Evolutionary Optimization Method for Approximating the Solution Set Hull of Parametric Linear Systems

Iwona Skalna

Department of Applied Computer Science,
University of Mining and Metallurgy AGH
ul. Gramatyka 10, 30-067 Cracow, Poland

Abstract. Systems of parametric interval equations are encountered in many practical applications. Several methods for solving such systems have been developed during last years. Most of them produce both outer and inner interval solutions, but the amount of overestimation, resp. underestimation is not exactly known. If a solution of a parametric system is monotonic and continuous on each interval parameter, then the method of combination of endpoints of parameter intervals computes its interval hull. Recently, a few polynomial methods computing the interval hull were developed. They can be applied if some monotonicity and continouity conditions are fulfilled. To get the most accurate inner approximation of the solution set hull for problems with any bounded solution set, an evolutionary optimization method is applied.

1 Introduction

The behavior of the loaded truss structure or analog linear circuit can be described by a system of parameter-dependent linear equations [2,5,6,7,16,17]. Assuming some model parameters are unknown and lie in given intervals will lead to a parametric system of linear interval equations. Several methods for solving such systems have been developed during last years [5,6,10,12,15,16]. Most of them compute both outer and inner interval approximations of the solution set hull, but the amount of overestimation, resp. underestimation is not exactly known. It can be estimated by a comparison of inner and outer approximations of the solution set hull [12,15].

If a solution of the parametric system is monotonic and continuous on each parameter interval, then the method of combination of endpoints of parameter intervals (CEPI in brief) [2,14] computes interval hull of the parametric solution set, that is the tightest interval vector containing this set. Unfortunately, because the complexity of the CEPI method – 2^k real systems have to be solved, where k is the number of interval parameters – increases at an exponential rate as a factor of the number of parameters, it can be applied to problems with small number of parameters. Recently, a few polynomial methods computing interval hull were developed [14]. They can be applied if the solution of the parametric system is continous and monotonic on each interval parameter. In [13] Popova

T. Boyanov et al. (Eds.): NMA 2006, LNCS 4310, pp. 361–368, 2007.

shows, that when some sufficient conditions are fulfilled, the interval hull (or some bounds) of the parametric solution set can be easily computed.

To get the best inner approximation of the solution set hull for problems with bounded (non-monotonic and non-continous) solution set, and a lot of interval parameters, an evolutionary optimization method (EOM in brief) is applied in Section 3. In Section 4 some numerical examples are presented and used to evaluate the results of the EOM method. They include two examples of parametric systems with non-monotonic solution set, and two test examples of truss structures. The computations performed show that the EOM method produces a high-quality approximation of the interval hull of a parametric solution set.

2 Preliminaries

Let $\mathbb{IR}$, $\mathbb{IR}^n$, $\mathbb{IR}^{n\times n}$ denote the set of real compact intervals, interval vectors, and interval square matrices, respectively [9]. Italic faces will denote real quantities, while bold italic faces will denote their interval counterparts.

Consider linear algebraic system

$$A(p)x = b(p) \tag{1}$$

with coefficients being functions that are linear in parameters

$$a_{ij}(p) = \omega_{ij0} + \sum_{\nu=1}^{k} \omega_{ij\nu} \cdot p_\nu, \quad b_j(p) = \omega_{0j0} + \sum_{\nu=1}^{k} \omega_{0j\nu} \cdot p_\nu, \tag{2}$$

$(i,\, j = 1, \ldots, n)$; where $p = \{p_1, \ldots, p_k\}^\top \in \mathbb{R}^k$ is a vector of parameters, $\omega \in (\mathbb{R}^{k+1})^{((n+1)\times n)}$ is a matrix of real $(k+1)$-dimensional vectors.

