

RESEARCH ARTICLE

Data-based polyhedron model for optimization of engineering structures involving uncertainties

Zhiping Qiu^{1,*} , Han Wu¹, Isaac Elishakoff² and Dongliang Liu¹

¹Institute of Solid Mechanics, Beihang University, Beijing 100191, China

²Department of Ocean and Mechanical Engineering, Florida Atlantic University, Boca Raton, Florida 33431-0991, USA

*Corresponding author. E-mail: zpqiu@buaa.edu.cn

Received: 11 December 2020; **Revised:** 20 April 2021; **Accepted:** 02 June 2021

Keywords: Chebyshev Inequality; convex polyhedron; uncertainty; uncertain linear programming; vertex solution method

Abstract

This paper studies the data-based polyhedron model and its application in uncertain linear optimization of engineering structures, especially in the absence of information either on probabilistic properties or about membership functions in the fuzzy sets-based approach, in which situation it is more appropriate to quantify the uncertainties by convex polyhedra. Firstly, we introduce the uncertainty quantification method of the convex polyhedron approach and the model modification method by Chebyshev inequality. Secondly, the characteristics of the optimal solution of convex polyhedron linear programming are investigated. Then the vertex solution of convex polyhedron linear programming is presented and proven. Next, the application of convex polyhedron linear programming in the static load-bearing capacity problem is introduced. Finally, the effectiveness of the vertex solution is verified by an example of the plane truss bearing problem, and the efficiency is verified by a load-bearing problem of stiffened composite plates.

Impact Statement

Uncertainty is everywhere in data of structural optimization and design. The goal of this paper is to help engineers and researchers efficiently solve uncertain linear optimization problems of engineering structures, in absence of information on probabilistic properties or membership functions. We quantify the uncertainties by convex polyhedron model, the generalization of the widely used interval model, which overcomes some shortages of the interval model. We present a model modification method and a vertex solution method. Our work demonstrates that the vertex solution method is effective and efficient in solving above uncertain linear optimizations. The idea is to design more reliable aerospace, mechanical, or ocean engineering structures via evaluation of the worst possible, maximum response due to uncertainties.

1. Introduction

There are many uncertainties in the design of engineering structures (Klir, 1994; Elishakoff, 1998a), such as material properties, geometric parameters, external loads, boundary conditions, and so on. If the above uncertainties are considered, the vectors and matrices in the optimization of engineering structures turn out to be no longer deterministic. The stochastic or probabilistic method, set-theoretical method, and fuzzy theory can be used to describe those uncertainties. If the uncertain parameter is the random variable

with a known probability distribution, the probabilistic methodology is preferred to deal with it. If the uncertainty parameters are complex and difficult to describe crisply, the fuzzy sets-based theory is adopted to deal with them. In the absence of such information or when there is no sufficient data available, the nonprobability convex set model is more practical and widely used in the analysis and design of engineering structures (Elishakoff, 1995; Elishakoff, 1998b).

Natural question arises: Why is there no sufficient data available? The response is as follows (Ley et al., 2020): “in 2012, IBM stated that 90% of the data available today have been generated over past 2 years,” according to Sagioglu and Sinanc (2013). When technical companies share the data, probably by creating international data banks, sufficient data will be available to utilize the proposed methodology. This applies to the case of both small data and big data. Why? This is since in both cases, in order to make structures more reliable, information on best and worst possible performances is crucial. Indeed, if the predicted worst response exceeds the allowable level, we can be convinced that the structure will fail.

With presence of uncertain parameters, the deterministic mathematical programming problems are transformed into relevant uncertain mathematical programming problems depending on the adopted modelling technique of uncertainty. There are many studies on uncertain linear programming. For example, there are active methods and passive methods for stochastic linear programming (Sengupta et al., 1963). The active approach solves the problem by introducing new design variables. The passive method accurately or approximately estimates the characteristics of the optimal value of the objective function. Evers (1967) proposed a new model to introduce the terms related to random constraints into the objective function, and the comparison results showed that taking the average of the elements could lead to considerable costs. For fuzzy linear programming, Zimmermann (1978) and Chanas (1989) used fuzzy linear programming to solve the multi-objective linear programming problem Ramík (1986) generalized the extension principle to set-to-set mappings to solve fuzzy optimization problems. Besides, multi-objective fuzzy linear programming has been studied in Stanculescu et al. (2003). Among various nonprobabilistic convex models, the interval model is more widely used (Qiu et al., 2006). For interval linear programming, Rohn (1980, 1981) presented a duality theorem and various forms of optimality criteria, as well as proved the strong solvability. Chinneck and Ramadan (2000) proposed an algorithm of interval linear programming.

However, as one of the most widely used nonprobabilistic convex models, the interval model has obvious defects in quantifying variables. In the interval method, each uncertain variable or parameter is quantified by a respective interval. In the situation of multiple variables, the whole region can be regarded as a hypercube, with surfaces of the hypercube being parallel or perpendicular to all coordinate planes. Therefore, the interval model fails to reflect the possible dependence between variables. As a consequence, the design results are supposed to be too conservative.

This study argues that if the interval model is generalized, for example, using convex polyhedron to quantify uncertainty, the results will not be as conservative as the interval model. It can also reflect the dependence between variables, because all the variables or parameters can be described by convex polyhedron as a whole, with each surface of the convex polyhedron not necessarily being parallel or perpendicular to the coordinate plane. The comparison between the interval model and the convex polyhedron model when quantifying uncertainty is shown in Figure 1. As is observed in Figure 1, the points are samples, the cuboid representing the interval model. Figure 1 also shows the polyhedron that includes all experimental points. Obviously, the interval model is anticipated to yield more conservative results than the convex polyhedron model. This is simply because the cuboid contains a bigger volume.

The availability of data which will be enclosed by polyhedron is central to this investigation. In this respect, two quotes appear to be pertinent; one belongs to Galileo Galilei: “Measure what is measurable, and make measurable what is not so.” Likewise, Lord Kelvin stated “When you can measure what you are speaking about, and express it in numbers, you know something about it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely, in your thoughts advanced to the stage of science.” Here, we use the available data without assumptions on probability density, or stationarity, or Gaussianity, or ergodicity which are frequently postulated in numerous works. According to Ley et al. (2020), “One of the biggest

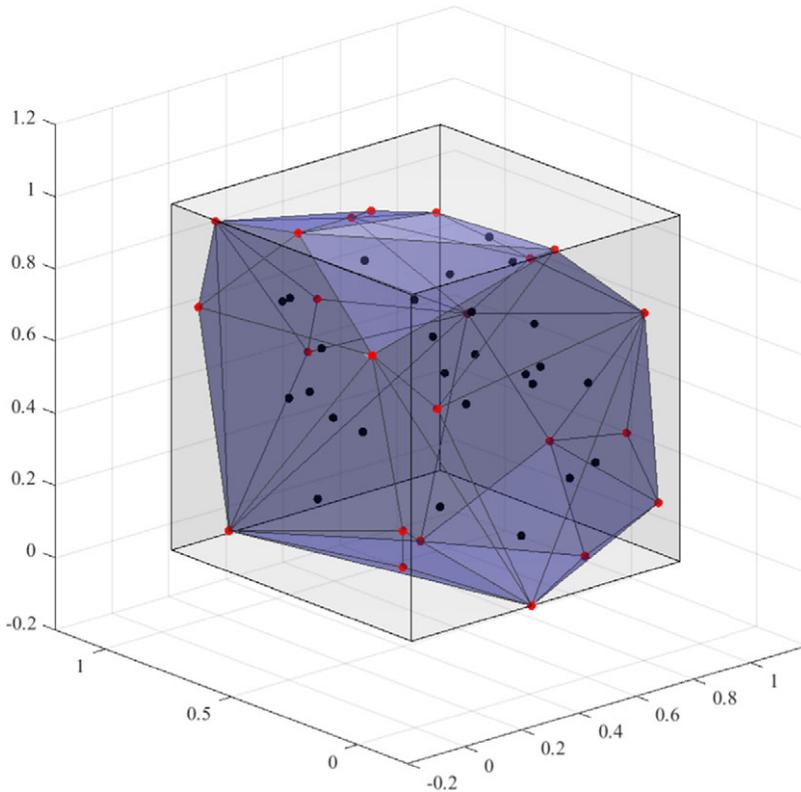


Figure 1. Comparison between interval model and convex polyhedron model.

misconceptions related to the Big Data phenomenon is that the sole fact of having massive data sets will solve all problems. Nothing is less true, because data do not speak by themselves (though this sentence is often incorrectly used).” In this paper, we suggest the data to be enclosed by a polyhedron and then the region to be inflated in order to anticipate future data.

The pertinent question arises on how to obtain the minimum convex polyhedron envelope of the sample points, that is, the convex hull. A variety of convex hull algorithms have been studied by present authors, such as Graham’s Scan, Divide and Conquer, Jarvis’ March, Andrew’s Monotone Chain, and Quick Hull Algorithm (Graham, 1972; Jarvis, 1973; Andrew, 1979; Preparata, 1979; Barber et al., 1996). Therefore, there is no difficulty, in principle, in introducing convex polyhedron into engineering as a new convex model to quantify uncertainty.

In addition, uncertainty problems often face the problem of solution efficiency. For the interval model with uncertainty, the vertex method is an effective method to improve the calculation efficiency. For example, Guo et al. (2009) found the bounds of structural response in the truss at the vertexes of the interval set. Qiu et al. (2006) and Qiu and Wang (2009) presented a vertex solution to estimate the static displacement bounds and dynamic response of structures with uncertain-but-bounded parameters. The interested reader can consult also with papers by Dong and Wong (1985) and Dong and Shah (1987). The interval variation region can be regarded as a hyper-cuboid, which is a special case of the convex polyhedron. However, there is no research conducted up to now on whether the vertex method is applicable to the convex polyhedron model.

We must emphasize that there is little research on how to use convex polyhedron approach to quantify uncertainty; likewise, there is no research on the uncertain linear programming with uncertain coefficients quantified by convex polyhedron models. This paper studies the uncertain linear optimization of engineering structure in the absence of detailed sample information and focuses on the characteristics

of the optimal solution set and the method of solving convex polyhedron linear programming. Section 2 introduces the uncertainty quantification method of the convex polyhedron models. Moreover, the model modification approach is employed via Chebyshev inequality, as introduced by Elishakoff and Sarlin, 2016. Section 3 defines the convex polyhedron-based linear programming. In Section 4, a practical vertex solution is discussed. Section 5 deals with the application of convex polyhedron linear programming in the static load-bearing capacity problem. In Section 6, some numerical examples are provided to verify the feasibility of the proposed vertex solution method.

2. Quantification of Sample Points by Convex Polyhedron Model

This section introduces the definition of a convex polyhedron and proposed the quantification method of uncertainties by convex polyhedron model.

Convex polyhedrons are common convex sets in the n -dimensional spaces. For example, convex polyhedrons in a two-dimensional space are polygons. In a three-dimensional space, a solid surrounded by four or more polygons is named a polyhedron, and when this polyhedron is on the same side of any face of itself, it can be called a convex polyhedron.

Convex polyhedrons can be expressed by the convex combination of vertex or by the intersection of appropriate half spaces (Rockafellar, 1970):

(a) If we know all the vertices x_i of a convex polyhedron, where x_i is a point in an n -dimensional space

$$x_i = (x_{1,i}, x_{2,i}, \dots, x_{n,i}), \quad i = 1, 2, \dots, m \tag{1}$$

then any point x in this convex polyhedron satisfies the following equality:

$$x = \sum_{i=1}^m \alpha_i x_i, \quad \left(\sum_{i=1}^m \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, m \right). \tag{2}$$

The convex polyhedron can be expressed by the following set:

$$\mathcal{X} = \left\{ x : x = \sum_{i=1}^m \alpha_i x_i, \sum_{i=1}^m \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, m \right\}. \tag{3}$$

Figure 2 gives the example of a convex polyhedron in a two-dimensional space, inside of which point x can be represented as

$$x = \sum_{i=1}^8 \alpha_i x_i, \quad \left(\sum_{i=1}^8 \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, 8 \right). \tag{4}$$

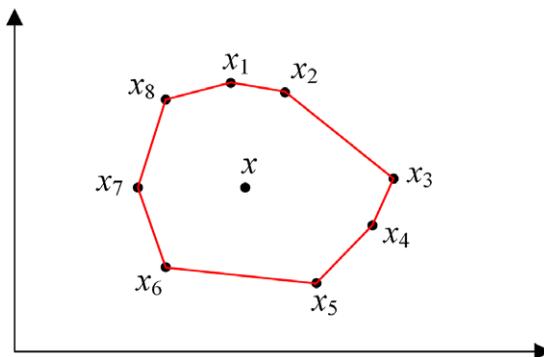


Figure 2. Example 1 of a convex polyhedron (two-dimension).

