Engineering Analysis with Boundary Elements 88 (2018) 26–40

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

Structural shape optimization by IGABEM and particle swarm optimization algorithm

S.H. Sun [a], T.T. Yu [a,*], T.T. Nguyen [b], E. Atroshchenko [c], T.Q. Bui [d,e,**]

[a] Department of Engineering Mechanics, Hohai University, Nanjing 211100, PR China
[b] Institute of Computational Engineering, University of Luxembourg, 6 Avenue de la Fonte, 4362 Esch-sur-Alzette, Luxembourg
[c] Department of Mechanical Engineering, Faculty of Physical and Mathematical Sciences, University of Chile, Chile
[d] Institute for Research and Development, Duy Tan University, Da Nang, Vietnam
[e] Department of Civil and Environmental Engineering, Tokyo Institute of Technology, 2-12-1-W8-22, Ookayama, Meguro-ku, Tokyo 152-8552, Japan

ARTICLE INFO

Keywords:
Shape optimization
NURBS
Isogeometric analysis
Boundary element method
Particle swarm optimization algorithm

ABSTRACT

In this paper, a new approach is developed for structural shape optimization, which consists in coupling the particle swarm optimization (PSO) algorithm and the isogeometric boundary element method (IGA-BEM). The IGA-BEM is based on the combination of the isogeometric analysis (IGA) and the boundary element method (BEM), where Non-Uniform Rational B-Splines (NURBS) are employed as shape functions for geometry parameterization and approximation of the field variables. The method inherits the main advantage of the IGA-based shape optimization, i.e., the control points are used as design variables, thus the design model, analysis model and optimization model are uniformly described with the NURBS, providing easy communication between the three models and resulting in a smooth optimized boundary. However, the main feature of the proposed method is the use of PSO, which provides an attractive gradient-free alternative to complicated sensitivity analysis. The efficiency and accuracy of the proposed approach are demonstrated through four 2D shape optimization examples.

1. Introduction

For the purpose of improving structural characteristics such as reducing stress concentration, structural weight, and maximum displacements, shape optimization is required to obtain suitable designs. In the early days of structural shape optimization analysis, the nodes on the boundaries in the finite element model were directly used as the design variables, which could lead to serrate shapes because of the independent changes of these nodes on the boundary [1], so the optimized boundary needed to be smoothed. The polynomial and spline functions were later successively used in the parametric characterization of the structure boundary to constrain the movement of the boundary nodes, most commonly with the use of NURBS (Non-uniform rational B-splines) [2] which are defined as the sole representation method of free curve and surface in the STEP standard. However, the gap between the finite element analysis (FEM) model and the CAD model consumes up to 80% of the solution time because of mesh generation, and the element-based polynomial approximation in the standard FEM inevitably leads to discretization errors in complex geometries.

In order to seamlessly integrate CAD and finite element analysis, Hughes et al. [3] introduced isogeometric analysis (IGA). The underlying principle of IGA is that NURBS used in CAD are employed as basis functions for the approximation of the unknown solution. Due to such features as the exact geometrical representation and high-order continuity of the shape functions, IGA has successfully been applied to many engineering problems, e.g., see Refs. [4–15].

NURBS only describe the boundaries of objects, so one of the key issues in the IGA FEM is generating volume meshes from the parameterization of the boundary surfaces. Since only the surfaces of the domain need to be meshed in the boundary element method (BEM), the BEM is very efficient for problems with a small surface/volume ratio. Additionally, the accuracy of BEM solutions heavily relies on accuracy of the boundary representation, which in IGA is always exact. These features make BEM an ideal partner for IGA (the concept known as IGA-BEM). The idea of IGA-BEM was first presented in [16]. The IGA-BEM has later been applied for various problems including elastostatic analysis [17–19], wave-resistance problems [20,21], flow problems [22,23], acoustic analysis [24], crack problems [25–27], elasto-plastic inclusions [28,29], and structural shape optimization [30–34].

* Corresponding author at: Department of Engineering Mechanics, Hohai University, Nanjing 211100, PR China.
** Corresponding author at: Institute for Research and Development, Duy Tan University, Da Nang, Vietnam & Tokyo Institute of Technology, Japan.
E-mail addresses: tiantangyu@hhu.edu.cn (T.T. Yu), buiquoctinh@duytan.edu.vn, bui.t.aa@m.titech.ac.jp (T.Q. Bui).

https://doi.org/10.1016/j.enganabound.2017.12.007
Received 25 May 2017; Received in revised form 29 September 2017; Accepted 15 December 2017
Available online 28 December 2017
0955-7997/

Some improvements on the original IGA-BEM were and continue to be investigated. In order to improve the computational speed, Takahashi and Matsumoto [35] and Simpson and Liu [36] combined IGA-BEM with the fast multipole method, while Marussig et al. [37] proposed a fast IGA-BEM based on an independent discretization for geometry, displacement and traction fields, and the application of H-matrices. Scott et al. [38] developed T-splines based IGA-BEM which allows local refinement and handles complex geometry. Peake et al. [39] developed extended IGA-BEM for 2D Helmholtz problems by introducing partition-of-unity based enrichments. Wang et al. [40] introduced a multi-patch nonsingular IGA-BEM for complicate structure analysis by coupling trimmed NURBS with a nonsingular BEM. Feischl et al. [41] integrated the adaptive concept into IGA-BEM in 2D.

Some researchers have studied structural topology optimization [42–48], shape optimization [49–54], and material optimization [55–57] by using IGA. Li and Qian [30] presented a boundary integral based approach to IGA and shape optimization where NURBS basis is used to parameterize the boundary shape and to approximate the physical fields in the analysis. They have extended their approach to both 2D and 3D elasticity problems successfully. Kostas et al. [31] presented a ship-hull optimization process combining a T-spline based parametric ship-hull model, and an IGA hydrodynamic solver for the calculation of ship wave resistance. Yoon et al. [32] and Yoon and Cho [33] derived a general configuration design sensitivity analysis method for steady state heat conduction problems using the isogeometric boundary integral equation method, and it was applied for configuration design optimization of heat conduction problems and 2D elastic structures. Lian et al. [34] presented shape sensitivity analysis and optimization in 2D elasticity with a regularized IGA-BEM. Recently, Lian et al. [58] developed a T-spline IGA-BEM to shape sensitivity analysis and gradient-based shape optimization in 3D linear elasticity.

In the majority of works related to isogeometric optimization of structures, gradient methods are used in optimization process. The advantage of gradient-based optimization methods is the fast convergence speed. However, gradient-based methods require data from sensitivity analysis, i.e., the object function is specified explicitly as a function of the optimized parameters and then differentiated, which is not easily applicable in various problems and its derivation and implementation are often complicated, unless automatic non-intuitive approaches. In general, gradient-based methods find a local optimum.

There are some evolutionary optimization algorithms, which are gradient-free methods and do not require that the optimization problem be differentiable as is required by classic optimization methods such as gradient descent and quasi-Newton methods. For example, Herath et al. [57] presented a layup optimization algorithm for composite marine propellers using NURBS based IGA coupled with real-coded genetic algorithm (GA). The particle swarm optimization (PSO) is one of the gradient-free optimization algorithms, originally proposed in [59]. Its main idea consists in optimizing a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality, and it makes few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. PSO has the following advantages: (1) search speed is fast, and the rule is simpler than that in GA as it avoids the “crossover” and “mutation” operations; (2) it possesses the memory; (3) few parameters are adjusted in the optimization process, and the algorithm structure is simple; (4) the PSO equations work on real numbers, so it can directly deal with the parameters in search space, while the traditional GA must convert the parameters of the problem space into chromosomes or individuals in genetic space by using the coding operator. The PSO was successfully applied to treating both the social and computational paradigms [60,61], as well as to IGA shape optimization in [62]. Global optimization methods using ‘soft computing’ tools such as PSO and GA are more suitable to multiple local minima. The most disadvantage of gradient-free optimization methods is that the convergence speed is slower that for gradient-based optimization methods. This disadvantage can be overcome by high performance computing. In this study, we combine the IGA-BEM and the PSO, which serves our main goal, to form a novel effective approach particularly for the structural shape optimization in 2D elasticity. The IGA-BEM in conjunction with the PSO is attractive to the shape optimization problem as the advantages of both the IGA-BEM and the PSO are taken, which can be listed as follows:

- In terms of structural shape optimization, the NURBS control points are used as design variables, thus design model, analysis model and optimization model can be fully described through the NURBS in a uniform and convenient manner;
- The interaction with the original CAD model is avoided during shape optimization;
- The dimensionality of the problem is reduced to the boundary only, avoiding the need for volume discretization;
- The optimized boundary is smooth;
- No sensitivity analysis is required;
- The implementation is simple.

The present paper is organized as follows. The problem is described in Section 2. The formulation of IGA-BEM for structure analysis is briefly outlined in Section 3. Section 4 presents the optimization algorithm and the numerical implementation. In Section 5, the advantages of the developed method are illustrated through numerical experiments. Some major conclusions drawn from the study are given in Section 6.

2. Problem description

The mathematical formulation of a shape optimization problem is described as follows:

$$\min_{x_k} F(\boldsymbol{x}) \tag{1}$$

$$\text{such that} \quad h_i(\boldsymbol{x}) = 0 \quad i=1,2,\ldots,n_h \text{ number of equality constraints} \tag{2a}$$

$$g_j(\boldsymbol{x}) \le 0, \quad j=1,2,\ldots,n_g \text{ number of inequality constraints} \tag{2b}$$

$$x_{k,\min} \le x_k \le x_{k,\max}, \quad k=1,2,\ldots,n_x \text{ number of design variables} \tag{2c}$$

where F is the objective function, depending on the design variables $\boldsymbol{x}$; the functions h_i and g_j represent the behavioral constraints; $x_{k,\max}$ and $x_{k,\min}$ are the upper and lower bounds of the design variable x_k, respectively.

A large number of mathematical algorithms have been proposed to solve the optimization problem (1). In this contribution, the PSO algorithm is used together with the IGA-BEM as the structural analysis method.

