 2^{n-1} vertex matrices of $H_{f,X}$. Furthermore, a simple rule was developed, allowing the generation of the necessary matrices.

By definition, λ_{min} can be expressed as

$$\lambda_{min} = \min_{H \in H_{f,X}} \left(\min_{\|\mathbf{r}\|=1} \mathbf{r}^T H \mathbf{r} \right) \tag{6}$$

where $\mathbf{r}$ is a real n–dimensional vector, and H is a real matrix with elements $a_{11}, \cdots, a_{nn}$, and belongs to the interval matrix family $H_{f,X}$. Note that the set of all vectors $\mathbf{r}$ spans the surface of the unit sphere.

The challenges presented by problem (6) can be circumvented by devising simple rules that identify *a priori* the real matrix $H^* \in H_{f,X}$ which minimizes the quadratic form $\mathbf{r}^T H \mathbf{r}$ for all vectors $\mathbf{r}$ satisfying $\| \mathbf{r} \| = 1$.

Equation (6) can be rewritten as

$$\lambda_{min} = \min_{H \in H_{f,X}} \left(\min_{\|\mathbf{r}\|=1} \sum_{i=1}^{n} a_{ii} r_i^2 + \sum_{\substack{1 \le i,j \le n \\ i \ne j}} a_{ij} r_i r_j \right) \tag{7}$$

$$= \min_{H \in H_{f,X}} \left(\left(\min_{\|\mathbf{r}\|=1} \sum_{i=1}^{n} a_{ii} r_i^2 \right) + \min_{\|\mathbf{r}\|=1} \left(\sum_{\substack{1 \le i,j \le n \\ i \ne j}} a_{ij} r_i r_j \right) \right) \tag{8}$$

$$= \min_{\|\mathbf{r}\|=1} \left(\sum_{i=1}^{n} a_{ii}^L r_i^2 \right) + \min_{H \in H_{f,X}} \left(\min_{\|\mathbf{r}\|=1} \sum_{\substack{1 \le i,j \le n \\ i \ne j}} a_{ij} r_i r_j \right) \tag{9}$$

Thus, the first term has been reduced to a minimization over the surface of the unit sphere and the diagonal elements of the desired matrix H^* have been identified as the lower bounds of the diagonal elements of the interval matrix $H_{f,X}$. The second term could be simplified in a similar manner if the sign of the product $r_i r_j$ were known. Thus, if $r_i r_j < 0$, the (ij) and (ji) elements of H^* should be equal to a_{ij}^U, while if $r_i r_j \ge 0$, they should be equal to a_{ij}^L. This reduction can be achieved by noting that the n–dimensional space can be divided into 2^n orthants inside of which the signs of the elements of vector $\mathbf{r}$ remain unchanged. Within each of these orthants, the sign of every

cross product $r_i r_j$, $(i,j) \in \{1,\cdots,n\}^2$, is also conserved. Moreover, due to symmetry about the origin of the coordinate system, there are only 2^{n-1} possible combinations of the signs of these cross products, so that 2^{n-1} orthants are sufficient to describe all the sign preserving domains of the $r_i r_j$ products. Let R_k denote the set of vectors $\mathbf{r}$ which belong to the kth orthant and for which $\| \mathbf{r} \| = 1$. Then, for each orthant k, the real matrix $H_k^* \in H_{f,X}$ which minimizes $\mathbf{r}^T H \mathbf{r}$ can be constructed according to the following rule :

$$H_k^* = [a_{ij}^k] \qquad \forall k \in \{1,\ldots,2^{n-1}\} \tag{10}$$

where

$$a_{ij}^k = \begin{cases} a_{ii}^L & \text{if } i = j \\ a_{ij}^L & \text{if } x_i x_j \geq 0, i \neq j \\ a_{ij}^U & \text{if } x_i x_j \leq 0, i \neq j \end{cases} \tag{11}$$

Therefore, the minimum eigenvalue of the family of matrices $H_{f,X}$ is given by

$$\begin{aligned} \lambda_{min} &= \min_{k=1,\cdots,2^{n-1}} \left(\min_{\mathbf{r} \in R_k} \mathbf{r}^T H_k^* \mathbf{r} \right) \\ &= \min_{k=1,\cdots,2^{n-1}} \lambda_{k,min} \end{aligned} \tag{12}$$

where $\lambda_{k,min}$ is the minimum eigenvalue of matrix H_k^* as defined in equation (11).