(b) On the other hand, any point in this convex polyhedron also satisfies the following inequality:

$$Ax \leq b, \quad (5)$$

where the faces of the convex polyhedron are given by

$$\begin{aligned} a_1x &= b_1, \\ a_2x &= b_2, \\ &\vdots \\ a_nx &= b_n, \end{aligned} \quad (6)$$

then the vectors in Equations (5) and (6)

$$A = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad (7)$$

in the convex polyhedron can be expressed by the following set:

$$\mathcal{X} = \{x : Ax \leq b\}. \quad (8)$$

Figure 3 gives the example of a convex polyhedron in a two-dimensional space.

2.1. Quantification of sample points by convex polyhedrons

Now we can use a convex polyhedron to quantify a limited number of sample points. For the sake of simplicity, the quantification of the following two-dimensional sample points by convex polyhedron model is first considered (note that different dimensions of each sample point (x_i, y_i) may be connected). The data points are shown in Figure 4. As is stated in the introduction, the key to the quantification is to find the convex hull of the sample points. By convex hull algorithms, we can easily find the minimum convex polyhedron that envelops all data points, which is illustrated in Figure 5.

2.2. Modification of the quantification results

Figure 3 provides data from conducted experiments. But there is a possibility that sample points from future experiments fall outside the existing geometric figures (Sarlin and Elishakoff, 2016). Figure 6 depicts some possible future experimental data, in which blue points fall inside the convex polyhedron while red points fall outside. To forecast the future data, the convex hull calculated from available data

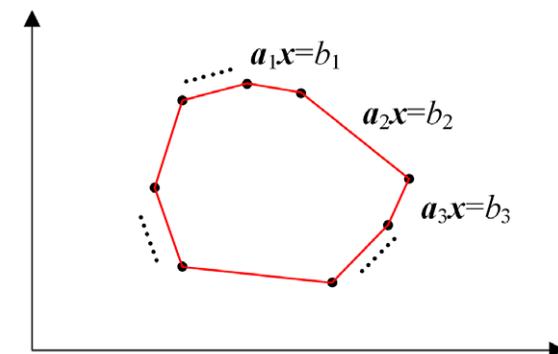


Figure 3. Example 2 of a convex polyhedron (two-dimension).

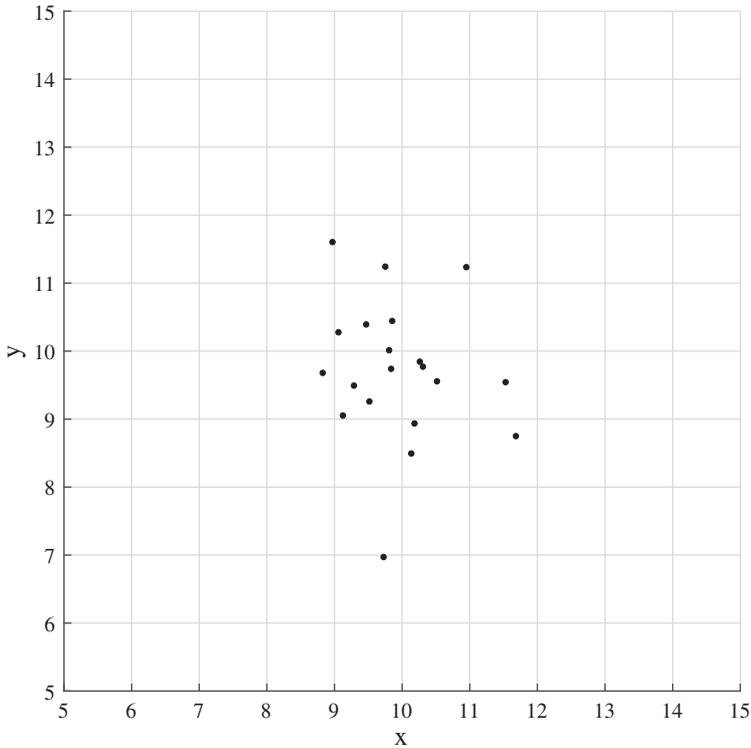


Figure 4. The set of sample points.

needs to be enlarged. A probability approach has been proposed to inflating the bounding figures by Chebyshev Inequality via suggestion by Elishakoff and Sarlin (2016):

(a) Produce the initial convex hull enclosing the set of points:

$$\mathcal{X} = \left\{ \mathbf{x} : \mathbf{x} = \sum_{i=1}^m \alpha_i \mathbf{x}_i, \sum_{i=1}^m \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, m \right\} \tag{9}$$

or

$$\mathcal{X} = \{ \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{b} \}, \tag{10}$$

where \mathbf{x} denotes all the set of points.

(b) Use Chebyshev inequality to decide the magnitude of the inflation coefficient of the initial convex hull:

$$\forall \varepsilon > 0, P\{ (\mathbf{S} - E[\mathbf{S}])^T \boldsymbol{\Sigma}^{-1} (\mathbf{S} - E[\mathbf{S}]) \geq \varepsilon \} \leq \frac{n}{\varepsilon}, \tag{11}$$

where ε is the inflation coefficient, \mathbf{S} denotes the set of points, $E[\mathbf{S}]$ represents the mean value of \mathbf{S} , $\boldsymbol{\Sigma}$ is the variance–covariance matrix, and n is the dimension of this problem. If ε has been chosen, we get an ellipse (or an n -dimensional ellipsoid) E_{ch} as follows

$$(\mathbf{x} - E[\mathbf{S}])^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - E[\mathbf{S}]) = \varepsilon \tag{12}$$

(c) With the mean value of data as the center, enlarge the initial convex hull until inscribed to the ellipse E_{ch} . The probability of future points falling outside the new convex polyhedron should be less than $\frac{n}{\varepsilon}$ according to Equation (11).

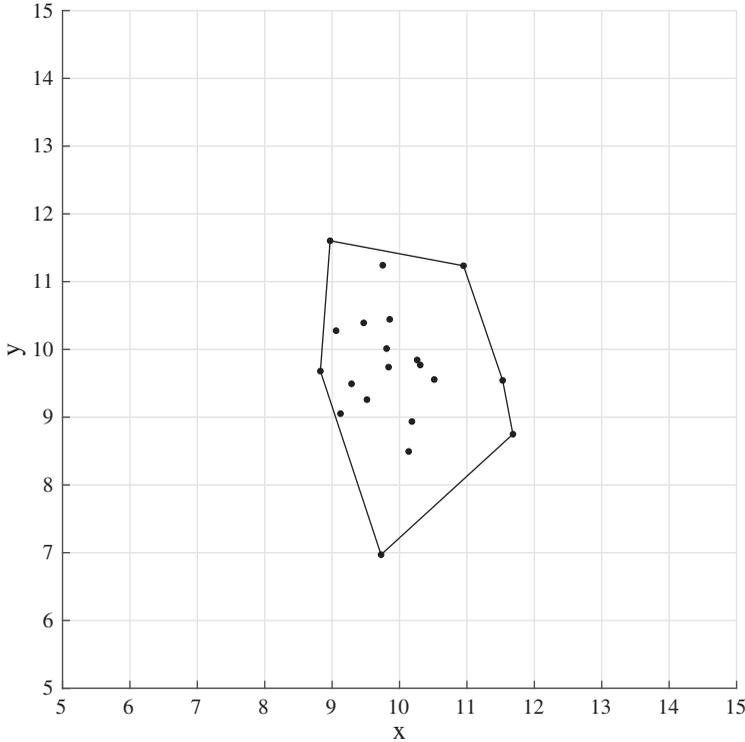


Figure 5. The quantification of sample points by convex polyhedron model.

Notice that it is not straightforward to find the exact enlarging coefficient especially for n -dimension conditions. If we denote the enlarging coefficient as k , the volume of the n -dimensional convex polyhedron as V_p , the volume of the n -dimensional ellipsoid as V_E , then the coefficient can be approximately computed by

$$k = \sqrt[n]{V_E/V_p}. \tag{13}$$

Although the enlarged convex polyhedron is not inscribed to the ellipse in this condition, it is a practical method to obtain k in engineering practice.

Now we apply this method to modify the original convex hull in Figure 5 and check the anticipated future data in Figure 6. The actions to be undertaken are as follows:

- (a) Produce an initial convex hull.

This convex hull has been produced in Figure 5.

- (b) Choose the magnitude of the inflation coefficient.

$E[S]$ and Σ can be obtained from the initial data in Figure 4:

$$E[S] = (4.9395, 4.7144)^T, \tag{14}$$

$$\Sigma = \begin{pmatrix} 0.62194 & -0.12696 \\ -0.12696 & 1.10190 \end{pmatrix}. \tag{15}$$

If we let $\varepsilon = 8$, the ellipse E_{ch} is as follows

$$(\mathbf{x} - E[S])^T \Sigma^{-1} (\mathbf{x} - E[S]) = 8 \tag{16}$$

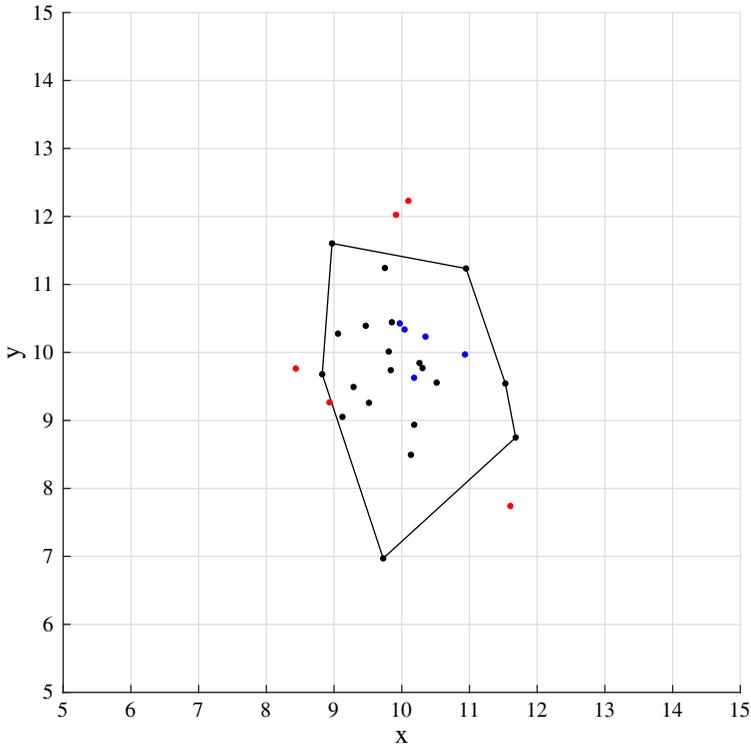


Figure 6. The position between hypothesized future sample points and the convex polyhedron.

and the probability of future points outside of the ellipse will be $P \leq 2/8 = 0.25$. Figure 7 shows the ellipse.

(c) Enlarge the initial convex polyhedron.

Use the approximate method to calculate the enlarging coefficient:

$$k = \sqrt[n]{V_E/V_p} \approx 1.538. \tag{17}$$

With the enlarging coefficient k , every new point \mathbf{x}' of the enlarged convex polyhedron can be expressed as

$$\mathbf{x}' = E[\mathbf{S}] + k(\mathbf{x} - E[\mathbf{S}]). \tag{18}$$

The equations of convex polyhedron will be the following, according to Equations (9) and (10)

$$\mathcal{X}' = \left\{ \mathbf{x}' : \mathbf{x}' = \sum_{i=1}^m \alpha_i \mathbf{x}'_i, \sum_{i=1}^m \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, m \right\} \tag{19}$$

or

$$\mathcal{X}' = \{ \mathbf{x}' : \mathbf{A} [k^{-1}(\mathbf{x}' - E[\mathbf{S}]) + E[\mathbf{S}]] \leq \mathbf{b} \}. \tag{20}$$

The result shows in Figure 8: All the future points, with certain probability, associated with the Chebyshev inequality, fall into the new convex polyhedron, as is depicted in Figure 9.