3. Formulation of IGA-BEM for structural analysis

3.1. NURBS basis functions

NURBS basis functions are defined by means of the B-splines in parametric space $\xi \in [0, 1]$ on a so-called knot vector $\mathbf{k}(\xi)$, which is a set of non-decreasing numbers between zero and one, $\mathbf{k}(\xi)=\{\xi_1=0,\ldots,\xi_i,\ldots,\xi_{n+p+1}=1\}^T$ with i representing the knot index, n being the number of basis functions, and p – the order of B-splines. By using knot vector $\mathbf{k}(\xi)$, the ith B-spline basis function of degree p, termed as $N_{i,p}(\xi)$, is defined recursively as follows [3]:

$$N_{i,0}(\xi) = \begin{cases} 1 & if\ \xi_i \le \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \text{ for } p=0 \tag{3a}$$

and

$$N_{i,p}(\xi) = \frac{\xi-\xi_i}{\xi_{i+p}-\xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1}-\xi}{\xi_{i+p+1}-\xi_{i+1}} N_{i+1,p-1}(\xi) \text{ for } p \ge 1 \tag{3b}$$

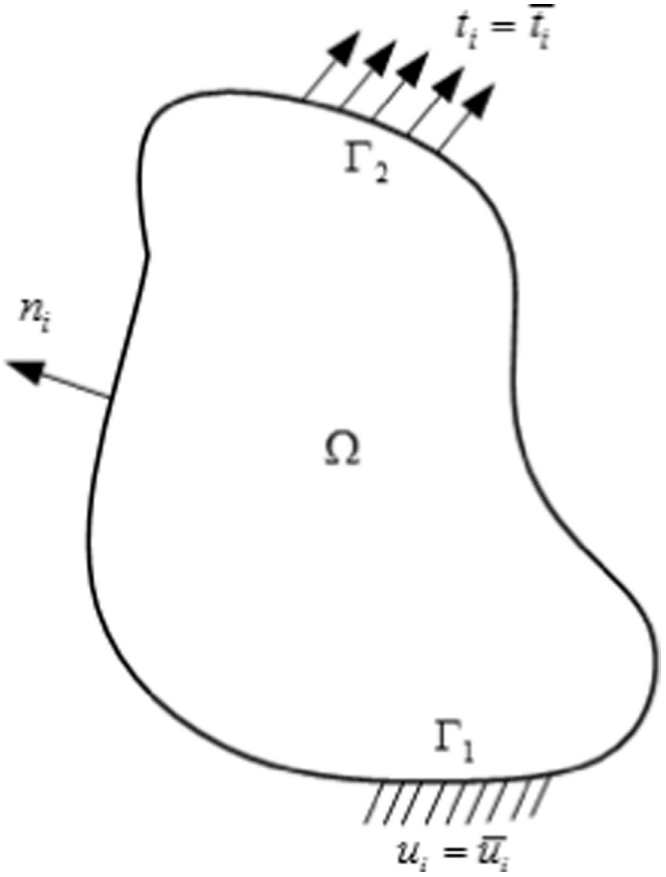

Fig. 1. A two-dimensional domain.

The NURBS basis function in one-dimension is given by [3]:

$$R_i(\xi) = \frac{N_{i,p}(\xi)w_i}{\sum_{i=1}^{n} N_{i,p}(\xi)w_i} \tag{4}$$

where w_i is referred to as the ith weight.

By using these basis functions, a NURBS curve of order p can be constructed with a set of control points $\mathbf{B}_i$ as

$$\boldsymbol{C}(\xi) = \sum_{i=1}^{n} R_i(\xi)\boldsymbol{B}_i \tag{5}$$

3.2. Boundary element method

The basic idea of the BEM is to transform the boundary value problem into a boundary integral equation, which can be subsequently reduced to a linear system of equations through discretization of the boundaries. Here, a brief introduction of linear elastic BEM is given, and for details, curious readers may refer to [63].

In absence of body forces, the boundary value problem for a two-dimensional elastic body occupying domain Ω with boundary Γ, as shown in Fig. 1, can be expressed as follows:

$$\lambda u_{k,ki} + G\left(u_{i,jj} + u_{j,ij}\right) = 0 \text{ in } \Omega \tag{6a}$$

$$u_i = \bar{u}_i \text{ on } \Gamma_1 \tag{6b}$$

$$t_i = \lambda u_{k,k} n_i + G\left(u_{i,j} n_j + u_{j,i} n_j\right) = \bar{t}_i \text{ on } \Gamma_2 \tag{6c}$$

where λ and G are Lame's constants; u_i is the displacement components; t_i is the surface tractions; $\bar{u}_i$ and $\bar{t}_i$ are the prescribed displacements and tractions on the Dirichlet (Γ_1) and Neumann (Γ_2) parts of boundary Γ ($\Gamma=\Gamma_1\cup\Gamma_2, \Gamma_1\cap\Gamma_2=\emptyset$) respectively, and n_i is the outward unit normal vector on Γ.

The boundary integral equation corresponding to Eq. (6) is given as

$$\chi u_l(\boldsymbol{x}) = \int_\Gamma \left[t_k(\boldsymbol{x}')u^*_{lk}(\boldsymbol{x},\boldsymbol{x}') - u_k(\boldsymbol{x}')t^*_{lk}(\boldsymbol{x},\boldsymbol{x}')\right] d\Gamma(\boldsymbol{x}') \tag{7}$$

where u^*_{li} and t^*_{lk} are called the fundamental solutions, which are expressed as

$$u^*_{li}(\boldsymbol{x},\boldsymbol{x}') = \frac{1}{8\pi G(1-\nu)}\left[(3-4\nu)\delta_{li}\ln\frac{1}{r} + \frac{\partial r}{\partial x_l}\frac{\partial r}{\partial x_i}\right] \tag{8a}$$

$$\begin{aligned} t^*_{li}(\boldsymbol{x},\boldsymbol{x}') \quad = \quad & -\frac{1}{4\pi(1-\nu)r}\left\{\frac{\partial r}{\partial n}\left[(1-2\nu)\delta_{li} + 2\frac{\partial r}{\partial x_l}\frac{\partial r}{\partial x_i}\right]\right. \\ & \left. -(1-2\nu)\left(\frac{\partial r}{\partial x_l}n_i - \frac{\partial r}{\partial x_i}n_l\right)\right\} \end{aligned} \tag{8b}$$

where $r = \|\boldsymbol{x}'-\boldsymbol{x}\|$, $\boldsymbol{x}$ and $\boldsymbol{x}'$ are the boundary source point and the boundary field point, respectively; ν is Poisson's ratio; δ_{li} is the Kronecker-delta function; $\chi = 0.5$ when the source point lies on a smooth surface.

3.3. Isogeometric boundary element method (IGA-BEM)

IGA-BEM adopts NURBS basis functions as the shape functions for discretization of Eq. (7). Based on the isoparametric concept, the boundary curve, as well as the boundary displacements and tractions are written as follows:

$$\boldsymbol{x}(\xi) = \sum_{i=1}^{n} R_i(\xi)\boldsymbol{B}_i \tag{9a}$$

$$\boldsymbol{u}(\xi) = \sum_{i=1}^{n} R_i(\xi)\boldsymbol{u}_i \tag{9b}$$

$$\boldsymbol{t}(\xi) = \sum_{i=1}^{n} R_i(\xi)\boldsymbol{t}_i \tag{9c}$$

where $R_i(\xi)$ is the NURBS basis function as shown in Eq. (4), $\mathbf{B}_i$ are the coordinates of control point i of the boundary surface, $\mathbf{u}_i$ and $\mathbf{t}_i$ are the ith control variables corresponding to boundary displacements and tractions respectively.

Discretizing the boundary into a non-overlapping set of N_e elements given by knot spans $[\xi_i,\xi_{i+1}]$, and introducing interpolation formula (Eq. (9)) in each element, yields the discretized boundary integral equation as follows:

$$\begin{aligned} & \chi(\boldsymbol{x}^c)\sum_{a_0=1}^{p+1} R_{a_0}(\hat{\xi}_c)u_j^{e_0 a_0} + \sum_{e=1}^{Ne}\sum_{a=1}^{p+1}\left[\int_{-1}^{+1} t^*_{ij}(\boldsymbol{x}^c,\boldsymbol{x}(\hat{\xi}))R_a(\hat{\xi})J^e(\hat{\xi})d\hat{\xi}\right]u_j^{ea} \\ & = \sum_{e=1}^{Ne}\sum_{a=1}^{p+1}\left[\int_{-1}^{+1} u^*_{ij}(\boldsymbol{x}^c,\boldsymbol{x}(\hat{\xi}))R_a(\hat{\xi})J^e(\hat{\xi})d\hat{\xi}\right]t_j^{ea} \end{aligned} \tag{10}$$

where u_k^{ei} and t_k^{ei} are displacements and tractions control variables, respectively, corresponding to shape function i on element e; $\hat{\xi}\in[-1,1]$ is the local coordinate system, e_0 denotes the element containing the collocation point $\mathbf{x}^c$, $\hat{\xi}_c$ is the local coordinate of the collocation point, and $J^e(\hat{\xi})$ is the Jacobian of transformation, is defined as [17]

$$J^e(\hat{\xi}) = \frac{d\Gamma}{d\xi}\frac{d\xi}{d\hat{\xi}} \tag{11}$$

with

$$\frac{d\Gamma}{d\xi} = \sqrt{\left(\frac{dx}{d\xi}\right)^2 + \left(\frac{dy}{d\xi}\right)^2} \tag{12a}$$

$$\frac{d\xi}{d\hat{\xi}} = \frac{\xi_2 - \xi_1}{2} \tag{12b}$$

where ξ_1 and ξ_2 represent the values of the both ends of the element in the parameter space.

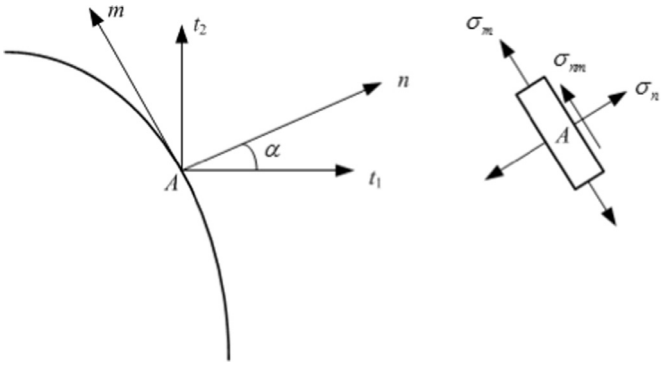

Fig. 2. Stress components of boundary points.

In IGA-BEM, a system of equations can be obtained by collocating at a series of control points. In this study, we selected the collocation scheme given by Greville abscissae, i.e. at the points [17]:

$$\xi_i^c = \frac{\xi_{i+1} + \xi_{i+2} + \cdots + \xi_{i+p}}{p} (i = 1, 2, \cdots, n-1) \tag{13}$$

In this study, singular integrals are evaluated using the subtraction of singularity scheme [26]. The weakly singular integrals, i.e. those containing the logarithmic terms in u_{ij}^*, are evaluated with transformation [64], which removes the singularity by ensuring that the Jacobian of the transformation approaches zero at the singular point.

3.4. Computation of displacements and stresses at any point on the boundary and interior region

After Eq. (10) is solved, the displacements and tractions of any point on the boundary can be obtained through Eq. (9). However, in engineering analysis, the stresses on the boundary are often required. We briefly introduce a method to calculate the stress at any point on the boundary. The detailed instruction can be found in [63]. As shown in Fig. 2, t_1 and t_2 are respectively the surface force components in the direction of x and y, m and n represent the tangential axis and normal axis of point A, α is the angle between normal axis n and t_1.