Assuming some model parameters are unknown and lie in given intervals $\boldsymbol{p}_i \ni p_i$ $(i = 1, \ldots, k)$ would give a family of systems (1) which is usually written in a symbolic compact form

$$A(\boldsymbol{p})x = b(\boldsymbol{p}), \tag{3}$$

and is called *parametric interval linear system.*

Parametric solution set of the system (3) is defined [4,15] as

$$S(\boldsymbol{p}) = \{\, x \mid A(p)x = b(p),\, p \in \boldsymbol{p}\}\,. \tag{4}$$

If the solution set is bounded, then its interval hull exists and is defined as

$$\square S(\boldsymbol{p}) = [\inf S(\boldsymbol{p}),\, \sup S(\boldsymbol{p})] = \bigcap \{\boldsymbol{y} \in \mathbb{IR}^n \mid S(\boldsymbol{p}) \subseteq \boldsymbol{y}\}\,. \tag{5}$$

In order to guarantee that the solution set is bounded, the matrix $A(\boldsymbol{p})$ must be regular, i.e. $A(p)$ must be regular for all $p \in \boldsymbol{p}$.

A vector $\boldsymbol{x} = [\underline{x}, \overline{x}] \in \mathbb{IR}^n$ is called inner approximation of $S \subseteq \mathbb{R}^n$ if

$$\inf_{s\in S} s_i \leqslant \underline{x}_i \quad \text{and} \quad \sup_{s\in S} s_i \geqslant \overline{x}_i, \quad i = 1, \ldots, n,$$

resp. outer approximation of $S \subseteq \mathbb{R}^n$ if

$$\inf_{s\in S} s_i \geqslant \underline{x}_i \quad \text{and} \quad \sup_{s\in S} s_i \leqslant \overline{x}_i, \quad i = 1, \ldots, n.$$

3 Evolutionary Optimization

The problem of computing optimal inner approximation of the solution set hull of the parametric linear system (3) can be written as a problem of solving $2n$ constrained optimization problems: for $i = 1, \ldots, n$,

$$\min \left\{ f(p) = \left(A(p)^{-1} b(p)\right)_i \mid p \in \boldsymbol{p} \right\} \tag{6}$$

and

$$\max \left\{ f(p) = \left(A(p)^{-1} b(p)\right)_i \mid p \in \boldsymbol{p} \right\}, \tag{7}$$

where $\boldsymbol{p} \in \mathbb{IR}^k$ is a vector of interval parameters.

Theorem 1. *Let $A(\boldsymbol{p})$ be regular, $p \in \mathbb{IR}^k$, and $x^i_{\min}$, $x^i_{\max}$ denote the global solutions of the i-th minimization (6), resp. maximization (7) problems. Then the interval vector $\boldsymbol{x} = [x_{\min}, x_{\max}] = \left(\left[x^i_{\min}, x^i_{\max}\right]\right)_{i=1}^n = \square S(\boldsymbol{p})$.*

Proof. $\subseteq$: $x^i_{\min}, x^i_{\max} \in S(\boldsymbol{p})_i$, hence $\left[x^i_{\min}, x^i_{\max}\right] \subseteq \square S(\boldsymbol{p})_i$. $\supseteq$: take any $x \in S(\boldsymbol{p})$, then $x = A(p)^{-1} b(p)$ for some $p \in \boldsymbol{p}$. Since for each $p \in \boldsymbol{p}$ $x^i_{\min} \leqslant (A(p)^{-1} b(p))_i \leqslant x^i_{\max}$, then $S(\boldsymbol{p})_i \subseteq \left[x^i_{\min}, x^i_{\max}\right]$ and hence

$$\square S(\boldsymbol{p})_i \subseteq \left[x^i_{\min}, x^i_{\max}\right].$$

$\square$

The optimization problems (6) and (7) will be solved using an evolutionary approach [3,8,11]. As a result of the minimization (maximization) problem one will obtain a value greater of equal (less or equal) to the actual minimum (maximum). The final result will be the inner approximation of the solution set hull.

3.1 Evolutionary Algorithm Description

Optimization is performed using the evolutionary algorithm shown in Fig. 1. Each evolutionary algorithm requires some input parameters. These are: population size (pop_{size}), crossover rate (c_r), mutation rate (m_r), number of generations (n_g). All of them have great influence on the result of the optimization, but the choice of the best values is still a matter of trial. Suggestions for parameter values can be found in the literature [1,8,11].