3. Problem Statement

The standard form of the linear programming model is as follows:

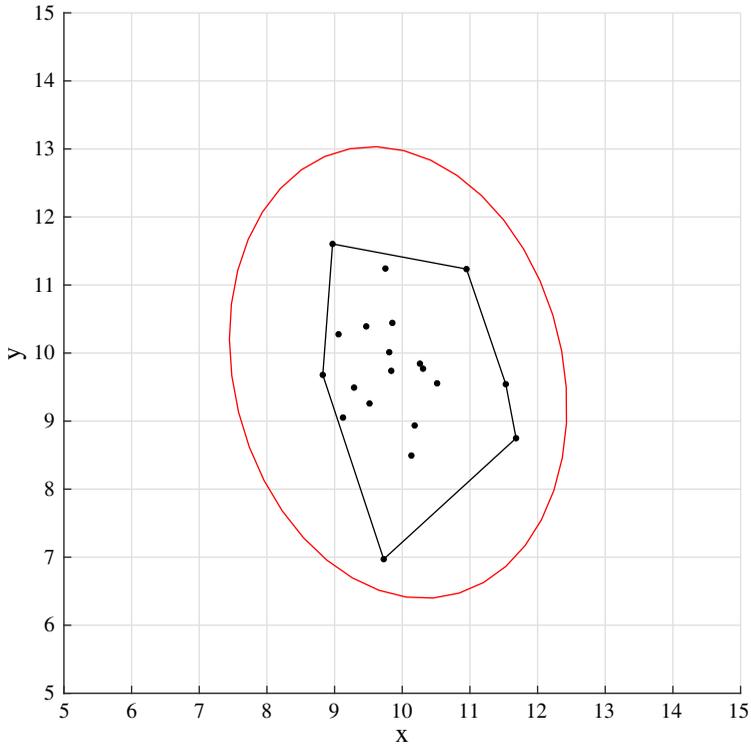


Figure 7. The ellipse enclosing the convex hull.

$$\begin{aligned}
 & \min \{ \mathbf{c}^T \mathbf{x} \} \\
 & \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}, \\
 & \mathbf{x} \geq \mathbf{0}
 \end{aligned} \tag{21}$$

where \mathbf{A} is a deterministic matrix, while \mathbf{b} and \mathbf{c} are deterministic vectors. In engineering applications, when the uncertainty of material properties, external load, and so on, cannot be neglected, uncertain parameters \mathbf{A} , \mathbf{b} and \mathbf{c} should be used instead of deterministic ones.

In this study, for parameters with unknown but bounded uncertainties, the convex polyhedron model is introduced for quantification:

$$\begin{aligned}
 \mathcal{A} &= \left\{ \mathbf{A} : \mathbf{A} = \sum_{i=1}^{n_A} \alpha_i \mathbf{A}_i, \sum_{i=1}^{n_A} \alpha_i = 1, \alpha_i \geq 0, i = 1, 2, \dots, n_A \right\}, \\
 \mathcal{B} &= \left\{ \mathbf{b} : \mathbf{b} = \sum_{j=1}^{n_b} \beta_j \mathbf{b}_j, \sum_{j=1}^{n_b} \beta_j = 1, \beta_j \geq 0, j = 1, 2, \dots, n_b \right\}, \\
 \mathcal{C} &= \left\{ \mathbf{c} : \mathbf{c} = \sum_{k=1}^{n_c} \gamma_k \mathbf{c}_k, \sum_{k=1}^{n_c} \gamma_k = 1, \gamma_k \geq 0, k = 1, 2, \dots, n_c \right\},
 \end{aligned} \tag{22}$$

where \mathbf{A}_i , \mathbf{b}_j , and \mathbf{c}_k denote vertexes of the polyhedrons \mathcal{A} , \mathcal{B} , and \mathcal{C} , respectively. For example, if \mathbf{A}_i is a $m \times n$ matrix, it can be regarded as a point in $R^{m \times n}$, therefore \mathcal{A} represented by \mathbf{A}_i is a polyhedron in space $R^{m \times n}$.

Then, the convex polyhedron model of linear programming problem is given by

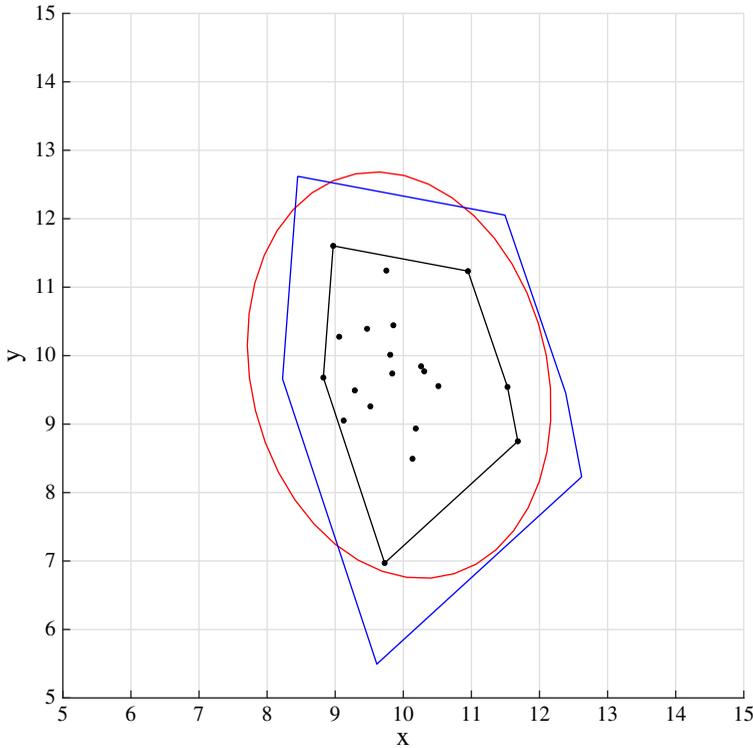


Figure 8. The enlarged convex polyhedron.

$$\begin{aligned}
 & \min \{c^T x\} \\
 & \text{s.t. } Ax = b \\
 & \quad c \in C, A \in \mathcal{A}, b \in \mathcal{B} \\
 & \quad x \geq 0
 \end{aligned} \tag{23}$$

In which \mathcal{A} , \mathcal{B} , and \mathcal{C} are quantified through Equation (22). The optimal solutions of convex polyhedron linear programming (23) should be a set denoted as X , which contains optimal solutions of determining linear programming for every A , b , and c of \mathcal{A} , \mathcal{B} , and \mathcal{C} , respectively:

$$X = \{x^* : x^* \text{ is an optimal solution of (1), } c \in C, A \in \mathcal{A}, b \in \mathcal{B}\}. \tag{24}$$

4. Vertex Solution Method of Convex Polyhedron Linear Programming

In this section, the vertex solution for efficiently solving convex polyhedron linear programming (23) is discussed.

4.1. Notes about optimal solutions

For any A , b , and c in Equation (23), they can be expressed by the linear combination vertexes of polyhedrons \mathcal{A} , \mathcal{B} , and \mathcal{C} :

$$A = \sum_{i=1}^{n_A} \alpha_i A_i, \quad b = \sum_{j=1}^{n_b} \beta_j b_j, \quad c = \sum_{k=1}^{n_c} \gamma_k c_k. \tag{25}$$

According to the fundamentals of the linear programming problem, the optimal solution is obtained at the vertices of the convex set of its feasible region. As the coefficient α_i , β_j , and γ_k change, A , b , and c as well

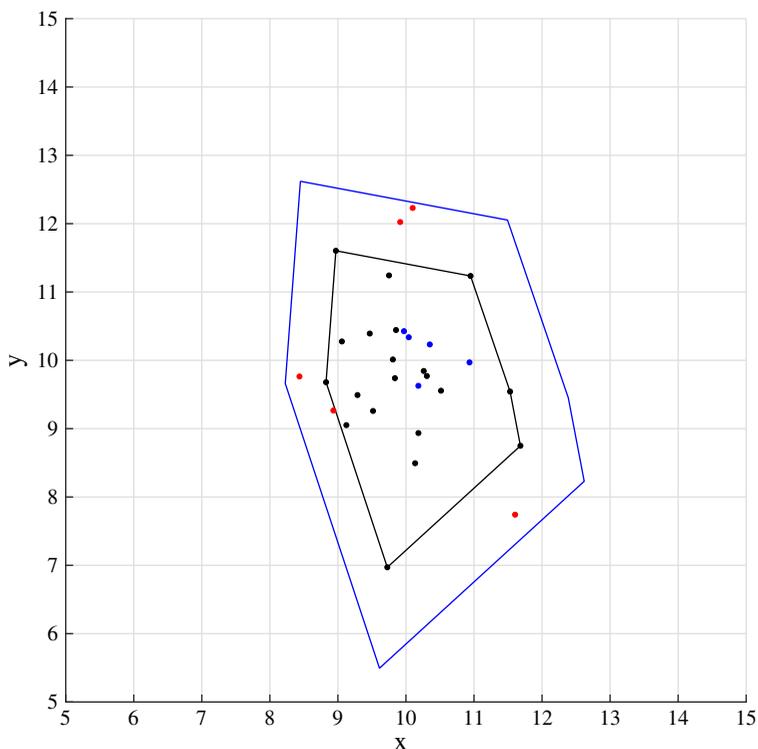


Figure 9. The position relationship between future sample points and the inflated convex polyhedron.

as the optimal solution change, but the optimal solution is still obtained at the vertices of the new feasible region.

Therefore, if we choose one vertex of \mathcal{A} , \mathcal{B} , and \mathcal{C} denoted as \mathbf{A}_i , \mathbf{b}_j , and \mathbf{c}_k respectively, the following single linear programming model is obtained:

$$\begin{aligned} \min \quad & \{ \mathbf{c}_k^T \mathbf{x} \} \\ \text{s.t.} \quad & \mathbf{A}_i \mathbf{x} = \mathbf{b}_j. \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{26}$$

One then represents the optimal solution of Equation (26) as \mathbf{x}_{ijk}^* , and solves the above linear programming problem for any combination of \mathbf{A}_i , \mathbf{b}_j , \mathbf{c}_k . A set of those optimal solutions is as follows:

$$\mathbf{X}^v = \left\{ \mathbf{x}_{ijk}^* : i = 1, 2, \dots, n_A, j = 1, 2, \dots, n_B, k = 1, 2, \dots, n_C \right\}. \tag{27}$$

Since the programming problem (23) is linear, it is natural to conjecture that the set of linear combinations of \mathbf{x}_{ijk}^* (denoted by \mathbf{X}') may be used to approximately quantify the solution set \mathbf{X} :

$$\mathbf{X}' = \left\{ \mathbf{x}^* : \sum_{i,j,k} \mu_i v_j v_k \mathbf{x}_{ijk}^*, \sum_i \mu_i = 1, \sum_j v_j = 1, \sum_k v_k = 1, \mu_i, v_j, v_k \geq 0, \right. \\ \left. i = 1, \dots, n_A, j = 1, \dots, n_B, k = 1, \dots, n_C \right\}. \tag{28}$$

However, the correctness of this conjecture should be rigorously established. In Section 4.2, this conjecture is proved. In Section 4.3, a vertex solution method of convex polyhedron linear programming problem is presented.

4.2. Properties of the optimal solution set

First of all, the necessary condition for every optimal solution is proved. In Appendix A, the relation between the optimal solution set X and the set X' is studied.

This section studies the properties of the optimal solution and gives a necessary condition that every optimal solution of (23) should satisfy. To simplify the discussion, the following assumption is made: Small variations ΔA , Δb , and Δc will change the position of the optimal solution without altering the numbers of vertices and the corresponding vertices of the optimal solution.