If the displacement and the surface force of point A have been obtained, σ_n, σ_{mn} and σ_m are computed with the following equations.

$$\sigma_n = t_1 \cos\alpha + t_2 \sin\alpha \tag{14a}$$

$$\sigma_{mn} = -t_1 \sin\alpha + t_2 \cos\alpha \tag{14b}$$

$$\sigma_m = \frac{E}{1-\nu}\varepsilon_m + \frac{\nu}{1-\nu}\sigma_n, \tag{14c}$$

where $E = 2G(1+\nu)$, ε_m is the strain in the m direction, which can be calculated by the displacement value in the m direction as follows

$$u_m = -u_1 \sin\alpha + u_2 \cos\alpha \tag{15a}$$

$$\varepsilon_m = \frac{\partial u_m}{\partial m} \tag{15b}$$

The displacement and stress at the internal point $\mathbf{x}'$ can be obtained with [63]:

$$u_i(\boldsymbol{x}') = \sum_{e_0=1}^{Ne}\sum_{a_0=1}^{m}\int_{-1}^{+1} u_{ij}^*\left(\boldsymbol{x}', \boldsymbol{x}(\hat{\xi})\right) R_{a_0}(\hat{\xi}) J^{e_0}(\hat{\xi}) t_j^{e_0 a_0} d\hat{\xi} - \sum_{e_0=1}^{Ne}\sum_{a_0=1}^{m}\int_{-1}^{+1} t_{ij}^*\left(\boldsymbol{x}', \boldsymbol{x}(\hat{\xi})\right) R_{a_0}(\hat{\xi}) J^{e_0}(\hat{\xi}) u_j^{e_0 a_0} d\hat{\xi} \tag{16a}$$

$$\sigma_{ij}(\boldsymbol{x}') = \sum_{e_0=1}^{Ne}\sum_{a_0=1}^{m}\int_{-1}^{+1} D_{kij}\left(\boldsymbol{x}', \boldsymbol{x}(\hat{\xi})\right) R_{a_0}(\hat{\xi}) J^{e_0}(\hat{\xi}) t_k^{e_0 a_0} d\hat{\xi} - \sum_{e_0=1}^{Ne}\sum_{a_0=1}^{m}\int_{-1}^{+1} S_{kij}\left(\boldsymbol{x}', \boldsymbol{x}(\hat{\xi})\right) R_{a_0}(\hat{\xi}) J^{e_0}(\hat{\xi}) u_k^{e_0 a_0} d\hat{\xi} \tag{16b}$$

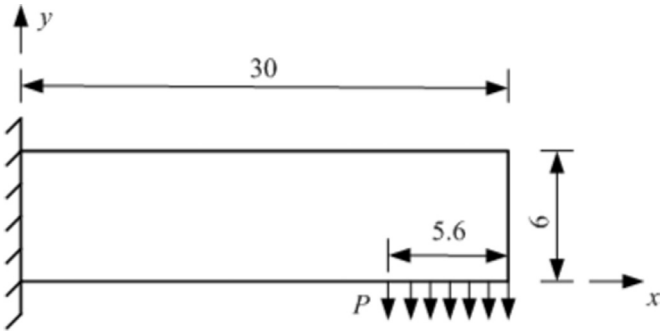

Fig. 3. Geometry and load of the cantilever beam.

with

$$D_{kij} = \frac{1}{4\pi(1-\nu)r}\left[(1-2\nu)\left(r_k\delta_{ij} + r_i\delta_{kj} - r_j\delta_{ki}\right) + 2r_k r_i r_j\right] \tag{17a}$$

$$S_{kij} = \frac{G}{2\pi(1-\nu)r^2}\left\{2\frac{\partial r}{\partial n}\left[(1-2\nu)r_j\delta_{ki} + \nu\left(r_i\delta_{kj} + r_k\delta_{ij}\right) - 4r_k r_i r_j\right] + 2\nu\left(n_k r_i + n_i r_k\right)r_j + (1-2\nu)\left(2n_j r_k r_i + n_i\delta_{kj} + n_k\delta_{ij}\right) - (1-4\nu)n_j\delta_{ki}\right\} \tag{17b}$$

4. Optimization algorithm

4.1. Particle swarm optimization

The particle swarm optimization (PSO) proposed by Kennedy and Eberhart [59] is an intelligent optimization algorithm, which is inspired by the foraging behavior of birds. In PSO algorithm, a swarm of particles is considered, i.e. every possible solution of the optimization problem is considered as a particle which corresponds to a bird in the bird flock, and the swarm is promoted to optimal solution by updating the position of particles based on their fitness. A candidate group of random solutions is initiated, and then optimal solution in the space of the problem is searched by updating the position and velocity of particles. The position of the particles is the desired solution of the optimization problem and their velocity corresponds to the rate of change of the position variations.

If $\boldsymbol{x}_i^k$ and v_i^k are the position and velocity of the ith particle in the kth iteration, respectively, then the new velocity and position of the ith particle in the kth iteration are updated as follows [65].

$$v_i^{k+1} = \omega v_i^k + c_1\,\mathrm{rand}_1^k\left(\mathrm{pbest}_i^k - x_i^k\right) + c_2\,\mathrm{rand}_2^k\left(\mathrm{gbest}^k - x_i^k\right) \tag{18a}$$

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k + v_i^{k+1} \tag{18b}$$

where ω is the inertia weight, which is a non-negative value which controls the effect of the former speed on the current speed; $v_i^k \in [-v_{\max}, v_{\max}]$, $v_{\max}$ is the maximum velocity of a particle and it is a constant for the given problem and it should be pointed out that if the velocity of the particle is faster than its maximum value $v_{\max}$, it will be set to $v_{\max}$; pbest_i^k is the best position of the ith particle (i.e., the position of the individual extreme point) and gbest^k is the position of the best particle, whose fitness value is the smallest, up to the kth iteration; c_1 and c_2 are learning factors or acceleration coefficients, which adjust the maximum step towards to the individual best particle and the global best particle, respectively. If they are too small, the particle may be away from the target area, if they are too large, it may cause a suddenly fly to the target area, or fly over the target area. c_1 and c_2 take the values between 1.5 and 2; rand_1^k and rand_2^k are random two numbers between 0 and 1.

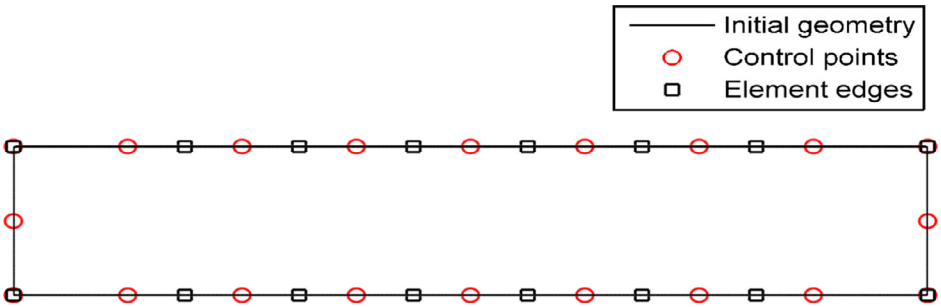

Fig. 4. The design mesh of the cantilever beam.

Table 1
The initial coordinates and weights of control points for the cantilever beam.

No. of control point	x	y	Weight	No. of control point	x	y	Weight
1	0	0	1	11	30	6	1
2	3.75	0	1	12	26.25	6	1
3	7.5	0	1	13	22.5	6	1
4	11.25	0	1	14	18.75	6	1
5	15	0	1	15	15	6	1
6	18.75	0	1	16	11.25	6	1
7	22.5	0	1	17	7.5	6	1
8	26.25	0	1	18	3.75	6	1
9	30	0	1	19	0	6	1
10	30	3	1	20	0	3	1

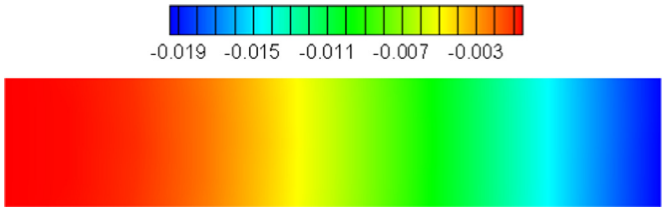

Fig. 5. Distribution of displacement in y-direction.

A fitness function is a particular type of objective function that is used to summarize, as a single figure of merit, how close a given design solution is to achieving the set aims. The fitness function for the shape optimization problem shown in Eq. (1) is expresses as follows

$$\text{Fitness} = F(\boldsymbol{x}) + \zeta \times \max\left(g_j, 0\right) + \zeta' \times \left|h_i\right| \tag{19}$$

where Fitness is the fitness function, F is the objective function, g_j and h_i, respectively, represent the inequality constraints and equality constraints, ζ and ζ' are the penalty factors.

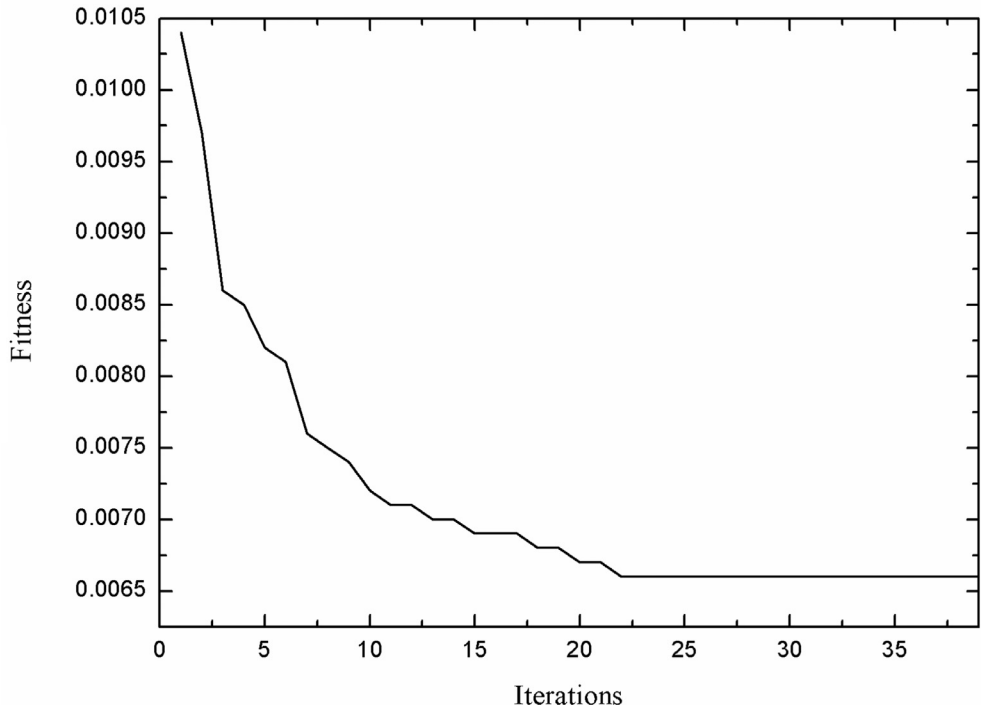

Fig. 6. Iteration process of optimization.

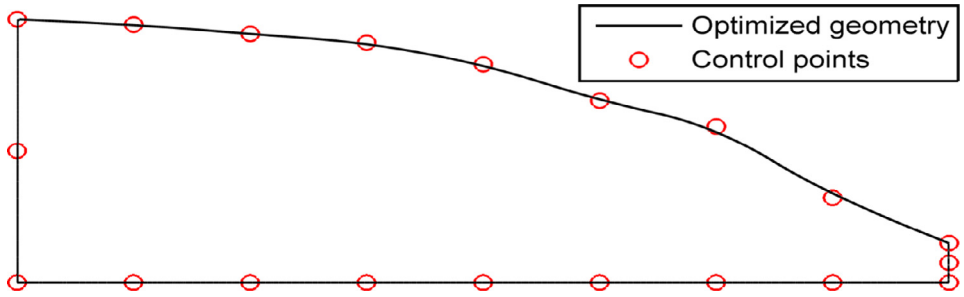

Fig. 7. Optimal shape of cantilever beam.

