Received: 26 April 2018 | Revised: 19 October 2018 | Accepted: 4 December 2018

DOI: 10.1002/nla.2229

RESEARCH ARTICLE

WILEY

Direct and iterative methods for interval parametric algebraic systems producing parametric solutions

Iwona Skalna[1] | Milan Hladík[2]

[1]Department of Applied Computer Science, AGH University of Science and Technology, Kraków, Poland

[2]Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Czech Republic

Correspondence
Milan Hladík, Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Malostranské nám. 25, 118 00 Prague, Czech Republic.
Email: milan.hladik@matfyz.cz

Funding information
Czech Science Foundation (Grantová agentura České republiky), Grant/Award Number: P403-18-04735S

Summary

This paper deals with interval parametric linear systems with general dependencies. Motivated by the so-called parameterized solution introduced by Kolev, we consider the enclosures of the solution set in a revised affine form. This form is advantageous to a classical interval solution because it enables us to obtain both outer and inner bounds for the parametric solution set and, thus, intervals containing the endpoints of the hull solution, among others. We propose two solution methods, a direct method called the generalized expansion method and an iterative method based on interval-affine Krawczyk iterations. For the iterative method, we discuss its convergence and show the respective sufficient criterion. For both methods, we perform theoretical and numerical comparisons with some other approaches. The numerical experiments, including also interval parametric linear systems arising in practical problems of structural and electrical engineering, indicate the great usefulness of the proposed methodology and its superiority over most of the existing approaches to solving interval parametric linear systems.

KEYWORDS

affine arithmetic, interval computation, linear equations, parametric system

1 | INTRODUCTION

In solving real-life problems, we often deal with uncertain data, that is, data that contain noise due to various kinds of inexactness (measurement errors, incomplete knowledge, data estimation, etc.). If we assume that the lower and upper bounds on uncertain data are given, then we encounter interval-valued quantities. In classical interval analysis, it is usually assumed that interval entries are independent from each other. This phenomena is known as the *dependency problem* and is the main source of overestimation in interval computation. To handle the dependency problem, we employ in this paper the interval parametric approach.

Formally, consider the $n \times n$ system of linear equations

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

in which constraint the matrix $A(p)$ and the right-hand side vector $b(p)$ depend on the vector of parameters $p = (p_1, \dots, p_K)^T$. If the parameters are unknown but vary within compact intervals, that is, $p_k \in \boldsymbol{p}_k = [\underline{p}_k, \overline{p}_k]$ for $k = 1, \dots, K$, then we obtain the following family of parametric linear systems:

$$\{A(p)x = b(p), \quad p \in \boldsymbol{p}\}. \tag{2}$$

As a special case, we will also discuss affine-linear dependencies, meaning that $A(p)$ and $b(p)$ depend linearly on p. In this case, $A(p)$ and $b(p)$ can be expressed as

$$A(p) = A^{(0)} + \sum_{k=1}^{K} A^{(k)} p_k, \quad b(p) = b^{(0)} + \sum_{k=1}^{K} b^{(k)} p_k, \tag{3}$$

where $A^{(k)} \in \mathbb{R}^{n\times n}$ and $b^{(k)} \in \mathbb{R}^n$ are fixed and known a priori.

The *united parametric solution set* of the system (2) is defined as a set of all possible solutions, that is,

$$\Sigma = \{x \in \mathbb{R}^n \mid \exists p \in \boldsymbol{p} : A(p)x = b(p)\}.$$

The parametric solution set is hard to characterize, even for particular classes of the affine-linear case. For example, the explicit description of the symmetric systems (where the symmetry of the constraint matrix defines the linear dependencies) was developed, for example, in the works of Hladík[1] and Mayer.[2] The general case of linear dependencies was characterized by Popova[3] in particular. In this case, the shape of the solution set is described by quadrics (cf. Figure 1). Handling Σ is computationally hard. Many questions, such as nonemptiness, boundedness, or computing tight approximation of Σ, are NP-hard even for very special subclasses of problems.[4-6]

Example 1. Consider the two-dimensional interval parametric linear system

$$\left\{ \begin{pmatrix} p_2 & 1+2p_1 \\ 3p_2 & -3p_2 \end{pmatrix} x = \begin{pmatrix} 2p_2 \\ 1 \end{pmatrix}, \quad p_1, p_2 \in [\delta, 2.05] \right\} \tag{4}$$

having affine-linear dependencies and where $\delta > 0$. The parametric solution set of the system (4), for $\delta = 0.13, 0.6$, is depicted in Figure 1. As we can see, the solution set on the left is included in the first quadrant of the coordinate plane and is nonconvex (let us recall that the solution set of a classical interval linear system is convex in each orthant).

Example 2. Consider the following three-dimensional interval parametric linear system:

$$\left\{ \begin{pmatrix} 1 & p_1 & p_2 \\ p_1 & 2 & p_1 \\ p_2 & p_1 & 3 \end{pmatrix} x = \begin{pmatrix} 1 \\ p_1^2 \\ p_2^2 \end{pmatrix}, \quad p_1 \in [0.0, 1.0], \quad p_2 \in [0.0, 0.9] \right\}, \tag{5}$$

which has nonlinear dependencies in the right-hand side vector. The parametric solution set of system (5) is depicted in Figure 2. As can be seen, it is represented by snail-shaped hypersurfaces.

Let us now introduce some interval notation. An interval vector is defined as

$$\boldsymbol{x} = \{x \in \mathbb{R}^n \mid \underline{x}_i \le x_i \le \overline{x}_i, \quad i = 1, \ldots, n\},$$

where $\underline{x}, \overline{x} \in \mathbb{R}^n$, $\underline{x} \le \overline{x}$, are given. The midpoint of an interval vector x is given by $x^c := \frac{1}{2}(\underline{x} + \overline{x})$, and its radius is given by $x^\Delta := \frac{1}{2}(\overline{x} - \underline{x})$. The set of all interval vectors is denoted by $\mathbb{IR}^n$. Similar notation is used for interval matrices. The

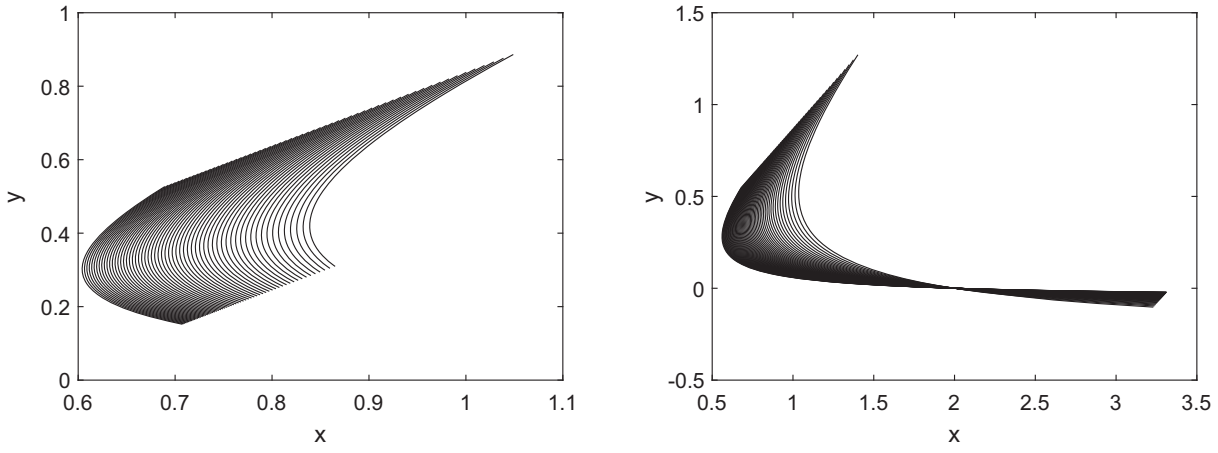

FIGURE 1 Parametric solution set of system (4): $\delta = 0.6$ (left) and $\delta = 0.13$ (right)

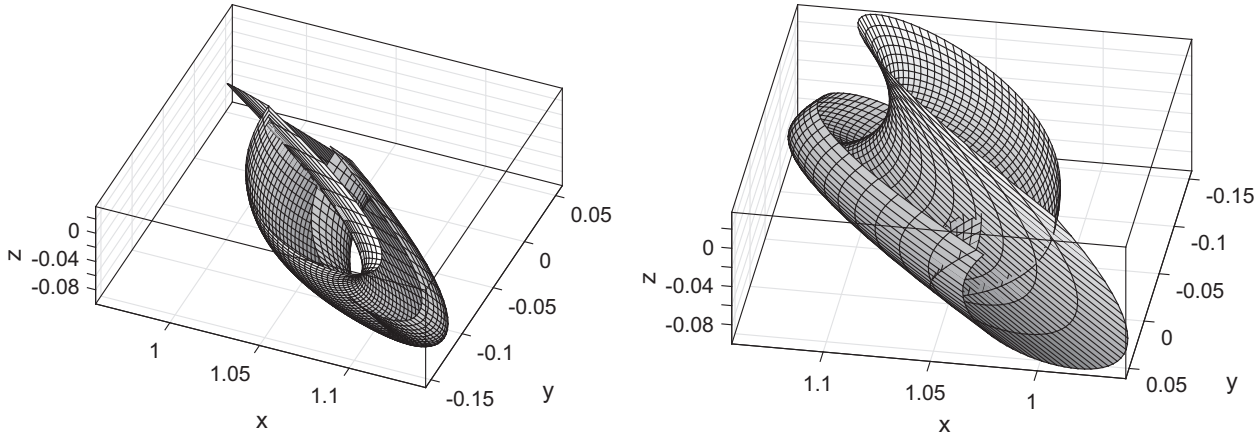

FIGURE 2 Parametric solution set of system (5) viewed from different perspectives

smallest (w.r.t. inclusion) interval vector containing Σ is called an interval *hull* of Σ and is denoted by $\square\Sigma$. Concerning noninterval notation, we use I for the identity matrix of arbitrary size, $\rho(\cdot)$ for the spectral radius, and $\overrightarrow{1}$ for the vector of ones of a suitable size.

The basic problem of interval linear algebra is to compute tight outer enclosure for the solution set Σ. An enclosure is understood as any interval vector $\boldsymbol{x} \in \mathbb{IR}^n$ such that $\Sigma \subseteq \boldsymbol{x}$. There exist various approaches to finding such an enclosure. For linear dependencies, various iterative methods were investigated.[7-14] The so-called direct methods were given in other works.[15-19] The problem of computing the tightest enclosure, the so-called interval hull of Σ, was addressed in the works of Kolev[18,20,21] and Skalna.[22] The general case of dependencies was discussed in several papers.[20,23,24]

1.1 | Revised affine form

Affine arithmetic[25,26] was introduced to keep track of dependencies between interval quantities and what follows to reduce overestimation in interval computation. The main disadvantage of affine arithmetic is that each nonlinear operation (in floating-point computing each operation) introduces a new noise symbol, so the length of affine forms increases during the computation. Therefore, the use of affine computation is limited in practical applications. There were several attempts to improve classical affine arithmetic.[27,28] Here, we utilize the *revised affine forms*.[29] Because it uses the so-called *accumulative error*, the number of noise symbols does not increase during the computation. Hence, the computation with revised affine forms are more efficient than computation with classical affine forms. Revised affine forms were successfully employed, for example, for solving parametric systems[30] by means of an interval–affine Gauss–Seidel iteration.

A revised affine form (affine form in brief) can be expressed as

$$\boldsymbol{x}(e) = \sum_{k=1}^{K} x^{(k)} \varepsilon_k + \boldsymbol{x},$$

where $x^{(1)}, \ldots, x^{(k)}$ are *partial deviations*, $\varepsilon_1, \ldots, \varepsilon_K$ are parameters (*noise symbols*) ranging independently within interval $[-1, 1]$, $e = (\varepsilon_1, \ldots, \varepsilon_K)^T$ is a vector of parameters, and $\boldsymbol{x} = [\underline{x}, \overline{x}]$ is an interval representing the accumulative error. The latter is used to store (accumulate) all errors introduced during a computation. The revised affine form $\boldsymbol{x}(e)$ implies the range $[\boldsymbol{x}(e)] = \sum_{i=1}^{K} |x^{(k)}|[-1, 1] + \boldsymbol{x}$. Conversely, an interval $\boldsymbol{x}$ can be represented by the affine form $\boldsymbol{x}(e) = x^c + x^{\Delta}\varepsilon_k$, where ε_k is a new noise symbol, independent from the noise symbols already used in a computation.