For $t = 0, \ldots n_g$, the population $P(t) = \left\{p^t_1, \ldots, p^t_{n_g}\right\}$ consists of individuals characterized by k-dimensional vectors of real numbers $p^t_i = \{p^t_{i1}, \ldots, p^t_{ik}\}^{\mathrm{T}}$ with $p^t_{ij} \in \boldsymbol{p}_j$, $i = 1, \ldots, pop_{size}, j = 1, \ldots, k$. The elements p^0_{ik} of the initial population $P(0)$ are generated randomly based on the uniform distribution.

The two following genetic operators are employed [8]:

- *non-uniform mutation* - this one-argument operator vouch for the system adaptation ability. If the element p_j of the individual p is chosen for mutation, then $p' = \left\{p_1, \ldots, p'_j, \ldots, p_k\right\}^\top$ with

$$p'_j = \begin{cases} p_j + \left(\overline{p}_j - p_j\right)\, r\, \left(1 - t/n_g\right)^b, & \text{if } q < 0.5 \\ p_j + \left(p_j - \underline{p}_j\right)\, r\, \left(1 - t/n_g\right)^b, & \text{if } q \geqslant 0.5, \end{cases}$$

where r, q are random numbers from [0, 1], t is a number of generation, n_g is a number of generations, and b is a parameter of the system describing the level of heterogeneity; the probability that mutation factor is close to zero increases as t increases from 0 to n_g;

- *arithmetic crossover* - this two-argument operator is defined as linear combination of two vectors. If the parents p^1 and p^2 are chosen for crossover, then the offsprings are

$$p^{1\prime} = rp^1 + (1-r)p^2, \qquad p^{2\prime} = (1-r)p^1 + rp^2,$$

where r is a random number from [0, 1]; arithmetic crossover guarantees that the elements of the vectors $p^{1\prime}$ and $p^{2\prime}$ lie in parameter intervals.

Let $P'(t)$ denote the population after the crossover process, and $P''(t)$ - the population after the mutation process. The best (wrt fitness function), (7)) pop_{size} individuals, from the combined population $P(t) \cup P''(t)$, form a new population $P(t+1)$.

```
Initialize parameters
t := 0
Initialize population P(t) of pop_size size
while t < n_g do
    P'(t) ⟵ Crossover_with_c_r_rate(P(t))
    P''(t) ⟵ Mutate_with_m_r_rate(P'(t))
    Evaluate the fitness f(P''(t))
    P(t+1) ⟵ Select_pop_size(P(t) ∪ P''(t))
    t:=t+1
end while
```

Fig. 1. Pseudo-code of an evolutionary algorithm

4 Numerical Examples

In this section the results of the EOM are presented. Two small parametric linear systems with non-monotonic solution set and two exemplary truss structures are included to evaluate the performance of the EOM method. Several variants of the input parameter values have been examined. The best results have been obtained for the following values: population size $pop_{size} = 10$ (exs. 1, 2), $pop_{size} = 16$ (exs. 3, 4), crossover rate $c_r = 0.1$, mutation rate $m_r = 0.9$, $b = 96$, number of generations $n_g = 80$ (exs. 1, 2), $n_g = 100$ (exs. 3, 4).

Example 1
Assume the two-dimensional parametric linear system is of the form:

$$\begin{bmatrix} p_1 & 1+p_2 \\ -2 & 3p_1-1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2p_1 \\ 0 \end{bmatrix}, \quad \boldsymbol{p}_1 = [0,\ 1], \boldsymbol{p}_2 = [0,\ 1]. \tag{8}$$

The numerical results are presented in Table 1. Column 2 contains the midpoint solution, columns 3 - the result of the EOM method, column 4 - the interval hull (IH). In case of two-dimensional system the interval hull can be easily computed. From (8) one gets two equations $p_1x_1 + (1 + p_2)x_2 = 2p_1$ and $-2x_1 + (3p_1 - 1)x_2 = 0$. Eliminating p_1 from these equations gives the equation $2x^2 + 3(1 + p_2)y^2 + xy - 4x - 2y = 0$ which, with $p_2 \in [0, 1]$, define the pencil of ellipses. The intersection of the pencil with the united solution set gives the parametric solution set.

