SOLVING LINEAR SYSTEMS WITH POLYNOMIAL PARAMETER DEPENDENCY IN THE RELIABLE ANALYSIS OF STRUCTURAL FRAMES

Garloff, J., Popova, E. D. & Smith, A. P.

University of Applied Sciences / HTWG Konstanz, Faculty of Computer Science, Postfach 100543, D-78405 Konstanz, Germany

Institute of Mathematics & Informatics, Bulgarian Academy of Sciences, Acad. G. Bonchev str., Bldg. 8, 1113 Sofia, Bulgaria

University of Applied Sciences / HTWG Konstanz, Institute of Applied Research, Postfach 100543, D-78405 Konstanz, Germany

garloff@htwg-konstanz.de

ABSTRACT. A wide range of scientific and engineering problems can be described by systems of linear algebraic equations involving uncertain model parameters. We consider such systems where the coefficients of the matrix and the right hand side are multivariate polynomials or rational functions of parameters varying within given intervals. A general-purpose parametric fixed-point iteration is combined with efficient tools for range enclosure based on the Bernstein expansion of multivariate polynomials. We discuss an advanced application of the general-purpose parametric method to linear systems obtained by standard finite element analysis of mechanical structures and thereby illustrate the efficiency of this new parametric solver.

KEYWORDS: parametric linear system, interval parameter, polynomial range, Bernstein expansion, mechanical structure

1 INTRODUCTION

Engineering analysis and design problems are often described by systems of linear algebraic equations involving uncertain model parameters, which may arise due to measurement imprecision, round-off errors, and various other kinds of inexact knowledge. Significant research in this field is directed towards the use of intervals to represent the uncertain quantities in such systems. When uncertain parameters are modelled by bounded intervals, the problem can be formulated as an interval linear system. Dependencies between such interval parameters may be linear or nonlinear in nature, with the former, simpler, case having been more extensively studied. In the latter case there may be highly nontrivial dependencies between the parameters.

A standard method for solving problems in structural mechanics, such as linear static problems, is the finite element method (FEM). The method leads to a system of algebraic equations, which in case of uncertain (interval) physical parameters becomes a linear system involving interval parameters, e.g., [2, 3, 7, 14]. In the following we assume that the FEM approximations are well tuned with respect to the discretisation errors. It is beyond the scope of this work to account for the discretisation error of the mathematical model in addition to the uncertainty in the parameters, although there are recent investigations in this direction [9, 20]. In [14], a parametric residual iteration [18] is applied to bounding the response of structural engineering systems involving rational dependencies between the model parameters. Corresponding software tools with result verification, implemented in the Wolfram *Mathematica*® environment, have been developed. This general-purpose interval approach imposes no restrictions on how the parametric system is generated and can be applied to linear parametric problems for which special methods have not yet been designed.

The method requires an enclosure of the range of nonlinear functions over the domain of the parameters. When the parameter dependencies are polynomial, tight bounds for the polynomial ranges can be obtained by the expansion of a multivariate polynomial into Bernstein polynomials [4, 21]. The goal of our work is to combine the generalised parametric residual iteration with range enclosure, based on Bernstein expansion, into a more efficient parametric linear system solver. For the sake of rapid development, run-time efficiency, and for exploiting the advantages of modern general-purpose software environments, such as *Mathematica*, our implementation is based on an advanced connectivity between *Mathematica* and an external C++ software via the *MathLink* communication protocol. The present parametric solver is illustrated by numerical solutions to three problems from structural mechanics which have been modelled by standard FEM and involve interval uncertainty in all material and load parameters. A discussion on the comparison between the present parametric solver, based on Bernstein polynomial ranges, and the former one is provided.

The paper is organised as follows. In Section 2, the parametric residual iteration method for linear interval systems is introduced, followed by an introduction to the Bernstein expansion and the implicit Bernstein form. In Section 3, the new parametric solvers are illustrated by three examples of one- and two-bay steel frames. Finally, some conclusions are given.

2 METHODOLOGY

We use the following notation: $\mathbb{R}^m, \mathbb{R}^{m\times n}$ denote the set of real vectors with m components and the set of real $m \times n$ matrices, respectively. A real compact interval is defined as $[a] = [\underline{a}, \overline{a}] := \{a \in \mathbb{R} \mid \underline{a} \le a \le \overline{a}\}$. By $\mathbb{IR}^m, \mathbb{IR}^{m\times n}$ we denote interval m-vectors and interval $m \times n$ matrices. Operations on interval values yield the smallest interval value containing the corresponding result when power set operations are used. We assume that the reader is familiar with the conventional interval arithmetic [1, 6].