Note that all 2^{n-1} H^* matrices are vertex matrices, that is their elements are endpoints of the intervals that appear in matrix $H_{f,X}$. Furthermore, these real matrices are symmetric ($r_i r_j = r_j r_i$), and their eigenvalues are therefore real.

Standard methods can be used in order to calculate the minimal eigenvalue of the each vertex matrix. For instance, the rational QR method with Newton corrections [19] can be applied on the matrix, after reducing it to a symmetric tridiagonal matrix.

Two simple examples are provided here in order to illustrate the procedure.

Example 1

The following third order polynomial in two variables is studied :

$f(x_1,x_2) = x_1^3 - x_1 x_2^2$ with $(x_1,x_2) \in X = [0,1]^2$.

Its Hessian matrix is

$$H_f(x_1,x_2) = \begin{pmatrix} 6x_1 & -2x_2 \\ -2x_2 & -2x_1 \end{pmatrix} \subseteq H_{f,X} = \begin{pmatrix} [0,6] & [-2,0] \\ [-2,0] & [-2,0] \end{pmatrix}$$

The four sign preserving regions for x_1 and x_2 are :

x_1	x_2	x_1x_2
+	+	+
+	−	−
−	+	−
−	−	+

Only the first two quadrants ($x_1, x_2 \geq 0$ and $x_1 \geq 0, x_2 \leq 0$) need to be taken into consideration. The two vertex matrices are then

$$H_1^* = \begin{pmatrix} 0 & -2 \\ -2 & -2 \end{pmatrix} \quad \text{and} \quad H_2^* = \begin{pmatrix} 0 & 0 \\ 0 & -2 \end{pmatrix}$$

and their smallest eigenvalues are $\lambda_{1,min} = -1 - \sqrt{5}$ and $\lambda_{2,min} = -2$. Hence $\lambda_{min} \geq -3.24$. The exact value of λ_{min} can also be calculated for this example (using GAMS [4], for example). It was found that $\lambda_{min} \approx -2.5$.

Example 2

The second example presented here is a function of two variables involving trigonometric terms :

$f(x_1, x_2) = x_1 \cos x_2 + x_2 \sin x_1$, where $(x_1, x_2) \in X = [0, 1]^2$.

The corresponding Hessian matrix is

$$H_f(x_1, x_2) = \begin{pmatrix} -x_2 \sin x_1 & -\sin x_2 + \cos x_1 \\ -\sin x_2 + \cos x_1 & -x_1 \cos x_2 \end{pmatrix}$$

$$\subseteq \quad H_{f,X} = \begin{pmatrix} [-0.842,0] & [-0.302,1] \\ [-0.302,1] & [-1,0] \end{pmatrix} \tag{13}$$

The following two vertex matrices are needed in order to evaluate the smallest eigenvalue :

$$H_1^* = \begin{pmatrix} -0.842 & -0.302 \\ -0.302 & -1 \end{pmatrix} \quad \text{and} \quad H_2^* = \begin{pmatrix} -0.842 & 1 \\ 1 & -1 \end{pmatrix}$$

The two minimum eigenvalues are $\lambda_{1,min} = -1.234$ and $\lambda_{2,min} = -1.925$. This yields a lower bound of -1.925 on the smallest eigenvalue of $H_f(x_1, x_2)$ over X. The exact value is -1.27.

2.3. Analysis of the Interval Hessian Method

The computational effort required by the interval Hessian matrix technique depends exponentially on the number of variables that participate in the original function. An approach based on the decomposition of the functions under consideration into the sum of several terms

seems especially valuable in this context. In addition, it is amenable to the exploitation of mathematical properties (concavity, bilinearity) that may be identified for some of the terms. Even for highly nonlinear functions, it is likely that each individual term only involves a fraction of the independent variables. Thus the generation of one α per term in a function is expected to be much faster than that of one α for the entire function. When using the αBB algorithm, functions can therefore be decomposed as :

$$f(\mathbf{x}) = LT(\mathbf{x}) + CT(\mathbf{x}) + \sum_{i=1}^{bt} BT_i(\mathbf{x}) + \sum_{i=1}^{ut} UT_i(\mathbf{x}) + \sum_{i=1}^{nt} NT_i(\mathbf{x}) \quad (14)$$

where LT is linear term, CT is a convex term, the BT's are bilinear terms, the UT's are univariate concave terms and the NT's are general nonconvex terms. bt, ut and nt denote the numbers of bilinear, univariate concave and general nonconvex terms respectively.