Table 1. Numerical results for Example 1

	x_0	EOM	IH
$\boldsymbol{x}_1$	0.1538	$[-0.087, 1]$	$[-0.087, 1]$
$\boldsymbol{x}_2$	0.6154	$[0, 1.026]$	$[0, 1.026]$

Example 2
The three-dimensional parametric linear system is of the form:

$$\begin{bmatrix} p_1 & p_2 + 1 & -p_3 \\ p_2 + 1 & -3 & p_1 \\ 2 - p_3 & 4p_2 + 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2p_1 \\ p_3 - 1 \\ -1 \end{bmatrix}, \tag{9}$$

where $p_1 \in \boldsymbol{p}_1 = [0, 1]$, $p_2 \in \boldsymbol{p}_2 = [0, 1]$, $p_3 \in \boldsymbol{p}_3 = [0, 1]$.

The numerical results of the EOM method, the direct method (DM) [17], and the Monte Carlo method (MC) (100000 samples), are presented in Table 2.

Table 2. Numerical results for Example 2

	x_0	MC	EOM	DM
$\boldsymbol{x}_1$	0.286	$[-0.866, 2.641]$	$[-1, 2.667]$	$[-2.184, 4.685]$
$\boldsymbol{x}_2$	0.048	$[-0.65, 0.328]$	$[-0.667, 0.333]$	$[-0.84, 1.337]$
$\boldsymbol{x}_3$	-1.571	$[-5.592, 0.679]$	$[-5.667, 1]$	$[-11.163, 2.663]$

Example 3. (*25-bars plane truss structure*)
Consider the truss structure shown in Fig. 2. Young's modulus E= 2.1×10^{11} [Pa], cross-section area C= 0.004 [m^2]. Assume the stiffness of all bars is uncertain by $\pm 5\%$. This gives 25 interval parameters. The vector of displacements $\boldsymbol{d}$ is a solution of the parametric system $K(\boldsymbol{p})\boldsymbol{d} = q(\boldsymbol{p})$, where $K(\boldsymbol{p})$ is parameter-dependent stiffness matrix, $q(\boldsymbol{p})$ is parameter-dependent vector of forces. Numerical results are presented in table 3. Column 2 contains the midpoint solution (d_0), columns 3, 4 - the result, resp. the relative error ($rerr = (\overline{d} - \underline{d})/(2 \cdot d_0)$) of the Monte Carlo method (100000 samples), columns 5, 6 - the result, resp. the relative error of the EOM method, columns 7, 8 - the result, resp. the relative error of the DM method.