First of all, A, b, c can be expressed by the sum of nominal value and variation:

$$A = A_0 + \Delta A, \quad b = b_0 + \Delta b, \quad c = c_0 + \Delta c \tag{29}$$

then vertices of polyhedrons \mathcal{A} , \mathcal{B} , and \mathcal{C} can be written as

$$A_i = A_0 + \Delta A_i, \quad b_j = b_0 + \Delta b_j, \quad c_k = c_0 + \Delta c_k. \tag{30}$$

By substitution of (30) into (22), every A , b , and c will be represented as follows

$$\begin{aligned} A &= \sum_i A_i = \sum_i \alpha_i (A_0 + \Delta A_i), \\ b &= \sum_j b_j = \sum_j \beta_j (b_0 + \Delta b_j), \\ c &= \sum_k c_k = \sum_k \gamma_k (c_0 + \Delta c_k). \end{aligned} \tag{31}$$

A linear programming problem for A_0 , b_0 , and c_0 is

$$\begin{aligned} \min \quad & \{c_0^T x\} \\ \text{s.t.} \quad & A_0 x = b_0. \\ & x \geq 0 \end{aligned} \tag{32}$$

The optimal solution of (32) is denoted as x_0^* . According to Equation (31), a linear programming problem concerning A , b , and c can be converted to

$$\begin{aligned} \min \quad & \left\{ \left[\sum_k \gamma_k (c_0 + \Delta c_k) \right]^T x \right\} \\ \text{s.t.} \quad & \left[\sum_i \alpha_i (A_0 + \Delta A_i) \right] x = \sum_j \beta_j (b_0 + \Delta b_j) \\ & x \geq 0 \end{aligned} \tag{33}$$

On the one hand, denote the optimal solution of (33) as x^* . The variation between x^* and x_0^* is Δx^*

$$x^* = x_0^* + \Delta x^*. \tag{34}$$

Substitution of (34) into the constraint condition of (33), and using the constraint condition of (32), the following equation is obtained

$$\sum_i \alpha_i A_i x^* = \sum_j \beta_j b_j. \tag{35}$$

On the other hand, according to the assumption in the second paragraph of this section, c_k does not affect the optimal solution. For any combination of A_i, b_j, c_k , the vertex solution of the corresponding linear programming problem is expressed as $x_{ij}^* = x_{0,ij}^* + \Delta x_{ij}^*$. The set of x_{ij}^* is denoted by

$$X^v = \left\{ x_{ij}^* : i = 1, 2, \dots, n_A, j = 1, 2, \dots, n_b \right\}. \tag{36}$$

The linear combination of x_{ij}^* is represented by

$$X' = \left\{ \mathbf{x} : \sum_{i,j} \mu_i v_j \mathbf{x}_{ij}^*, \sum_i \mu_i = 1, \sum_j v_j = 1, \mu_i \geq 0, v_j \geq 0, i = 1, 2, \dots, n_A, j = 1, 2, \dots, n_B \right\}. \quad (37)$$

The solution \mathbf{x}_{ij}^* should satisfy the constraint condition:

$$A_i \mathbf{x}_{ij}^* = \mathbf{b}_j. \quad (38)$$

It can be shown that every optimal solution \mathbf{x}^* can be expressed by the linear combination of vertex optimal solutions \mathbf{x}_{ij}^* . The detailed demonstration is given in [Appendix A](#). The following conclusions are drawn:

The optimal solution set X of convex polyhedron linear programming (23) is contained in set X' :

$$X \subseteq X' \quad (39)$$

which means that for any $A \in \mathcal{A}, \mathbf{b} \in \mathcal{B}$, and $\mathbf{c} \in \mathcal{C}$, the optimal solution of the corresponding linear programming can be regarded as the linear combination of \mathbf{x}_{ij}^* . Therefore, we can use X' to approximate the optimal solution set X . Furthermore, notice that \mathbf{x}_{ij}^* is not necessarily the vertex of X' , but the convex hull of X^v can be easily calculated to express X' in practical application:

$$X' = co(X^v). \quad (40)$$

Determining X by [Equation \(40\)](#) is more efficient with the help of various convex hull algorithms.

4.3. Vertex solution method

In short, the following vertex solution can be formulated for convex polyhedron linear programming:

Vertex Solution Statement: For a convex polyhedron linear programming problem (23), the optimal solution set of is approximately a polyhedron X , the vertices of X are elements of the set X^v , and X can be expressed by the convex hull of X^v .

Note that we use “approximately” in the vertex solution statement, because the proof in [Appendix A](#) is based on [Equation \(A.4\)](#), which is a “necessary condition” of optimal solutions. Although the vertex solution statement holds approximately, the proof in [Appendix A](#) indicates that all the optimal solutions should be contained in set X' , which means no optimal solutions will be missed when using this statement.

When using the vertex solution, first we need to solve linear programming for any combination of $A_i, \mathbf{b}_j, \mathbf{c}_k$ and obtain optimal solution \mathbf{x}_{ijk}^* :

$$\begin{aligned} \min \quad & \{ \mathbf{c}_k^T \mathbf{x} \} \\ \text{s.t.} \quad & A_i \mathbf{x} = \mathbf{b}_j. \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \quad (41)$$

Then set X^v can be expressed as:

$$X^v = \left\{ \mathbf{x}_{ijk}^* : i = 1, 2, \dots, n_A, j = 1, 2, \dots, n_B, k = 1, 2, \dots, n_C \right\}. \quad (42)$$

Finally, the optimal solution set of [Equation \(23\)](#) can be calculated by any convex hull algorithm:

$$X \approx co(X^v). \quad (43)$$

[Figure 10](#) gives the flow chart of the vertex solution algorithm.

Note that the material in [Section 4](#) is based on the standard form of convex polyhedron linear programming problem, which means the constraint conditions should be linear equations instead of linear inequalities. Therefore, in the subsequent numerical examples, each established convex polyhedron linear programming will be converted into the standard form.

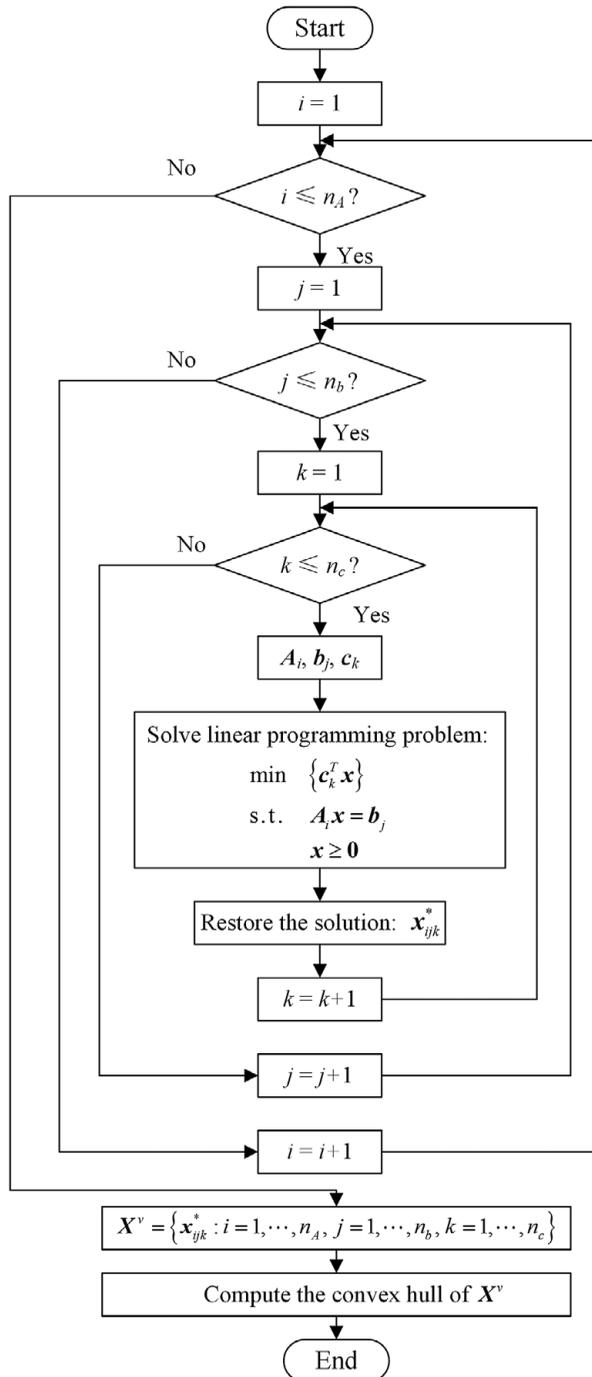


Figure 10. Flow chart of vertex solution.

5. Application of Convex Polyhedron Linear Programming: Static Load-Bearing Capacity Problem

In this section, the static load-bearing capacity problem of a structure is taken as an example to show how to construct a model of convex polyhedron linear programming.

The static response equation of structure is given in the standard form:

$$\mathbf{K}\mathbf{u} = \mathbf{f}, \quad (44)$$

where \mathbf{K} is a $n \times n$ matrix and \mathbf{u}, \mathbf{f} are $n \times 1$ vectors. Considering the uncertainties in material and geometry, the stiffness matrix and load vector can be quantified as belonging to the convex polyhedra (the dimension should be smaller than n^2 and n , respectively). Moreover, for linear elastic structures, if only elastic modulus belongs to a convex polyhedron \mathcal{E} , the convex polyhedral dimensions of the stiffness matrix also belong to a convex polyhedron \mathcal{K} , and the dimension of \mathcal{K} is the same as \mathcal{E} , since the stiffness matrix is linear with respect to the elastic modulus. Therefore, in the problem where only the elastic modulus is uncertain, the convex polyhedron of stiffness matrices can be converted to the convex polyhedron of the elastic modulus to improve efficiency.

Assume that the strength requirement of the structure is satisfied, and the displacements are constrained by inequality $\mathbf{u} \leq \mathbf{u}_0$. The load-bearing capacity problem can be modeled as following uncertain linear programming:

$$\begin{aligned} \min \quad & \{\mathbf{c}^T \mathbf{f}\} \\ \text{s.t.} \quad & \mathbf{D}\mathbf{f} \leq \mathbf{u}_0 \\ & \mathbf{D} \in \mathcal{D}, \\ & \mathbf{f} \geq \mathbf{0} \end{aligned} \quad (45)$$

where \mathbf{D} is the flexibility matrix. It can be proved that \mathcal{D} can also be regarded as a convex polyhedron. A detailed demonstration is given in Appendix B. Therefore, the flexibility matrix can be approximated as a convex polyhedron, and (45) then constitutes a convex polyhedron linear programming problem. In addition, Equation (B.7) indicates that \mathbf{D} is approximately linear with respect to elastic modulus since Δk_{s_j} is linear concerning the elastic modulus. This reminds us of using the convex polyhedron of the elastic modulus instead of that of a flexibility matrix in practice when only the elastic modulus is uncertain.

Furthermore, (45) can be converted to a standard form of convex polyhedron linear programming:

$$\begin{aligned} \min \quad & \{\mathbf{c}'^T \mathbf{f}'\} \\ \text{s.t.} \quad & \mathbf{G}\mathbf{f}' = \mathbf{u}_0 \\ & \mathbf{G} \in \mathcal{G}, \\ & \mathbf{f}' \geq \mathbf{0} \end{aligned} \quad (46)$$

$$\mathbf{G} = (\mathbf{D}\mathbf{I}), \mathbf{f}' = \begin{pmatrix} \mathbf{f} \\ \mathbf{f}_0 \end{pmatrix}, \mathbf{c}' = \begin{pmatrix} \mathbf{c} \\ \mathbf{c}_0 \end{pmatrix}, \quad (47)$$

where \mathcal{G} is also a convex polyhedron because \mathbf{I} is an identity matrix.

In the subsequent section, two static load-bearing capacity problems of plane truss and stiffened composite plate, are provided as numerical examples.

6. Numerical Examples

Two examples will be presented to illustrate the structural optimization method through the convex polyhedron model in Section 6. The first example focuses on the modeling approach by studying the load-bearing capacity of a plane truss. The second case provides an uncertain optimization of a more complex engineering structure to show the feasibility of the proposed convex polyhedron model.

6.1. The load-bearing capacity of plane truss

The first example considers the load-bearing capacity of plane truss with displacement constraint, as shown in Figure 11. Nodes 4, 5, 8, and 9 are fixed. The y -direction load is applied at node 1. The nominal elastic modulus and cross-sectional area of each rod are 210 GPa and $1 \times 10^{-4} \text{ m}^2$, respectively.

Due to the scatter in the material properties and manufacturing errors, the elastic modulus and cross-sectional area are uncertain. The sample points of elastic modulus and cross-sectional area are plotted in Figure 12, where the blue convex polyhedron is an inflated model by the method of 2.2 with the inflation coefficient $\varepsilon = 15$. To facilitate comparison with another method, the data in Figure 12 comes from two normal distribution:

$$E \sim N(210 \text{ GPa}, 10^2 \text{ GPa}^2), A \sim N(1 \times 10^{-4} \text{ m}^2, 1 \times 10^{-14} \text{ m}^4). \tag{48}$$

The elastic moduli and cross-sectional areas can be different for each rod in this simple example.