The value calculated for λ_{min} corresponds to the exact minimum eigenvalue for the interval matrix family $H_{f,X}$. However, the fact that the elements of the Hessian matrix of an arbitrary function are not in general independent is overlooked when deriving $H_{f,X}$. This results in an overestimate of the family of matrices encompassed by $H_f(\mathbf{x})$ as $\mathbf{x}$ varies in X. λ_{min} is then a lower bound on the eigenvalue needed, whose quality depends on the functionality of the elements of $H_f(\mathbf{x})$, as well as on the tightness of the bounds on the variable vector $\mathbf{x}$. An update of the bounds on the variables, as well as the calculation of new α values at each iteration of the algorithm can therefore lead to significant speed improvements.

3. A Case Study

The α calculations for a highly nonlinear function are studied in order to test the performance of the interval Hessian matrix method.

The illustration presented here involves a single variable, so that attention can be focused on the graphical analysis of the proposed approach. As the solution space is partitioned into several subdomains, the change in α value is recorded, and the improvement in the quality of the lower bounding function is monitored. The piecewise convex underestimator of the overall function, generated by constructing one lower bounding function per subspace according to equation (2), is plotted for different domain sizes. The exact α value can be obtained by repeatedly solving the NLP described in (4) from different starting

points. Comparison of the values thus calculated with those provided by the interval Hessian method also yields valuable information.

The chosen example is that of the molecular conformation of pseudoethane, an ethane molecule in which all the hydrogen atoms have been replaced by C, N or O atoms. It is taken from [15], where the global minimum potential energy conformation of small molecules is studied. The Lennard–Jones potential is expressed in terms of a single dihedral angle.

The potential energy of the molecule is given by

$$
\begin{aligned}
f(t) = & \frac{588600}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t-\frac{2\pi}{3}) - \cos^2\theta\right)r_0^2\right)^6} \\
& - \frac{1079.1}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t-\frac{2\pi}{3}) - \cos^2\theta\right)r_0^2\right)^3} \\
& + \frac{600800}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t) - \cos^2\theta\right)r_0^2\right)^6} \\
& - \frac{1071.5}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t) - \cos^2\theta\right)r_0^2\right)^3} \\
& + \frac{481300}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t+\frac{2\pi}{3}) - \cos^2\theta\right)r_0^2\right)^6} \\
& - \frac{1064.6}{\left(3r_0^2 - 4\cos\theta r_0^2 - 2\left(\sin^2\theta\cos(t+\frac{2\pi}{3}) - \cos^2\theta\right)r_0^2\right)^3}
\end{aligned}
$$

where r_0 is the covalent bond length ($1.54\mathring{A}$),
θ is the covalent bond angle (109.5°),
t is the dihedral angle ($0 \leq t \leq 2\pi$).

Note that the contribution from 1–2 and 1–3 atom interactions have not been included in the above formulation since they are constant.

It can be seen from Figure 1 that this highly nonlinear function exhibits three minima in the range $[0, 2\pi]$. The two local minima occur at $t = 61.42^\circ$ and $t = 296.12^\circ$, where the potential energy takes on values of -0.7973 kcal/mol and -1.0399 kcal/mol respectively. The global minimum of -1.0711 kcal/mol is observed at $t = 183.45^\circ$.

Table I shows the exact and computed α's for the first four partitions of the solution space, while Table II summarizes the results for subsequent levels. The accuracy of the calculated α values is seen to improve rapidly as the size of the search domain is reduced, confirming the importance of supplying tight variables bounds. The numerical value of α is not sufficient to assess the performance of the method since the convergence of the algorithm is largely determined by the separation distance between the function and its underestimator, which depends both on α and the size of the solution space. Tables III and IV give the maximum separation distance for the exact and interval Hessian calculations. Although the former provide much tighter underestimators at the initial levels of the tree, the latter decreases at a faster rate. By plotting the piecewise convex lower bounding function at every level in

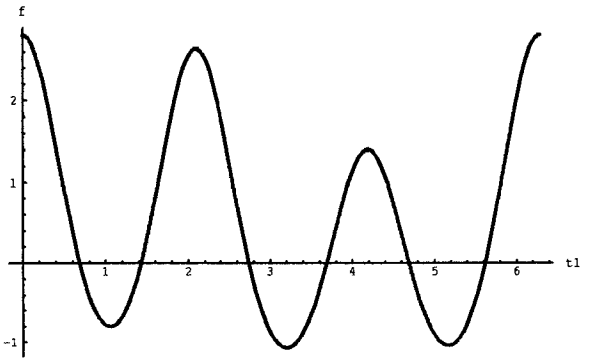

Figure 1. Potential energy of pseudoethane as a function of the dihedral angle.

the tree, the rapid improvement in the quality of the underestimator can be visualized (Figures 2 and 3).