Affine-linear combinations of affine forms are naturally defined as follows. Given $\alpha, \beta, \gamma \in \mathbb{R}$ and two affine forms $\boldsymbol{x}(e), \boldsymbol{y}(e)$, their affine-linear combination is given by

$$\alpha\boldsymbol{x}(e) + \beta\boldsymbol{y}(e) + \gamma = \sum_{k=1}^{K} \left(\alpha x^{(k)} + \beta y^{(k)}\right) \varepsilon_k + (\alpha\boldsymbol{x} + \beta\boldsymbol{y} + \gamma),$$

where the residual interval $\alpha\boldsymbol{x}+\beta\boldsymbol{y}+\gamma$ is calculated by standard interval arithmetic.[31] The product of two affine forms is not an affine form, so it must be approximated by a suitable affine form. There are several ways to do it. The trivial approximation[26] yields

$$\begin{aligned}\boldsymbol{x}(e)\boldsymbol{y}(e) :=& \sum_{k=1}^{K}\left(x^{(k)}y^c + x^c y^{(k)}\right)\varepsilon_k + x^c y^c + (|x^c|y^\Delta + x^\Delta|y^c|)[-1,1]\\ &+\left(\sum_{k=1}^{K}|x^{(k)}| + x^\Delta\right)\left(\sum_{k=1}^{K}|y^{(k)}| + y^\Delta\right)[-1,1].\end{aligned}$$

There are several better approximations.[32,33] Even the best Chebyshev minimum-error approximation can be computed in linear time.[32] Because the division of affine forms is usually defined as multiplication by the reciprocal,[30] it can also be performed in linear time.

1.2 | System and solutions in affine forms

In the case of affine-linear dependencies, the system (1) can be straightforwardly[34] expressed as

$$\boldsymbol{A}(e)x = \boldsymbol{b}(e), \tag{6}$$

where

$$\begin{aligned}\boldsymbol{A}(e) &= \sum_{k=1}^{K} A^{(k)}\varepsilon_k + \boldsymbol{A},\\ \boldsymbol{b}(e) &= \sum_{k=1}^{K} b^{(k)}\varepsilon_k + \boldsymbol{b}.\end{aligned}$$

In the nonlinear case, the functions $\boldsymbol{A}_{ij}(e)$ and $\boldsymbol{b}_i(e)$ are usually described by some analytical expressions. The evaluation of these expressions by means of affine arithmetic[26–30] yields again the revised affine forms (6). Therefore, we can assume that form anyway.

Example 3. Consider the parametric system of interval linear equations

$$\left\{\begin{pmatrix} p_2 & 1+2p_1^2\\ 3p_2 & -3p_2\end{pmatrix}x = \begin{pmatrix}2p_2\\ 1\end{pmatrix},\quad p_1,p_2\in[0.6,1.05]\right\} \tag{7}$$

burdened by nonlinear dependencies. The transformed system of revised affine forms is given by

$$\left(\begin{pmatrix} 0.225\varepsilon_2 & 0.7425\varepsilon_1\\ 0.675\varepsilon_2 & -0.675\varepsilon_2\end{pmatrix} + \begin{pmatrix} 0.825 & 2.412+0.051[-1,1]\\ 2.475 & -2.475\end{pmatrix}\right)x = \begin{pmatrix}1.65+0.45\varepsilon_2\\ 1\end{pmatrix}. \tag{8}$$

The solution sets of the systems (7) and (8) are depicted in Figure 3. As we can see, the solution set of the system (8) is only slightly bigger than the parametric solution set of the original system, so the gain from using the affine transformation in this case is significant.

So far, when solving interval parametric linear equations, most of the developed methods yield an enclosure for the solution set Σ in the form of an interval vector $\boldsymbol{x}\supseteq\Sigma$. Kolev[18] introduced a novel type of a solution called a *parameterized solution* or simply a *p*-solution defined as

$$\boldsymbol{x}(e) = Le + \boldsymbol{x} \tag{9}$$

and satisfying the condition that, for each $e\in[-1,1]^K$, the solution of the system (6) is included in $\boldsymbol{x}(e)$. The evaluation of the expression (9) by means of interval arithmetic yields the outer interval enclosure

$$[\boldsymbol{x}(e)]^{OI} := \left[-|L|\overrightarrow{1}+\underline{x},\ |L|\overrightarrow{1}+\overline{x}\right]$$

for Σ. Notice, however, that the parametric form (9) is a more subtle approximation of Σ having the geometric shape of a zonotope. Therefore, it is better to work with the form (9) itself. Among other advantages of the *p*-solution is that it enables us as well to easily compute an inner estimation of $\square\Sigma$[18,30]:

$$\boldsymbol{x}^{IEH} := \left[-|L|\overrightarrow{1}+\overline{x},\ |L|\overrightarrow{1}+\underline{x}\right]\subseteq\square\Sigma.$$

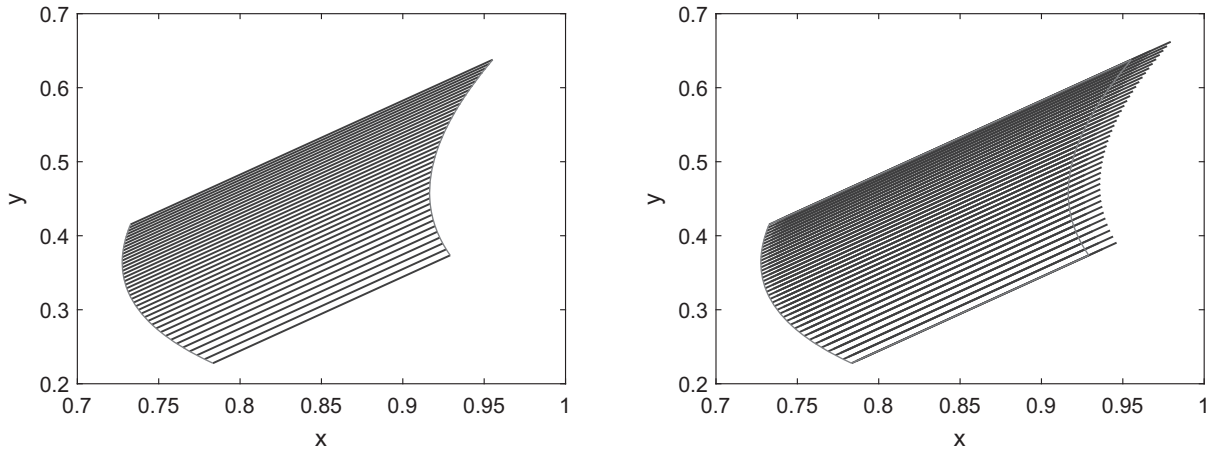

FIGURE 3 Solution set of the interval parametric linear system with nonlinear dependencies (left) and solution set of the transformed system of revised affine forms (right)

After the introduction of the *p*-solution concept by Kolev,[18] there were several follow-up works. Skalna et al.[30] proposed a method based on interval–affine Gauss–Seidel iteration. Kolev presented several approaches, including a direct parametric method[35] (described briefly in Section 2.1), which extended the direct method proposed by Skalna,[19] or a class of iterative methods[36,37] such as the linear iterative method (producing linear *p*-solution) and the quadratic iterative method (producing quadratic *p*-solution). Popova[38] generalized Kolev's direct parametric method for any parametric *AE*-solution set.

Let us consider the system (6). By substituting $y = x - \tilde{x}$, we rewrite the system into an equivalent form

$$\boldsymbol{A}(e)y = \boldsymbol{b}(e) - \boldsymbol{A}(e)\tilde{x}. \tag{10}$$

Before we present our approaches, we will transform the system in a more convenient form first. The preconditioning of the system (10) with $(A^c)^{-1}$ yields the new system

$$\boldsymbol{V}(e)y = \boldsymbol{v}(e), \tag{11}$$

where

$$\begin{aligned}
\boldsymbol{V}(e) &= \sum_{k=1}^{K} V^{(k)}\varepsilon_k + \boldsymbol{V},\\
V^{(k)} &= (A^c)^{-1}A^{(k)}, \quad k = 1, \ldots, K,\\
\boldsymbol{V} &= I + \left|(A^c)^{-1}\right| A^{\Delta}[-1, 1],\\
\boldsymbol{v}(e) &= \sum_{k=1}^{K} v^{(k)}\varepsilon_k + \boldsymbol{v},\\
v^{(k)} &= (A^c)^{-1}(b^{(k)} - A^{(k)}\tilde{x}), \quad k = 0, \ldots, K,\\
\boldsymbol{v} &= (A^c)^{-1}(\boldsymbol{b} - \boldsymbol{A}\tilde{x}).
\end{aligned}$$

2 | SOLUTION METHODS

2.1 | Kolev's parametric direct method

Recently, the parametric direct method (PDM) for computing the parametric solution to systems with affine linear dependencies has been proposed by Kolev.[35] We briefly recall this method here (cf. Skalna[34]). Consider the parametric system of the form (11). The vector $\tilde{x}$ is chosen as the solution of the midpoint system $A^c x = b^c$. Then, affine form matrix $\boldsymbol{V}(e)$ is relaxed into the interval matrix $\boldsymbol{C}$ by using interval arithmetic. This way, we arrive at the (mixed) interval-parametric system

$$\boldsymbol{C}y = \boldsymbol{v}(e)$$

with $v^c = 0$. Because $C^c = I$, the system is easy to solve, but the solution set is overestimated because of the relaxation of the constraint matrix. Suppose $\rho(C^\Delta) < 1$, which is both a sufficient and necessary condition[34] for nonsingularity of all matrices in $\boldsymbol{C}$. Now, the interval hull of the inverses of matrices from $\boldsymbol{C}$ has the form[39]

$$\boldsymbol{H} := \square\{C^{-1} | C \in \boldsymbol{C}\} = [-M + \operatorname{diag}(z), M], \tag{12}$$

where

$$M := (I - C^\Delta)^{-1} \geqslant 0 \tag{13}$$

and $\operatorname{diag}(z)$ is the diagonal matrix with entries

$$z_i := \frac{2M_{ii}^2}{2M_{ii} - 1}, \quad i = 1, \ldots, n.$$

The resulting enclosure of the solution set simply reads

$$\boldsymbol{x}(e) := \tilde{x} + \boldsymbol{H}\boldsymbol{v}(e). \tag{14}$$

The p-solution form of this enclosure is

$$\boldsymbol{x}(e) = Le + \boldsymbol{x},$$

where the kth column of the matrix $L \in \mathbb{R}^{n \times K}$ is $H^c v^{(k)}$ and the interval residual is

$$\boldsymbol{x} = \tilde{x} + \boldsymbol{H}\boldsymbol{v} + H^\Delta \sum_{k=1}^{K} |v^{(k)}|[-1, 1].$$

2.2 | Generalized expansion method

The *expansion method* (EM) was originally proposed by Degrauwe et al.[40] as a method for solving the parametric linear equations (6). Its name comes from the fact that the method utilizes the Neumann series to expand an interval parametric matrix inverse. Degrauwe et al.[40] also presented the first- and second-order EM formulae.

Skalna[34] proposed a modified version of the EM method, which employed revised affine forms and involved the residual correction of the right-hand side vector. The method, called the *generalized EM* (GEM), showed several advantages over the EM method. First of all, thanks to the use of revised affine forms, an arbitrary-order expansion could be obtained and the produced solutions have a parametric form. Additionally, the residual correction of the right-hand side vector significantly improved the accuracy of the results. Below, we present the preliminaries of the GEM method and show novel properties.