The Iteration Method Consider a linear system

$$A(x) \cdot s \;=\; d(x), \tag{1a}$$

where the coefficients of the $m \times m$ matrix $A(x)$ and the vector $d(x)$ are functions of n parameters varying within given intervals

$$a_{ij}(x) \;=\; a_{ij}(x_1, \ldots, x_n), \quad d_i(x) \;=\; d_i(x_1, \ldots, x_n), \quad i, j = 1, \ldots, m, \tag{1b}$$

$$x \;\in\; [x] = ([x_1], \ldots, [x_n])^\top. \tag{1c}$$

The set of solutions to the above system, called the *parametric solution set*, is

$$\Sigma = \Sigma\,(A(x), d(x), [x]) \;:=\; \{s \in \mathbb{R}^m \mid A(x) \cdot s = d(x) \text{ for some } x \in [x]\}\,.$$

The set Σ is compact if $A(x)$ is nonsingular for every $x \in [x]$. For a nonempty bounded set $\mathcal{S} \subseteq \mathbb{R}^m$, define its interval hull by $\square\,\mathcal{S} \;:=\; \cap\{[s] \in \mathbb{IR}^m \mid \mathcal{S} \subseteq [s]\}$. Since it is quite expensive to obtain Σ or $\square\,\Sigma$, we seek an interval vector $[w]$ for which it is guaranteed that $[w] \supseteq \square\,\Sigma \supseteq \Sigma$.

In this section we consider a self-verified method for bounding the solution set of a parametric linear system. This is a general-purpose method since it does not assume any particular structure among the parameter dependencies. The method originates in the inclusion theory for nonparametric problems, which is discussed in many works (cf. [18] and the literature cited therein). In [18, Theorem 4.8] a straightforward generalisation to linear systems with linear parameter dependencies is given. With obvious modifications, the corresponding theorems can also be applied directly to linear systems involving nonlinear parameter dependencies, as demonstrated in [11, 14]. The following theorem is a general formulation of the enclosure method for linear systems involving arbitrary parametric dependencies.

Theorem 2.1. *Consider a parametric linear system defined by Eqs. 1a – 1c. Let* $R \in \mathbb{R}^{m\times m}$, $[y] \in \mathbb{IR}^m$, $\tilde{s} \in \mathbb{R}^m$ *be given and define* $[z] \in \mathbb{IR}^m$, $[C] \in \mathbb{IR}^{m\times m}$ *by*

$$[z] \;:=\; \square\{z(x) = R\,(d(x) - A(x)\tilde{s}) \mid x \in [x]\},$$

$$[C] \;:=\; \square\{C(x) = I - R \cdot A(x) \mid x \in [x]\},$$

where I denotes the identity matrix. Define $[v] \in \mathbb{IR}^m$ *by means of the following Gauss-Seidel iteration*

$$[v_i] := \{[z] + [C] \cdot ([v_1], ..., [v_{i-1}], [y_i], \ldots, [y_m])^\top\}_i, \quad 1 \le i \le m.$$

If $[v] \subsetneq [y]$, *then R and every matrix* $A(x)$ *with* $x \in [x]$ *are regular, and for every* $x \in [x]$ *the unique solution* $\widehat{s} = A^{-1}(x)d(x)$ *of the system defined by Eqs. 1a–1c satisfies* $\widehat{s} \in \tilde{s} + [v]$.

The above theorem generalises [18, Theorem 4.8] by stipulating a sharp enclosure of $C(x) := I - R \cdot A(x)$ for $x \in [x]$, instead of using the interval extension $C([x])$. A sharp enclosure of the iteration matrix $C(x)$ is also required by other authors (who do not refer to [18]), e.g., [3], without addressing the issue of rounding errors. Examples demonstrating the expanded scope of application of the generalized inclusion theorem can be found in [11, 14, 16]. It should be noted that the above theorem provides strong regularity (cf. [10]), which is a weaker but sufficient condition for regularity of the parametric matrix.

When aiming to compute a self-verified enclosure of the solution to a parametric linear system by the above inclusion method, a fixed-point iteration scheme is proven to be very useful. A detailed presentation of the computational algorithm can be found in [11].