When the αBB algorithm is used to identify the global optimum solution, convergence to within 10^{-6} is achieved in 21 iterations and 1.1 CPU seconds on an HP/730. Figure 4 shows the branch and bound tree as explored during the αBB run. Although the search is initially breadth–first, most nodes on level 5 are fathomed and the solution is quickly identified as the underestimator tightens. Figure 5 provides the branch and bound tree obtained by using the exact α value for each region. It can be seen that the search also starts as a breadth–first process, which then changes to follow depth–first pattern. The global solution is identified at level 6, one level higher than with the interval Hessian matrix method. For an unconstrained optimization problem in one variable such as this one, the exact eigenvalue at each level can be identified with the help of graphical analysis.

Due to the use of interval arithmetic in the Hessian matrix method, it is important to derive a mathematical expression for the second–order

derivatives which will reduce the occurence of interval over–estimation. In the case of pseudoethane, it is interesting to note that better estimates of the eigenvalue are generated if the potential function is not differentiated in the dihedral angle space but in the r space, where r is the distance between two atoms. Once the r–space Hessian has been obtained, it can be expressed in terms of the dihedral angle through a transformation of coordinates. At the first level, for example, a value of 620 is obtained for α, as compared with approximately 38,000 using differentiation in the dihedral angle space. The algorithm terminates after 15 iterations and 0.9 CPU seconds. In some instances where explicit expressions for the eigenvalues are not available, the user may therefore be able to provide expressions superior to those obtained through direct differentiation.

4. Summary of Computational Experience

The αBB algorithm may be used to solve a variety of problems, and this section presents preliminary results obtained for a selection of nonconvex problems with different mathematical structures. The computational results are summarized in table V. HP1 [6] is a Haverly pooling problem in which the nonconvexities arise exclusively in bilinear terms. Branin, Hartman and Goldstein [5] are highly nonlinear standard unconstrained test problems. The first involves a combination of polynomial and trigonometric terms, the second is the sum of two exponential terms and has two minima. The latter is an eighth degree polynomial in two variables. CNC [17] is a highly nonconvex constrained test problem. Finally, HEN [6] is a heat exchanger network optimization problem, while RN [20] is a reactor network configuration problem.

5. Conclusions

The αBB algorithm presented in [3] is a valuable tool for the global optimization of nonconvex twice–differentiable problems. The main difficulty encountered is the generation of convex underestimating problems, which requires the calculation the minimum eigenvalues of the functions involved, or of a valid lower bound on these eigenvalues. In this paper, a rigorous method for the generation of a lower bound on the eigenvalues of a function defined over a given interval has been described. This technique is characterized by the use of interval Hessian matrices and a mathematical complexity of the order of 2^{n-1} where n is the number of variables in the function being studied. Using this

procedure, a small highly nonlinear example has been extensively studied and the convex underestimators were seen to tighten significantly as the solution space is partitioned. As a result, convergence was achieved within a few levels of the branch and bound tree. The algorithm has also been used successfully to solve a number of highly nonconvex problems.

Acknowledgements

Financial support from the National Science Foundation, the Air Force Office of Scientific Research and the Binational Science Foundation is gratefully acknowledged.