Any matrix $V(e) \in \boldsymbol{V}(e)$ has the form $V(e) = I + \sum_{k=1}^{K} V^{(k)}\varepsilon_k + V$ for some $e \in [-1, 1]^K$ and $V \in V^\Delta[-1, 1]$. Denote $V_0(e) := -\sum_{k=1}^{K} V^{(k)}\varepsilon_k - V$. If $\rho(V_0(e)) < 1$, then the inverse of $V(e)$ has the expression by Neumann series

$$\begin{aligned} V(e)^{-1} &= I + \sum_{i=1}^{\infty} V_0(e)^i \\ &= I + \sum_{i=1}^{m} V_0(e)^i + \left(\sum_{i=0}^{\infty} V_0(e)^i\right) V_0(e)^{m+1} \\ &= I + \sum_{i=1}^{m} V_0(e)^i + V(e)^{-1} V_0(e)^{m+1}. \end{aligned} \tag{15}$$

In the above expression, we enclose the right-hand occurrence of $V(e)^{-1}$ by the interval matrix $\boldsymbol{H}$ from (12). Thus, we arrive at the enclosure $\boldsymbol{V}(e)^{-1} \subseteq \boldsymbol{G}(e)$, where

$$\boldsymbol{G}(e) := I + \sum_{i=1}^{m} \left(-\sum_{k=1}^{K} V^{(k)}\varepsilon_k + V^\Delta[-1, 1]\right)^i + \boldsymbol{H}\left(-\sum_{k=1}^{K} V^{(k)}\varepsilon_k + V^\Delta[-1, 1]\right)^{m+1}.$$

We call it an expansion of the inverse matrix of order m. Put

$$\boldsymbol{g}(e) := \boldsymbol{G}(e)\boldsymbol{v}(e).$$

Then, the p-solution of the system (6) is given by

$$\boldsymbol{x}(e) := \tilde{x} + \boldsymbol{g}(e). \tag{16}$$

Equivalently, we can express it as

$$\boldsymbol{x}(e) = Le + \boldsymbol{x},$$

where $L = \left(g^{(1)} | \dots | g^{(K)}\right) \in \mathbb{R}^{n \times K}$ and $\boldsymbol{x} = \tilde{x} + \boldsymbol{g}$.

Proposition 1 (See the work of Skalna[34])**.**
Let $\boldsymbol{x}(e)$ be defined by (16). Then, it is a p-solution, that is, for every $e \in [-1,1]^K$, the unique solution of (6) is included in $\boldsymbol{x}(e)$.

Proof. Let $e \in [-1,1]^K$. If x is a solution to the system (6), then $u = x - \tilde{x}$ is a solution to the system (10) and, thus, to system (11) as well. Therefore,

$$x \in \tilde{x} + \boldsymbol{V}(e)^{-1}\boldsymbol{v}(e) \subseteq \tilde{x} + \boldsymbol{G}(e)\boldsymbol{v}(e) \subseteq \boldsymbol{x}(e). \qquad \square$$

In the following results, we will use also the notation

$$R := \sum_{k=1}^{K} |V^{(k)}| + V^{\Delta}.$$

Proposition 2. *Consider the system (11) with $\rho(R) < 1$ and the trivial affine multiplication. Let $\boldsymbol{x}'(e)$ be given by formula (14) and $\boldsymbol{x}(e)$ be given by formula (16). Then,*

$$[\boldsymbol{x}(e)] \subseteq [\boldsymbol{x}'(e)]. \tag{17}$$

Notice that the proposition does not claim that $\boldsymbol{x}(e) \subseteq \boldsymbol{x}'(e)$ for every $e \in [-1,1]^K$. We have the inclusion only for the ranges of the affine forms.

Proof. We have

$$\begin{aligned}\boldsymbol{x}'(e) &= \tilde{x} + \boldsymbol{H}\boldsymbol{v}(e) = \tilde{x} + \boldsymbol{H}\left(\sum_{k=1}^{K} v^{(k)}\varepsilon_k + [-1,1]v^{\Delta}\right)\\ &= \tilde{x} + \sum_{k=1}^{K} H^c v^{(k)}\varepsilon_k + [-1,1]\left(|H^c|v^{\Delta} + H^{\Delta}\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right)\right).\end{aligned}$$

Then,

$$[\boldsymbol{x}'(e)] = \tilde{x} + [-1,1]\left(\sum_{k=1}^{K}|H^c v^{(k)}| + |H^c|v^{\Delta} + H^{\Delta}\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right)\right).$$

Because $H^c = \frac{1}{2}\mathrm{diag}(z)$ is nonnegative and diagonal, we have $|H^c v^{(k)}| = H^c|v^{(k)}|$, whence

$$\begin{aligned}[\boldsymbol{x}'(e)] &= \tilde{x} + [-1,1]\left(H^c\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right) + H^{\Delta}\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right)\right)\\ &= \tilde{x} + [-1,1]M\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right).\end{aligned}$$

For the second solution, we have

$$[\boldsymbol{x}(e)] = \tilde{x} + [\boldsymbol{G}(e)\boldsymbol{v}(e)] \subseteq \tilde{x} + [\boldsymbol{G}(e)][\boldsymbol{v}(e)].$$

Using the fact that $\boldsymbol{C} = [I - R, I + R]$, and so the matrix M has the form $M = (I-R)^{-1} = \sum_{i=0}^{\infty} R^i$, we obtain

$$\begin{aligned}[\boldsymbol{G}(e)] &\subseteq I + \textstyle\sum_{i=1}^{m}[-1,1]R^i + \boldsymbol{H}[-1,1]R^{m+1}\\ &\subseteq [-1,1]\left(I + \textstyle\sum_{i=1}^{m} R^i + MR^{m+1}\right)\\ &= [-1,1]\left(I + \textstyle\sum_{i=1}^{\infty} R^i\right)\\ &= [-1,1]M.\end{aligned}$$

Because

$$[\boldsymbol{v}(e)] = [-1,1]\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right),$$

we arrive at

$$[\boldsymbol{x}(e)] \subseteq \tilde{x} + [-1,1]M\left(\sum_{k=1}^{K}|v^{(k)}| + v^{\Delta}\right) = [\boldsymbol{x}'(e)].$$

□

2.3 | Alternative approach

The main deficiency of the GEM method is its time complexity. We propose here a modification that aims to decrease time consumption. Let us notice that the p-solution given by the formula (16)

$$\boldsymbol{x}(e) := \tilde{x} + \boldsymbol{G}(e)\boldsymbol{v}(e)$$

can also be evaluated in an alternative way. Denote

$$\boldsymbol{V}_0(e) := -\sum_{k=1}^{K} V^{(k)}\varepsilon_k + V^{\Delta}[-1,1].$$

Then, the product $\boldsymbol{G}(e)\boldsymbol{v}(e)$ can be expressed also as

$$\boldsymbol{v}(e) + \sum_{i=1}^{m}\boldsymbol{V}_0(e)^i\boldsymbol{v}(e) + \boldsymbol{H}\boldsymbol{V}_0(e)^{m+1}\boldsymbol{v}(e).$$

We can save the computational time if, instead of computing matrix powers, we will use matrix-vector products only, that is, we will calculate the particular terms successively as follows:

$$\boldsymbol{v}(e), \quad \boldsymbol{V}_0(e)\boldsymbol{v}(e), \quad \boldsymbol{V}_0(e)(\boldsymbol{V}_0(e)\boldsymbol{v}(e)), \quad \ldots, \quad \boldsymbol{V}_0(e)(\boldsymbol{V}_0(e)^{m-1}\boldsymbol{v}(e)), \quad \boldsymbol{H}(\boldsymbol{V}_0(e)^{m+1}\boldsymbol{v}(e)).$$

3 | KRAWCZYK-TYPE ITERATIONS

Consider a parametric system in the form of (11), in which $V^c = I$. It naturally yields Krawczyk-type iterations

$$\boldsymbol{y}(e) \mapsto \boldsymbol{v}(e) + (I - \boldsymbol{V}(e))\boldsymbol{y}(e). \tag{18}$$

Suppose that $\boldsymbol{y}^0(e)$ is an initial p-solution, that is, for every $e \in [-1,1]^K$, the solution of (11) is contained in $\boldsymbol{y}^0(e)$. Then,

$$\boldsymbol{y}(e) := \boldsymbol{v}(e) + (I - \boldsymbol{V}(e))\boldsymbol{y}^0(e)$$

is again a p-solution. In other words, solution enclosing property is satisfied during the iterations of (18). By continuing the iterations, we can possibly reduce a large initial enclosure to a smaller one. In the following, we state a convergence property.

Proposition 3. *If $\rho(R) < 1$, then the iterations (18) converge to a unique fixed point for each initial $\boldsymbol{y}^0(e)$.*

Proof. In the work of Skalna et al.,[30] it was shown (for a more general class of iterations) that, by using the trivial affine multiplication, the Krawczyk-type iteration (18) can be expressed as real linear iterations of the form

$$\tilde{y} \mapsto \tilde{y} + \tilde{W}_y\tilde{y} \subseteq \tilde{y} + [-W, W]\tilde{y},$$

where

$$W := \begin{pmatrix} |V^c| & & |V^{(1)}| \\ & \ddots & \vdots \\ & |V^c| & |V^{(K)}| \\ & & |V^c| \\ R \quad \ldots & R & V^{\Delta} \quad R + |V^c| \end{pmatrix}, \quad \tilde{y} := \begin{pmatrix} y^{(1)} \\ \vdots \\ y^{(K)} \\ y^c \\ y^{\Delta} \end{pmatrix}.$$

The iterations then converge provided $\rho(W) < 1$.

In our case, $V^c = 0$, whence W takes the form

$$W := \begin{pmatrix} 0 & & |V^{(1)}| \\ & \ddots & \vdots \\ & 0 & |V^{(K)}| \\ & & 0 \\ R \ \ldots & R & V^{\Delta} \quad R \end{pmatrix}.$$

We claim that $\rho(R) = \rho(W)$, and this claim obviously finishes the proof. To show the claim, let $u \geq 0, u \neq 0$ be the Perron vector corresponding to R^T, that is, $R^T u = \rho(R)u$. Then,

$$\begin{pmatrix} 0 & & |V^{(1)}| & \\ & \ddots & \vdots & \\ & & 0 \; |V^{(K)}| & \\ & & 0 & \\ R & \dots \; R & V^\Delta & R \end{pmatrix}^T \begin{pmatrix} u \\ \vdots \\ u \end{pmatrix} = \begin{pmatrix} R^T u \\ \vdots \\ R^T u \end{pmatrix} = \rho(R) \begin{pmatrix} u \\ \vdots \\ u \end{pmatrix}.$$

Therefore, $(u^T, \dots, u^T)^T$ is the Perron vector of W. □

We now show that the limit of the Krawczyk iterations can be explicitly determined.

Proposition 4. *The fixed point of the iterations (18) is $\boldsymbol{y}(e)$, where*

$$y^c = v^c, \tag{19a}$$

$$y^{(k)} = v^{(k)} - V^{(k)} v^c, \quad k = 1, \dots, K, \tag{19b}$$

$$y^\Delta = M\left(v^\Delta + V^\Delta |v^c| + R \sum_{k=1}^{K} |y^{(k)}| \right). \tag{19c}$$

Proof. The fixed point satisfies

$$\boldsymbol{y}(e) = \boldsymbol{v}(e) + (I - \boldsymbol{V}(e))\boldsymbol{y}(e).$$

By using the trivial affine multiplication, we have

$$\begin{aligned} (I - \boldsymbol{V}(e))\boldsymbol{y}(e) &= \left(-\sum_{k=1}^{K} V^{(k)} \varepsilon_k - \boldsymbol{V} \right) \left(\sum_{k=1}^{K} y^{(k)} \varepsilon_k + \boldsymbol{y} \right) \\ &= -\sum_{k=1}^{K} V^{(k)} y^c \varepsilon_k + [-1, 1] \left(V^\Delta |y^c| + R \left(\sum_{k=1}^{K} |y^{(k)}| + y^\Delta \right) \right), \end{aligned}$$

whence

$$\sum_{k=1}^{K} y^{(k)} \varepsilon_k + \boldsymbol{y} = \sum_{k=1}^{K} v^{(k)} \varepsilon_k + \boldsymbol{v} - \sum_{k=1}^{K} V^{(k)} y^c \varepsilon_k + [-1, 1] \left(V^\Delta |y^c| + R \left(\sum_{k=1}^{K} |y^{(k)}| + y^\Delta \right) \right),$$

Comparing the midpoint values, we obtain (19a). Comparing the coefficients by ε_k, we get (19b). Comparing the radii of the residual intervals, we have

$$y^\Delta = v^\Delta + V^\Delta |v^c| + R \left(\sum_{k=1}^{K} |y^{(k)}| + y^\Delta \right),$$

from which

$$(I - R) y^\Delta = v^\Delta + V^\Delta |v^c| + R \sum_{k=1}^{K} |y^{(k)}|.$$

Recall that the matrix R is the same as C^Δ from (13). Thus, $M = (I - R)^{-1}$ and (19c) follows. □

Notice that this formula was derived for a general system in the form of (11) with $V^c = I$. The residual correction was not utilized. Provided we employ the residual correction as well, then $v^c = 0$ and the formula (19) is further simplified to

$$y^c = 0, \tag{20a}$$

$$y^{(k)} = v^{(k)}, \quad k = 1, \dots, K, \tag{20b}$$

$$y^\Delta = M\left(v^\Delta + R \sum\nolimits_{k=1}^{K} |v^{(k)}| \right). \tag{20c}$$

Now, we show an interesting relation to the parametric Bauer–Skeel (PBS) enclosure[16] or the direct method (DM) enclosure.[19] The PBS enclosure was derived for the system (11) with residual correction. It also assumes linear dependencies already in the original system so that $V^\Delta = 0$ and $v^\Delta = 0$.