In case of arbitrary nonlinear dependencies between the uncertain parameters, computing $[z]$ and $[C]$ in Theorem 2.1 requires a sharp range enclosure of nonlinear functions. This is a key problem in interval analysis and there exists a huge number of methods and techniques devoted to this problem, with no one method being universal. In this work we restrict ourselves to linear systems where the elements of $A(x)$ and $d(x)$ are rational functions of the uncertain parameters. In this case the coefficients of $z(x)$ and $C(x)$ are also rational functions of x. The quality of the range enclosure of $z(x)$ will determine the sharpness of the parametric solution set enclosure. In [11] the above inclusion theorem is combined with a simple interval arithmetic technique providing inner and outer bounds for the range of monotone rational functions. The arithmetic of generalised (proper and improper) intervals is considered as an intermediate computational tool for eliminating the dependency problem in range computation and for obtaining inner estimations by outwardly rounded interval arithmetic. Since this methodology is not efficient in the general case of non-monotone rational functions, in this work we combine the parametric fixed-point iteration with range enclosing tools based on the Bernstein expansion of multivariate polynomials.

Bernstein Enclosure of Polynomial Ranges In this section we recall some properties of the Bernstein expansion which are fundamental to our approach, cf. [4, 21] and the references therein.

Firstly, some notation is introduced. We define multiindices $i = (i_1, \ldots, i_n)^T$ as vectors, where the n components are nonnegative integers. The vector 0 denotes the multiindex with all components equal to 0. Comparisons are used entrywise. Also the arithmetic operators on multiindices are defined componentwise such that $i \odot l := (i_1 \odot l_1, \ldots, i_n \odot l_n)^T$, for $\odot = +, -, \times$, and / (with $l > 0$). For instance, i/l, $0 \le i \le l$, defines the Greville abscissae. For $x \in \mathbb{R}^n$ its multipowers are $x^i := \prod_{\mu=1}^{n} x_\mu^{i_\mu}$. For the n-fold sum we use the notation $\sum_{i=0}^{l} := \sum_{i_1=0}^{l_1} \ldots \sum_{i_n=0}^{l_n}$. The generalised binomial coefficient is defined by $\binom{l}{i} := \prod_{\mu=1}^{n} \binom{l_\mu}{i_\mu}$.

An n-variate polynomial p of degree $l = (l_1, \ldots, l_n)$,

$$p(x) = \sum_{i=0}^{l} a_i x^i, \quad x = (x_1, \ldots, x_n), \tag{2}$$

can be represented over $[x]$ (Eq. 1c) with $\underline{x} = (\underline{x}_1, \ldots, \underline{x}_n)$, $\overline{x} = (\overline{x}_1, \ldots, \overline{x}_n)$ as

$$p(x) = \sum_{i=0}^{l} b_i B_i(x),$$

where B_i is the *i-th Bernstein polynomial of degree l*

$$B_i(x) = \binom{l}{i} \frac{(x - \underline{x})^i (\overline{x} - x)^{l-i}}{(\overline{x} - \underline{x})^l}$$

and the so-called *Bernstein coefficients* b_i are given by

$$b_i = \sum_{j=0}^{i} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \sum_{\kappa=j}^{l} \binom{\kappa}{j} \underline{x}^{\kappa-j} a_\kappa, \quad 0 \leq i \leq l.$$

The essential property of the Bernstein expansion is the *range enclosing property*, namely that the range of p over $[x]$ is contained within the interval spanned by the minimum and maximum Bernstein coefficients: $\min_i\{b_i\} \leq p(x) \leq \max_i\{b_i\}, \ x \in [x]$.

It is also worth noting that the values attained by the polynomial at the vertices of $[x]$ are identical to the corresponding vertex Bernstein coefficients, for example $b_0 = p(\underline{x})$ and $b_l = p(\overline{x})$. The *sharpness property* states that the lower (resp. upper) bound provided by the minimum (resp. maximum) Bernstein coefficient is sharp, i.e. there is no underestimation (resp. overestimation), if and only if this coefficient occurs at a vertex of $[x]$.

The traditional approach (see, for example, [4, 21]) assumes that all of the Bernstein coefficients are computed, and their minimum and maximum is determined. By use of an algorithm (cf. [4, 21]) which is similar to de Casteljau's algorithm (see, for example, [17]), this computation can be made efficient, with time complexity $O(n\hat{l}^{n+1})$ and space complexity (equal to the number of Bernstein coefficients) $O((\hat{l} + 1)^n)$, where $\hat{l} = \max_{i=1}^n l_i$. This exponential complexity is a drawback of the traditional approach, rendering it infeasible for polynomials with moderately many (typically, 10 or more) variables.