References

1. C. A. Adjiman, I. P. Androulakis, C. D. Maranas and C. A. Floudas, A Global Optimisation Method, αBB, for Process Design, *Proc. of the Sixth European Symposium on Computer Aided Process Engineering*, To be published, May 26–29 1996.
2. F. A. Al-Khayyal and J.E. Falk, Jointly Constrained Biconvex Programming, *Maths Ops Res.*, 8:273–286, 1983.
3. I. P. Androulakis, C. D. Maranas and C. A. Floudas, αBB : A Global Optimization Method for General Constrained Nonconvex Problems, *J. Global. Opt.*, 7:337–363, 1995.
4. A. Brooke, D. Kendrick and A. Meeraus, *GAMS – Release 2.25. A User's Guide*, boyd & fraser publishing company,1992.
5. L. Dixon and G. P. Szegö, Towards Global Optimization, *Proc. of a Workshop at the University of Cagliari, Italy*, 1990.
6. C. A. Floudas and P. M. Pardalos, *A Collection of Test Problems for Constrained Global Optimization Algorithms*, Springer-Verlag, 1990.
7. C. A Floudas and V. Visweswaran, A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs : I. Theory, *Comp. Chem. Engng.*, 14:1397–1417, 1990.
8. C. A Floudas and V. Visweswaran, A Primal-Relaxed Dual Global Optimization Approach, *J. Opt. Theory and App.*, 78:187–225, 1993.
9. E. Hansen, *Global Optimization Using Interval Analysis*, Marcel Dekkar, New York, 1992.
10. D. Hertz, The Extreme Eigenvalues and Stability of Real Symmetric Interval Matrices, *IEEE Transactions on Automatic Control*, 37:532–535, 1992.
11. C. V. Hollot and A. C. Bartlett, On the Eigenvalues of Interval Matrices, *Proc. 1987 Conf. Decision Contr.*, 794–799,1987.
12. V. L. Kharitonov, Asymptotic Stability of an Equilibrium Position of a Family of Systems of Linear Differential Equations, *Differential Equations*, 78:1483–1485, 1979.
13. C. D. Maranas and C. A. Floudas, A Global Optimization Approach for Lennard-Jones Microclusters, *J. Chem. Phys.*, 97:7667–7677,1992.
14. C. D. Maranas and C. A. Floudas, Global Optimization for Molecular Conformation Problems, *Annals of Operations Research*, 42:85–117, 1993.
15. C. D. Maranas and C. A. Floudas, Global Minimum Potential Energy Conformations of Small Molecules, *J. Global. Opt.*, 4:135–170, 1994.

16. C. D. Maranas and C. A. Floudas, Finding All Solutions of Nonlinearly Constrained Systems of Equations, *J. Global. Opt.*, 7:143–182, 1995
17. B. A. Murtagh and M. A. Saunders, *MINOS 5.0 User's Guide*, Systems Optimization Laboratory, Dept. of Operations Research, Stanford University, CA., 1988.
18. A. Neumaier, *Interval Methods for Systems of Equations*, Cambridge University Press, 1990.
19. C. Reinsch and J. H. Wilkinson, *Linear Algebra*, Handbook for Automatic Computation, Vol. 2, Springer–Verlag, 1971.
20. H. S. Ryoo and N. V. Sahinidis, Global Optimization of Nonconvex NLPs and MINLPs with Applications in Process Design, *Comp. Chem. Engng.*, 19:551–566, 1995.
21. S.A. Vavasis, *Nonlinear Optimization – Complexity Issues*, Oxford Science Publications, 1991.
22. V. Visweswaran and C. A. Floudas, New Formulations and Branching Strategies for the GOP Algorithm, *Global Optimization in Chemical Engineering*, I.E. Grossmann, Editor, Kluwer Academic Publishers, Chapter 3, 75–110, 1996.
23. V. Visweswaran and C. A. Floudas, Computational Results for an Efficient Implementation of the GOP Algorithm and its Variants, *Global Optimization in Chemical Engineering*, I.E. Grossmann, Editor, Kluwer Academic Publishers, Chapter 4, 111–154, 1996.

Table I. Exact and calculated α s for the pseudoethane potential energy function

Level	Search Domain	Exact α	Calculated α
1	$[0, 2\pi]$	10.7	$3.8\ 10^4$
2	$[0, \pi]$	10.7	$9.5\ 10^3$
2	$[\pi, 2\pi]$	10.7	$8.6\ 10^3$
3	$[0, \frac{\pi}{2}]$	10.7	944.2
3	$[\frac{\pi}{2}, \pi]$	10.4	298.9
3	$[\pi, \frac{3\pi}{2}]$	7.9	262.0
3	$[\frac{3\pi}{2}, 2\pi]$	10.7	869.2
4	$[0, \frac{\pi}{4}]$	10.7	56.8
4	$[\frac{\pi}{4}, \frac{\pi}{2}]$	0	93.5
4	$[\frac{\pi}{2}, \frac{3\pi}{4}]$	10.4	41.3
4	$[\frac{3\pi}{4}, \pi]$	5.5	87.0
4	$[\pi, \frac{5\pi}{4}]$	4.0	75.8
4	$[\frac{5\pi}{4}, \frac{3\pi}{2}]$	7.9	34.5
4	$[\frac{3\pi}{2}, \frac{7\pi}{4}]$	0	84.4
4	$[\frac{7\pi}{4}, 2\pi]$	10.7	56.7