Assume that the truss structure meets strength requirements, and the displacements of node 1 in x and y -directions are constrained. The displacement vector, load vector, and the corresponding stiffness matrix are $\mathbf{u} = (u_x, u_y)^T$, $\mathbf{f} = (0, f_y)^T$ and $\mathbf{K} = \{K_{11}, K_{12}; K_{21}, K_{22}\}$, respectively, then we have:

$$\mathbf{K}\mathbf{u} = \mathbf{f} \tag{49}$$

Using flexibility matrix:

$$\mathbf{D} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} = \mathbf{K}^{-1} = \begin{bmatrix} K_{22} & -K_{12} \\ -K_{21} & K_{11} \end{bmatrix} / \det(\mathbf{K}) \tag{50}$$

then Equation (49) can be transformed into:

$$\mathbf{D}\mathbf{f} = \mathbf{u} \tag{51}$$

As is shown in Section 5, the flexibility matrix can be approximated as a convex polyhedron. Let the displacement constraint be $\mathbf{u} \leq \mathbf{u}_0$, and the convex polyhedron model of the linear programming problem is as follows:

$$\begin{aligned} \min & f_y \\ \text{s.t.} & \mathbf{D}\mathbf{f} \leq \mathbf{u}_0 \\ & \mathbf{D} \in \mathcal{D} \\ & f_y \geq 0 \end{aligned} \tag{52}$$

Let $\mathbf{c} = (-1, 0, 0)^T$ and transform (52) into the form of (23), then we have

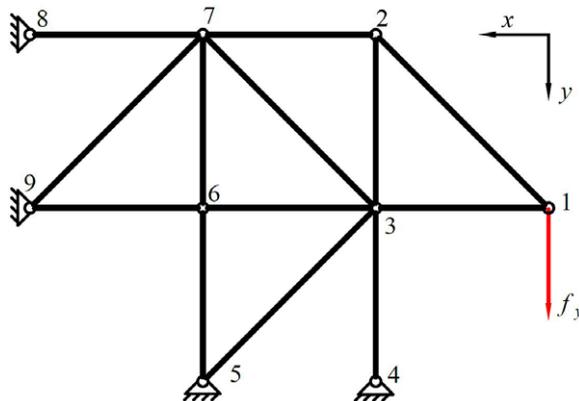


Figure 11. Plane truss.

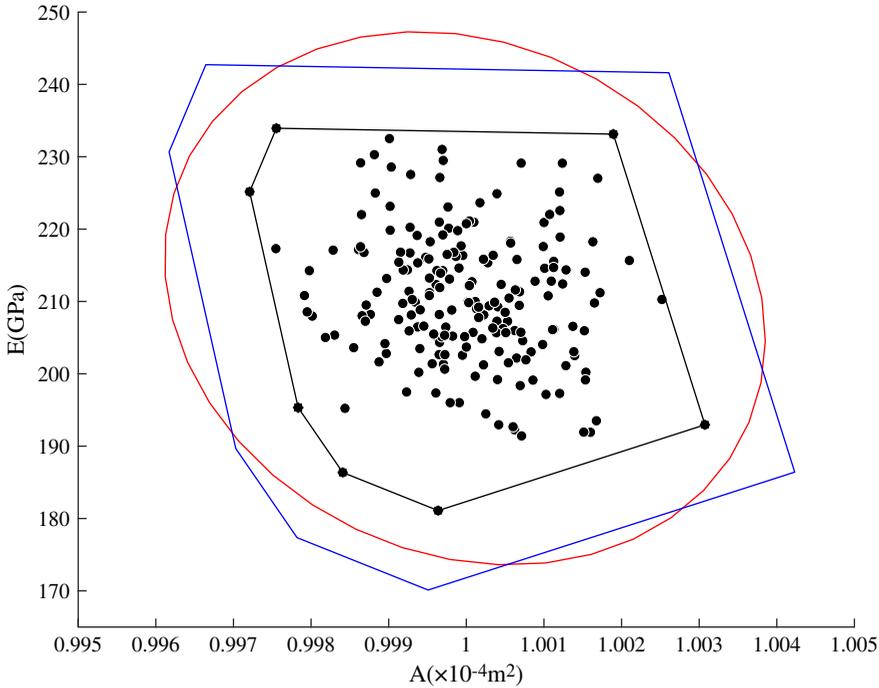


Figure 12. Elastic modulus and cross-sectional area sample points.

$$\begin{aligned}
 & \min \quad \mathbf{c}^T (f_y, x_1, x_2)^T \\
 & \text{s.t.} \quad \mathbf{G} (f_y, x_1, x_2)^T = \mathbf{u}_0 \\
 & \quad \mathbf{G} \in \mathcal{G} \\
 & \quad (f_y, x_1, x_2) \geq \mathbf{0}
 \end{aligned} \tag{53}$$

In which $\mathbf{G} = [\mathbf{G}_1 \quad \mathbf{G}_2]$, $\mathbf{G}_1 = \begin{bmatrix} D_{12} \\ D_{22} \end{bmatrix}$, $\mathbf{G}_2 = \begin{bmatrix} 1 \times 10^{-6} & 0 \\ 0 & 1 \times 10^{-6} \end{bmatrix} m/N$.

Next, we solve convex polyhedron linear programming (53) by vertex solution. First of all, using the data in Figure 11 to evaluate the convex hull of \mathbf{G}_1 . Then use the vertex solution to obtain the vertexes of the optimal solutions and the objective function values, which are listed in Table 1.

As is shown in Table 2, the lower bound of objective function value is $f_{\min}^* = -117.74 \text{ kN}$ and the best optimal solution is $\mathbf{x}^* = (1.17743, 1.09E - 14, 0.001381)^T \text{ kN}$, therefore the maximum load-bearing capacity is 117.74 kN ; the upper bound of objective function value is $f_{\max}^* = -79.29 \text{ kN}$ and the worse optimal value is $\mathbf{x}^* = (0.792884, 3.39E - 10, 0.065673)^T \text{ kN}$, therefore the minimum load-bearing capacity is 79.29 kN . Figure 13 plots the convex hull of the optimal solution set. In conclusion, to meet the design requirements, the maximum load should not be higher than 79.29 kN .

To verify the effectiveness of our method, the Monte-Carlo simulation method was used for comparison. The elastic modulus and cross-sectional area are normally distributed in Equation (48). The load f_y is chosen as the random variable, and the distribution function is obtained after 10,000 simulations as shown in Figure 14.

Table 2 gives a comparison between convex polyhedron vertex solution method (CPVSM), interval method (IM), and Monte-Carlo method (MCM). In the CPVSM method, the geometric gravity center is

Table 1. Vertices of optimal solutions and objective function values ($\times 10^5 N$).

| f_y | x | | f_{\min} |
|----------|----------|----------|------------|
| | x_1 | x_2 | |
| 1.176636 | 1.21E-14 | 0.001637 | -1.17664 |
| 1.100991 | 3.23E-15 | 0.021255 | -1.10099 |
| 1.069615 | 4.66E-17 | 0.028607 | -1.06961 |
| 0.866496 | 1.52E-12 | 0.079934 | -0.8665 |
| 0.796667 | 1.41E-15 | 0.075442 | -0.79667 |
| 0.792884 | 1.6E-14 | 0.072516 | -0.79288 |
| 0.792884 | 5.73E-13 | 0.08144 | -0.79288 |
| 0.792884 | 3.39E-10 | 0.065673 | -0.79288 |
| 0.792884 | 2.38E-17 | 0.027569 | -0.79288 |
| 0.803715 | 3.69E-15 | 0.016411 | -0.80371 |
| 0.825314 | 2.29E-14 | 0.001878 | -0.82531 |
| 0.825872 | 1.61E-14 | 0.001633 | -0.82587 |
| 0.833115 | 0.001022 | 2.35E-14 | -0.83311 |
| 0.837404 | 0.002552 | 2.34E-14 | -0.8374 |
| 0.874386 | 0.011795 | 2.29E-14 | -0.87439 |
| 0.898555 | 0.015576 | 2.27E-14 | -0.89855 |
| 0.903728 | 0.015592 | 2.27E-14 | -0.90373 |
| 1.131539 | 0.004654 | 2.57E-14 | -1.13154 |
| 1.161805 | 0.002164 | 2.61E-14 | -1.1618 |
| 1.17743 | 1.09E-14 | 0.001381 | -1.17743 |

Table 2. Comparison between convex polyhedron vertex solution method, interval method, and Monte-Carlo method.

| | Nominal value | Lower bound | Upper bound |
|--------------------|---------------|-------------|-------------|
| | (kN) | (kN) | (kN) |
| CPVSM | 92.27 | 79.29 | 117.74 |
| IM | 99.62 | 85.28 | 113.95 |
| MCM | 102.17 | 84.73 | 120.97 |
| Relative error (%) | -7.53 | 0.65 | -6.18 |

Abbreviations: CPVSM, convex polyhedron vertex solution method; IM, interval method; MCM, Monte-Carlo method.

regarded as the nominal value of the optimal solution. In the interval method, the midpoint of the optimal solution interval is regarded as the nominal value. It is obvious that the optimal solution set of CPVSM is included in the result of the MCM from Table 2, and there is a difference between the nominal values of CPVSM and IM. Table 3 provided the quantiles of the Monte-Carlo method. It is found that the lower bound 79.29 kN and nominal value 92.27 kN obtained by the convex polyhedron method are both conservative for truss design.

6.2. The load-bearing capacity of stiffened composite plate

The stiffened composite plates composed of laminated plates and stiffeners have been widely used in aircraft structures, which not only ensures the structural strength, but also reduces the manufacturing cost. The stiffened composite plate considered in this paper is shown in Figure 15. The plate consists of one

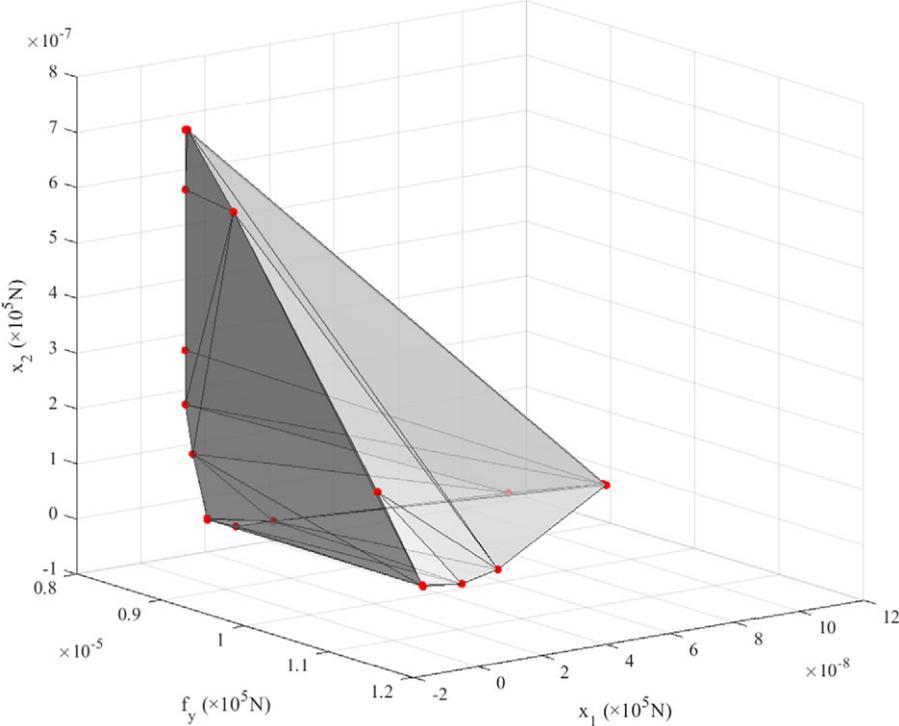


Figure 13. Convex hull of optimal solutions ($\times 10^5 N$).

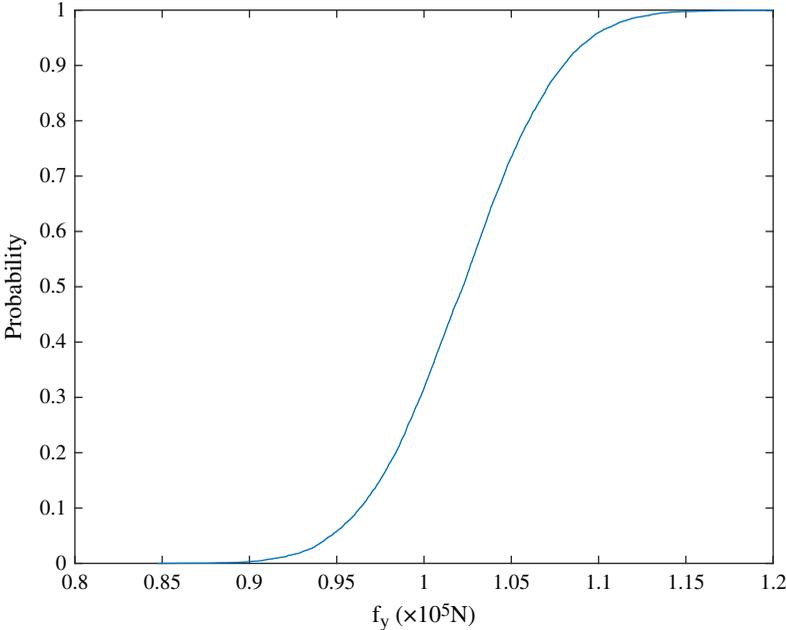


Figure 14. Distribution function of f_y .