In [19] a new method for the representation and computation of the Bernstein coefficients is presented, which is especially well suited to sparse polynomials. With this method the computational complexity typically becomes nearly linear with respect to the number of the terms in the polynomial, instead of exponential with respect to the number of variables. This improvement is obtained from the results surveyed in the following subsections. For details and examples the reader is referred to [19].

Bernstein Coefficients of Monomials Let $q(x) = x^r$, $x = (x_1, \ldots, x_n)$, for some $0 \leq r \leq l$. Then the Bernstein coefficients of q (of degree l) over $[x]$ (Eq. 1c) are given by

$$b_i = \prod_{m=1}^{n} b_{i_m}^{(m)},$$

where $b_{i_m}^{(m)}$ is the i_mth Bernstein coefficient (of degree l_m) of the univariate monomial x^{r_m} over $[\underline{x}_m, \overline{x}_m]$. If the box $[x]$ is restricted to a single orthant of $\mathbb{R}^n$ then the Bernstein coefficients of q over $[x]$ are monotone with respect to each variable x_j, $j = 1, \ldots, n$.

With this property, for a single-orthant box, the minimum and maximum Bernstein coefficients must occur at a vertex of the array of Bernstein coefficients. This also implies that the bounds provided by these coefficients are sharp; see the aforementioned sharpness property. Finding the minimum and maximum Bernstein coefficients is therefore straightforward; it is not necessary to explicitly compute the whole set of Bernstein coefficients. Computing the component univariate Bernstein coefficients for a multivariate monomial has time complexity $O(n(\hat{l} + 1)^2)$. Given the exponent r and the orthant in question, one can determine whether the monomial (and its Bernstein coefficients) is increasing or decreasing with respect to each coordinate direction, and one then merely needs to evaluate the monomial at these two vertices.

Without the single orthant assumption, monotonicity does not necessarily hold, and the problem of determining the minimum and maximum Bernstein coefficients is more complicated. For boxes which intersect two or more orthants of $\mathbb{R}^n$, the box can be bisected, and the Bernstein coefficients of each single-orthant sub-box can be computed separately.

The Implicit Bernstein Form Firstly, we can observe that since the Bernstein form is linear, if a polynomial p consists of t terms, as follows,

$$p(x) = \sum_{j=1}^{t} a_{i_j} x^{i_j}, \quad 0 \leq i_j \leq l, \quad x = (x_1, \ldots, x_n),$$

then each Bernstein coefficient is equal to the sum of the corresponding Bernstein coefficients of each term, as follows:

$$b_i = \sum_{j=1}^{t} b_i^{(j)}, \quad 0 \leq i \leq l,$$

where $b_i^{(j)}$ are the Bernstein coefficients of the jth term of p. (Hereafter, a superscript in brackets specifies a particular term of the polynomial. The use of this notation to indicate a particular coordinate direction, as in the previous subsection, is no longer required.)

Therefore one may implicitly store the Bernstein coefficients of each term, and compute the Bernstein coefficients as a sum of t products, only as needed. The implicit Bernstein form thus consists of computing and storing the n sets of univariate Bernstein coefficients (one set for each component univariate monomial) for each of t terms. Computing this form has time complexity $O(nt(\hat{l}+1)^2)$ and space complexity $O(nt(\hat{l}+1))$, as opposed to $O((\hat{l}+1)^n)$ for the explicit form. Computing a ingle Bernstein coefficient from the implicit form requires $(n+1)t-1$ arithmetic operations.

Determination of the Bernstein Enclosure for Polynomials We consider the determination of the minimum Bernstein coefficient; the determination of the maximum Bernstein coefficient is analogous. For simplicity we assume that $[x]$ is restricted to a single orthant.

We wish to determine the value of the multiindex of the minimum Bernstein coefficient in each direction. In order to reduce the search space (among the $(\hat{l}+1)^n$ Bernstein coefficients) we can exploit the monotonicity of the Bernstein coefficients of monomials and employ uniqueness, monotonicity, and dominance tests, cf. [19] for details. As the examples in [19] show, it is often possible in practice to dramatically reduce the number of Bernstein coefficients that have to be computed.

3 NUMERICAL EXAMPLES

In this section we illustrate the usage of the new parametric solvers based on bounding polynomial ranges by Bernstein expansion. The improved efficiency of the new polynomial solvers is demonstrated by comparing both the computing time and the quality of the solution enclosure for the new solvers and the former one. The examples were run on a PC with AMD Athlon-64 3GHz processor.