Proposition 5. *The iterations (18) converge to a p-solution, whose interval hull is the same as the PBS enclosure.*

Proof. The Bauer–Skeel method assumes the same condition $\rho(R) < 1$. The enclosure has the form

$$\tilde{x} + [-1,1]M\sum_{k=1}^{K}|v^{(k)}|,$$

where $\tilde{x} = (A^c)^{-1}b^c$. The limiting Krawczyk enclosure has the form

$$\tilde{x} + \sum_{k=1}^{K} v^{(k)}\varepsilon_k + [-1,1]MR\sum_{k=1}^{K}|v^{(k)}|.$$

Its interval hull is

$$\tilde{x} + [-1,1]\left(\sum_{k=1}^{K}|v^{(k)}| + MR\sum_{k=1}^{K}|v^{(k)}|\right).$$

The midpoints of both enclosures are the same, so we compare their radii. The radius of the latter is

$$\begin{aligned}\sum_{k=1}^{K}|v^{(k)}| + MR\sum_{k=1}^{K}|v^{(k)}| &= \sum_{k=1}^{K}|v^{(k)}| + \sum_{i=0}^{\infty}R^iR\sum_{k=1}^{K}|v^{(k)}| \\ &= M\sum_{k=1}^{K}|v^{(k)}|,\end{aligned}$$

using $M = \sum_{i=0}^{\infty}R^i$. We see that radii are the same, too. □

This result, however, does not justify to suppress the Krawczyk iteration method. Compared with the DM enclosure, it still have two major advantages. First, we have an enclosing solution in a parametric form. Therefore, despite the fact that it has the same span in each coordinate, the enclosing set itself can be much smaller. Second, the result holds for the most trivial implementation of affine multiplication. Employing more advanced techniques, we may hope for tighter resulting enclosures. Indeed, the use of the Chebyshev minimum-error multiplication of revised affine forms significantly improves enclosures produced by the Krawczyk iteration method (see the next section).

4 | NUMERICAL EXPERIMENTS

In this section, we compare the PDM, EM, GEM, the alternative version of GEM (GEMA; the order of the EMs is given in parentheses, for example, GEM(2) denotes the second-order GEM method), and Krawczyk iteration (KRI) in terms of speed and accuracy of produced enclosures. The KRI uses the PBS bounds as a starting point $\boldsymbol{y}^0$, and the iteration process terminates when there is no significant improvement of the bounds, that is, when the distance

$$d\left(\boldsymbol{y}^{i+1},\boldsymbol{y}^i\right) = \max_j\left\{\left|\underline{y}_j^{i+1} - \underline{y}_j^i\right|, \left|\overline{y}_j^{i+1} - \overline{y}_j^i\right|\right\} \leqslant \epsilon,$$

where ϵ is a prescribed threshold. In our experiments, the threshold is $\epsilon = 10^{-8}$.

The test examples include square interval parametric linear systems of equations having affine-linear dependencies, nonlinear dependencies, an overdetermined system with affine-linear dependencies, and systems arising from practical problems of structural and electrical engineering. We use the Chebyshev minimum-error multiplication in computation on revised affine forms.

The quality of outer interval enclosures will can be assessed by using the following measures. Given two intervals $\boldsymbol{x}, \boldsymbol{y}$ such that $\boldsymbol{x} \subseteq \boldsymbol{y}$

– *standard overestimation measure*

$$O_w(\boldsymbol{x},\boldsymbol{y}) = \left(1 - \frac{x^\Delta}{y^\Delta}\right)100\%, \tag{21}$$

TABLE 1 Comparison of outer enclosures for Example 4

GEM(2)
[0.6969257716047762, 1.333333333333575]
[0.9999999999998003, 1.000000000000143]
GEMA(2)
[0.6956768088491990, 1.333333333333575]
[0.9999999999998003, 1.000000000000143]
KRI
[0.6666666666664801, 1.333333333333577]
[0.9999999999997983, 1.000000000000145]
PDM
[0.6666666666664812, 1.333333333333575]
[0.9999999999998003, 1.000000000000143]
Result from the work of Popova[41]
[0.6666666666661077, 1.333333333338497]
[0.9999999999995374, 1.000000000004625]

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

– *sharpness measure*[12]

$$O_s(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} 1, & y^\Delta = 0, \\ 0, & \boldsymbol{x} = \emptyset, \\ \frac{x^\Delta}{y^\Delta}, & \text{otherwise.} \end{cases} \tag{22}$$

For interval vectors, we take the mean of O_w and the minimal and maximal values of O_s, over all entries. If the hull solution is available, then it can be used to assess the quality of outer enclosures. However, as already mentioned, the problem of computing the hull solution is NP-hard. That is why, outer enclosure should be compared with inner estimate of the hull solution, which is provided with a little additional effort by methods that are based strictly on revised affine forms. The obtained results are compared with the results from the literature, whenever the latter are available.

Example 4. Consider the interval parametric linear system[41]

$$\left\{ \begin{pmatrix} p_1 & p_1 \\ p_1 & p_1 + 0.01 \end{pmatrix} x = \begin{pmatrix} p_2 \\ p_2 + 0.01 \end{pmatrix}, \quad p_1 \in [0.9, 1.1], \quad p_2 \in [1.9, 2.1] \right\}. \tag{23}$$

The obtained results (given with 16-digit mantissa) and the result from the work of Popova[41] are presented in Table 1. As we can see, the GEM, GEMA, and KRI methods produced guaranteed tight enclosures for the exact solution, which is $([8/11, 4/3], [1, 1])^T$. The second-order GEM method turned out to be the best in this case. The second-order GEMA produced slightly worse results, whereas KRI produced worse results than GEM and GEMA, however still better than the results from the work of Popova.[41]

Example 5. Consider the interval parametric linear system of equations defined by

$$\begin{pmatrix} p_1 + p_6 & -p_6 & 0.0 & 0.0 & 0.0 \\ -p_6 & p_2 + p_6 + p_7 & -p_7 & 0.0 & 0.0 \\ 0.0 & -p_7 & p_3 + p_7 + p_8 & -p_8 & 0.0 \\ 0.0 & 0.0 & -p_8 & p_4 + p_8 + p_9 & -p_9 \\ 0.0 & 0.0 & 0.0 & -p_9 & p_5 + p_9 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 10 \\ 0 \\ 10 \\ 0 \\ 0 \end{pmatrix},$$

having affine-linear dependencies. The parameters $p_k \in [1 - \delta, 1 + \delta]$, $k = 1, \ldots, 9$, where $\delta > 0$. This system emerges when Okumura's problem of a linear resistive network is solved.[11,34,42] Table 2 shows the sharpness of the obtained enclosures for various tolerances. The maximal inner bounds (produced by KRI) are used to compute the values of O_s measure.

TABLE 2 Sharpness (O_s) of outer enclosures for Example 5

δ	GEM-A(1) min–max	GEM(3) min–max	GEMA(3) min–max	KRI min–max	PDM min–max	EM(1) min–max	EM(2) min–max
0.01	0.97–0.98	0.97–0.98	0.97–0.98	0.97–0.98	0.95–0.97	0.93–0.96	0.95–0.97
0.05	0.82–0.89	0.82–0.89	0.82–0.89	0.82–0.89	0.77–0.85	0.70–0.79	0.77–0.84
0.1	0.63–0.76	0.64–0.77	0.63–0.76	0.64–0.77	0.56–0.70	0.46–0.62	0.54–0.68
0.15	0.43–0.63	0.44–0.64	0.44–0.63	0.44–0.64	0.36–0.56	0.28–0.46	0.33–0.52
0.2	0.22–0.48	0.23–0.50	0.23–0.49	0.23–0.50	0.18–0.41	0.13–0.32	0.15–0.37
0.25	0.01–0.32	0.01–0.33	0.01–0.33	0.01–0.34	0.00–0.26	0.00–0.19	0.00–0.22

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method; EM = expansion method.

TABLE 3 Computational times (in seconds) for Example 5

δ	GEMA(1)	GEM(3)	GEMA(3)	KRI	PDM	EM(1)	EM(2)
0.01	0.01	0.035	0.014	0.016(6)	0.006	0.002	0.002
0.05	0.01	0.035	0.014	0.021(9)	0.006	0.002	0.002
0.1	0.01	0.035	0.014	0.029(13)	0.006	0.002	0.002
0.15	0.01	0.035	0.014	0.038(18)	0.006	0.002	0.002
0.2	0.01	0.035	0.014	0.052(25)	0.006	0.002	0.002
0.25	0.01	0.035	0.014	0.068(35)	0.006	0.002	0.002

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method; EM = expansion method.

TABLE 4 Enclosures obtained using GEM(3) and KRI methods for Example 5 ($\delta = 0.1$)

x	GEM(3)		GEMA(3)		KRI	
	outer	inner	outer	inner	outer	inner
x_1	[6.301, 8.004]	[6.503, 7.802]	[6.290, 8.005]	[6.500, 7.795]	[6.302, 8.004]	[6.498, 7.808]
x_2	[3.489, 4.946]	[3.678, 4.757]	[3.479, 4.950]	[3.680, 4.749]	[3.487, 4.949]	[3.678, 4.758]
x_3	[4.811, 6.206]	[5.001, 6.015]	[4.799, 6.207]	[5.000, 6.007]	[4.810, 6.207]	[4.998, 6.018]
x_4	[1.694, 2.710]	[1.845, 2.559]	[1.688, 2.714]	[1.847, 2.555]	[1.692, 2.713]	[1.845, 2.560]
x_5	[0.732, 1.466]	[0.867, 1.332]	[0.729, 1.469]	[0.867, 1.330]	[0.732, 1.467]	[0.864, 1.334]

Note. GEM = generalized expansion method; KRI = Krawczyk iteration.

As can be seen, the GEM(3) and KRI methods produced the sharpest enclosures. However, the results of GEMA(3) do not differ much from the GEM(3) and KRI results, whereas GEMA(3) is much faster than GEM(3) and KRI (see Table 3). It can be seen as well that the first-order GEM is better than the second-order expansion method, which confirms the great usefulness of our modifications.

If we take a closer look at the results of the GEM(3), GEMA(3), and KRI methods for $\delta = 0.1$ (Table 4), we can see that GEM(3) produced slightly better outer enclosures (except for x_1) than GEMA(3) and KRI, whereas KRI produced better inner bounds.

Example 6. Consider the interval parametric linear system of equations with affine-linear dependencies and data defined by[43]

$$\begin{aligned} A^{(k)} &= (k+1)L, \quad k = 0, \ldots, K, \\ b^{(k)} &= 1, \end{aligned} \tag{24}$$

where $L \in \mathbb{R}^n$ is the `Lehmer` matrix defined, for $i, j = 1, \ldots, n$, by

$$L(i,j) = \begin{cases} i/j, & i \leqslant j, \\ j/i, & i > j. \end{cases}$$

The parameters come from the interval domains as follows: $p_k \in [1-\delta, 1+\delta], k = 1, \ldots, K$. We solve the system for various n, K, and δ. The sharpness of outer enclosures and the computational times are presented in Tables 5 and 6, respectively.