Table 3. Quantiles of the Monte-Carlo method ($\times 10^5$ N).

| | | | | |
|------------------------------|--------|--------|--------|--------|
| | 0.001 | 0.005 | 0.01 | 0.1 |
| $f_y (\times 105 \text{ N})$ | 0.8813 | 0.9070 | 0.9162 | 0.9632 |

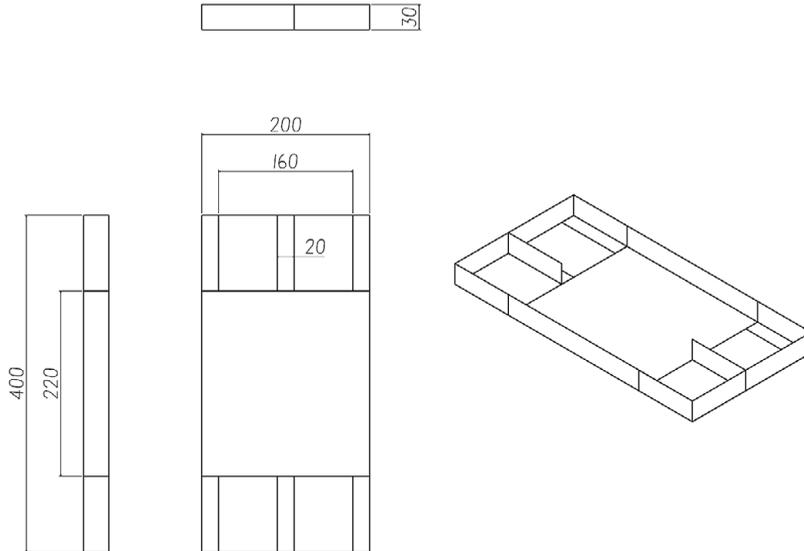


Figure 15. Stiffened composite plate.

Table 4. Laminated plate.

| | Orientations | Single layer thickness/mm |
|-----------|---|---------------------------|
| Stringers | $[(45^\circ/0^\circ)_2/45^\circ/45^\circ/\bar{0}^\circ]_S$ | 0.2 |
| Panel | $[(45^\circ/0^\circ)_2/45^\circ/45^\circ/0^\circ/\bar{45}^\circ]_S$ | 0.2 |

panel, two stringers, and eight stiffeners, of which two stiffeners are lower and upper ends. The lower end of the stiffened plate is fixed, and the rotational degrees of freedom and displacement perpendicular to the panel of the upper end is constrained. Uniform load acts on the upper end along the long side of the panel. The stringers and the panel are laminated plates, the sequence of laying is shown in Table 4. The stiffener material is aluminum alloy.

Considering the scatter of properties of laminated plates, the elastic modulus of 0° and 90° were obtained by testing a batch of standard parts, as shown in Figure 16, whose nominal values are 128 GPa and 10 GPa, respectively. In addition, suppose other properties are $G_{12} = G_{13} = 5.50$ GPa, $G_{23} = 3.77$ GPa, $\nu_{12} = \nu_{13} = 0.316$, $\nu_{23} = 0.3$. At the same time, considering the scatter of aluminum alloy material, the elastic modulus interval is $[70, 75]$ GPa and the Poisson's ratio is 0.33. The initial (red) convex polyhedron and the inflating ellipsoid of uncertain elastic moduli with 18 vertexes are plotted in Figure 17, and the inflation coefficient is $\varepsilon = 5$. The initial (red) and enlarging (blue) convex polyhedron are plotted in Figure 18. Parts of the same material have the same elastic moduli in this example.

Assuming that the strength of the stiffened wall meets the requirements, but the compression along the long side of the panel should be limited. The plate is divided into 7,160 shell elements and the finite

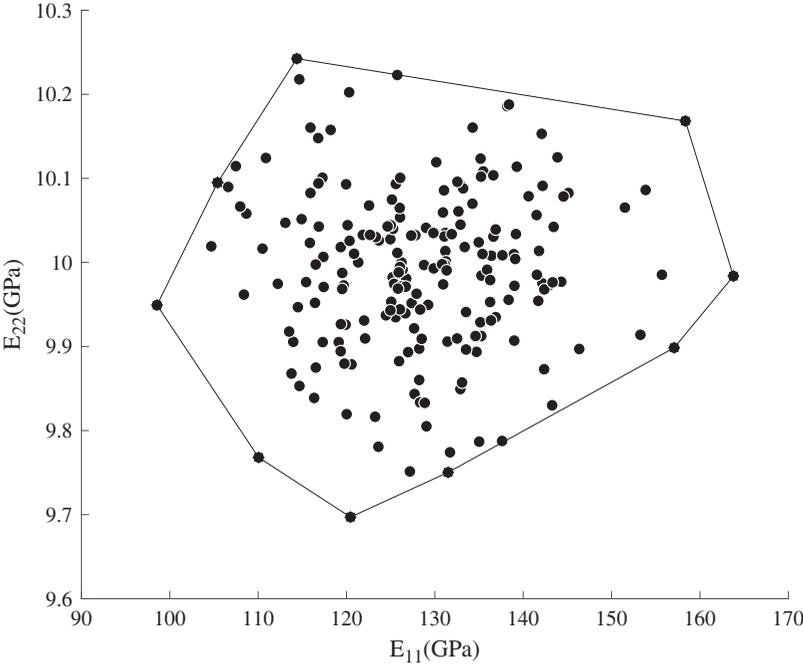


Figure 16. Elastic modulus test results of laminated plates.

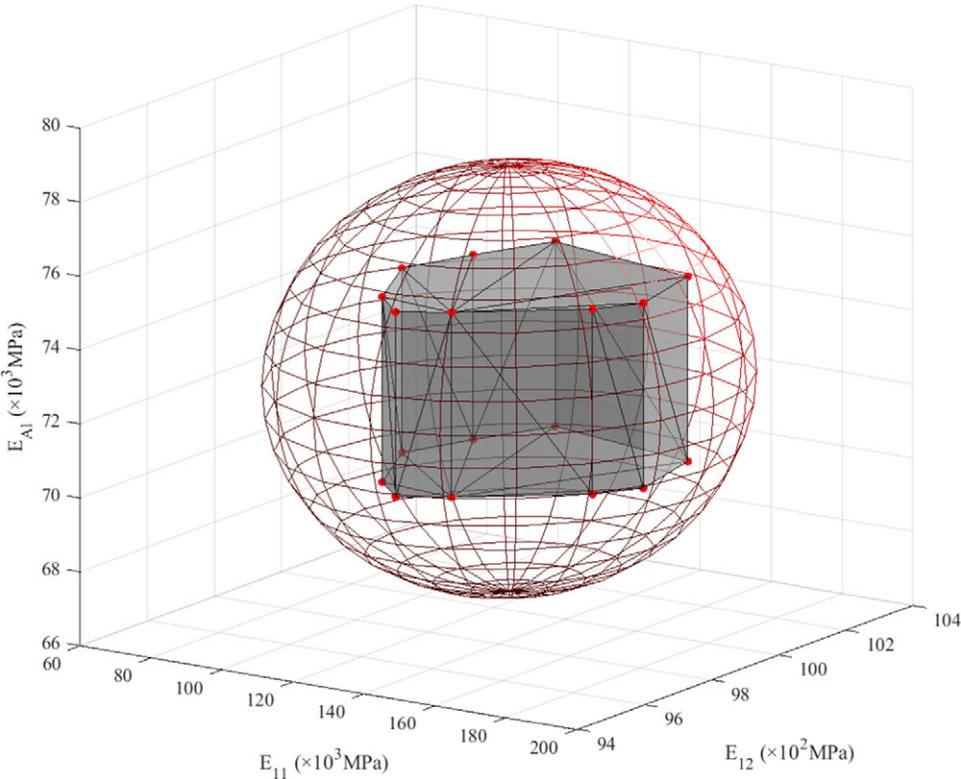


Figure 17. Initial convex polyhedron and the inflating ellipsoid of uncertain elastic moduli.

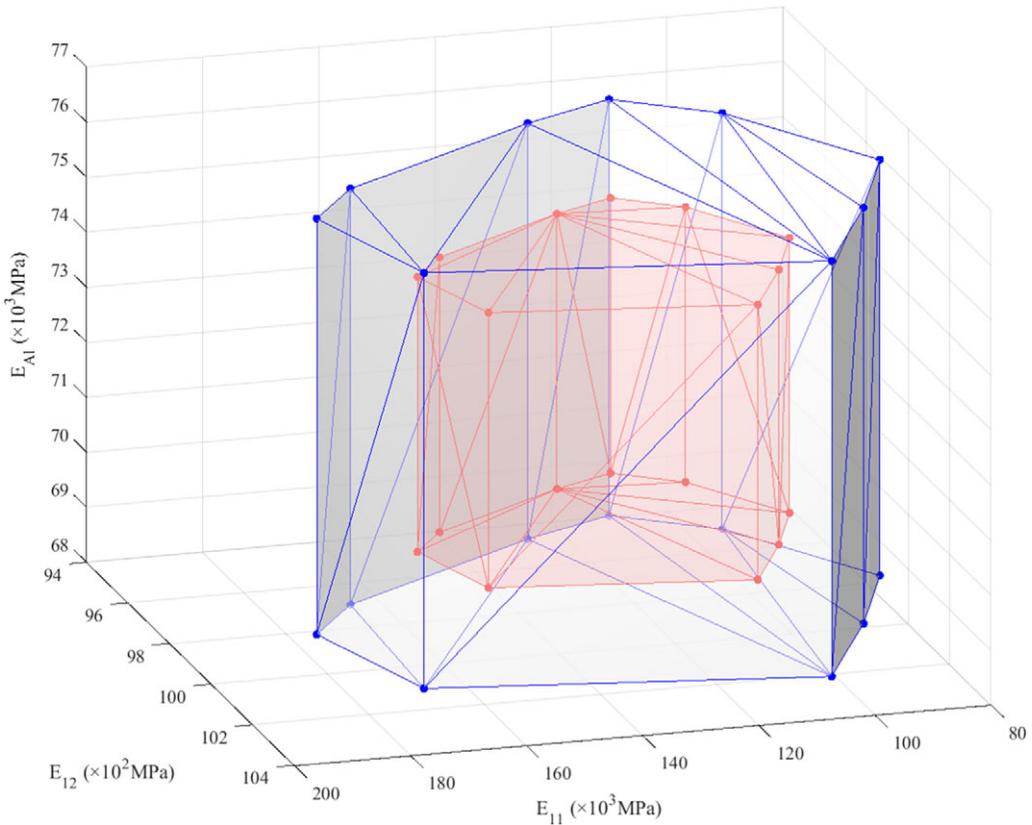


Figure 18. Initial and enlarging convex polyhedron of uncertain elastic moduli.

element model mesh is plotted in Figure 19. Figure 20 shows the displacement contour image of this stiffened composite plate with nominal parameters and the uniform load on the upper end is 100 MPa.

Now concentrate on the load-bearing capacity of this plate. Taking the degrees of freedom of the upper end along the long side of the panel into consideration. Let \mathbf{D} be the flexibility matrix, after the displacement constraints are considered, \mathbf{f} and \mathbf{u}_0 be the column vectors of equivalent load and permissible displacement respectively. \mathbf{D} , \mathbf{f} , and \mathbf{u}_0 only consist of components at the above degrees of freedom.