Table II. Range for exact and calculated α s for the pseudoethane potential energy function

Level	Interval width	Range for exact α	Range for Calculated α
5	$\frac{\pi}{8}$	0 – 10.7	10.2 – 23.5
6	$\frac{\pi}{16}$	0 – 10.7	2.8 – 14.0
7	$\frac{\pi}{32}$	0 – 10.7	0.0 – 10.7
8	$\frac{\pi}{64}$	0 – 10.7	0.0 – 10.7

Table III. Maximum separation distance between function and underestimator. $\mathcal{L}^*$ denotes the exact lower bounding function while $\mathcal{L}$ corresponds to the Hessian matrix calculation

Level	Search Domain	$\max(f-\mathcal{L}^*)$	$\max(f-\mathcal{L})$
1	$[0, 2\pi]$	105.6	$3.8\ 10^5$
2	$[0, \pi]$	26.4	$2.3\ 10^4$
2	$[\pi, 2\pi]$	26.4	$2.1\ 10^4$
3	$[0, \frac{\pi}{2}]$	6.6	582.4
3	$[\frac{\pi}{2}, \pi]$	6.4	184.4
3	$[\pi, \frac{3\pi}{2}]$	4.9	161.6
3	$[\frac{3\pi}{2}, 2\pi]$	6.6	536.2
4	$[0, \frac{\pi}{4}]$	1.7	8.8
4	$[\frac{\pi}{4}, \frac{\pi}{2}]$	0	14.4
4	$[\frac{\pi}{2}, \frac{3\pi}{4}]$	1.6	6.4
4	$[\frac{3\pi}{4}, \pi]$	0.8	13.4
4	$[\pi, \frac{5\pi}{4}]$	0.6	11.7
4	$[\frac{5\pi}{4}, \frac{3\pi}{2}]$	1.2	5.3
4	$[\frac{3\pi}{2}, \frac{7\pi}{4}]$	0	13.0
4	$[\frac{7\pi}{4}, 2\pi]$	1.7	8.7

Table IV. Range of maximum separation distance between function and underestimator. $\mathcal{L}^*$ denotes the exact lower bounding function while $\mathcal{L}$ corresponds to the Hessian matrix calculation

Level	Interval width	Range of $\max(f-\mathcal{L}^*)$	Range of $\max(f-\mathcal{L})$
5	$\frac{\pi}{8}$	0 – 0.413	0.393 – 0.906
6	$\frac{\pi}{16}$	0 – 0.103	0.027 – 0.135
7	$\frac{\pi}{32}$	0 – 0.026	0 – 0.026
8	$\frac{\pi}{64}$	0 – 0.006	0 – 0.006

Table V. Summary of Computational Results

Problem Name	Number of Variables	Number of Constraints	Number of Iterations	CPU time (s)
HP1	9	6	17	2.1
Branin	2	0	44	2.0
Hartman	1	0	22	0.8
Goldstein	2	0	1092	181
CNC	5	6	587	258
HEN	8	6	286	677
RN	6	5	24	22