TABLE 5 Sharpness of outer interval enclosures for Example 6

n	K	δ	GEM(3) min–max	GEMA(3) min–max	KRI min–max	DM min–max
20	10	0.05	0.96–0.96	0.96–0.96	0.96–0.96	0.93–0.93
20	20	0.05	0.96–0.96	0.96–0.96	0.96–0.96	0.93–0.93
20	10	0.1	0.91–0.91	0.91–0.91	0.92–0.92	0.86–0.86
20	20	0.1	0.91–0.91	0.91–0.91	0.91–0.91	0.86–0.86
20	10	0.3	0.71–0.71	0.73–0.73	0.74–0.74	0.61–0.61
20	20	0.3	0.70–0.70	0.72–0.72	0.73–0.73	0.60–0.60
100	10	0.05	0.96–0.96	0.96–0.96	0.96–0.96	0.93–0.93
100	20	0.05	0.96–0.96	0.96–0.96	0.96–0.96	0.93–0.93
100	10	0.1	0.91–0.91	0.91–0.91	0.92–0.92	0.86–0.86
100	20	0.1	0.91–0.91	0.91–0.91	0.91–0.91	0.86–0.86
100	10	0.3	0.71–0.71	0.73–0.73	0.74–0.74	0.61–0.61
100	20	0.3	0.70–0.70	0.72–0.72	0.73–0.73	0.60–0.60

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; DM = direct method.

TABLE 6 Computational times for Example 6 (the number of iterations taken by the KRI method is given in parentheses)

n	K	δ	GEM(3)	GEM-A(3)	KRI	PDM
20	10	0.05	2.33	0.48	1.04(6)	0.35
20	20	0.05	3.49	0.75	1.59(5)	0.54
20	10	0.1	2.33	0.48	1.06(7)	0.35
20	20	0.1	3.49	0.75	1.66(7)	0.59
20	10	0.3	2.33	0.48	1.23(12)	0.35
20	20	0.3	3.49	0.75	1.96(12)	0.56
100	10	0.05	284.88	41.031	118.75(6)	35.87
100	20	0.05	419.7	60.64	174.50(5)	57.94
100	10	0.1	284.88	41.031	116.71(7)	35.87
100	20	0.1	419.7	60.64	167.04(7)	54.911
100	10	0.3	284.88	41.031	114.23(12)	35.87
100	20	0.3	419.7	60.64	181.48(12)	63.333

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

From the obtained results, it can be seen that, for $\delta = 0.05$ and $\delta = 0.1$, the GEM(3), GEMA(3), and KRI enclosures are very similar. For $\delta = 0.3$, the KRI produced slightly better enclosure than GEMA(3); however, the latter was significantly faster. Therefore, in this case, GEMA(3) turned out to be generally the most efficient among the proposed approaches. Notice as well that the computational times of GEMA(3) and PDM are very close, and in the last case, GEMA(3) outperformed PDM.

Example 7. This example shows that the proposed methods can be successfully used to solve rectangular systems of linear equations. Consider the overdetermined interval parametric linear system[44,45] defined by

$$\begin{pmatrix} p_1 & p_1+1 & p_1+2 & p_1 \\ p_1 & p_1+2 & p_1+3 & p_1+1 \\ p_1+1 & p_1+2 & p_1+3 & p_1+2 \\ p_1+2 & p_1+3 & p_1+4 & p_1+3 \\ p_1+3 & p_1+4 & p_1+5 & p_1+5 \\ p_1+5 & p_1+5 & p_1+6 & p_1+7 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} p_2 \\ 1 \\ -2 \\ -3 \\ 0 \\ 0 \end{pmatrix}.$$

The nominal value of parameter $p_1 = 1$. The matrix of the system is a full rank matrix ($\text{rank}(A(p)) = 4$) and the right-hand side vector $b(p)$ is constructed so that, for every p_1 and $p_2 = 3$, the exact solution of the normal equation

TABLE 7 Comparison of enclosures for Example 7

GEM(3)
[0.9999999999998332, 1.000000000000171]
[− 1.000000000001248, − 0.9999999999987297]
[0.9999999999991258, 1.00000000000086]
[− 1.000000000000342, − 0.9999999999996656]
GEMA(3)
[0.9999999999998283, 1.000000000000176]
[− 1.000000000001308, − 0.9999999999986695]
[0.9999999999990755, 1.00000000000091]
[− 1.000000000000351, − 0.9999999999996565]
KRI
[0.9999999998833602, 1.000000000116647]
[− 1.000000003149011, − 0.9999999968509893]
[0.9999999970844601, 1.000000002915569]
[− 1.000000000213859, − 0.9999999997861468]
PDM
[0.9999999999998271, 1.000000000000177]
[− 1.000000000001321, − 0.999999999998656]
[0.9999999999990638, 1.000000000000922]
[− 1.000000000000353, − 0.9999999999996546]

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

system $A(p)^TA(p)x = A(p)^Tb$ is $x = (1, -1, 1, -1)^T$ (see the work of Zielke[46]). We solve the interval parametric linear system with $p_1 \in [0.9, 1.1]$ and $p_2 \in [2.995, 3.005]$. The obtained enclosures are presented in Table 7 (the results are given with 16-digit mantissa).

As we can see, all the methods produced guaranteed sharp enclosures for the parametric solution set. In this case, KRI was slightly outperformed by GEM(3), GEMA(3), and PDM in terms of the sharpness of produced enclosures.

Example 8. This example illustrates that the proposed methods can be directly used to solve parametric systems with nonlinear dependencies. Consider the interval parametric system of linear equations

$$\begin{pmatrix} p_1^2(p_2+p_3)-2 & p_1p_2^2 & p_1^3 & -4 & p_3^3 \\ -p_1^2p_2+4 & p_1^2+p_2^2 & 3+p_3^2 & 3p_3p_4-1 & 0 \\ (p_1-p_3)p_2 & 3 & p_2p_3 & p_1p_2p_5 & 1+p_5 \\ p_4p_5-p_1 & p_2^3+p_4 & p_2p_3 & p_2p_3p_4+p_5^2 & -p_4 \\ p_4^2+1 & -p_4 & p_3^2 & p_2p_3p_4+p_5^2 & -p_2^2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} p_1 \\ p_1^2-p_2p_3 \\ -2p_3 \\ -2 \\ 1 \end{pmatrix},$$

where $p_1 \in 4\boldsymbol{\delta}, p_2 \in 1.8\boldsymbol{\delta}\, p_3 \in 2.5\boldsymbol{\delta}, p_4 \in 2.5\boldsymbol{\delta}, p_5 \in 1.0\boldsymbol{\delta}, \boldsymbol{\delta} = [1-\delta, 1+\delta], \delta > 0$. The results for the values of $\delta = 0.01, 0.03, 0.05$ are displayed in Tables 8-10.

In this example, the KRI method produced the best enclosures; however, they are only slightly better than the GEM(3) enclosures. Whereas GEM(3) was much faster than KRI. GEMA(3) was again much faster than KRI and GEM(3), whereas their enclosures were only slightly worse.

TABLE 8 Sharpness of outer enclosures for Example 8

δ	GEM(3) min–max	GEMA(3) min–max	KRI min–max	PDM min–max
0.01	0.76–0.88	0.76–0.88	0.76–0.88	0.70–0.84
0.03	0.30–0.59	0.29–0.57	0.31–0.59	0.23–0.49
0.05	0.00–0.17	0.00–0.16	0.00–0.19	0.00–0.13

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 9 Computational times (in seconds) and number of iterations taken by KRI (in parentheses) for Example 8

δ	GEM(3)	GEMA(3)	KRI	PDM
0.01	0.031	0.014	0.033(9)	0.006
0.03	0.031	0.014	0.055(18)	0.006
0.05	0.031	0.014	0.081(41)	0.006

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 10 Enclosures obtained using GEM(3) and KRI methods for Example 8

δ	GEM-3		KRI	
	outer	inner	outer	inner
0.01	$[-0.9384, -0.8447]$	$[-0.9274, -0.8557]$	$[-0.9385, -0.8448]$	$[-0.9275, -0.8558]$
	$[-0.7617, -0.5965]$	$[-0.7482, -0.6099]$	$[-0.7618, -0.5965]$	$[-0.7483, -0.6100]$
	$[1.3267, 1.5014]$	$[1.3435, 1.4845]$	$[1.3268, 1.5014]$	$[1.3436, 1.4846]$
	$[-0.6681, -0.5275]$	$[-0.6591, -0.5365]$	$[-0.6681, -0.5275]$	$[-0.6592, -0.5364]$
	$[-1.4613, -1.1600]$	$[-1.4430, -1.1783]$	$[-1.4615, -1.1601]$	$[-1.4432, -1.1784]$
0.03	$[-1.0946, -0.7033]$	$[-0.9519, -0.8460]$	$[-1.0935, -0.7116]$	$[-0.9611, -0.8441]$
	$[-0.9952, -0.3658]$	$[-0.8225, -0.5385]$	$[-0.9936, -0.3704]$	$[-0.8300, -0.5340]$
	$[1.0858, 1.7840]$	$[1.3041, 1.5657]$	$[1.0930, 1.7803]$	$[1.2980, 1.5753]$
	$[-0.8700, -0.3541]$	$[-0.7528, -0.4714]$	$[-0.8654, -0.3563]$	$[-0.7572, -0.4644]$
	$[-1.8677, -0.7786]$	$[-1.6351, -1.0112]$	$[-1.8714, -0.7884]$	$[-1.6490, -1.0109]$

Note. GEM = generalized expansion method; KRI = Krawczyk iteration.

TABLE 11 Sharpness of outer enclosures for Example 9

δ	GEM(3) min–max	GEMA(3) min–max	KRI min–max	PDM min–max
0.005	0.94–0.96	0.94–0.96	0.94–0.96	0.93–0.95
0.01	0.88–0.92	0.88–0.92	0.88–0.92	0.86–0.89
0.05	0.42–0.57	0.41–0.56	0.41–0.57	0.36–0.50
0.1	0.00–0.09	0.00–0.08	0.00–0.09	0.00–0.07

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

Example 9. Consider the interval parametric linear system defined by

$$\begin{pmatrix} 1/p_1 - 2 & \sqrt{p_1 p_2^2} & p_1^3 & -4 \\ -p_1^2 p_2 + 4 & 1/\left(p_1^2 + p_2^2\right) & \sqrt{3 + p_3} & 3p_3 p_4 - 1 \\ 3 & (p_1 - p_3)p_2 & \sqrt{p_2 p_3} & p_1 p_2 p_5 \\ p_4 p_5 - p_1 & (2p_4 - p_3)^2 p_2^2 & p_2 p_3 & p_2 p_3 p_4 + p_5^2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} p_1 \\ p_1^2 - p_2 p_3 \\ -2p_3 \\ -2 \end{pmatrix},$$

where $p_1 \in 1.2\boldsymbol{\delta}$, $p_2 \in 0.8\boldsymbol{\delta}$, $p_3 \in 0.51\boldsymbol{\delta}$, $p_4 \in 2.51\boldsymbol{\delta}$, $p_5 \in 1.01\boldsymbol{\delta}$, $\boldsymbol{\delta} = [1 - \delta, 1 + \delta]$. The results for $\delta = 0.005, 0.01, 0.05, 0.1$ are presented in Tables 11–13. As we can see, the GEM(3), GEMA(3), and KRI methods produced very similar enclosures, but GEMA(3) was the least time consuming.

Example 10. All the above described methods can be easily extended to solve interval parametric linear systems with multiple right-hand sides.[43] Consider the following interval parametric system of linear equations with multiple right-hand sides[43]:

$$\left\{ \begin{pmatrix} 2p_1 & p_2 \\ -p_2 & 2p_1 \end{pmatrix} \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} = \begin{pmatrix} 0 & p_3 \\ p_3 & 0 \end{pmatrix},\ p_1, p_2, p_3 \in [0.9, 1.1] \right\}.$$

TABLE 12 Computational times (in seconds) and number of iterations taken by KRI (in parentheses) for Example 9

δ	GEM	GEMA	KRI	PDM
0.005	0.017	0.008	0.018(5)	0.004
0.01	0.017	0.008	0.021(7)	0.004
0.05	0.017	0.008	0.027(14)	0.004
0.1	0.017	0.008	0.042(31)	0.004

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 13 Enclosures obtained using the GEM(3) and KRI methods for Example 9

δ	GEM(3)		KRI	
	outer	inner	outer	inner
0.03	$[-0.6815, -0.6054]$	$[-0.6694, -0.6175]$	$[-0.6815, -0.6055]$	$[-0.6693, -0.6177]$
	$[-0.2143, -0.1345]$	$[-0.2040, -0.1447]$	$[-0.2143, -0.1345]$	$[-0.2041, -0.1447]$
	$[1.0171, 1.1537]$	$[1.0411, 1.1296]$	$[1.0168, 1.1538]$	$[1.0411, 1.1295]$
	$[0.2704, 0.3745]$	$[0.2850, 0.3599]$	$[0.2705, 0.3744]$	$[0.2852, 0.3597]$
0.05	$[-0.7163, -0.5704]$	$[-0.6774, -0.6092]$	$[-0.7165, -0.5714]$	$[-0.6775, -0.6103]$
	$[-0.2541, -0.1035]$	$[-0.2207, -0.1369]$	$[-0.2540, -0.1040]$	$[-0.2215, -0.1365]$
	$[0.9552, 1.2201]$	$[1.0325, 1.1428]$	$[0.9539, 1.2206]$	$[1.0320, 1.1425]$
	$[0.2240, 0.4207]$	$[0.2709, 0.3738]$	$[0.2247, 0.4204]$	$[0.2717, 0.3734]$

Note. GEM = generalized expansion method; KRI = Krawczyk iteration.