According to the finite element modal, \mathbf{D} is a 287-by-287 matrix, and \mathbf{f} and \mathbf{u}_0 are 287-dimensional column vectors. With the finite element method, the relationship of the uniform load P and the column vector of equivalent load \mathbf{f} is established with a 287-dimensional column vector \mathbf{l} or \mathbf{c} as

$$\mathbf{f} = \mathbf{P}\mathbf{l} \text{ or } \mathbf{P} = \mathbf{c}^T \mathbf{f} \tag{54}$$

Similar to example 1, the load-bearing capacity of this stiffened composite plate can be determined by the convex polyhedron linear programming problem:

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{f} \\ \text{s.t.} \quad & \mathbf{D}\mathbf{f} \leq \mathbf{u}_0 \\ & \mathbf{D} \in \mathcal{D} \\ & \mathbf{f} \geq \mathbf{0} \end{aligned} \tag{55}$$

And (55) can be transformed into the following problem by applying formula (54):

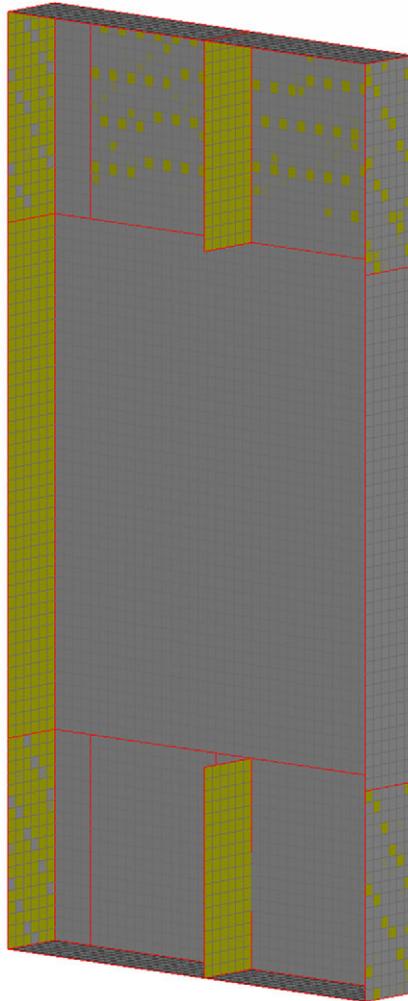


Figure 19. Finite element model mesh.

$$\begin{aligned}
 & \min P \\
 & \text{s.t. } \mathbf{DlP} \leq \mathbf{u}_0 \\
 & \quad \mathbf{Dl} \in \mathcal{D}' , \\
 & \quad P \geq 0
 \end{aligned} \tag{56}$$

where \mathcal{D}' is the convex hull of \mathbf{Dl} . Furthermore, by introducing new variables \mathbf{P}_0 , (55) can be transformed into the following equivalent form:

$$\begin{aligned}
 & \min \mathbf{c}'^T \mathbf{p}' \\
 & \text{s.t. } \mathbf{Gp}' = \mathbf{u}_0 , \\
 & \quad \mathbf{G} \in \mathcal{G} , \\
 & \quad \mathbf{p}' \geq 0
 \end{aligned} \tag{57}$$

where

$$\mathbf{G} = (\mathbf{Dl} \ I), \mathbf{p}' = \begin{pmatrix} \mathbf{P} \\ \mathbf{P}_0 \end{pmatrix}, \mathbf{c}' = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{58}$$

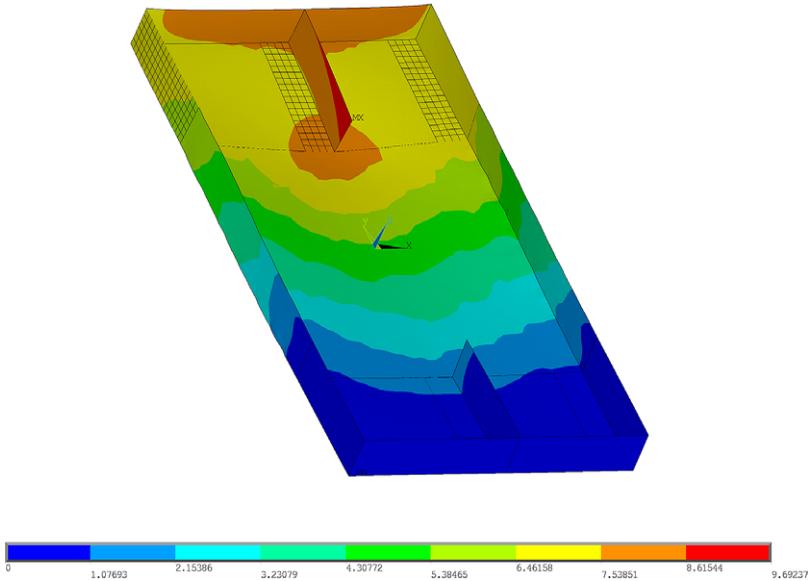


Figure 20. Displacement contour image of this stiffened composite plate with nominal parameters (mm).

Table 5. Uniform loads.

| E11 (GPa) | E22 (GPa) | EAl_alloy (GPa) | P (MPa) |
|-----------|-----------|-----------------|---------|
| 93.41 | 10.16 | 68.72 | 92.40 |
| 83.06 | 9.95 | 68.72 | 88.87 |
| 100.43 | 9.68 | 68.72 | 93.56 |
| 116.15 | 9.57 | 68.72 | 97.53 |
| 132.86 | 9.65 | 68.72 | 101.72 |
| 171.54 | 9.87 | 68.72 | 110.42 |
| 181.67 | 10.00 | 68.72 | 112.65 |
| 173.46 | 10.28 | 68.72 | 111.63 |
| 106.98 | 10.39 | 68.72 | 96.63 |
| 93.41 | 10.16 | 76.28 | 95.72 |
| 83.06 | 9.95 | 76.28 | 92.01 |
| 100.43 | 9.68 | 76.28 | 96.92 |
| 116.15 | 9.57 | 76.28 | 101.07 |
| 132.86 | 9.65 | 76.28 | 105.47 |
| 171.54 | 9.87 | 76.28 | 114.57 |
| 181.67 | 10.00 | 76.28 | 116.91 |
| 173.46 | 10.28 | 76.28 | 115.86 |
| 106.98 | 10.39 | 76.28 | 100.17 |

In which I is a 287-dimensional identity matrix and P_0 is a 287-dimensional column vector. Similar to example 1, it is easy to prove that \mathcal{G} is a convex polyhedron. Therefore (57) is a standard form of a convex polyhedron linear programming problem.

Assuming that the permissible displacement of the plate is 10 mm, the components of u_0 can be set to 10 mm. By applying the vertex solution, only 18 linear programming procedures need to be solved. Notice that the uniform load is the target function of (57), so the convex polyhedron of optimal solutions need not be calculated.

Table 6. Nominal value and bounds of uniform load.

| Nominal value (MPa) | Lower bound (MPa) | Upper bound (MPa) |
|---------------------|-------------------|-------------------|
| 102.89 | 88.87 | 116.91 |

The 18 results are listed in Table 5. The nominal value of P is calculated from the geometric gravity center. The nominal value and bounds of P are listed in Table 6. From Tables 5 and 6, the upper bound of loading bearing capacity is 116.91 MPa, and the lower bound of loading bearing capacity is 88.87 MPa. Therefore, to meet the design requirements in most cases, the maximum load should not be higher than 88.87 MPa.

7. Conclusion

As is explained in Section 1, there is usually no sufficient data available in engineering. Among the models quantifying insufficient data, the interval model—one of the most widely used nonprobabilistic convex models—usually fails to reflect the dependence between variables. As a result, the design results turn out to be too conservative. The data-based convex polyhedron model is a natural extension of the interval model, overcoming the shortcomings of interval models and helping obtain reliable design results.

The convex polyhedron model for uncertain linear optimization of engineering structure is studied in this paper. Firstly, the convex polyhedron model for uncertainty quantification is studied. In order to prevent the data from future experiments fall outside of the convex hull, beyond certain probability, Chebyshev inequality has been applied to inflate the convex polyhedron.

Secondly, the characteristics of the optimal solution of the convex polyhedron linear programming problem are investigated. The vertex solution of convex polyhedron linear programming is discussed. The modeling method of convex polyhedron linear programming problem with the example of a load-bearing capacity problem of the structure is then introduced.

Finally, in the first example of the load-bearing problem of plane truss, the effectiveness of the vertex solution is verified by comparison with the Monte-Carlo method. In the second example of stiffened composite plates, one complex convex polyhedron linear programming is simplified into 18 linear programming tasks, which demonstrates the calculation efficiency of the vertex solution method.

Abbreviations

| | |
|-------|--|
| CPVSM | convex polyhedron vertex solution method |
| IM | interval method |
| MCM | Monte-Carlo method |

Author Contributions. Conceptualization, Z.Q.; Methodology, Z.Q., H.W., I.E., and D.L.; Data curation, H.W.; Data visualization, H.W. and D.L.; Writing-original draft, H.W.; Writing-review & editing, Z.Q., H.W., and I.E. All authors approved the final submitted draft.

Data Availability Statement. The data used in this work is proprietary and could not be made publicly available.

Competing Interests. The authors declare no competing interests.

Funding Statement. This work received no specific grant from any funding agency, commercial or not-for-profit sectors.

References

- Andrew AM (1979) Another efficient algorithm for convex hulls in two dimensions. *Information Processing Letters* 9(5), 216–219. [https://doi.org/10.1016/0020-0190\(79\)90072-3](https://doi.org/10.1016/0020-0190(79)90072-3)
- Barber C, Dobkin D and Huhdanpaa H (1996) The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software* 22(4), 469–483. <https://doi.org/10.1145/235815.235821>

- Chanas S** (1989) Fuzzy programming in multiobjective linear programming—a parametric approach. *Fuzzy Sets and Systems* 29(3), 303–313. [https://doi.org/10.1016/0165-0114\(89\)90042-0](https://doi.org/10.1016/0165-0114(89)90042-0)
- Chinneck JW and Ramadan K** (2000) Linear programming with interval coefficients. *Journal of the Operational Research Society* 51(2), 209–220. <https://doi.org/10.1057/palgrave.jors.2600891>
- Dong W and Shah HC** (1987) Vertex method for computing functions of fuzzy variables. *Fuzzy Sets and Systems* 24(1), 65–78.
- Dong W and Wong F** (1985) The Vertex Method and Its Use in Earthquake Engineering. *Proceedings of NSF Workshop on Civil Engineering Applications of Fuzzy Sets*, 173–192.
- Elishakoff I** (1995). Essay on uncertainties in elastic and viscoelastic structures: From A. M. Freudenthal's criticisms to modern convex modeling. *Computers & Structures* 56(6), 871–895. [https://doi.org/10.1016/0045-7949\(94\)00499-S](https://doi.org/10.1016/0045-7949(94)00499-S)
- Elishakoff I** (1998a) Three facets of uncertainty. *Computers and Structures* 67(1), 1–2.
- Elishakoff I** (1998b) Three versions of the finite element method based on concepts of either stochasticity, fuzziness, or anti-optimization. *Applied Mechanics Reviews* 51(3), 209–218.
- Elishakoff I and Sarlin N** (2016) Uncertainty quantification based on pillars of experiment, theory, and computation, part II: Theory and computation. *Mechanical Systems and Signal Processing* 74, 54–72.
- Evers W** (1967) A new model for stochastic linear programming. *Management Science* 13, 680–693. <https://doi.org/10.1287/mnsc.13.9.680>
- Graham RL** (1972) An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters* 1(4), 132–133. [https://doi.org/10.1016/0020-0190\(72\)90045-2](https://doi.org/10.1016/0020-0190(72)90045-2)
- Guo X, Bai W, Zhang W and Gao X** (2009) Confidence structural robust design and optimization under stiffness and load uncertainties. *Computer Methods in Applied Mechanics and Engineering* 198(41), 3378–3399. <https://doi.org/10.1016/j.cma.2009.06.018>
- Jarvis RA** (1973) On the identification of the convex hull of a finite set of points in the plane. *Information Processing Letters* 2(1), 18–21. [https://doi.org/10.1016/0020-0190\(73\)90020-3](https://doi.org/10.1016/0020-0190(73)90020-3)
- Klir GJ** (1994) On the alleged superiority of probabilistic representation of uncertainty. *IEEE Transactions on Fuzzy Systems* 2(1), 27–31.
- Ley C, Tibolt M and Fromme D** (2020) Data-centric engineering in modern science from the perspective of a statistician, an engineer, and a software developer. *Data-Centric Engineering* 1, E2. <https://doi.org/10.1017/dce.2020.2>
- Preparata FP** (1979) An optimal real-time algorithm for planar convex hulls. *Communications of the ACM* 22(7), 402–405. <https://doi.org/10.1145/359131.359132>
- Qiu Z and Wang X** (2009) Vertex solution theorem for the upper and lower bounds on the dynamic response of structures with uncertain-but-bounded parameters. *Acta Mechanica Sinica* 25(3), 367–379. <https://doi.org/10.1007/s10409-008-0223-5>
- Qiu Z, Wang X and Chen J** (2006) Exact bounds for the static response set of structures with uncertain-but-bounded parameters. *International Journal of Solids and Structures*, 43(21), 6574–6593. <https://doi.org/10.1016/j.ijsostr.2006.01.012>
- Ramik J** (1986) Extension principle in fuzzy optimization. *Fuzzy Sets & Systems*, 19(1), 29–35.
- Rockafellar R** (1970) *Convex analysis*. Princeton: Princeton University Press. <https://doi.org/10.1515/9781400873173>.
- Rohn J** (1980) Duality in interval linear programming. In Nickel KLE (ed), *Interval Mathematics 1980*. Cambridge: Academic Press, pp. 521–529.
- Rohn J** (1981) Strong solvability of interval linear programming problems. *Computing* 26, 79–82. <https://doi.org/10.1007/BF02243426>
- Sagiroglu S and Sinanc D** (2013) *Big data: A review*. Paper presented at the 2013 International Conference on Collaboration Technologies and Systems (CTS).
- Sarlin N and Elishakoff I** (2016) Uncertainty quantification based on pillars of experiment, theory, and computation. Part I: Data analysis. *Mechanical Systems and Signal Processing* 74, 29–53.
- Sengupta J, Tintner G and Millham C** (1963). On some theorems of stochastic linear programming with applications. *Management Science*, 10, 143–159. <https://doi.org/10.1287/mnsc.10.1.143>
- Stanculescu C, Fortemps P, Installé M and Wertz V** (2003) Multiobjective fuzzy linear programming problems with fuzzy decision variables. *European Journal of Operational Research* 149(3), 654–675. [https://doi.org/10.1016/S0377-2217\(02\)00449-6](https://doi.org/10.1016/S0377-2217(02)00449-6)
- Zimmermann HJ** (1978) Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems* 1 (1), 45–55. [https://doi.org/10.1016/0165-0114\(78\)90031-3](https://doi.org/10.1016/0165-0114(78)90031-3)

Appendix A: Relation Between Optimal Solutions and Vertex Solutions

Sample appendix content. This section analyzes whether every optimal solution \mathbf{x}^* can be expressed by the linear combination of vertex optimal solutions \mathbf{x}_{ij}^* .

