Generalised Interval Methods in Optimal Design

Julius Zilinskas* and Ian David Lockhart Bogle†
Centre for Process Systems Engineering, Department of Chemical Engineering, University College London, Torrington Place, London WC1E 7JE, UK

Abstract
A global optimisation branch and bound algorithm with bounds evaluated using probabilistic generalised interval methods is presented in the present paper. The speed of the algorithm is evaluated by means of the number of objective function evaluations and time of the optimisation. The reliability of the algorithm is evaluated by means of the probability that the global minimum is found. The performance of the algorithm is evaluated and compared with algorithm based on standard interval methods. On the two case problems the optimisation algorithm is faster in general than the standard algorithm when a 96% success rate is required and can be as much as six times faster for 80%.

Keywords: global optimisation, interval methods, random interval arithmetic, multidimensional scaling

1. Introduction
In process engineering it is frequently necessary to solve global optimisation problems (Floudas, 1999, Xu et al., 2002, Byrne and Bogle, 1999). For some problems interval methods cannot produce acceptable bounds of functions. Such inefficiency is caused by the large ranges resulting from interval operations. Balanced random interval arithmetic has been proposed in (Zilinskas and Bogle, 2003) extending the ideas of random interval arithmetic (Alt and Lamotte, 2001) by using different probabilities of standard and inner interval operations. Test results seem promising for the construction of global optimisation algorithms based on the ideas of probabilistic generalised interval methods.

A global optimisation branch and bound algorithm with bounds evaluated using probabilistic generalised interval methods is presented in the present paper. The speed of the algorithm is evaluated by means of number of objective function evaluations and time of the optimisation. The reliability of the algorithm is evaluated by means of the probability that the global minimum is found. The influence of the predefined probabilities of standard and inner interval operations for balanced random interval arithmetic on the speed and reliability of the algorithm is experimentally investigated on a multidimensional scaling problem. The performance of the algorithm is evaluated and compared with an algorithm based on standard interval methods.

* On leave from Faculty of Informatics, Kaunas University of Technology, Lithuania
† Author to whom correspondence should be addressed: d.bogle@ucl.ac.uk
2. Interval Global Optimisation

Interval arithmetic (Moore, 1966) operates with real intervals \( x=[x_1, x_2]=\{x \in \mathbb{R} | x_1 \leq x \leq x_2 \} \), where \( x_1 \) and \( x_2 \) are real numbers. Upper and lower bounds of function values may be found using interval operations and used to solve global optimisation problems. A disadvantage of interval methods is the dependency problem: when a given variable occurs more than once in an interval computation, it is treated as a different variable in each occurrence and the estimated bounds of an objective function are not tight.

Alt and Lamotte (2001) proposed the idea of random interval arithmetic which is obtained by choosing standard or inner interval operations randomly with the same probability at each step of the computation. They showed that random interval arithmetic provides ranges of functions over small intervals which are much closer to the exact range of function values than the standard interval arithmetic.

Balanced random interval arithmetic proposed by Zilinskas and Bogle (2003) is obtained by choosing standard and inner interval operations at each step of the computation randomly with predefined probabilities. Balanced random interval arithmetic provides wider or narrower bounds depending on the predefined probabilities. The predefined probabilities should be defined by the user before any computation. Before each interval operation (addition, subtraction, multiplication or division) a random number (uniformly distributed between 0 and 1) is generated and if it is larger than the predefined probability of standard interval operations, inner interval operation is performed instead of the standard operation. Using balanced random interval arithmetic a number of sample intervals is evaluated. It is assumed that the distributions of centres and radii of the evaluated intervals are normal. An approximate range of the function is evaluated using the mean values and the standard deviations of centres (\( \mu_{\text{centres}} \) and \( \sigma_{\text{centres}} \)) and radii (\( \mu_{\text{radii}} \) and \( \sigma_{\text{radii}} \)) of the evaluated intervals:

\[
[\mu_{\text{centres}} - \alpha \sigma_{\text{centres}} - \mu_{\text{radii}} - \alpha \sigma_{\text{radii}}, \mu_{\text{centres}} + \alpha \sigma_{\text{centres}} + \mu_{\text{radii}} + \alpha \sigma_{\text{radii}}].
\]

Sometimes, when larger values of the predefined probability of standard interval operations are used, the result of balanced random interval arithmetic may exceed the standard interval. In this case the evaluated interval is intersected with the standard interval to get a tighter interval while not reducing the probability that the evaluated interval will contain all function values, as the standard interval gives guaranteed bounds.