TABLE 14 Enclosures and computational times (in seconds) for Example 10

outer		inner	
GEM(3) (CPU: 0.003)			
$[-0.27321, -0.13149]$	$[0.31211, 0.48950]$	$[-0.26007, -0.14464]$	$[0.32949, 0.47211]$
$[0.3121, 0.4895]$	$[0.1315, 0.2732]$	$[0.3295, 0.4721]$	$[0.1446, 0.2601]$
GEMA(3) (CPU: 0.005)			
$[-0.27368, -0.13068]$	$[0.31137, 0.49008]$	$[-0.25906, -0.14530]$	$[0.32993, 0.47151]$
$[0.3114, 0.4901]$	$[0.1307, 0.2737]$	$[0.3299, 0.4715]$	$[0.1453, 0.2591]$
KRI (CPU: 0.014(11))			
$[-0.27353, -0.13113]$	$[0.31177, 0.48966]$	$[-0.25952, -0.14513]$	$[0.32952, 0.47192]$
$[0.3118, 0.4897]$	$[0.1311, 0.2735]$	$[0.3295, 0.4719]$	$[0.1451, 0.2595]$
PDM (CPU: 0.002)			
$[-0.27964, -0.12036]$	$[0.30403, 0.49597]$	$[-0.24984, -0.15016]$	$[0.33411, 0.46589]$
$[0.3040, 0.4960]$	$[0.1204, 0.2796]$	$[0.3341, 0.4659]$	$[0.1502, 0.2498]$
Best result from Dehghani-Madiseh et al.[43]			
$[-0.27970, -0.12720]$	$[0.30750, 0.49600]$		
$[0.3075, 0.4960]$	$[0.1272, 0.2797]$		

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

Table 14 shows the enclosures obtained by using the considered methods and the enclosure obtained by taking the intersection of all results from the work of Dehghani-Madiseh et al.[43] As can be seen, for three components, the GEM(3) produced tighter bounds than all methods from the work of Dehghani-Madiseh et al.[43]

Example 11. Consider the following interval parametric linear system with multiple right-hand sides[43]:

$$\left\{ \begin{pmatrix} 3p_1 & p_2+p_3 \\ -1 & 5p_1 \end{pmatrix} \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} = \begin{pmatrix} p_3 & p_4 \\ 0 & p_3 \end{pmatrix},\ p_1, p_3 \in [2, 2.5], p_2 \in [1, 2], p_4 \in [2, 3] \right\}.$$

From the results presented in Table 15, it can be seen that GEM(3) produced the best result out of the considered methods. It is also better than the results of all the methods from the work of Dehghani-Madiseh et al.[43] for four components.

TABLE 15 Enclosures and computational times (in seconds) for Example 11

outer		inner	
GEM(3) (CPU: 0.003)			
[0.24619, 0.39419]	[0.10717, 0.38180]	[0.25719, 0.38320]	[0.14297, 0.34599]
[0.0187, 0.0394]	[0.1656, 0.2840]	[0.0207, 0.0374]	[0.1753, 0.2744]
GEMA(3) (CPU: 0.004)			
[0.24581, 0.39434]	[0.10703, 0.38236]	[0.25733, 0.38281]	[0.14470, 0.34469]
[0.0186, 0.0394]	[0.1653, 0.2840]	[0.0208, 0.0372]	[0.1755, 0.2738]
KRI (CPU: 0.008(11))			
[0.24606, 0.39420]	[0.10710, 0.38175]	[0.25718, 0.38307]	[0.14413, 0.34473]
[0.0187, 0.0394]	[0.1656, 0.2840]	[0.0207, 0.0373]	[0.1755, 0.2741]
PDM (CPU: 0.001)			
[0.23923, 0.39606]	[0.10249, 0.39163]	[0.25794, 0.37736]	[0.15092, 0.34320]
[0.0170, 0.0395]	[0.1594, 0.2845]	[0.0208, 0.0357]	[0.1751, 0.2689]
Enclosures from Dehghani-Madiseh et al.[43]			
[0.24960, 0.39610]	[0.10670, 0.39170]		
[0.0204, 0.0394]	[0.1739, 0.2835]		

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

4.1 | Practical applications

The case studies presented in this section illustrate practical applications of the methods described in the previous sections. We consider small but realistic problems coming from structural and electrical engineering.

Example 12. Consider a simple one-bay structural steel frame, shown in Figure 4, which was initially analyzed by Corliss et al.[47] By applying conventional methods for frame structures analysis, the following parametric linear system is obtained[23,47]:

$$\begin{pmatrix}
\frac{A_bE_b}{L_b}+\frac{12E_cI_c}{L_c^3} & 0 & \frac{6E_cI_c}{L_c^2} & 0 & 0 & -\frac{A_bE_b}{L_b} & 0 & 0 \\
0 & \frac{A_cE_c}{L_c}+\frac{12E_bI_b}{L_b^3} & 0 & \frac{6E_bI_b}{L_b^2} & \frac{6E_bI_b}{L_b^2} & 0 & -\frac{12E_bI_b}{L_b^3} & 0 \\
\frac{6E_cI_c}{L_c^2} & 0 & \alpha+\frac{4E_cI_c}{L_c} & -\alpha & 0 & 0 & 0 & 0 \\
0 & \frac{6E_bI_b}{L_b^2} & -\alpha & \alpha+\frac{4E_bI_b}{L_b} & \frac{2E_bI_b}{L_b} & 0 & -\frac{6E_bI_b}{L_b^2} & -\alpha \\
0 & \frac{6E_bI_b}{L_b^2} & 0 & \frac{2E_bI_b}{L_b} & \alpha+\frac{4E_cI_c}{L_c} & 0 & -\frac{6E_bI_b}{L_b^2} & 0 \\
-\frac{A_bE_b}{L_b} & 0 & 0 & 0 & 0 & \frac{A_bE_b}{L_b}+\frac{12E_cI_c}{L_c^3} & 0 & \frac{6E_cI_c}{L_c^2} \\
0 & -\frac{12E_bI_b}{L_b^3} & 0 & -\frac{6E_bI_b}{L_b^2} & -\frac{6E_bI_b}{L_b^2} & 0 & \frac{A_cE_c}{L_c}+\frac{12E_bI_b}{L_b^3} & -\frac{6E_bI_b}{L_b^2} \\
0 & 0 & 0 & 0 & 0 & \frac{6E_cI_c}{L_c^2} & -\frac{6E_bI_b}{L_b^2} & \alpha+\frac{4E_cI_c}{L_c}
\end{pmatrix}
\begin{pmatrix} d2_x \\ d2_y \\ r2_z \\ r5_z \\ r6_z \\ d3_x \\ d3_y \\ r3_z \end{pmatrix}
=
\begin{pmatrix} H \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{25}$$

As can be seen, the elements of the system (25) are rational functions of Young modulus E_b, E_c; the second moment of area I_b, I_c; the cross-sectional area A_b, A_c; length L_b, L_c; and joint stiffness α. The right-hand side vector depends only on the horizontal force H. In the work of Corliss et al.,[47] all the parameters, except the lengths, were assumed

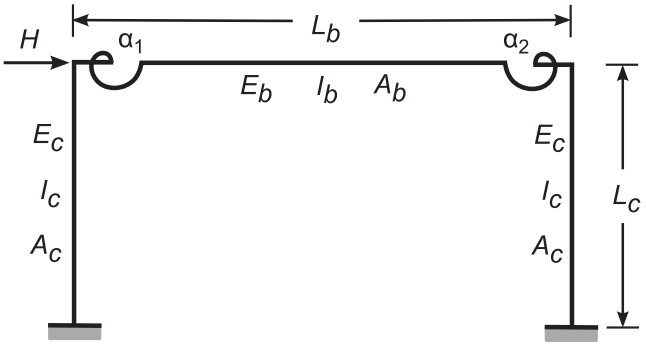

FIGURE 4 One-bay structural steel frame[23,47] (Example 12)

TABLE 16 Parameters of one-bay structural steel frame: nominal values and worst-case uncertainties (Example 12)

Parameter	Nominal value	Uncertainty
E_b, E_c	$29 \cdot 10^6$ lbs/in^2	$\pm 348 \cdot 10^4$
I_b	510 in^4	±51
I_c	272 in^4	±27.2
A_b	10.3 in^2	±1.3
A_c	14.4 in^2	±1.44
H	5,305.5 lbs	±2203.5
α	$2.77461 \cdot 10^8$ lb-in/rad	$\pm 1.26504 \cdot 10^8$
L_b	288 in	
L_c	144 in	

TABLE 17 Enclosures and computational times (last row) for one-bay steel frame (Example 12)

Solution component	GEM(3)	GEMA(3)	KRI	PDM	Popova[23]
$d2_x$	6.44%	6.44%	6.44%	6.66%	1.66%
$d2_y$	4.81%	4.81%	4.81%	4.99%	1.15%
$r2_z$	6.76%	6.76%	6.76%	7.11%	9.17%
$r5_z$	7.73%	7.73%	7.73%	8.31%	17.31%
$r6_z$	7.78%	7.78%	7.79%	8.42%	27.08%
$d3_x$	6.52%	6.52%	6.52%	6.70%	1.68%
$d3_y$	4.96%	4.96%	4.96%	5.20%	1.59%
$r3_z$	6.79%	6.79%	6.79%	7.16%	6.80%
time [s]	0.12	0.04	0.09(4)	0.02	

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

to be uncertain and varying within given intervals. The nominal values of the model parameters and the worst-case uncertainties are given in Table 16.

In order to compare the results produced by different methods, we solve system (25) with parameter uncertainties, which are 1% of the values presented in the last column of Table 16.[23,47] Similarly as in the work of Popova,[23] we assess the quality of the obtained enclosures by using O_w measure (see Table 17; the inner estimation of the hull solution is used for the comparison purposes). Additionally, we provide the results from the work of Popova,[23] obtained by combining the parametric fixed-point iteration method with generalized interval arithmetic.

Table 17 shows that GEMA(3) and KRI produced similar results; however, GEMA(3) turned out to be the fastest. Moreover, for solution components $r2_z$, $r5_z$, $r6_z$, the PDM, GEM(3), GEMA(3), and KRI methods produced significantly better

TABLE 18 Hull solution and results of the KRI method for the one-bay steel frame example

	Hull	KRI	
		outer	inner
$d2_x$	[0.152233, 0.154307]	[0.152198, 0.154341]	[0.152268, 0.154271]
$d2_y \times 10^3$	[0.323803, 0.329781]	[0.323721, 0.329846]	[0.323870, 0.329698]
$r2_z \times 10^3$	[−0.971680 −0.957698]	[−0.971916 −0.957444]	[−0.971427 −0.957933]
$r5_z \times 10^3$	[−0.469077 −0.462296]	[−0.469207 −0.462153]	[−0.468935 −0.462425]
$r6_z \times 10^3$	[−0.430183 −0.423872]	[−0.430308 −0.423740]	[−0.430052 −0.423995]
$d3_x$	[0.149693, 0.151739]	[0.149658, 0.151773]	[0.149728, 0.151703]
$d3_y \times 10^3$	[−0.677375 −0.664490]	[−0.677522 −0.664309]	[−0.677195 −0.664636]
$r3_z \times 10^3$	[−0.939613 −0.925978]	[−0.939847 −0.925732]	[−0.939368 −0.926210]

Note. Results are rounded outwardly to 10-digit accuracy (Example 12). KRI = Krawczyk iteration.