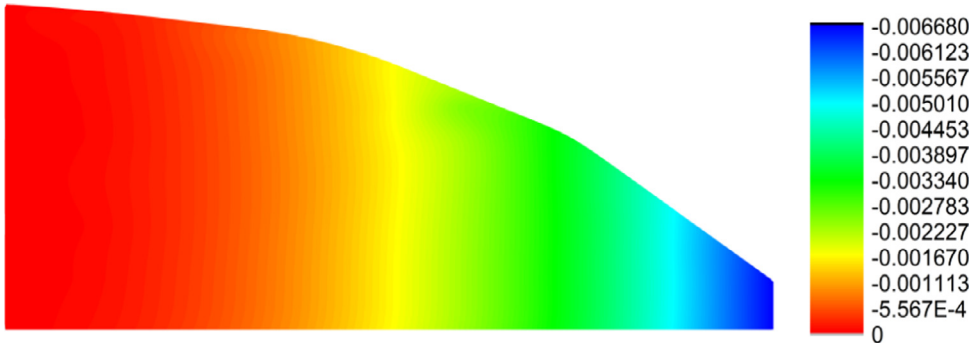

Fig. 8. Distribution of displacement in *y*-direction for optimized shape.

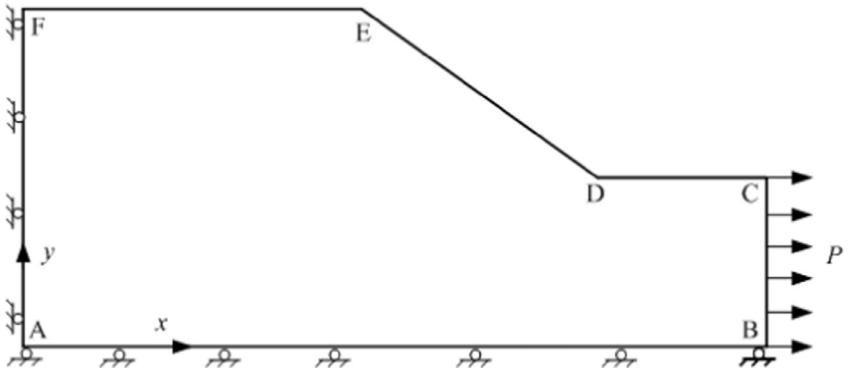

Fig. 9. Geometry and load of the fillet.

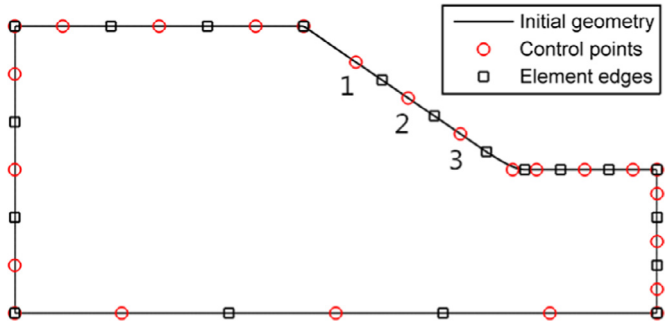

Fig. 10. The design mesh of the fillet.

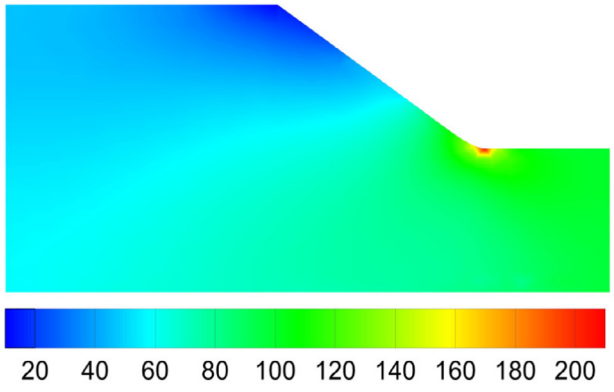

Fig. 11. The von Mises stress distribution in the initial structure.

In this study, the equality constraints are not considered, so the fitness function shown in Eq. (19) is simplified as

$$\text{Fitness} = F(\boldsymbol{x}) + \zeta \times \max\left(g_j, 0\right) \tag{20}$$

It is noted that too large value of ζ may lead to a local optimum, while the optimized target may not meet the constraints for a too small value of ζ. The criteria of choosing the value of the penalty factor value is to set the second term on the right hand side of Eq. (19) to be one or two orders of magnitude greater than the first term. The detailed value of the penalty factor is determined with the trial method according

Table 2
Variation of design variables in the cantilever beam optimization procedure.

No. of design variable	Lower bound	Upper bound	Initial value	Optimal value (PSO)	Optimal value (MMA) [34]
1	1.5	10	6	1.5000	1.5001
2	1.5	10	6	3.2247	3.0951
3	1.5	10	6	5.9346	5.5876
4	1.5	10	6	6.9153	6.9434
5	1.5	10	6	8.2886	8.2222
6	1.5	10	6	9.1151	9.1364
7	1.5	10	6	9.4473	9.9619
8	1.5	10	6	9.8001	9.9999
9	1.5	10	6	10.000	10.0000

Table 3
The initial coordinates and weights of control points for the fillet.

No. of control point	x	y	Weight	No. of control point	x	y	Weight
1	0	0	1	13	15.5	4.5	1
2	3.333	0	1	14	13.875	5.625	1
3	10	0	1	15	12.25	6.75	1
4	16.667	0	1	16	10.625	7.875	1
5	20	0	1	17	9	9	1
6	20	0.75	1	18	7.5	9	1
7	20	2.25	1	19	4.5	9	1
8	20	3.75	1	20	1.5	9	1
9	20	4.5	1	21	0	9	1
10	19.25	4.5	1	22	0	7.5	1
11	17.75	4.5	1	23	0	4.5	1
12	16.25	4.5	1	24	0	1.5	1

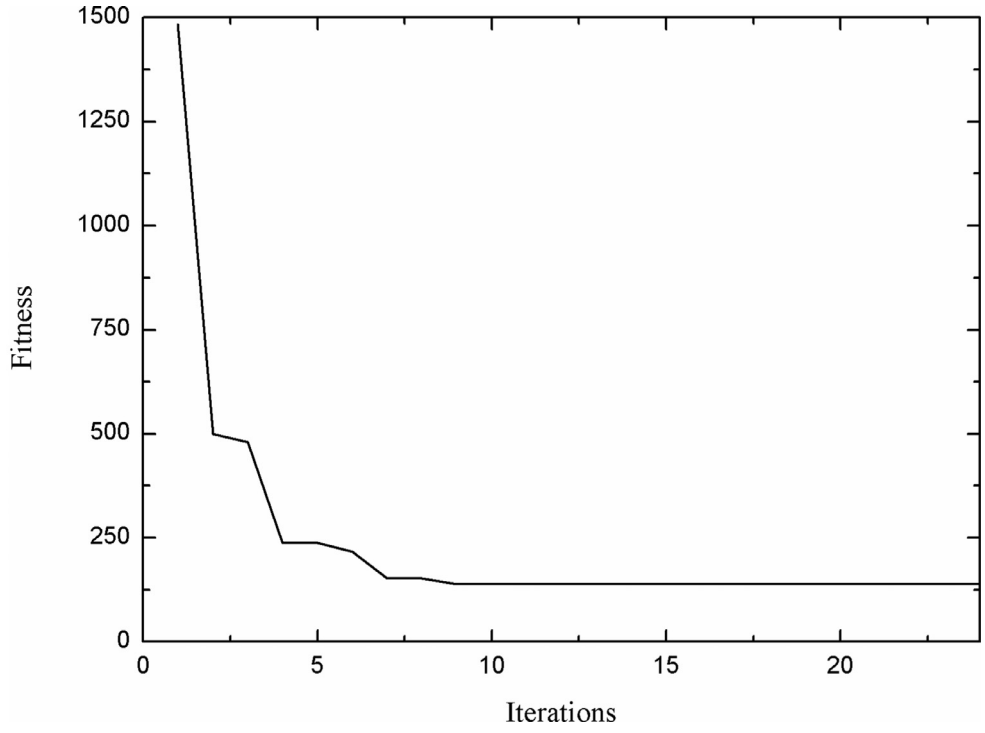

Fig. 12. Iteration process of optimization.

to the convergence curve. If premature convergence of the results is obtained, then the value of the penalty factor is reduced. However, when the convergence ratio is slow, the penalty factor value is increased.

4.2. Implementation of the PSO

The key steps for implementation of the PSO algorithm are described as follows:

Step 1: Initialize the parameters: the maximum number of iterations N_1, the number of consecutive iterations N_2, the learning factors c_1 and c_2, the inertia weight ω, the initial position $\boldsymbol{x}_i^0$ and the initial velocity v_i^0 of particles, and the error tolerance. The initial position $\boldsymbol{x}_i^0$ and initial velocity v_i^0 of particles are randomly generated in the allowable range, the optimal coordinate of each particle pbest is its current position. Compute the corresponding individual extremum (i.e., the fitness value of individual extreme point), the global extremum (i.e., the fitness value of global ex-

Table 4
Variation of design variables.

No. of design variable	Lower bound	Upper bound	Initial value	Optimal value (PSO)	Optimal value (MMA) [34]
1	4.5	9	7.875	6.2517	6.0814
2	4.5	9	6.75	5.2158	5.1486
3	4.5	9	5.625	4.7057	4.6895

Table 5
Variation of design variables.

Method	Parameters	Optimal value		
		Design variable 1	Design variable 2	Design variable 3
PSO	c_1=1.5, c_2=1.5, ω=0.6, δ=0.01, N_2=15	6.2517	5.2158	4.7057
	c_1=2, c_2=2, ω=0.6, δ=0.01, N_2=15	6.2521	5.2156	4.7057
	c_1=1.5, c_2=1.5, ω=1.0, δ=0.01, N_2=15	6.2635	5.2150	4.7147
MMA [34]		6.0814	5.1486	4.6895

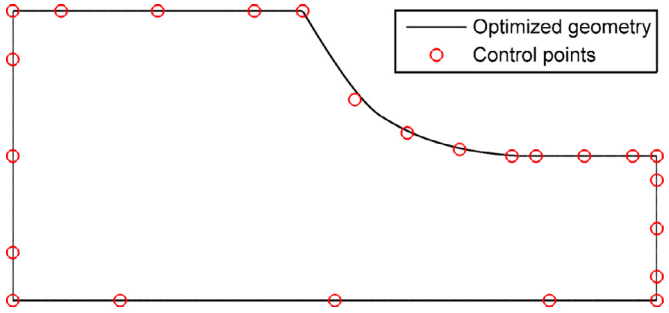

Fig. 13. The optimized shape.