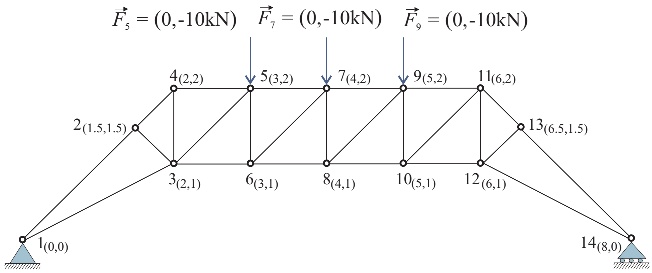

Fig. 2. 25-bar plane truss structure

Table 3. Numerical results for example 3

	d_0 [$\times10^{-5}$]	MC [$\times10^{-5}$]	$rerr$[%]	EOM [$\times10^{-5}$]	$rerr$[%]	DM [$\times10^{-5}$]	$rerr$[%]
d_2^x	62.2	[60.26, 64.34]	3.27	[59.12, 65.57]	5.19	[56.47, 67.9]	9.2
d_2^y	−74.32	[−76.95, −71.95]	3.37	[−78.33, −70.66]	5.16	[−79.99, −68.63]	7.64
d_3^x	50.83	[49.16, 52.69]	3.46	[48.33, 53.57]	5.16	[47.27, 54.37]	6.98
d_3^y	−85.68	[−88.6, −83.05]	3.24	[−90.33, −81.45]	5.18	[−92.86, −78.49]	8.39
d_4^x	66.66	[64.66, 68.96]	3.22	[63.34, 70.31]	5.23	[61.01, 72.3]	8.47
d_4^y	−82.83	[−85.6, −80.31]	3.19	[−87.33, −78.73]	5.19	[−89.45, −76.18]	8.01
d_5^x	63.81	[61.85, 65.98]	3.23	[60.62, 67.31]	5.24	[58.56, 69.04]	8.21
d_5^y	−102.7	[−106.03, −99.81]	3.03	[−108.17, −97.74]	5.08	[−111.82, −93.56]	8.89
d_6^x	55.12	[53.28, 57.15]	3.51	[52.41, 58.08]	5.15	[51.41, 58.8]	6.7
d_6^y	−103.18	[−106.5, −100.29]	3.01	[−108.65, −98.21]	5.06	[−112.53, −93.81]	9.07
d_7^x	59.52	[57.59, 61.56]	3.33	[56.53, 62.79]	5.26	[54.57, 64.46]	8.31
d_7^y	−108.93	[−112.32, −105.9]	2.95	[−114.66, −103.74]	5.01	[−119.04, −98.8]	9.29
d_8^x	59.88	[57.88, 62.1]	3.52	[56.95, 63.1]	5.14	[56.03, 63.71]	6.41
d_8^y	−108.45	[−111.85, −105.41]	2.97	[−114.18, −103.26]	5.03	[−118.57, −98.32]	9.33
d_9^x	54.76	[53.01, 56.89]	3.54	[51.99, 57.78]	5.28	[50.05, 59.45]	8.58
d_9^y	−101.98	[−105.03, −99.10]	2.90	[−107.42, −97.06]	5.08	[−111.56, −92.41]	9.39
d_{10}^x	64.16	[62.11, 66.53]	3.44	[61.03, 67.61]	5.13	[60.14, 68.17]	6.28
d_{10}^y	−100.56	[−103.63, −97.69]	2.95	[−105.96, −95.64]	5.13	[−109.91, −91.19]	9.31
d_{11}^x	50.47	[48.64, 52.45]	3.77	[47.81, 53.38]	5.52	[45.89, 55.04]	9.06
d_{11}^y	−82.83	[−85.62, −80.19]	3.27	[−87.36, −78.69]	5.23	[−90.46, −75.21]	9.21
d_{12}^x	67.02	[64.9, 69.4]	3.35	[63.75, 70.62]	5.12	[62.73, 71.3]	6.39
d_{12}^y	−84.25	[−87.06, −81.69]	3.18	[−88.87, −80.05]	5.23	[−92.33, −76.18]	9.58
d_{13}^x	56.01	[54, 58.28]	3.82	[53.13, 59.16]	5.38	[51.43, 60.58]	8.17
d_{13}^y	−73.25	[−75.8, −70.86]	3.37	[−77.23, −69.62]	5.20	[−80.18, −66.3]	9.47
d_{14}^x	117.13	[113.57, 121.14]	3.23	[111.55, 123.29]	5.01	[110.18, 124.07]	5.93

Example 4. (*Baltimore bridge 1820*)
Every bar of the bridge (Fig. 3) has Young's modulus E= 2.1×10^{11} [Pa], cross-section area C= 0.004 [m^2]. Assuming the stiffness of all bars is uncertain by $\pm5\%$ will give 45 interval parameters. The computational time of the EOM method increased about 7 times. Most coordinates of the vector solution, produced by the

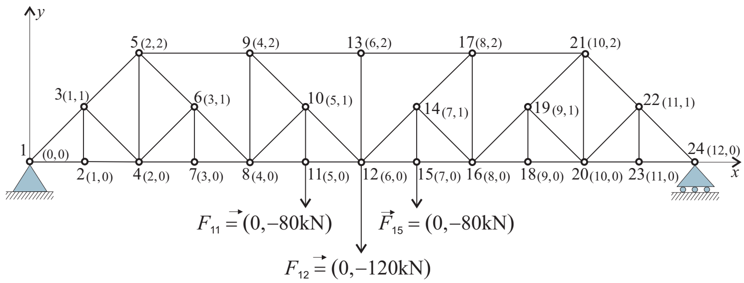

Fig. 3. Baltimore bridge 1820

EOM method, were exactly equal to the interval hull. The remaining coordinates differed only slightly from the exact hull solution. The result of the EOM method is not included because of the limit for the number of pages.