We multiply both sides of by (35) $\sum_j \beta_j$ and $\sum_i \alpha_i$ respectively. Equation (35) will be transformed into

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^* = \sum_i \sum_j \alpha_i \beta_j \mathbf{b}_j. \tag{A.1}$$

We then multiply both sides of Equation (38) by $\alpha_i \beta_j$, and calculate the sum over i and j , the following relation is obtained

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* = \sum_i \sum_j \alpha_i \beta_j \mathbf{b}_j. \tag{A.2}$$

Then, comparing Equation (A.1) with (A.2), we obtain

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^* = \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* \tag{A.3}$$

or in another form

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^* - \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* = 0. \tag{A.4}$$

Equations (A.3) and (A.4) are the necessary conditions that every optimal solution \mathbf{x}^* of (23) should satisfy.

It is noteworthy that \mathbf{x}^* is a function of α_i, β_j , therefore, the first term of (A.3) and (A.4) cannot be simplified. Considering the assumption at the beginning of 4.2, \mathbf{x}^* can be uniquely determined if α_i, β_j are given by using (A.3) or (A.4).

This work can be done by analyzing every $\mathbf{x}^{*l} \in \mathbf{X}'$ and every $\mathbf{x}^{*l} \notin \mathbf{X}'$. The necessary condition (A.4) will be used to check whether \mathbf{x}^{*l} may be one optimal solution of the problem (23) or not.

1. $\forall \mathbf{x}^{*l} \in \mathbf{X}' \subseteq \mathbf{X}'$.

it is obvious that $\mathbf{x}^{*l} \in \mathbf{X}$.

2. $\forall \mathbf{x}^{*l} \in \mathbf{X}'$.

According to (37), \mathbf{x}^{*l} can be determined as $\mathbf{x}^{*l} = \sum_{i_1, j_1} \mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^*$. Therefore, check the necessary condition as follow

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^{*l} - \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* = \sum_{i_1, j_1} \left[\sum_i (\alpha_i \mathbf{A}_i) \mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^* - \frac{1}{n_A n_b} \sum_i \sum_j (\alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^*) \right]. \tag{A.5}$$

Applying the equation $\mathbf{A}_i \mathbf{x}_{ij}^* = \mathbf{b}_j$ and $\sum_i \alpha_i \mathbf{A}_i \mathbf{x}^{*l} = \mathbf{b}$, the second term in Equation (A.5) can be transformed into

$$\begin{aligned} \frac{1}{n_A n_b} \sum_i \sum_j (\alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^*) &= \frac{1}{n_A n_b} \sum_i \sum_j \alpha_i \beta_j \mathbf{b}_j \\ &= \frac{1}{n_A n_b} \sum_i \alpha_i \mathbf{A}_i \mathbf{x}^{*l} \\ &= \frac{1}{n_A n_b} \sum_i \alpha_i \mathbf{A}_i \sum_{i_2, j_2} \mu_{i_2} v_{j_2} \mathbf{x}_{i_2 j_2}^*, \end{aligned} \tag{A.6}$$

where n_A and n_b identify the number of vertices of polyhedrons \mathcal{A} and \mathcal{B} , respectively. Substituting the equation above into Equation (A.5), the following result is obtained

$$\sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^{*l} - \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* = \sum_i \alpha_i \mathbf{A}_i \left[\sum_{i_1, j_1} (\mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^*) - \frac{1}{n_A n_b} \sum_{i_1, j_1, i_2, j_2} \mu_{i_2} v_{j_2} \mathbf{x}_{i_2 j_2}^* \right] \tag{A.7}$$

Therefore, Equation (A.4) holds. That is, if $\mathbf{x}^{*l} \in \mathbf{X}'$, \mathbf{x}^{*l} maybe an optimal solution of (23). On the other hand, if Equation (A.4) does not hold, \mathbf{x}^{*l} does not belong to \mathbf{X}' . We use “maybe” here because (A.4) is a “necessary condition,” not a “sufficient and necessary condition”: Equation (A.4) is necessary for all optimal solutions, and there exist \mathbf{x}^{*l} satisfying Equation (A.4) is not an optimal solution of (23).

3. $\forall \mathbf{x}^* \notin \mathbf{X}'$.

Assuming that $\mathbf{x}^{*l} = \sum_{i_1, j_1} \mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^* + \mathbf{x}_{i_0 j_0}$, where $\mathbf{x}_{i_0 j_0} \neq \mathbf{0}$ and $\sum_{i_1, j_1} \mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^* + \mathbf{x}_{i_0 j_0} \geq \mathbf{0}$. The constraint condition holds as

$$\sum_i \alpha_i \mathbf{A}_i \left(\sum_{i_1, j_1} \mu_{i_1} v_{j_1} \mathbf{x}_{i_1 j_1}^* + \mathbf{x}_{i_0 j_0} \right) = \mathbf{b}. \text{ Similar to ii), the necessary condition is checked as}$$

$$\begin{aligned}
 & \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}^{*j} - \sum_i \sum_j \alpha_i \beta_j \mathbf{A}_i \mathbf{x}_{ij}^* \\
 &= \sum_i (\alpha_i \mathbf{A}_i) \left[\sum_{j_1, j_2, \dots, j_n} \left(\mu_{i_1} \nu_{j_1} \mathbf{x}_{i_1 j_1}^* + \frac{1}{n_A n_b} \mathbf{x}_{i_0 j_0} \right) - \frac{1}{n_A n_b} \sum_{i_1, j_1, i_2, j_2} \mu_{i_2} \nu_{j_2} \mathbf{x}_{i_2 j_2}^* \right] \\
 &= \sum_i (\alpha_i \mathbf{A}_i) \mathbf{x}_{i_0 j_0} \\
 &= \left(\mathbf{A}_0 + \sum_i \Delta \mathbf{A}_i \right) \mathbf{x}_{i_0 j_0}.
 \end{aligned} \tag{A.8}$$

To satisfy Equation (A.4), it is required that $\left(\mathbf{A}_0 + \sum_i \Delta \mathbf{A}_i \right) \mathbf{x}_{i_0 j_0} = \mathbf{0}$. Consider small quantities of zero order and first order:

$$\begin{cases} \mathbf{A}_0 \mathbf{x}_{i_0 j_0} = \mathbf{0} \\ \Delta \mathbf{A}_i \mathbf{x}_{i_0 j_0} = \mathbf{0}, i = 1, 2, \dots, n_A \end{cases} \tag{A.9}$$

In practice, the number of vertices of polyhedron \mathcal{A} calculated from samples often larger than the equation number in (A.9), which means that simultaneous Equations (A.9) have no solution. Therefore, case (3) is not true. That is, if $\mathbf{x}^{*j} \notin \mathcal{X}^j$, \mathbf{x}^{*j} is not an optimal solution of (23).

Appendix B: Demonstration that Flexibility Matrices can be Quantified as a Convex Polyhedron

The perturbation method will be used to demonstrate that \mathcal{D} can also be regarded as a convex polyhedron in Appendix B.

The elements of \mathbf{K} and \mathbf{D} are denoted by k_{ij} and d_{ij} , and k_{ij} can be expressed by

$$k_{ij} = k_{0,ij} + \Delta k_{ij} \tag{B.1}$$

The determinant of \mathbf{K} is as follows:

$$\det(\mathbf{K}) = \sum_{j_1, j_2, \dots, j_n} (-1)^{\tau(j_1 j_2 \dots j_n)} k_{1j_1} k_{2j_2} \dots k_{nj_n}, \tag{B.2}$$

where $\tau(j_1 j_2 \dots j_n)$ represent the number of inversions in j_1, j_2, \dots, j_n . Denote the algebraic cofactor of k_{ij} as K_{ij} , then K_{ij} can be computed as:

$$K_{ij} = (-1)^{i+j} \sum_{j_1, j_2, \dots, j_{n-1} \neq j} (-1)^{\tau(j_1 j_2 \dots j_{n-1})} k_{1j_1} \dots k_{(i-1)j_{i-1}} k_{(i+1)j_{i+1}} \dots k_{nj_n} \tag{B.3}$$

We substitute (B.1) into (B.2) and (B.3). The determinant of \mathbf{K} and the algebraic cofactor of k_{ij} are transformed into

$$\begin{aligned}
 \det(\mathbf{K}) &= \sum_{j_1, j_2, \dots, j_n} (-1)^{\tau(j_1 j_2 \dots j_n)} \left[k_{0,1j_1} k_{0,2j_2} \dots k_{0,nj_n} + \sum_{s=1}^n \left(\Delta k_{sj_s} \prod_{l \neq s} k_{0,lj_l} \right) \right. \\
 &\quad \left. + \sum_{s,t=1}^n \left(\Delta k_{sj_s} \Delta k_{tj_t} \prod_{l \neq s, l \neq t} k_{0,lj_l} \right) + \dots + \Delta k_{1j_1} \Delta k_{2j_2} \dots \Delta k_{nj_n} \right]
 \end{aligned} \tag{B.4}$$

$$\begin{aligned}
 K_{ij} &= \sum_{j_1, j_2, \dots, j_{n-1} \neq j} (-1)^{i+j+\tau(j_1 j_2 \dots j_{n-1})} \left[k_{0,1j_1} \dots k_{0,(i-1)j_{i-1}} k_{0,(i+1)j_{i+1}} \dots k_{0,nj_n} + \sum_{s=1}^n \left(\Delta k_{sj_s} \prod_{l \neq i, l \neq s} k_{0,lj_l} \right) \right. \\
 &\quad \left. + \sum_{s,t=1}^n \left(\Delta k_{sj_s} \Delta k_{tj_t} \prod_{l \neq i, l \neq s, l \neq t} k_{0,lj_l} \right) + \dots + \Delta k_{1j_1} \dots \Delta k_{(i-1)j_{i-1}} \Delta k_{(i+1)j_{i+1}} \dots \Delta k_{nj_n} \right],
 \end{aligned} \tag{B.5}$$

where the zero-order items in (B.4) and (B.5) are represented as $\det(\mathbf{K}_0)$ and $K_{0,ij}$. Combining Equations (B.4) and (B.5), d_{ij} can be represented as:

$$d_{ij} = \frac{K_{ij}}{\det(\mathbf{K})} \tag{B.6}$$

We retain the first-order small quantities of Equation (B.6). The following expression is obtained:

$$\begin{aligned}
 d_{ji} = & \frac{K_{0,ij}}{\det(\mathbf{K}_0)} + \frac{1}{\det(\mathbf{K}_0)} \sum_{j_1, j_2, \dots, j_{n-1} \neq j} \sum_{s=1}^n (-1)^{i+j+\tau(j_1 j_2 \dots j_{n-1})} \left(\Delta k_{s j_s} \prod_{l \neq i, l \neq s} k_{0, j_l} \right) \\
 & - \frac{K_{0,mn}}{\det^2(\mathbf{K}_0)} \sum_{j_1, j_2, \dots, j_n} \sum_{s=1}^n (-1)^{\tau(j_1 j_2 \dots j_n)} \left(\Delta k_{s j_s} \prod_{l \neq s} k_{0, j_l} \right)
 \end{aligned} \tag{B.7}$$

We denote the zero-order term and first-order small quantity as $d_{0,ji}$ and Δd_{ji} . Signify $\mathbf{D}_0 = (d_{0,ij})_{n \times n}$, then the flexibility matrix is written as

$$\mathbf{D} = \mathbf{D}_0 + (\Delta d_{ij})_{n \times n} \tag{B.8}$$

The former term of Equation (B.8) is the inverse of the nominal matrix \mathbf{K}_0 , while the latter term is a linear combination of matrices that only contains the first-order small quantity Δd_{ij} . It is obvious that the latter term is convex according to the convex property of $\Delta k_{s j_s}$.