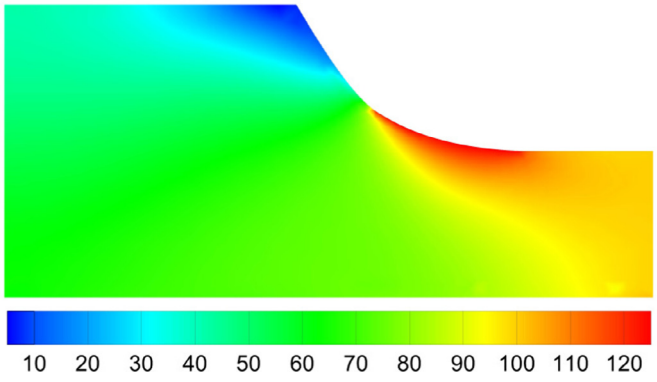

Fig. 14. The von Mises stress contour of the optimized structure.

treme point) is the best of individual extremums, record the optimal particle number, and set the current position of the best particle as gbest.

Step 2: Assess each particle. Calculate the fitness value of each particle, if it is better than that of the current individual extremum of the particle (i.e., the fitness value is getting smaller), set the position of the particle as pbest and update the individual extremum. If the best individual extremum of all particles is better than that of the current global extremum, set its location as gbest, record the number of the particle and update the global extremum.

Step 3: Update the particles. The position $\mathbf{x}_i$ and velocity v_i of each particle are updated using Eq. (18).

Step 4: Verify the end condition. If the current iteration number reaches the pre-set maximum number or the relative difference of fitness is less than the error tolerance δ in consecutive N_2 iterations, stop the iteration and output the optimal solution, otherwise go to Step 2.

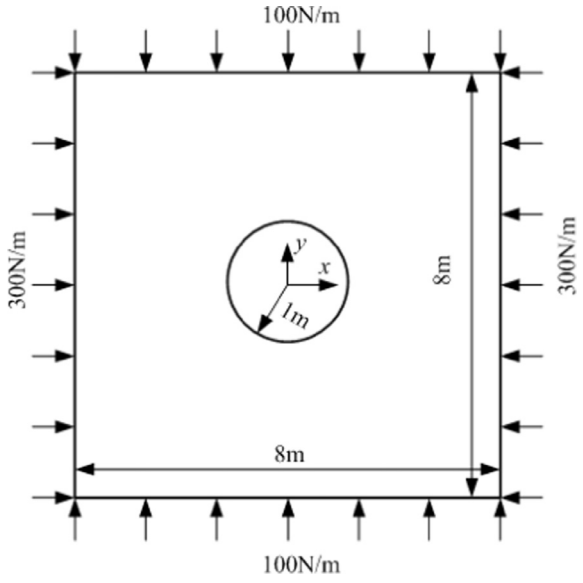

Fig. 15. Geometry and load of a square plate with one circular cutout.

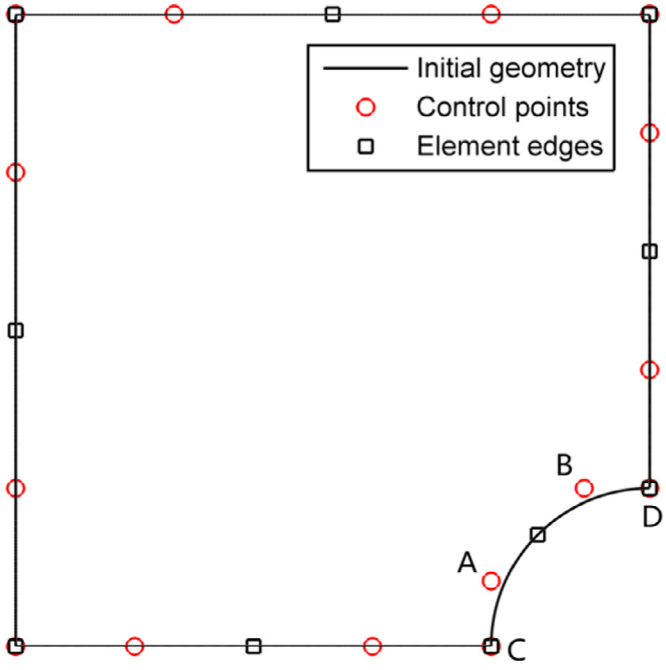

Fig. 16. The design mesh.

Table 6
The initial coordinates and weights of control points for a square plate with one circular cutout.

No. of control point	x	y	Weight	No. of control point	x	y	Weight
1	−4.0000	0.0000	1.0000	19	0.0000	2.7500	1.0000
2	−3.7500	0.0000	1.0000	20	0.0000	3.2500	1.0000
3	−3.2500	0.0000	1.0000	21	0.0000	3.7500	1.0000
4	−2.7500	0.0000	1.0000	22	0.0000	4.0000	1.0000
5	−2.2500	0.0000	1.0000	23	−0.3333	4.0000	1.0000
6	−1.7500	0.0000	1.0000	24	−1.0000	4.0000	1.0000
7	−1.2500	0.0000	1.0000	25	−1.6667	4.0000	1.0000
8	−1.0000	0.0000	1.0000	26	−2.3333	4.0000	1.0000
9	−1.0000	0.1239	0.9512	27	−3.0000	4.0000	1.0000
10	−0.9373	0.3730	0.8861	28	−3.6667	4.0000	1.0000
11	−0.8047	0.6095	0.8536	29	−4.0000	4.0000	1.0000
12	−0.6095	0.8047	0.8536	30	−4.0000	3.6667	1.0000
13	−0.3730	0.9373	0.8861	31	−4.0000	3.0000	1.0000
14	−0.1239	1.0000	0.9512	32	−4.0000	2.3333	1.0000
15	0.0000	1.0000	1.0000	33	−4.0000	1.6667	1.0000
16	0.0000	1.2500	1.0000	34	−4.0000	1.0000	1.0000
17	0.0000	1.7500	1.0000	35	−4.0000	0.3333	1.0000
18	0.0000	2.2500	1.0000				

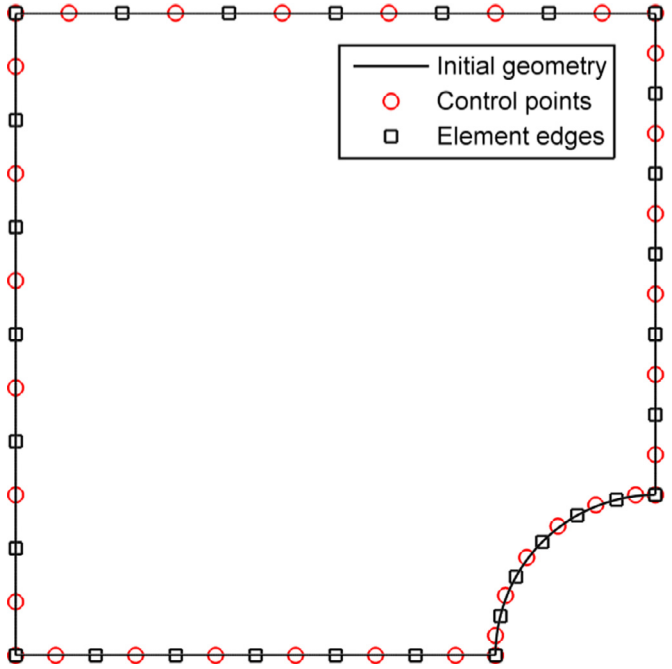

Fig. 17. The analysis mesh.

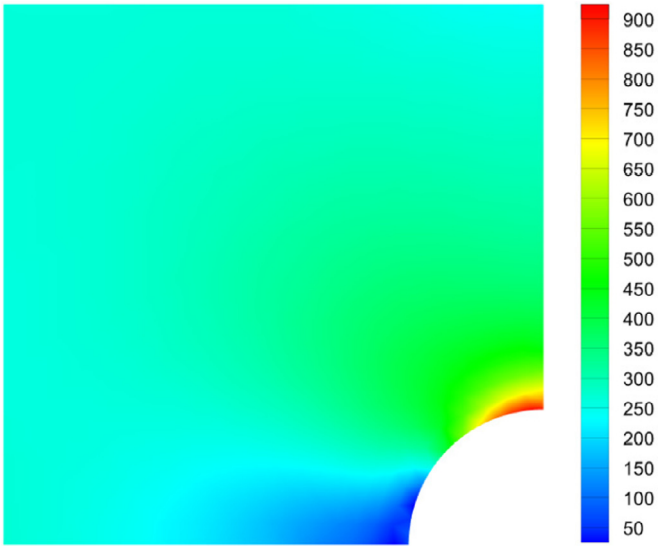

Fig. 18. The von Mises stress contour of the initial structure.

4.3. Comparative analysis of PSO and GA

GA is a metaheuristic method inspired by the process of natural selection [66], and it can generate high-quality solution to optimization problem by conducting bio-inspired operators such as mutation, crossover and selection. In a typical GA, a genetic representation and a fitness function are defined, which represents the solution domain and evaluates the solution domain, respectively. A GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators. Commonly, the solutions are represented in binary as strings of 0 s and 1 s. The optimization process of GA is governed by several control parameters including the size of the population and the probability that a given encoded trial solution undergoes crossover and mutation.

The structure of PSO is simple and easy to control, so its performance is more stable for the dynamic objective function [67]. In solving some typical function optimization problems, PSO can get better results than GA [68]. GA is a mature function optimization algorithm, but it is often difficult to meet the desired requirements in convergence speed of optimization and precision for the super high-dimensional and multi-local extreme complex function problems. PSO has not the complex operations of select, crossover, and mutation operator, so it is more suited to the solution of such problems, it can achieve the global optimal value faster [69]. The speed and accuracy of the PSO has improved for most of the nonlinear benchmark functions optimization problems than GA [70]. As an effective new algorithm, PSO can be used to solve all problems to which the GA does.