Software Software for the solution of parametric interval linear systems, for the C-XSC [16] and *Mathematica* [11] environments has been developed. Recently, the *Mathematica* parametric linear solvers were upgraded to handle linear systems involving arbitrary rational dependencies [11, 14]. The enclosures of $z(x)$ and $C(x)$ from Theorem 2.1 were computed by a technique based on generalised intervals, which provides sharp range enclosures for monotone rational functions. The goal of this work is to further upgrade the parametric solvers for systems involving polynomial and/or arbitrary rational dependencies, by integrating more powerful and efficient tools for range computation into the corresponding *Mathematica* functions.

Given a polynomial p (Eq. 2) and a box $[x]$ (Eq. 1c), we wish to compute a guaranteed tight enclosure for $p([x])$. The existing C++ software routines of the last author, which implement the aforementioned implicit Bernstein form, are utilised. Interval arithmetic is used extensively throughout, for which the C++ interval library `filib++` [5] is employed. As an additional benefit, all computational results can also be guaranteed in the presence of rounding errors.

In order to shorten the development time and to preserve the beneficial properties of both implementation environments, the authors have connected the generalized parametric fixed-point iteration and the Bernstein enclosure of polynomial ranges into a new parametric solver via the *MathLink* communication protocol. *MathLink* allows the external `filib++` function for polynomial range computation to be called from within *Mathematica* as required. More details about *MathLink* technology and the connectivity between *Mathematica* and external `filib++` based interval programs can be found in [13].

The new solver computes guaranteed outer bounds for the solution set of a parametric linear system, where the matrix and/or the right hand side vector involve polynomial dependencies between uncertain parameters. The general parametric residual iteration, cf. Theorem 2.1, implemented in this function, uses some algebraic manipulations and *MathLink* communication with the `filib++` software for bounding the ranges of multivariate polynomials involved in the computation of $z(x)$ and $C(x)$.

The `filib++` software for bounding the range of a multivariate polynomial over a box can also be used for bounding the range of a function involving arbitrary rational dependencies between its variables, if we represent the rational function as a quotient of two multivariate polynomials which are to be bounded separately. This motivates the development of a further routine applicable to linear systems involving arbitrary rational parameter dependencies.

For the sake of comparison, we also test a routine with the same usage as the previous solvers but applying the former range computation method based on generalised interval arithmetic.

One-Bay Steel Frame Consider a simple one-bay structural steel frame, as shown in Figure 1, which was initially studied by interval methods in [2]. Following standard practice, the authors have assembled a parametric linear system of order eight and involving eight uncertain parameters. The typical nominal parameter values and the corresponding worst case uncertainties, as proposed in [2] but converted to SI-units, are shown in Table 1. The explicit analytic form of the given system involving polynomial parameter dependencies can be found in [2, 14].

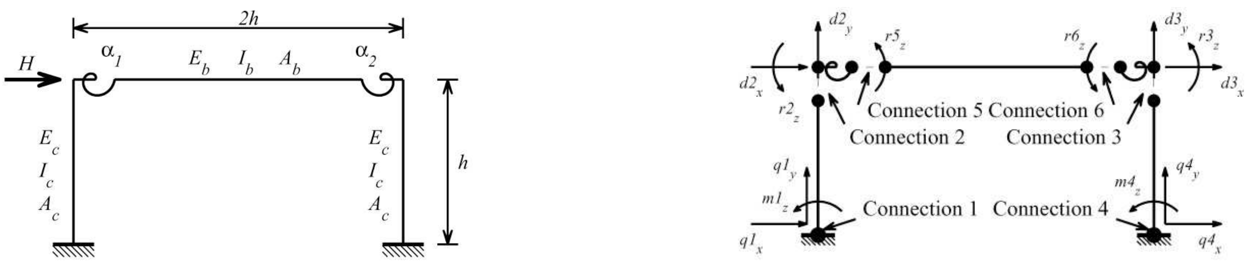

Figure 1: One-bay structural steel frame [2].