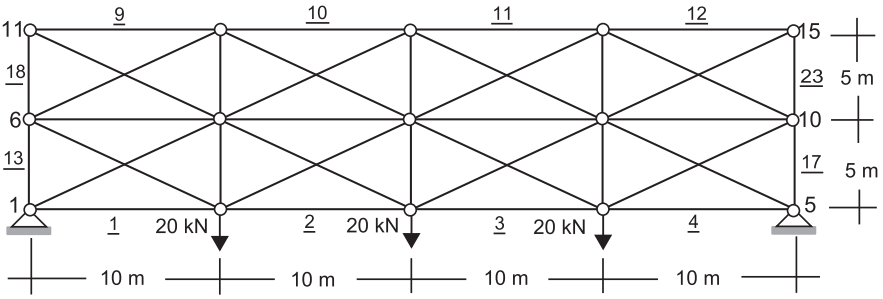

FIGURE 5 Four-bay, two-floor truss structure (Example 13)

enclosures than the method from the work of Popova.[23] Table 18 shows the hull solution (which can be obtained by using combinatorial approach) and the result of the KRI method.

Example 13. Consider the plane four-bay two-floor truss structure depicted in Figure 5. The truss is subjected to equal downward forces of 20 kN at nodes 2, 3, and 4 and has fixed supports at nodes 1 and 5.

The conventional finite element method for trusses gives the following parametric system of equations:

$$K(p)x = F, \tag{26}$$

where p is a vector of model parameters, $K(p)$ is a stiffness matrix, F is a vector of nodal loads, and u is a vector of nodal displacements. The variation in the loading is ±10% from the nominal value. Each beam has the cross-sectional area $A = 0.005\ \text{m}^2$, and the uncertain modulus of elasticity is varying within a tolerance interval around the nominal value $E = 2.0 \cdot 10^{11}$ Pa (the modulus of elasticity of each element are assumed to be varied independently). The relative uncertainty in the modulus of elasticity is varied to be able to assess the degradation of the bounds as the uncertainty increases. The lengths of the horizontal and vertical beams are, respectively, $L_H = 10$ m and $L_V = 5$ m. There are 15 nodes, 38 elements, and three loads, which results in 26 variables and 41 uncertain parameters. The width of the tolerance interval for the modulus of elasticity is varied to be able to assess the degradation of the bounds for displacements as the uncertainty increases. The sharpness of outer enclosures is presented in Table 19 and computational times in Table 20.

TABLE 19 Sharpness of outer enclosures for Example 13

Stiff. uncert.	GEM(3)	GEMA(3)	KRI	PDM
1%	0.44–0.92	0.44–0.92	0.44–0.92	0.43–0.91
2%	0.26–0.83	0.26–0.83	0.26–0.83	0.25–0.83
3%	0.13–0.75	0.13–0.75	0.13–0.75	0.12–0.73

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 20 Computational times (in seconds) and number of iterations for Example 13

Stiff. uncert.	GEM(3)	GEMA(3)	KRI	PDM
1%	8.13	0.88	1.66(7)	0.52
2%	8.13	0.88	1.846(9)	0.52
3%	8.13	0.88	2.186(12)	0.52

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

As can be seen, the GEM(3), GEMA(3), and KRI methods are slightly better than the PDM method, but PDM is the fastest. The GEMA(3) is the most competitive to PDM, whereas GEM(3) is the least efficient in this case.

Example 14. Consider interval parametric linear system (27), which occurs in worst-case tolerance analysis of linear AC (alternate current) electrical circuits.[48-51] The circuit studied is shown in Figure 6 (cf. the work of Kolev[48]). It has eleven branches and five nodes (not including the datum node). The parameters of the model have the following nominal values:

$$
\begin{aligned}
&e_1 = e_2 = 100V,\ e_5 = e_7 = 10V,\\
&Z_j = R_j + iX_j \in \mathbb{C},\ R_j = 100\Omega,\ X_j = \omega L_j - \frac{1}{\omega C_j},\ j = 1,\ \dots, 11,\\
&\omega = 50,\ X_{1,2,5,7} = \omega L_{1,2,5,7} = 20, X_3 = \omega L_3 = 30,\\
&X_4 = -\frac{1}{\omega C_4} = -300,\ X_{10} = -\frac{1}{\omega C_{10}} = -400,\ X_{6,8,9,11} = 0.
\end{aligned}
$$

The electric parameters: resistance R_j, inductance L_j, and capacitance $C_j, j = 1, \dots, 11$, of the branch elements are considered to be unknown but vary within given intervals. The amplitudes e_1, e_2, e_5, e_7 of the sine voltages are assumed to have zero tolerances. The goal here is to find bounds for the real and imaginary parts of the node voltages $V_1, \dots, V_5$.

The nodal analysis of the considered circuit leads to the following complex parametric linear system[49,52]:

$$
\begin{pmatrix}
\frac{1}{Z_1}+\frac{1}{Z_3}+\frac{1}{Z_6} & -\frac{1}{Z_3} & 0 & 0 & -\frac{1}{Z_6}\\
-\frac{1}{Z_3} & \frac{1}{Z_2}+\frac{1}{Z_3}+\frac{1}{Z_4}+\frac{1}{Z_5} & -\frac{1}{Z_4}-\frac{1}{Z_5} & 0 & 0\\
0 & -\frac{1}{Z_4}-\frac{1}{Z_5} & \frac{1}{Z_4}+\frac{1}{Z_5}+\frac{1}{Z_7}+\frac{1}{Z_{10}} & -\frac{1}{Z_7} & 0\\
0 & 0 & -\frac{1}{Z_7} & \frac{1}{Z_7}+\frac{1}{Z_8}+\frac{1}{Z_9} & -\frac{1}{Z_9}\\
-\frac{1}{Z_6} & 0 & 0 & -\frac{1}{Z_9} & \frac{1}{Z_6}+\frac{1}{Z_9}+\frac{1}{Z_{11}}
\end{pmatrix}
\begin{pmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5 \end{pmatrix}
=
\begin{pmatrix} \frac{e_1}{Z_1}\\ \frac{e_2}{Z_2}-\frac{e_5}{Z_5}\\ \frac{e_5}{Z_5}+\frac{e_7}{Z_7}\\ -\frac{e_7}{Z_7}\\ 0 \end{pmatrix}. \tag{27}
$$

Without loss of generality, we change the system parameters and substitute complex parameters Z_j to $p_j = 1/Z_j$, $j = 1, \dots, 11$. This way, the parametric system involves affine-linear dependencies in the matrix. The system (27) is then replaced with an equivalent twice-larger real parametric linear system with 18 real parameters by separating real and imaginary parts.[34,53] We solve the latter system with parameter tolerances ±5%, ±10%, and 20%. The results are presented in Tables 21–23.

For 15% tolerance, GEM and KRI were the only methods that produced positive intervals for all real parts and negative intervals for all imaginary parts of the voltages. However, for 20% tolerance, the GEM and KRI methods were the only methods that produced the positive intervals for all real parts of voltages.

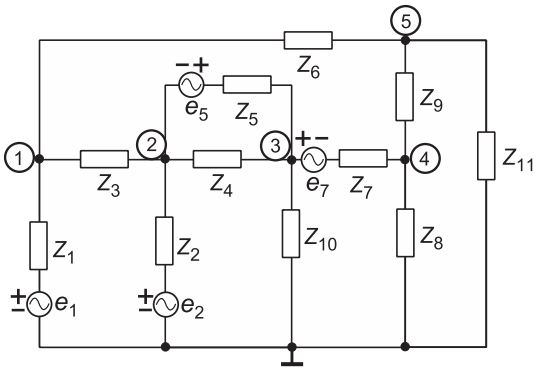

FIGURE 6 Linear electrical AC circuit with five nodes and eleven branches (Example 14)

TABLE 21 Comparison of enclosures for Example 14

Uncertainty	GEM(3) min–max	GEMA(3) min–max	KRI min–max	PDM min–max
5%	0.64–0.84	0.64–0.84	0.64–0.85	0.59–0.79
10%	0.28–0.65	0.28–0.65	0.28–0.66	0.24–0.57
15%	0.00–0.41	0.00–0.41	0.00–0.43	0.00–0.33
20%	0.00–0.07	0.00–0.08	0.00–0.08	0.00–0.06

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 22 Computational times (in seconds) and number of iterations for Example 14

Uncertainty	GEM(3)	GEMA(3)	KRI	PDM
5%	0.35	0.076	0.198(13)	0.04
10%	0.35	0.076	0.282(22)	0.04
15%	0.35	0.076	0.419(36)	0.04
20%	0.35	0.076	0.787(69)	0.04

Note. GEM = generalized expansion method; KRI = Krawczyk iteration; PDM = parametric direct method.

TABLE 23 Enclosures of the Krawczyk iteration (KRI) method for Example 14

Solution component	10%
V_1	$[61.679863, 72.826731] + i[-9.119969, -4.012243]$
V_2	$[65.719834, 77.206612] + i[-10.990121, -4.188371]$
V_3	$[48.551099, 63.689408] + i[-16.610873, -6.360692]$
V_4	$[19.434815, 30.478724] + i[-9.236386, -3.967931]$
V_5	$[25.285505, 36.057956] + i[-6.412950, -2.382641]$
Solution component	15%
V_1	$[57.561123, 77.231049] + i[-11.814045, -1.465630]$
V_2	$[61.052096, 82.109487] + i[-14.640627, -0.762774]$
V_3	$[41.693255, 70.403484] + i[-22.177650, -1.094732]$
V_4	$[14.754410, 34.857138] + i[-12.227419, -1.131749]$
V_5	$[21.147759, 39.980043] + i[-8.628727, -0.295893]$
Solution component	20%
V_1	$[50.134084, 85.158509] + i[-17.767611, 4.302055]$
V_2	$[51.837402, 91.684731] + i[-22.774211, 6.943761]$
V_3	$[27.690381, 84.245783] + i[-34.761587, 10.939194]$
V_4	$[5.775446, 43.338500] + i[-19.183341, 5.640981]$
V_5	$[13.857992, 46.873914] + i[-13.691724, 4.607110]$

TABLE 24 Comparison of theoretical time costs of considered methods, where n is the size of problem, $\mathcal{K} = (K + 1)$ is the number of parameters plus one for accumulation error, κ is the cost of multiplication of revised affine forms, m is the order of the GEM(A) method, and τ is the number of iterations taken by the KRI method

Method	Time cost
PDM	$n^3(2+\mathcal{K}) + n^2(1+2\mathcal{K}+\kappa) + 2n$
EM(1)	$n^3(3+\mathcal{K}) + n^2(\mathcal{K}+\kappa)$
EM(2)	$n^3(6+\mathcal{K}+\mathcal{K}^2) + n^2(\mathcal{K}+\kappa)$
GEM	$n^3(2+\mathcal{K}+(m+2)\kappa) + n^2(1+2\mathcal{K}+\kappa) + n$
GEMA	$n^3(2+\mathcal{K}) + n^2(1+2\mathcal{K}+(m+2)\kappa) + n$
KRI	$n^3(1+\mathcal{K}) + n^2(\mathcal{K}+\tau\kappa)$ + cost of computing $\boldsymbol{y}^0(e)$

Note. PDM = parametric direct method; EM = expansion method; GEM = generalized expansion method; KRI = Krawczyk iteration.

TABLE 25 Time costs in details

	PDM	EM(1)	EM(2)	GEM	GEMA	KRI
R	n^3	n^3	n^3	n^3	n^3	n^3
$\tilde{x}$	n^2	–	–	n^2	n^2	–
$\boldsymbol{v}(e)$	$2n^2\mathcal{K}$	$n^2\mathcal{K}$	$n^2\mathcal{K}$	$2n^2\mathcal{K}$	$2n^2\mathcal{K}$	$n^2\mathcal{K}$
$\boldsymbol{V}(e)$	$n^3\mathcal{K}$	–	–	$n^3\mathcal{K}$	$n^3\mathcal{K}$	$n^3\mathcal{K}$
M	n^3	–	–	n^3	n^3	–
$\boldsymbol{H}$	$2n$	–	–	n	n	–
$\boldsymbol{G}(e)$	–	–	–	$n^3(m+2)\kappa$	$n^2(m+2)\kappa$	–
$\boldsymbol{V}(e)^{-1}$	–	$n^3\mathcal{K}+2n^3$	$n^3(5+\mathcal{K}+\mathcal{K}^2)$	–	–	–
$\boldsymbol{x}(e)$	$n^2\kappa$	$n^2\kappa$	$n^2\kappa$	$n^2\kappa$	0	$n^2\kappa\tau$

Note. PDM = parametric direct method; EM = expansion method; GEM = generalized expansion method; KRI = Krawczyk iteration.