5. Numerical applications

In this section, four 2D shape optimization problems under static loading are considered to show the accuracy and performance of the present approach. In all numerical calculations, quadratic order NURBS basis functions are used, while a set of 12 Gaussian quadrature points is taken for the integration of each element. The parameters in the PSO algorithm are given as follows: the maximum number of iterations N_1=100; the learning factor c_1=1.5 and c_2=1.5; the inertia weight ω=0.6; the error tolerance is δ=0.01, and the number of consecutive iterations N_2=15. The larger inertia weight can enhance the global searching ability of PSO, while the smaller inertia weight can enhance the local

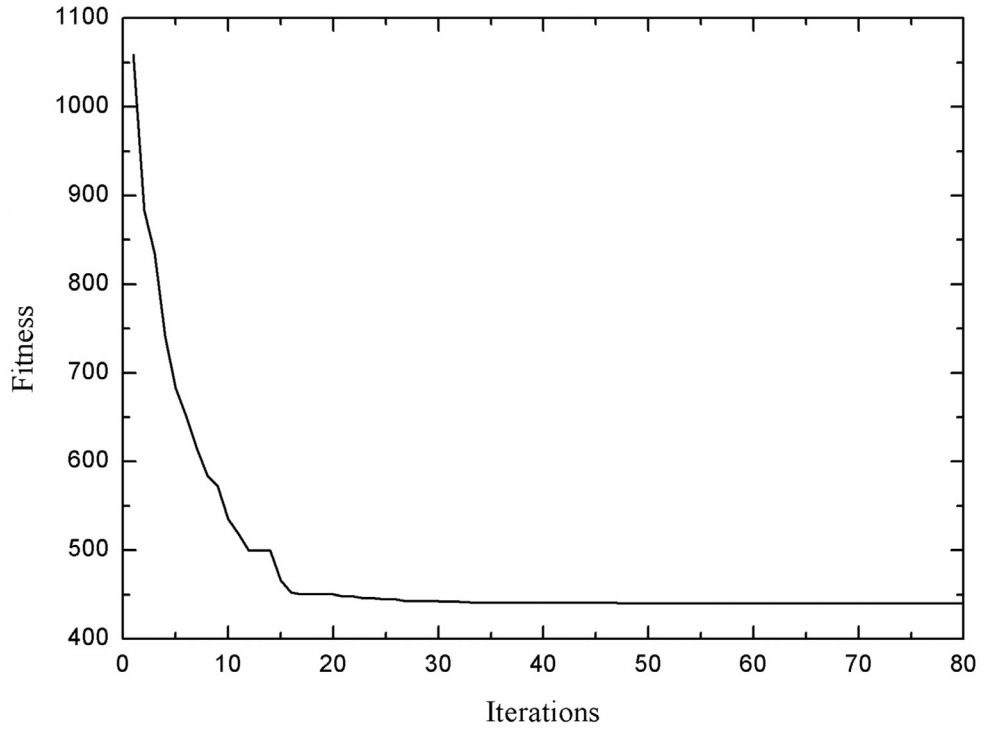

Fig. 19. Iteration process of optimization.

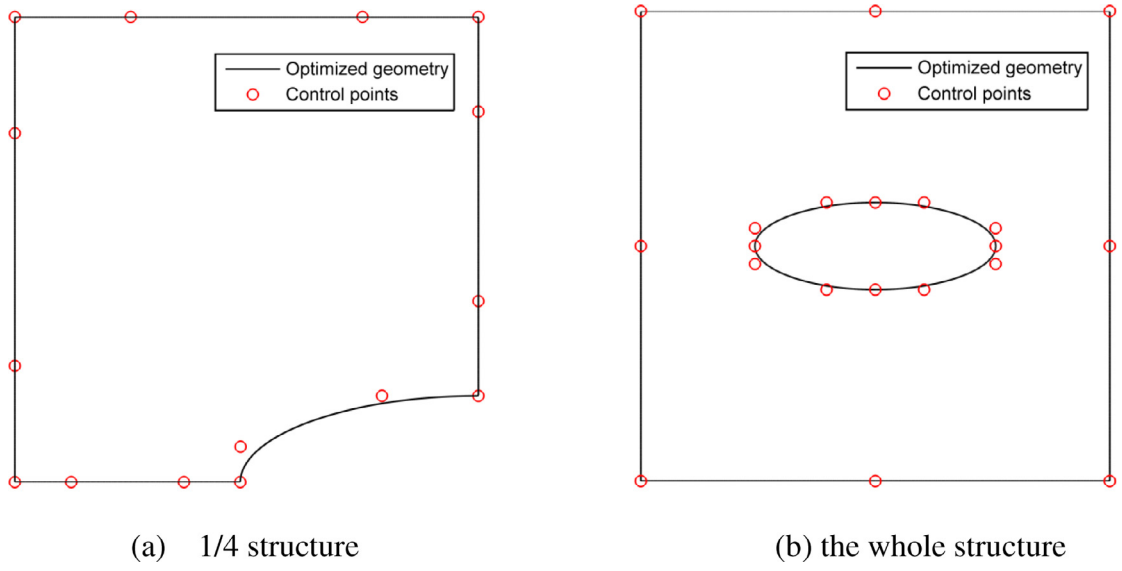

Fig. 20. Optimized shape.

Table 7
Variation of design variables.

No. of design variable	Lower bound	Upper bound	Initial value	Optimal value
A	(−4,0)	(0,4)	(−1, 0.4142)	(−2.0523 , 0.3046)
B	(−4,0)	(0,4)	(−0.4142,1)	(−0.8322 , 0.7422)
x coordinate of point C	−4	0	−1	−2.0546
y coordinate of point D	0	4	1	0.7416

Table 8
Variation of design variables.

No. of design variable	Lower bound	Upper bound	Initial value	Optimal value
a	27	58.25	50.4375	54.7899
b	27	58.25	42.6250	46.8142
c	27	58.25	34.8125	31.9088

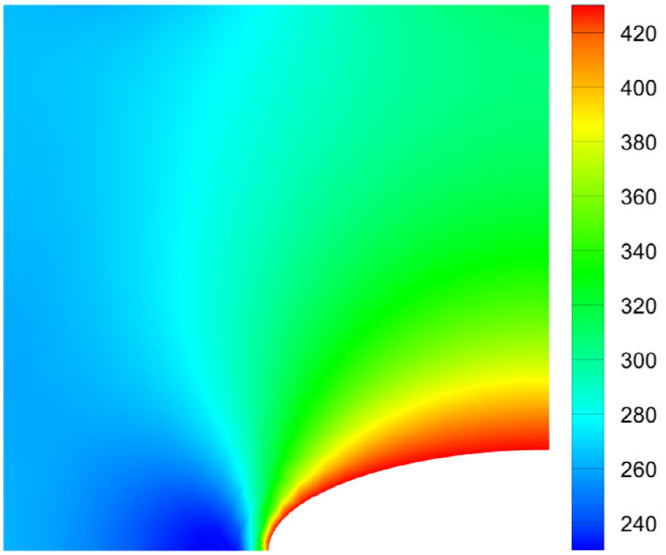

Fig. 21. The von Mises stress contour of the optimized structure.

searching ability. The experimental results show that PSO has faster convergence speed for the inertia weight between [0.6, 1.2]. Our test results show that the good convergence speed of the PSO can be obtained when the inertia weight is chosen as 0.6.

5.1. Example #1: cantilever beam

A cantilever beam of length 30 units and height 6 units, as schematically depicted in Fig. 3, is considered. The bottom surface of the beam is subjected to a uniform traction of $P=2$ units, applied to the end segment of the bar with the length 5.6 units. The material parameters are the Young's modulus $E=2.1\times10^5$ units and the Poisson ratio $\nu=0.3$.

The initial design mesh is shown in Fig. 4, and it is also the analysis mesh. Table 1 shows that the initial coordinates and weights of control points. The displacements in y-direction are presented in Fig. 5, the maximum value occurs at the beam end. The optimization problem consists in minimizing the displacement in y-direction at the beam end while keeping the beam area not less than 220 units. The design variables are

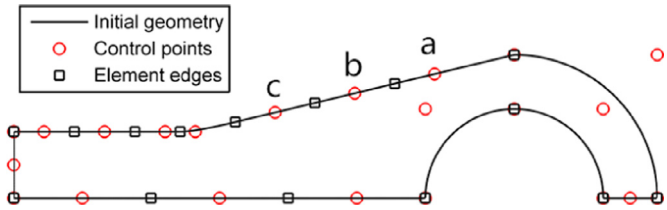

Fig. 23. The design mesh of the connecting rod.

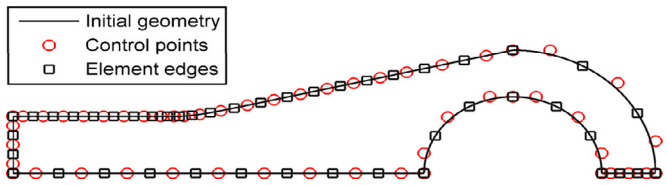

Fig. 24. The analysis mesh of the connecting rod.

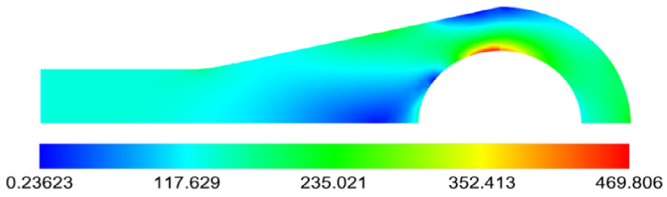

Fig. 25. The von Mises stress contour of the initial structure.

the vertical coordinates of nine control points on the top edge, the lower bound is 1.5 units and the upper bound is 10 units for each variable.

The fitness function is defined as follows

$$\text{Fitness} = |F(\boldsymbol{x})| + \zeta \times \max(A - 220, 0) \tag{21}$$

where $F(\mathbf{x})$ is the objective function representing the displacement in y-direction at the beam end, A is the area of the structure, ζ=100 is adopted in this study.

Fig. 6 shows the iteration process of optimization. As it can be seen from Fig. 6, the process converges after generation 22, when the fitness function remains quasi constant. Table 2 shows the variation of design variables. The optimized shape is shown in Fig. 7, and the area of the optimized beam is 219.2 units. From Figs. 5 and 8 it can be seen that the optimization objective, i.e. maximal vertical displacement of the initial

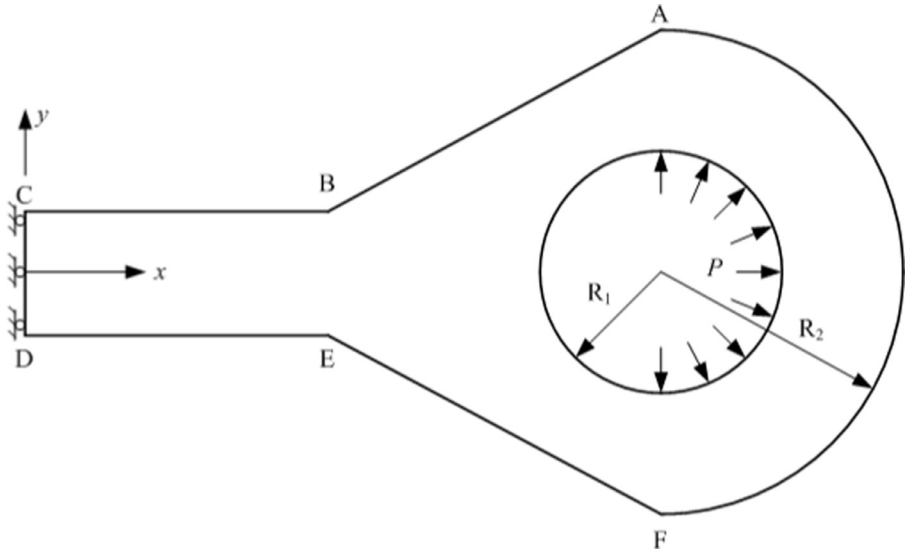

Fig. 22. Geometry and load of a connecting rod.