The choice of the predefined probabilities of standard and inner interval operations dictates the balance to be struck between the speed and reliability of the global optimisation algorithm. The global optimum may be missed by the optimisation algorithm if the predefined probability of standard interval operations is too small and the evaluated ranges of the function are too narrow. However, the optimisation is faster when the ranges are narrower.

A branch and bound technique has been used to construct interval global optimisation algorithm. Lower and upper bounds of function values over a sub-region were evaluated using balanced random interval arithmetic: \([f^L(X), f^U(X)]=f(X)\). 5 samples and \( \alpha=3.0 \) have been used for balanced random interval arithmetic. The calculations of the balanced random interval function takes approximately 6 times longer than the
calculation of the standard interval function when 5 samples are used. Scalar function values at middle points of sub-regions \( f(\text{middle}(X)) \) were used as upper bounds for the minimum value of function over the sub-region.

The performance of algorithms with standard interval arithmetic and with balanced random interval arithmetic have been compared optimising a multidimensional scaling function and a separation design problem. The functions were optimised using the algorithm with balanced random interval arithmetic with different predefined probabilities of standard interval operations. 100 runs were used for each value of predefined probability.

The success rate for each value of predefined probability was evaluated showing the rate of runs finding the true optimum found by the algorithm with standard interval arithmetic. The ratio between the mean number of scalar objective function evaluations of the algorithm with balanced random interval arithmetic \( \text{mean}(nfe_{\text{sub}}) \) and the number for standard interval arithmetic \( nfe_{\text{sub}} \) \( \text{nfe}=\text{mean}(nfe_{\text{sub}})/nfe_{\text{sub}} \) and the ratio between the mean number of interval objective function evaluations of the algorithm with balanced random interval arithmetic \( \text{mean}(nfe_{\text{sub}}) \) and the number for standard interval arithmetic \( nfe_{\text{sub}} \) \( \text{nfe}=\text{mean}(nfe_{\text{sub}})/nfe_{\text{sub}} \) are also evaluated for each value of predefined probability.

3. Case Studies

Two case studies have been used to explore and compare the performances of algorithms with standard interval arithmetic and with balanced random interval arithmetic: soft drinks similarity analysis using multidimensional scaling and separation process synthesis optimisation.

3.1. Multidimensional Scaling

Multidimensional scaling is a widely used technique to analyse the structure and inherent dimensionality of sets of multidimensional data as well as visualisation of data by means of their mapping into spaces of low dimensionality. Multidimensional scaling addresses the problem of how \( N \) objects represented by proximity data can be faithfully visualised as points in a low-dimensional Euclidean space. The proximity data are represented as pair-wise dissimilarity values \( \delta_{ij} \). A spatial representation of the objects in the \( m \)-dimensional space should be determined. In our research we use data from soft drinks testing (Mathar, 1996). The goal of this multidimensional scaling problem is to find the best configuration of ten objects representing ten soft drinks in the two-dimensional space which would help to interpret the data. The objective function of the problem is

\[
f(X) = \sum_{j \neq i} \left( \sqrt{(x_{i,1} - x_{j,1})^2 + (x_{i,2} - x_{j,2})^2} - \delta_{ij} \right)^2,\]

where \( x_{i,1}, x_{i,2} \) are the co-ordinates of the \( i \)-th object, \( i=1..10 \) and \( j=1..10 \). To reduce the subsets of local minimizers to the points, the first object is fixed at the origin and the second is allowed to lie on the x-axis: \( x_{1,1}=x_{1,2}=x_{2,2}=0 \). To reduce the number of mirrored
solutions, the second object is not allowed to lie on the negative x-axis and the third object is not allowed to lie below the x-axis: \(x_2,1 \geq 0, x_3,2 \geq 0\). Non fixed co-ordinates are variables of the problem. The number of variables of the problem is 17. The feasible region is \(D = (0,4]^2, [-4,4]^{n-2}\).

Experiments (Zilinskas and Bogle, 2003) have shown that the distributions of the centres and radii of the evaluated balanced random intervals are normal and so balanced random interval arithmetic can be used to evaluate the ranges of this function giving tighter intervals than standard interval arithmetic.

The overall global optimisation of the multidimensional scaling function is very computationally intensive so only a small sub-region has been explored at this stage. The sub-region is chosen so that the best known function value would be in the sub-region and optimisation of the function over the sub-region using the algorithms would not take too long such that experiments may be made and performance may be evaluated. Optimisation of the function over the sub-region using the algorithm with standard interval arithmetic on an AIX computer takes 31 second, evaluating the scalar objective function 41723 times and the interval objective function 49789 times.