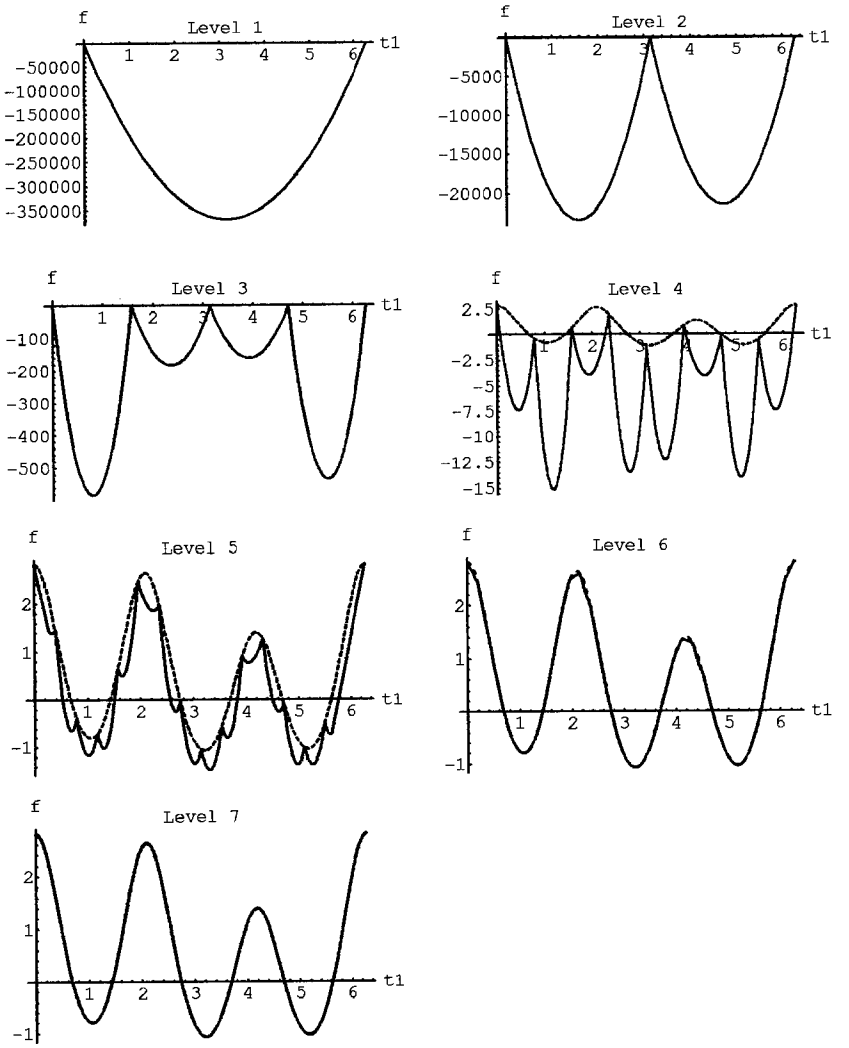

Figure 2. Function and underestimator at different levels of the branch-and-bound tree using calculated α values