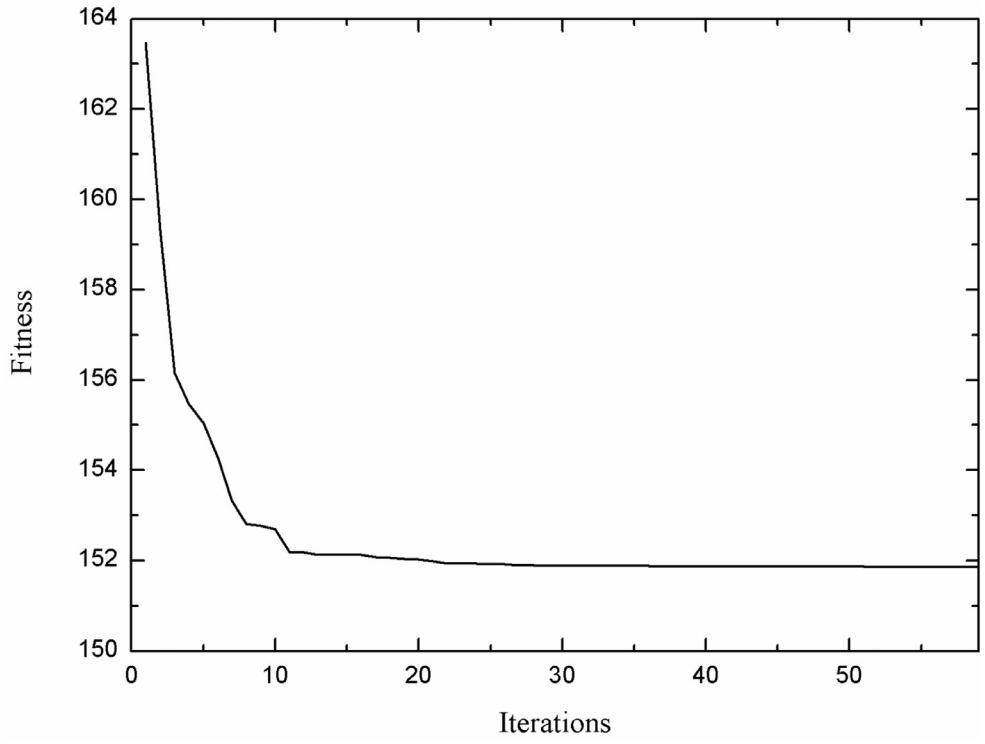

Fig. 26. Iteration process of optimization.

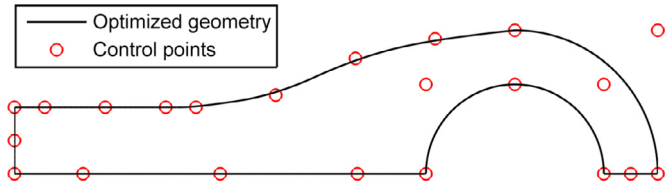

Fig. 27. Optimized shape.

shape is 0.019 units, while the maximal vertical displacement of the optimal shape is 0.0066 units. This corresponds to reduction in almost 35% while preserving a smooth geometry and satisfying the constraints. In Table 2 the obtained numerical results are also compared with the data from [34], obtained by the IGA-BEM based on the method of moving asymptotes (MMA). The results are in good agreement, however, the MMA requires complicated shape sensitivity analysis, i.e. the evaluation of the derivatives of quantities of interest with respect to design variables. It is noted that the convergence speed of gradient-based algorithms is generally faster than that of intelligent optimization algorithms. The iteration number is 6 with MMA in [34], while it is 22 with PSO, however, more coarse mesh with PSO is used than the mesh used with sensitivity analysis method like in [34].

5.2. *Example #2: fillet*

In the next example we consider a fillet subjected to uniform traction $P = 100$ units in the x-direction. Due to symmetry, only a half model is needed, as shown in Fig. 9. The geometry is given as follows: AB = 20 units, BC = CD = 4.5 units, AF = FE = 9 units. The material parameters are Young's modulus $E = 10^7$ units and the Poisson ratio $\nu = 0.3$.

The initial design mesh is shown in Fig. 10, and it is also the analysis mesh. Table 3 shows that the initial coordinates and weights of control points. The von Mises stress distribution in the initial structure is presented in Fig. 11. The stress concentration occurs at the point D, and the maximum value is 208.16 units. The optimization problem consists in minimizing the area of the fillet while keeping the von Mises stress below 125 units by changing the shape of section DE. The design variables are the vertical coordinates of control points 1–3 (Fig. 10). The lower bound is 4.5 units and the upper bound is 9 units for each variable.

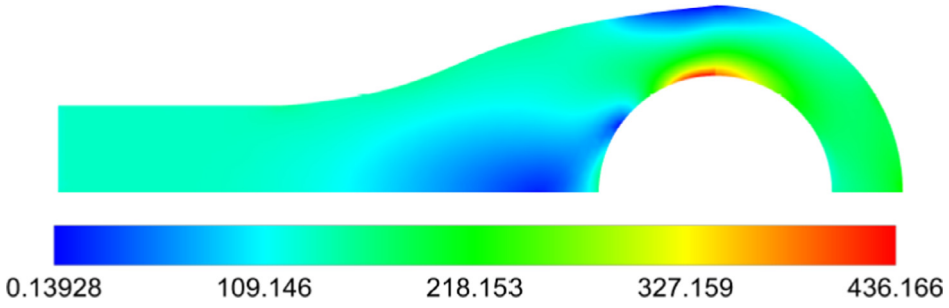

Fig. 28. The von Mises stress contour of the optimized structure.

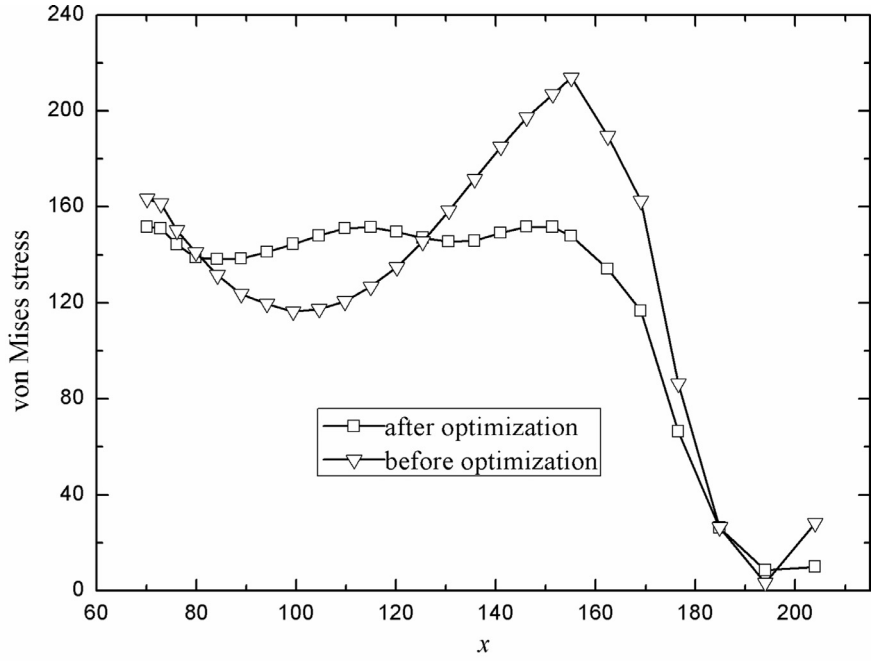

Fig. 29. The von Mises stress distribution along BA boundary.

The fitness function is defined as follows:

$$\text{Fitness} = F(\boldsymbol{x}) + \zeta \times \max\left(\sigma_{\max} - 125, 0\right) \tag{22}$$

where $F(\mathbf{x})$ is the objective function representing the area of the structure, $\sigma_{\max}$ is the maximum von Mises stress on CE segment, and penalty factor ζ=100 is adopted.

Fig. 12 shows the iteration process of optimization, where it can be seen that the process converged after 10 iterations. The optimal shape is shown in Fig. 13. Table 4 presents the variation of optimization variables. The area of the initial shape is 145.1602 units, while the area of the optimal shape is 138.7947 units. Fig. 14 shows the von Mises stress distribution in the optimal structure, which varies smoothly on segment DE. In Table 4 the results are compared with the data from [34] obtained with IGABEM with MMA and a good agreement between the results is shown.

In order to verify the robustness of the proposed approach, the effect of user-defined parameters in the PSO algorithm on optimized values is investigated. Table 5 presents the variation of design variables under different user-defined parameters in the PSO algorithm. From Table 5, it is found that the effect of user-defined parameters in the PSO algorithm on optimized values is small.

5.3. Example #3: a square plate with one circular cutout

A square plate with one circular cutout is shown in Fig. 15. The material parameters are the Young's modulus $E = 200$GPa and the Poisson ratio $\nu = 0.3$. Due to the symmetry of geometry, 1/4 annulus is considered for the investigation.

The initial design mesh is shown in Fig. 16, and the analysis mesh is shown in Fig. 17. Table 6 shows that the initial coordinates and weights of control points. The von Mises stress distribution in the initial structure is presented in Fig. 18. The stress concentration occurs at the boundary of the cutout, and the maximum value is 921 MPa. The optimization problem is stated as that the von Mises stress is minimized by changing the shape of the cutout, while keeping its area not smaller. The optimization variables are the coordinates of control points A and B, the x coordinate of control point C and the y coordinate of control point D.

The fitness function is defined as follows

$$\text{Fitness} = F(\boldsymbol{x}) + \zeta \times \max(0.25\pi - A, 0) \tag{23}$$

where $F(\mathbf{x})$ is the objective function representing the maximum von Mises stress on the cutout boundary, A is the area of the 1/4 cutout, penalty factor ζ=5000 is adopted in this study.

Fig. 19 shows the iteration process of optimization, which required 25 iterations to converge. The optimal shape is shown in Fig. 20, and the optimal shape is an ellipse, which is consistent with analytical solution. Table 7 presents the variation of optimization variables. Fig. 21 shows the von Mises stress distribution in the optimal structure, with the maximum value of 439.7 MPa.

5.4. Example #4: connecting rod

A simplified connecting rod problem is shown in Fig. 22. The geometry size is given as follows: AB = 133.7 units, BC = 74 units, CD = 27 units, $R_1 = 36.25$ units, $R_2 = 58.25$ units. The uniform pressure $P = 100$ units is applied at the inner edge, and the material parameters are $E = 10^7$ units and $\nu = 0.3$.

The objective of optimization is to reduce the stress peak value of section AB. The initial design mesh is shown in Fig. 23, where only 1/2 of the structure is considered, and the analysis mesh is shown in Fig. 24. The initial coordinates and weights of control points are given in Table A1 in the Appendix A. The optimization variables are the vertical coordinates of control points a, b and c (Fig. 23).

The fitness function is defined as follows

$$\text{Fitness} = F(\boldsymbol{x}) \tag{24}$$

where $F(\mathbf{x})$ is the objective function representing the maximum von Mises stress on BA segment.

Fig. 25 shows the von Mises stress distribution in the initial structure. Fig. 26 shows the iterative process of the optimization. The optimal shape is achieved after 20 iterations and it is shown in Fig. 27. Table 8 presents the variation of optimization variables. Fig. 28 shows the von Mises stress in the optimized structure. The variation of the von Mises stress of section BA before and after optimization is shown in Fig. 29. The stress in the optimized boundary is more uniform, and the stress peak at section BA is reduced from 213.8 units to 151.8 units.