Table 1: Parameters involved in the steel frame example.

parameter		nominal value	uncertainty
Young modulus	E_b	$1.999 * 10^8$ kN/m^2	$\pm 2.399 * 10^7$ kN/m^2
	E_c	$1.999 * 10^8$ kN/m^2	$\pm 2.399 * 10^7$ kN/m^2
Second moment	I_b	$2.123 * 10^{-4}$ m^4	$\pm 2.123 * 10^{-5}$ m^4
	I_c	$1.132 * 10^{-4}$ m^4	$\pm 1.132 * 10^{-5}$ m^4
Area	A_b	$6.645 * 10^{-3}$ m^2	$\pm 6.645 * 10^{-4}$ m^2
	A_c	$9.290 * 10^{-3}$ m^2	$\pm 9.290 * 10^{-4}$ m^2
External force	H	23.600 kN	± 9.801 kN
Joint stiffness	α	$3.135 * 10^5$ kNm/rad	$\pm 1.429 * 10^5$ kNm/rad
Length	L_c	3.658 m, L_b 7.316 m	

As in [2, 14], we solved the system with parameter uncertainties which are 1% of the values presented in the last column of Table 1.

We run the parametric solver and measure a total time of 0.05 seconds for execution of the main steps. Running the previous parametric solver, we get a time measurement of 0.34 seconds, showing that the compiled external code for range computation was considerably faster than the interpretative *Mathematica* code. For this example, the quality of the solution set enclosures, provided by both solvers, was comparable. As shown in [11, 14], the solution enclosure obtained by the parametric solver is by more than one order of magnitude better than the solution enclosure obtained in [2].

Based on the runtime efficiency of the new parametric solver, we next attempt to solve the same parametric linear system for the worst case parameter uncertainties in Table 1 ranging between about 10% and 46%. Firstly, we notice that the parametric solution depends linearly on the parameter H, so that we can obtain a better solution enclosure if we solve two parametric systems with the corresponding end-points for H. Secondly, enclosures of the hull of the solution set are obtained by subdivision of the worst case parameter intervals $(E_b, E_c, I_b, I_c, A_b, A_c, \alpha)^\top$ into $(2, 2, 2, 2, 1, 1, 6)^\top$ subintervals of equal width, respectively. We use more subdivision with respect to α since α is subject to the greatest uncertainty. The solution enclosure, obtained within 11 sec., is given in Table 2. Moreover, the quality of the solution enclosure $[u]$ of the respective eight quantities is compared to the combinatorial solution $[\tilde{h}]$, i.e. the convex hull of the solutions to the point linear systems obtained when the parameters take all possible combinations of the interval end-points. The combinatorial solution serves as an *inner* estimation of the solution enclosure. The quality of the solution enclosure is measured by O_ω, defined by $O_\omega([\tilde{h}], [u]) := 100(1 - \omega([\tilde{h}])/\omega([u]))$, where ω denotes the width of the interval.

Table 2: One-bay steel frame example with worst-case parameter uncertainties (Table 1). Solution enclosure $[u]$ found by dividing the parameter intervals $(E_b, E_c, I_b, I_c, A_b, A_c, \alpha)^\top$ into $(2, 2, 2, 2, 1, 1, 6)^\top$ subintervals of equal width, respectively. $[u]$ is compared to the combinatorial solution $[\tilde{h}]$.

	$10^5 *$ solution enclosure $[u]$	$O_\omega([\tilde{h}], [u])$
$d2_x$:	[138.54954789, 627.59324779]	12.5
$d2_y$:	[0.29323100807, 2.1529383383]	8.0
$r2_z$:	[-129.02427835, -22.381136355]	23.7
$r5_z$:	[-113.21398401, -17.95789860]	25.6
$r6_z$:	[-105.9680866, -17.64526946]	25.0
$d3_x$:	[135.25570695, 616.85512710]	12.7
$d3_y$:	[-3.7624790816, -0.41629803684]	13.2
$r3_z$:	[-122.3361772, -21.69878778]	23.5

These results show that by means of a small number of subdivisions the new parametric solver provides a good solution enclosure very quickly for the difficult problem of worst-case parameter uncertainties. Note that sharper bounds, close to the exact hull, can be obtained by proving the monotonicity properties of the parametric solution [12].

Two-Bay Two-Story Frame Model with 13 Parameters Consider a two-bay two-story steel frame with IPE 400 beams and HE 280 B columns, as shown in Figure 2, after [14]. The frame is subjected to lateral static forces and vertical uniform loads. Beam-to-column connections are considered to be semi-rigid and they are modelled by single rotational spring elements. Applying conventional methods for the analysis of frame structures, a system of 18 linear equations is obtained, where the elements of the stiffness matrix and of the right hand side vector are rational functions of the model parameters. We consider the parametric system resulting from a finite element model involving the following 13 uncertain parameters: A_c, I_c, E_c, A_b, I_b, E_b, c, $w_1, \ldots, w_4$, F_1, F_2. Their nominal values, taken according to the European Standard Eurocode3, are given in Table 3. The explicit analytic form of the given parametric system can be found in [15].