5 Conclusions

The problem of solving parametric linear systems has been considered in Section 2. In Section 3 the evolutionary optimization method EOM for approximating from below the solution set hull of parametric linear systems has been described. Computations performed in Section 4 show that the EOM is a powerful tool for solving such systems. The EOM method produced a very accurate approximation of the interval hull of all parametric solution sets considered. It is simple and quite efficient. The main advantage of the EOM method is that it can be applied to any parametric linear system with bounded solution set. The EOM method can be used to solve problems with a lot of interval parameters. However, since the accuracy of the EOM method is not exactly known it should be used in conjunctions with methods that compute inner and outer approximations. The comparison of the EOM method with existing methods solving parametric linear systems will be a subject of future work.

References

1. T. Bäck, D.B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computations.* Oxford University Press, 1997.
2. A. Dreyer. Combination of symbolic and interval-numeric methods for analysis of analog circuits. *Proc. 8 International Workshop on Symbolic Methods and Applications in Circuit Design (SMACD 2004)*, september 2004. Wroclaw, Poland.
3. D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley, Reading, Massachusetts, 1989.
4. C. Jansson. Interval linear systems with symmetric matrices, skew-symmetric matrices and dependencies in the right hand side. *Computing*, 46(3):265–274, 1991.
5. L.V. Kolev. Outer solution of linear systems whose elements are affine functions of interval parameters. *Reliable Computing*, 6:493–501, 2002.

6. L.V. Kolev. A method for outer interval solution of linear parametric systems. *Reliable Computing*, 10:227–239, 2004.
7. Z. Kulpa, A. Pownuk, and I. Skalna. Analysis of linear mechanical structures with uncertainties by means of interval methods. *Computer Assisted Mechanics and Engineering Sciences*, 5(4):443–477, 1998.
8. Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer-Verlag, Berlin, Germany, 1996.
9. A. Neumaier. *Interval Methods for Systems of Equations.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge, UK, 1990.
10. A. Neumaier and A. Pownuk. Linear systems with large uncertainties, with applications to truss structures. *(submitted for publication).*
11. A. Osyczka. *Evolutionary Algorithm for Single and Multicriteria Design Optimization.* Studies in Fuzziness and Soft Computing Physica-Verlag, Heidelberg New York, 2002.
12. E.D. Popova. On the solution of parametrised linear systems. *Scientific Computing, Validated Numerics, Interval Methods*, pages 127–138, 2001.
13. E.D. Popova. Quality of the solution of parameter-dependent interval linear systems. *ZAMM*, 82(10):723–727, 2002.
14. A. Pownuk. Efficient method of solution of large scale engineering problems with interval parameters. *NSF workshop on Reliable Engineering Computing*, September 15-17, 2004, Savannah, Georgia, USA.
15. S.M. Rump. Verification methods for dense and sparse systems of equations. In Jürgen Herzberger, editor, *Topics in validated computations: proceedings of IMACS-GAMM International Workshop on Validated Computation, Oldenburg, Germany, 30 August–3 September 1993*, volume 5 of *Studies in Computational Mathematics*, pages 63–135, Amsterdam, The Netherlands, 1994. Elsevier.
16. I. Skalna. Methods for solving systems of linear equations of structure mechanics with interval parameters. *Computer Assisted Mechanics and Engineering Sciences*, 10(3):281–293, 2003.
17. I. Skalna. A method for outer interval solution of parametrized systems of linear interval equations. *Reliable Computing*, 12(2):107–120, 2006.