The ratios of the numbers of function evaluations depending on the desired rate of success for the algorithm with balanced random interval arithmetic are shown in Figure 1. The ratios of numbers of functions evaluations \(nfe\) and \(nife\) are very similar to each other. The success rate approaches 1.0 in this case when the ratios approach approximately 0.33. In this case the numbers of function evaluations for the algorithm with balanced random interval arithmetic are three times less than for the algorithm with standard interval arithmetic, but time for optimisation is longer. When the desired success rate is greater than 0.99 the algorithm with balanced random interval arithmetic is slower than the algorithm with standard interval arithmetic. When the desired success rate is less than 0.99 the algorithm with balanced random interval arithmetic is faster. When 80% success rate is required the algorithm is two times faster and when 60% five times faster.

![Figure 1](image.png)

*Figure 1. The ratios of numbers of functions evaluations depending on the desired rate of success for the algorithm with balanced random interval arithmetic optimising multidimensional scaling function.*
3.2 Process Network Synthesis
The second case study is a process network synthesis problem. An example problem from (Csendes, 1998) is used. In this problem a separation network has two raw materials each consisting of three components. The network produces three products consisting of the same three components with a different component distribution. There are four separators and four blending controls in the network. The goal of the problem is to find values of control variables $x_1$, $x_2$, $x_3$ and $x_4$ minimizing the cost of separation:

$$f(X) = (a_1 + b_{11} + c_{11})^{0.6} + (a_{21} + b_{21} + c_{21})^{0.6} + (a_{12} + b_{12})^{0.6} + (b_{22} + c_{22})^{0.6},$$

where

$$a_{11} = x_1 A_1 + (1 - x_2) A_2 + x_3 (1 - x_1) A_1 + x_4 A_2, \quad b_{11} = x_1 B_1 + (1 - x_2) B_2 + x_4 (1 - x_1) B_1 + x_3 B_2)/(1 - x_4 x_3),$$

$$c_{11} = x_1 C_1 + (1 - x_2) C_2, \quad a_{21} = (1 - x_1) A_1 + x_2 A_2, \quad b_{21} = (1 - x_1) B_1 + x_2 B_2 + x_3 b_{11},$$

$$c_{21} = (1 - x_1) C_1 + x_2 C_2 + x_3 c_{11}, \quad a_{12} = (1 - x_4) a_{21}, \quad b_{12} = (1 - x_3) b_{21}, \quad b_{22} = (1 - x_3) b_{11}, \quad c_{22} = (1 - x_3) c_{11},$$

and $A_1 = 120$, $B_1 = 1$, $C_1 = 20$, $A_2 = 100$, $B_2 = 1$, $C_2 = 200$.

Optimisation of the separation cost function using the algorithm with standard interval arithmetic on an AIX computer uses 5.52 seconds of CPU time, evaluating the scalar objective function 41170 times and the interval objective function 68473 times.

The ratios of numbers of function evaluations depending on the desired rate of success for the algorithm with balanced random interval arithmetic are shown in Figure 2. The ratios of numbers of function evaluations $nfe$ and $nife$ are again very similar. The success rate approaches 1.0 in this case when the ratios approach approximately 0.5. In this case the numbers of function evaluations for the algorithm with balanced random interval arithmetic are two times less than for the algorithm with standard interval arithmetic, but time for optimisation is longer. When the desired success rate is greater than 0.96 the algorithm with balanced random interval arithmetic is slower than the algorithm with standard interval arithmetic. When the desired success rate is less than 0.96 the algorithm with balanced random interval arithmetic is faster. When 90% success rate is required the algorithm is four times faster, when 80% – six times faster and when 70% – ten times faster.

![Figure 2. The ratios of numbers of functions evaluations depending on the desired rate of success for the algorithm with balanced random interval arithmetic optimising separation cost function.](image)
4. Conclusions
The algorithm with balanced random interval arithmetic could be used when speed of optimisation is more important than guaranteed reliability instead of the algorithm with standard interval arithmetic – when the desired success rate is less than 95% the algorithm with balanced random interval arithmetic is faster than the algorithm with standard interval arithmetic and provides good, although not always optimal, values. The algorithm with 60% reliability is five times faster than the algorithm with guaranteed reliability. Experiments with two case studies have shown that the use of balanced random interval arithmetic in global optimisation is promising. The performance of the global optimisation algorithm based on balanced random interval arithmetic should be further studied and compared with the performance of stochastic global optimisation algorithms.

5. Acknowledgement
We wish to acknowledge the financial support through the Royal Society/NATO Postdoctoral Fellowship for the research “Stochastic Interval Methods for Process Optimisation”.

6. References