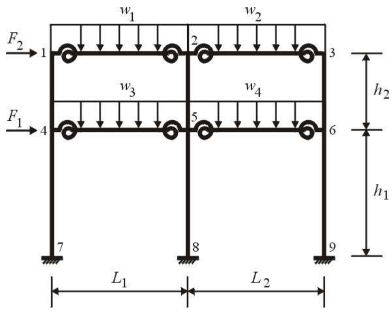

Figure 2: Two-bay two-story steel frame [14].

The parametric system is solved for the element material properties ($A_c, \ldots, E_b$), which are taken to vary within a tolerance of 1% (that is $[x - x/200,\ x + x/200]$, where x is the corresponding parameter nominal value from Table 3) while the spring stiffness and all applied loadings are taken to vary within 10% tolerance intervals.

parameter	Columns (HE 280 B)	Beams (IPE 400)
Cross-sectional area	$A_c = 0.01314\ \text{m}^2$	$A_b = 0.008446\text{m}^2$
Moment of inertia	$I_c = 19270 * 10^{-8}\ \text{m}^4$	$I_b = 23130 * 10^{-8}\ \text{m}^4$
Modulus of elasticity	$E_c = 2.1 * 10^8 \text{kN/m}^2$	$E_b = 2.1 * 10^8\ \text{kN/m}^2$
Length	$L_c = 3$ m	$L_b = 6$ m
Rotational spring stiffness	$c = 10^8$ kN	
Uniform vertical load	$w_1 = \ldots = w_4 = 30$ kN/m	
Concentrated lateral forces	$F_1 = F_2 = 100$ kN	

Table 3: Parameters involved in the two-bay two-story frame example with their nominal values.

The new parametric solver requires a total of 1.3 seconds for execution of the main steps, compared to 7.4 seconds for the former one; it is about six times faster. An algebraic simplification, applied to functional expressions in computer algebra environments, may reduce the occurrence of interval variables which could result in a sharper range enclosure. Such an algebraic simplification is expensive and when applied to complicated rational expressions usually does not result in a sharper range enclosure. For the sake of comparison, we have run the former parametric solver in two ways: applying intermediate simplification during the range computation, and without any algebraic simplification. The above results were obtained when the range computation does not use any algebraic simplification. When the range computation of the previous solver uses intermediate algebraic simplification, we obtain the results in 14.4 seconds. This is much slower, but provided a tighter enclosure of the solution set than the rational solver, based on polynomial ranges, which did not account for all the parameter dependencies.

Two-Bay Two-Story Frame Model with 37 Parameters As a larger problem of a parametric system involving rational parameter dependencies, we consider the finite element model of the two-bay two-story steel frame from the previous example, where each structural element has properties varying independently within 1% tolerance intervals. This does not change the order of the system but it now depends on 37 interval parameters. The explicit analytic form of the given parametric system can be found in [15].

First, the polynomial solver is run, and one observes a considerable increase in the computing time (245 seconds) compared to the time needed for the 13 parameters example, caused by the larger number of parameters. The former parametric solver, based on range computation without algebraic simplification, exhibits approximately three times slower performance (755 seconds) than the new one. The quality of the solution enclosure, provided by the new polynomial solver, is also much better than the solution enclosure provided by the former solver.

Note that when the previous range computation uses algebraic simplification, it is much slower. However, the quality of the solution enclosure does not improve by more than a tiny amount, probably due to the more complicated parameter dependencies. This demonstrates the merit of the general-purpose parametric iteration, combined with Bernstein enclosure of polynomial ranges, for solving parametric systems involving complicated dependencies between many parameters.

The method presented in [8] is highly efficient for truss structures but it is not applicable to the complicated dependencies arising in the two-bay two-story frame models considered above. The general-purpose interval approach we consider is suitable for linear parametric problems involving arbitrary polynomial dependencies for which special methods have not yet been designed.

4 CONCLUSIONS

In this paper, we demonstrated the advanced application of a general-purpose parametric method, combined with the Bernstein enclosure of polynomial ranges, to linear systems obtained by standard FEM analysis of mechanical structures, and illustrated the efficiency of the new parametric solver.

It is shown that powerful techniques for range enclosure are necessary to provide tight bounds on the solution set, in particular when the parameters of the system are subject to large uncertainties and the dependencies are complicated.