4.2 | Computational complexity

The revised affine matrix requires less memory to be stored than a parametric matrix represented as a linear combination of real matrices (3), at least when using naive (full matrix) representation. Therefore, methods using this naive representation will be in general more memory consuming than methods operating on revised affine matrices.

Let us now analyze the theoretical time cost of the proposed methods. Because addition and subtraction of floating-point numbers are less time consuming than multiplication and division; therefore, we express the time cost by the number of the two latter operations only.

It is not hard to see that the recorded computational times are consistent with theoretical results; see Table 24 for the upper bounds on the costs and Table 25 for detailed analysis. The GEM, GEMA, and KRI methods are generally more time consuming than the PDM or EM method, but this is the cost of obtaining better enclosures. As it was mentioned before, the problem of computing the hull solution is NP-hard; therefore, it is natural that getting closer to the hull requires more computational time. The time cost of all the methods depends on the time cost of computing the product of revised affine forms; however, GEM is mostly affected. The time cost of KRI depends additionally on the number of iterations and the cost of the method used to compute the initial solution. Therefore, the time cost of KRI can be decreased by reducing the threshold and using a less time-consuming method to compute the initial solution.

5 | CONCLUSION

Summarizing the obtained results, it can be seen that the proposed methods are very useful in solving interval parametric linear systems. Moreover, the GEM, GEMA, and KRI methods are competitive, in terms of the quality of produced enclosures, to most of the known methods for solving interval parametric linear systems. The proposed generalizations of the

EM significantly improve the quality of its results. The third-order method seems to be the most useful because it yields relatively good results at a relatively low computational time cost. The proposed affine Krawczyk iteration method is able to produce tight bounds thanks to the use of revised affine forms, which enable us to propagate the dependencies through iterations. The results of all three methods are generally very similar. Because GEMA is usually the fastest (having the computational time close to, for example, PDM), it is the most recommended. On the other side, the KRI method usually produces the best inner bounds, so it is also worth considering, especially for the comparison purposes. The main advantage of the proposed methods over other methods for solving interval parametric linear systems is that they produce parametric solution given as a revised affine form, which has many useful properties. In particular, it can be used to solve various other problems involving interval parametric linear systems. The proposed methods are more time consuming than, for example, PDM or EM methods, but this is the cost of obtaining better enclosures. However, because further improvements are possible, therefore, our future works will be directed, among others, toward decreasing time costs.

ACKNOWLEDGEMENT

M. Hladík was supported through the Czech Science Foundation (Grantová agentura České republiky) Grant P403-18-04735S.

ORCID

Milan Hladík https://orcid.org/0000-0002-7340-8491

REFERENCES

1. Hladík M. Description of symmetric and skew–symmetric solution set. SIAM J Matrix Anal Appl. 2008;30(2):509–521.
2. Mayer G. Three short descriptions of the symmetric and of the skew-symmetric solution set. Linear Algebra Appl. 2015;475:73–79.
3. Popova ED. Solvability of parametric interval linear systems of equations and inequalities. SIAM J Matrix Anal Appl. 2015;36(2):615–633.
4. Kreinovich V, Lakeyev AV. Linear interval equations: computing enclosures with bounded relative or absolute overestimation is NP-hard. Reliab Comput. 1996;2(4):341–350.
5. Kreinovich V, Lakeyev AV, Rohn J, Kahl PT. Computational complexity and feasibility of data processing and interval computations. Dordrecht, The Netherlands: Kluwer; 1998.
6. Poljak S, Rohn J. Checking robust nonsingularity is NP-hard. Math Control Signals Syst. 1993;6(1):1–9.
7. El-Owny H. Outer interval solution of linear systems with parametric interval data. Int J Sci Eng Res. 2013;4(9):1432–1436.
8. Garloff J, Popova ED, Smith AP. Solving linear systems with polynomial parameter dependency with application to the verified solution of problems in structural mechanics. In: Optimization, simulation, and control. New York, NY: Springer; 2013. p. 301–318
9. Hladík M. Optimal preconditioning for the interval parametric Gauss–Seidel method. In: Scientific computing, computer arithmetic, and validated numerics: 16th International Symposium, SCAN 2014, Würzburg, Germany, September 21-26 2014. Revised selected papers. Cham, Switzerland: Springer International Publishing Switzerland; 2016. p. 116–125.
10. Kolev LV. A method for outer interval solution of linear parametric systems. Reliab Comput. 2004;10(3):227–239.
11. Popova ED. On the solution of parametrised linear systems. In: Krämer W, von Gudenberg JW, editors. Scientific computing, validated numerics, interval methods. Boston, MA: Springer; 2001. p. 127–138.
12. Popova ED, Krämer W. Inner and outer bounds for the solution set of parametric linear systems. J Comput Appl Math. 2007;199(2):310–316.
13. Rump SM. Verification methods for dense and sparse systems of equations. In: Topics in validated computations. Amsterdam, The Netherlands: Elsevier; 1994. p. 63–136.
14. Rump SM. Verification methods: rigorous results using floating-point arithmetic. Acta Numerica. 2010;19:287–449.
15. Akhmerov RR. Interval-affine Gaussian algorithm for constrained systems. Reliab Comput. 2005;11:323–341.
16. Hladík M. Enclosures for the solution set of parametric interval linear systems. Int J Appl Math Comput Sci. 2012;22(3):561–574.
17. Kolev LV. Improvement of a direct method for outer solution of linear parametric systems. Reliab Comput. 2006;12(3):193–202.
18. Kolev LV. Parameterized solution of linear interval parametric systems. Appl Math Comput. 2014;246:229–246.
19. Skalna I. A method for outer interval solution of systems of linear equations depending linearly on interval parameters. Reliab Comput. 2006;12(2):107–120.
20. Kolev LV. Solving linear systems whose elements are non-linear functions of intervals. Reliab Comput. 2004;10(3):227–239.
21. Kolev LV. Componentwise determination of the interval hull solution for linear interval parameter systems. Reliab Comput. 2014;20(1):1–24.

22. Skalna I. A global optimization method for solving parametric linear systems whose input data are rational functions of interval parameters. In: Parallel processing and applied mathematics: 8th International Conference, PPAM 2009, Wroclaw, Poland, September 13-16, 2009, revised selected papers, part II. Berlin, Germany: Springer-Verlag Berlin Heidelberg. 2010. p. 475–484. Lecture notes in computer science, No. 6068.
23. Popova ED. Solving linear systems whose input data are rational functions of interval parameters. In: Boyanov T, Dimova S, Georgiev K, Nikolov G, editors. Numerical methods and applications: 6th International Conference, NMA 2006, Borovets, Bulgaria, August 20-24, 2006. Revised papers. Berlin, Germany: Springer; 2007. p. 345–352. Lecture notes in computer science, No. 4310.
24. Skalna I. Enclosure for the solution set of parametric linear systems with non-affine dependencies. In: Parallel processing and applied mathematics. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2012. p. 513–522.
25. Comba JLD, Stolfi J. Affine arithmetic and its applications to computer graphics. SIBGRAPI'93 VI Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens (Recife, BR); 1993.
26. de Figueiredo LH, Stolfi J. Self-validated numerical methods and applications. Brazilian mathematics colloquium monograph. Rio de Janeiro, Brazil: IMPA; 1997.
27. Kolev LV. Automatic computation of a linear interval enclosure. Reliab Comput. 2001;7:17–28.
28. Messine F. New affine forms in interval branch and bound algorithms. Pyrénées-Atlantiques, France: Université de Pau et des Pays de l'Adour (UPPA); 1999. Technical Report No. R2I 99-02.
29. Vu X-H, Sam-Haroud D, Faltings BV. A generic scheme for combining multiple inclusion representations in numerical constraint propagation. Lausanne, Switzerland: Swiss Federal Institute of Technology in Lausanne (EPFL); 2004. Technical Report No. IC/2004/39. Available from: http://liawww.epfl.ch/Publications/Archive/vuxuanha2004a.pdf
30. Skalna I, Hladík M. A new method for computing a *p*-solution to parametric interval linear systems with affine-linear and nonlinear dependencies. BIT Numer Math. 2017;57(4):1109–1136.
31. Neumaier A. Interval methods for systems of equations. Cambridge, UK: Cambridge University Press; 1990.
32. Skalna I, Hladík M. A new algorithm for Chebyshev minimum-error multiplication of reduced affine forms. Numer Algorithms. 2017;76(4):1131–1152.
33. Sun R, Zhang Y, Cui A. A refined affine approximation method of multiplication for range analysis in word-length optimization. EURASIP J Adv Signal Process. 2014;2014(36).
34. Skalna I. Parametric interval algebraic systems. Cham, Switzerland: Springer International; 2018.
35. Kolev L. A direct method for determining a p-solution of linear parametric systems. J Appl Comput Math. 2016;5(1):1–5.
36. Kolev LV. A class of iterative methods for determining p-solutions of linear interval parametric systems. Reliab Comput. 2016;22:26–46.
37. Kolev LV. Iterative algorithms for determining a p-solutions of linear interval parametric systems. Advanced aspects of theoretical electrical engineering. Sofia, Bulgaria. 2016.
38. Popova ED. Parameterized outer estimation of AE-solution sets to parametric interval linear systems. Appl Math Comput. 2017;311:353–360.
39. Rohn J. Explicit inverse of an interval matrix with unit midpoint. Electron J Linear Algebra. 2011;22:138–150.
40. Degrauwe D, Lombaert G, De Roeck G. Improving interval analysis in finite element calculations by means of affine arithmetic. Comput Struct. 2010;88(3–4):247–254.
41. Popova ED. Enclosing the solution set of parametric interval matrix equation $A(p)X = B(p)$. Numer Algorithms. 2017;78:423–447.
42. Okumura K. An application of interval operation to electric network analysis. Bull Jpn Soc Ind Appl Math. 1993;3(2):115–127.
43. Dehghani-Madiseh M, Dehghan M. Parametric AE-solution sets to the parametric linear systems with multiple right-hand sides and parametric matrix equation $A(p)X = B(p)$. Numer Algorithms. 2016;73(1):245–279.
44. Popova ED. Improved solution enclosures for over- and underdetermined interval linear systems. In: Lirkov I, Margenov S, Waśniewski J, editors. Large-scale scientific computing 5th International Conference, LSSC 2005, Sozopol, Bulgaria, June 6-10, 2005. Revised papers. Berlin, Germany: Springer; 2006. p. 305–312.
45. Zielke G. Report on test matrices for generalized inverses. Computing. 1986;36(1):105–162.
46. Zielke G, Drygalla V. Genaue Lösung linearer Gleichungssyteme. Mitteilungen der GAMM. 2003;26(1/2):7–107.
47. Corliss G, Foley C, Kearfott RB. Formulation for reliable analysis of structural frames. Reliab Comput. 2007;13(2):125–147.
48. Kolev LV. Interval methods for circuit analysis. Singapore: World Scientific; 1993.
49. Kolev LV. Worst-case tolerance analysis of linear DC and AC electric circuits. IEEE Trans Circuits Syst. I: Fundam Theory Appl. 2002;49(12):1693–1702.
50. Zimmer M, Krämer W, Popova ED. Solvers for the verified solution of parametric linear systems. Computing. 2012;94(2):109–123.
51. Dreyer A. Interval analysis of analog circuits with component tolerances. Aachen, Germany: Shaker Verlag; 2005.
52. Popova ED, Kolev L, Krämer W. A solver for complex-valued parametric linear systems. Serdica J Comput. 2010;4(1).
53. Hladík M. Solution sets of complex linear interval systems of equations. Reliab Comput. 2010;14:78–87.

How to cite this article: Skalna I, Hladík M. Direct and iterative methods for interval parametric algebraic systems producing parametric solutions. *Numer Linear Algebra Appl*. 2019;e2229. https://doi.org/10.1002/nla.2229