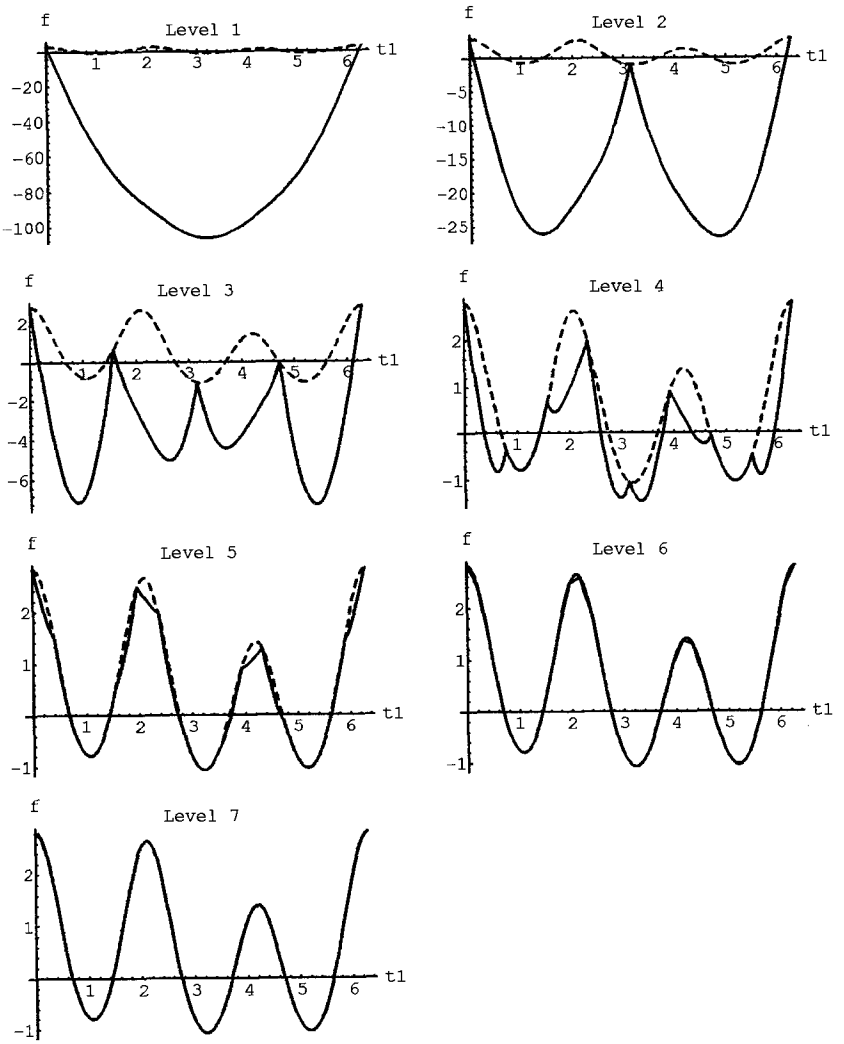

Figure 3. Function and underestimator at different levels of the branch–and–bound tree using exact α values

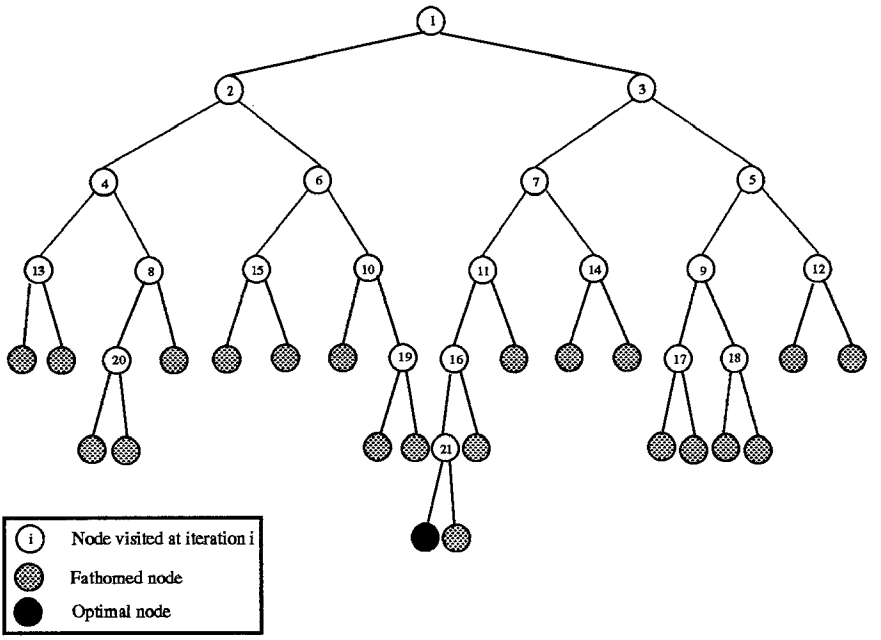

Figure 4. αBB branch and bound tree for pseudoethane example, using calculated α values

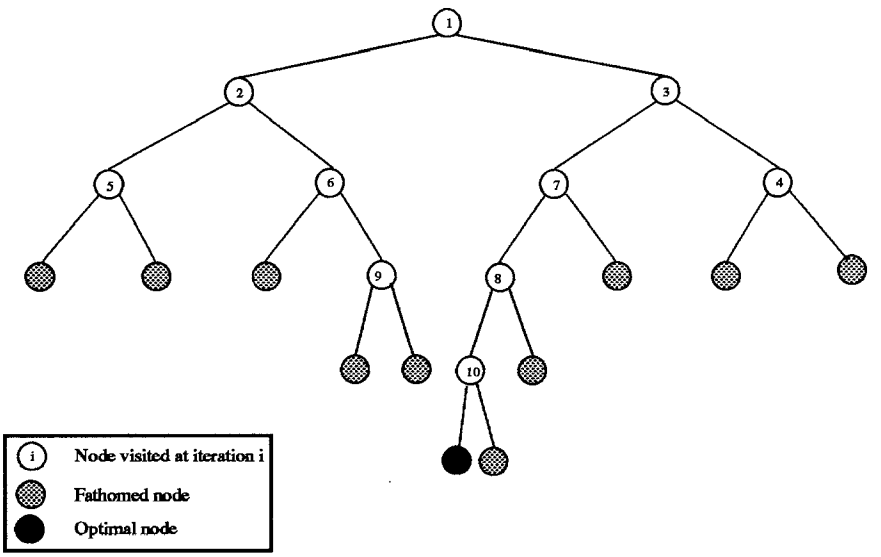

Figure 5. αBB branch and bound tree for pseudoethane example, using exact α values