The new self-verified parametric solvers can be incorporated into a general framework for the computer-assisted proof of global and local monotonicity properties of the parametric solution. Based on these properties, a guaranteed and highly accurate enclosure of the interval hull of the solution set can be computed [12]. The parametric solvers for square systems facilitate the guaranteed enclosures of the solution sets to over- and underdetermined parametric linear systems.

REFERENCES

[1] Alefeld G, Herzberger J. Introduction to Interval Computations, Academic Press, New York; 1983.

[2] Corliss G, Foley C, Kearfott RB. Formulation for reliable analysis of structural frames, Reliab. Comp. 2007;13:125–147.

[3] Dessombz O, Thouverez F, Laîné JP, Jézéquel L. Analysis of mechanical systems using interval computations applied to finite element methods, Journal of Sound and Vibration 2001;239(5):949–968.

[4] Garloff J. Convergent bounds for the range of multivariate polynomials. In: Nickel K, editor. Interval Mathematics 1985, Lecture Notes in Computer Science, Vol. 212, Springer, Berlin; 1986. p. 37–56.

[5] Lerch M, Tischler G, Wolff von Gudenberg J, Hofschuster W, Krämer W. `filib++`, a fast interval library supporting containment computations, ACM Trans. on Math. Software 2006;32(2):299–324.

[6] Moore RE. Methods and Applications of Interval Analysis, SIAM, Philadelphia; 1979.

[7] Muhanna RL, Zhang H, Mullen RL. Interval finite elements as a basis for generalized models of uncertainty in engineering mechanics, Reliab. Comp. 2007;13:173–194.

[8] Neumaier A, Pownuk A. Linear systems with large uncertainties, with applications to truss structures. Reliable Computing 2007;13:149–172.

[9] Neumaier A. Certified error bounds for uncertain elliptic equations, Journal of Computational and Applied Mathematics 2008;218:125–136.

[10] Popova ED. Strong regularity of parametric interval matrices. In Dimovski I et al., editors. Mathematics & Education in Mathematics, IMI-BAS, Sofia, 2004. p. 446–451.

[11] Popova ED. Solving linear systems whose input data are rational functions of interval parameters. In Boyanov T et al., editors. NMA 2006, Lecture Notes in Computer Science, Vol. 4310, Springer; 2007. p. 345–352. Expanded version in Preprint 3/2005, Institute of Mathematics and Informatics, BAS, Sofia; 2005. http://www.math.bas.bg/~epopova/papers/05PreprintEP.pdf

[12] Popova ED. Computer-assisted proofs in solving linear parametric problems. In: Conference Post-Proceedings of SCAN 2006, IEEE Computer Society Press, Library of Congress Number 2007929345; 2007. p. 35.

[13] Popova ED. Mathematica connectivity to interval libraries filib++ and C-XSC. In: Cuyt AAM et al. editors. Numerical Validation in Current Hardware Architectures, Lecture Notes in Computer Science, Vol. 5492, Springer; 2009. p. 117-132.

[14] Popova ED, Iankov R, Bonev Z. Bounding the response of mechanical structures with uncertainties in all the parameters. In Muhanna RL, Mullen RL, editors. Proceedings of the NSF Workshop on Reliable Engineering Computing, Savannah, Georgia; 2006. p. 245–265.

[15] Popova ED, Iankov R, Bonev Z. FEM model of a two-bay two-story steel frame – 2 benchmark examples. http://www.math.bas.bg/~epopova/papers/2bay2storyProblems.pdf

[16] Popova ED, Krämer W. Inner and outer bounds for the solution set of parametric linear systems, J. of Computational and Applied Mathematics 2007;199(2):310–316.

[17] Prautzsch H, Boehm W, Paluszny M. Bezier and B-Spline Techniques, Springer, Berlin, Heidelberg; 2002.

[18] Rump S. Verification methods for dense and sparse systems of equations. In: Herzberger J, editor. Topics in Validated Computations, North-Holland, Amsterdam; 1994. p. 63–135.

[19] Smith AP. Fast construction of constant bound functions for sparse polynomials, J. Global Optimization 2009;43(2–3):445–458.

[20] Zalewski BF, Mullen RL. Worst case bounds on the point-wise discretization error in boundary element method for elasticity problem. In Muhanna RL, Mullen RL, editors. Proceedings of the 3rd Int. Workshop on Reliable Engineering Computing, Savannah, Georgia; 2008. p. 429–457.

[21] Zettler M, Garloff J. Robustness analysis of polynomials with polynomial parameter dependency using Bernstein expansion, IEEE Trans. Automat. Contr. 1998;43:425–431.