6. Conclusions and outlook

An approach coupling the isogeometric boundary element method (IGA-BEM) and the particle swarm optimization (PSO) algorithm is developed for 2D structural shape optimization. The NURBS basis functions are employed to represent the boundary shape and approximate the physical fields in analysis. NURBS control points are used as design variables, thus design model, analysis model and optimization model can be uniformly described with the NURBS, and the optimized boundary is smooth. In addition, due to the use of boundary integrals, the IGA-BEM avoids the need for domain discretization, which remains a bottleneck in current volume-based IGA. The PSO, which is one of the gradient-free optimization algorithms, does not require complicated sensitivity analysis in comparison with the gradient-based optimization algorithms. Compared with the genetic algorithm (GA), the PSO avoids the "crossover" and "mutation" operations, and works on real numbers, so the rule and coding for the PSO is simpler than that for the GA.

In general, the convergence speed of gradient-based optimization algorithms is faster than that of the intelligent optimization algorithms such as PSO, but more coarse mesh with PSO can be used than the mesh with gradient-based optimization. The performance and accuracy of the present approach are verified through four 2D shape optimization examples.

The proposed methodology is a legitimate candidate to study structural shape optimization problems. However, the present formulation should be extended to handle 3D structure optimization. The present IGA-BEM method will also be improved using T-splines [71], or LR B-splines [72], which allow local refinement and handle complex geometry. In that circumstance, modeling of 3D structure optimization would be more efficient.

Acknowledgments

The authors would like to thank all the reviewers for their time and valuable comments/suggestions.

Appendix A. The control points data used for the connecting rod

Table A1
The initial coordinates and weights of control points for the connecting rod.

No. of control point	x	y	Weight	No. of control point	X	y	Weight
1	0.0000	0.0000	1.0000	27	204.0000	58.2500	1.0000
2	9.3193	0.0000	1.0000	28	193.1667	55.6458	1.0000
3	27.9581	0.0000	1.0000	29	175.1111	51.3056	1.0000
4	46.5970	0.0000	1.0000	30	160.6667	47.8333	1.0000
5	65.2360	0.0000	1.0000	31	149.8333	45.2292	1.0000
6	83.8750	0.0000	1.0000	32	139.0000	42.6250	1.0000
7	102.5140	0.0000	1.0000	33	128.1667	40.0208	1.0000
8	121.1530	0.0000	1.0000	34	117.3333	37.4167	1.0000
9	139.7919	0.0000	1.0000	35	106.5000	34.8125	1.0000
10	158.4307	0.0000	1.0000	36	95.6667	32.2083	1.0000
11	167.7500	0.0000	1.0000	37	84.8333	29.6042	1.0000
12	167.7500	9.4686	0.9024	38	76.2411	27.8681	1.0000
13	177.3711	26.6289	0.8373	39	69.8900	27.0000	1.0000
14	194.5314	36.2500	0.9024	40	65.7800	27.0000	1.0000
15	204.0000	36.2500	1.0000	41	60.2989	27.0000	1.0000
16	213.4686	36.2500	0.9024	42	53.4467	27.0000	1.0000
17	230.6289	26.6289	0.8373	43	45.2233	27.0000	1.0000
18	240.2500	9.4686	0.9024	44	37.0000	27.0000	1.0000
19	240.2500	0.0000	1.0000	45	28.7767	27.0000	1.0000
20	243.9167	0.0000	1.0000	46	20.5533	27.0000	1.0000
21	251.2500	0.0000	1.0000	47	12.3311	27.0000	1.0000
22	258.5833	0.0000	1.0000	48	4.1100	27.0000	1.0000
23	262.2500	0.0000	1.0000	49	0.0000	22.5000	1.0000
24	262.2500	15.2151	0.9024	50	0.0000	13.5000	1.0000
25	246.7899	42.7899	0.8373	51	0.0000	4.5000	1.0000
26	219.2151	58.2500	0.9024	52	0.0000	27.0000	1.0000

References

[1] Braibant V, Fleury C. Shape optimal design using B-splines. Comput Methods Appl Mech Eng 1984;44:247–67.
[2] Piegl L, Tiller W. The NURBS book. New York: Springer-Verlag, 1997.
[3] Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput Methods Appl Mech Eng 2005;194:4135–95.
[4] Yu TT, Bui TQ, Yin SH, Do TV, Tanaka S, Doan DH. On the thermal buckling analysis of functionally graded plates with internal defects using extended isogeometric analysis. Compos Struct 2016;136:684–95.
[5] Yu TT, Yin SH, Bui TQ, Liu Ch, Wattanasakulpong N. Buckling isogeometric analysis of functionally graded plates under combined thermal and mechanical loads. Compos Struct 2017;162:54–69.
[6] Bui TQ. Extended isogeometric dynamic and static fracture analysis for cracks in piezoelectric materials using NURBS. Comput Methods Appl Mech Eng 2015;295:470–509.
[7] Bazilevs Y, Calo MV, Zhang Y, Hughes RJT. Isogeometric fluid–structure interaction analysis with applications to arterial blood flow. Comput Mech 2006;38:310–22.
[8] Gomez H, Calo MV, Bazilevs Y, Hughes RJT. Isogeometric analysis of the Cahn–Hilliard phase-field model. Comput Methods Appl Mech Eng 2008;197:4333–52.
[9] Irzal F, Remmers JJ, Verhoosel CV, Borst R. Isogeometric finite element analysis of poroelasticity. Int J Numer Anal Meth Geomech 2013;37:1891–907.
[10] Nguyen MN, Bui TQ, Yu TT, Hirose S. Isogeometric analysis for unsaturated flow problems. Comput Geotech 2014;62:257–67.
[11] Bui QT, Hirose S, Zhang Ch, Rabczuk T, Wu CT, Saitoh T, et al. Extended isogeometric analysis for dynamic fracture in multiphase piezoelectric/piezomagnetic composites. Mech Mater 2016;97:135–63.
[12] Fischer P, Klassen M, Mergheim J, Steinmann P, Müller R. Isogeometric analysis of 2D gradient elasticity. Comput Mech 2011;47(3):325–34.
[13] Yin SH, Yu TT, Bui QT, Zheng XJ, Tanaka S. In-plane material inhomogeneity of functionally graded plates: a higher-order shear deformation plate isogeometric analysis. Compos Part B Eng 2016;106:273–84.
[14] Wall WA, Frenzel MA, Cyron C. Isogeometric structural shape optimization. Comput Methods Appl Mech Eng 2008;197(33):2976–88.
[15] Seo Y, Kim H, Youn S. Isogeometric topology optimization using trimmed spline surfaces. Comput Methods Appl Mech Eng 2010;199(49):3270–96.
[16] Cottrell J, Hughes TJR, Bazilevs Y. Isogeometric analysis: toward integration of CAD and FEA. Chichester: Wiley; 2009.
[17] Simpson RN, Bordas SPA, Trevelyan J, Rabczuk T. A two-dimensional isogeometric boundary element method for elastostatic analysis. Comput Methods Appl Mech Eng 2012;209-212(324):87–100.
[18] Simpson RN, Bordas SPA, Lian H, Trevelvan J. An isogeometric boundary element method for elastostatic analysis: 2D implementation aspects. Comput Struct 2013;118(6):2–12.
[19] Gu J, Zhang J, Sheng X, Li G. B-spline approximation in boundary face method for three-dimensional linear elasticity. Eng Anal Bound Elem 2011;35(11):1159–67.
[20] Belibassakis KA, Gerostathis TP, Kostas KV, Politis CG, Kaklis PD, Ginnis AI, et al. A BEM-isogeometric method for the ship wave-resistance problem. Ocean Eng 2013;60(2):53–67.
[21] Ginnis AI, Kostas KV, Politis CG, et al. Isogeometric boundary-element analysis for the wave-resistance problem using T-splines. Comput Methods Appl Mech Eng 2014;279:425–39.
[22] Politis C, Ginnis AI, Kaklis PD, Belibassakis K, Feurer C. An isogeometric BEM for exterior potential-flow problems in the plane. In: Proceedings of the 2009 SIAM/ACM joint conference on geometric and physical modeling. Springer; 2009. p. 349–54.
[23] Heltai L, Arroyo M, Desimone A. Nonsingular isogeometric boundary element method for stokes flows in 3D. Comput Methods Appl Mech Eng 2014;268:514–39.
[24] Simpson RN, Scott MA, Taus M, Thomas DC, Lian H. Acoustic isogeometric boundary element analysis. Comput Methods Appl Mech Eng 2014;269(2):265–90.
[25] Nguyen BH, Tran HD, Anitescu C, Zhuang X, Rabczuk T. An isogeometric symmetric Galerkin boundary element method for two-dimensional crack problems. Comput Methods Appl Mech Eng 2016;306:252–75.
[26] Peng X, Atroshchenko E, Kerfriden P, Bordas SPA. Linear elastic fracture simulation directly from CAD: 2D NURBS-based implementation and role of tip enrichment. Int J Fract 2017;204(1):55–78.
[27] Peng X, Atroshchenko E, Kerfriden P, Bordas SPA. Isogeometric boundary element methods for three dimensional static fracture and fatigue crack growth. Comput Methods Appl Mech Eng 2016;316:151–85.
[28] Beer G, Marussig B, Zechner J, Dünser C, Fries TP. Isogeometric boundary element analysis with elasto-plastic inclusions. Part 1: plane problems. Comput Methods Appl Mech Eng 2016;308:552–70.
[29] Beer G, Mallardo V, Ruocco E, Marussig B, Zechner J, Dünser C, et al. Isogeometric boundary element analysis with elasto-plastic inclusions. Part 2: 3-D problems. Comput Methods Appl Mech Eng 2017;315:418–33.
[30] Li K, Qian X. Isogeometric analysis and shape optimization via boundary integral. Comput Aided Des 2011;43(11):1427–37.
[31] Kostas KV, Ginnis AI, Politis CG, Kaklis PD. Ship-hull shape optimization with a T-spline based BEM–isogeometric solver. Comput Methods Appl Mech Eng 2015;284:611–22.
[32] Yoon M, Choi M, Cho S. Isogeometric configuration design optimization of heat conduction problems using boundary integral equation. Int J Heat Mass Transf 2015;89:937–49.
[33] Yoon M, Cho S. Isogeometric shape design sensitivity analysis of elasticity problems using boundary integral equations. Eng Anal Bound Elem 2016;66:119–28.

[34] Lian HJ, Kerfriden P, Bordas SPA. Implementation of regularized isogeometric boundary element methods for gradient-based shape optimization in two-dimensional linear elasticity. Int J Numer Meth Eng 2016;106:972–1017.
[35] Takahashi T, Matsumoto T. An application of fast multipole method to isogeometric boundary element method for Laplace equation in two dimensions. Eng Anal Bound Elem 2012;36(12):1766–75.
[36] Simpson RN, Liu Z. Acceleration of isogeometric boundary element analysis through a black-box fast multipole method. Eng Anal Bound Elem 2016;66:168–82.
[37] Marussig B, Zechner J, Beer G, Fries TP. Fast isogeometric boundary element method based on independent field approximation. Comput Methods Appl Mech Eng 2015;284:458–88.
[38] Scott MA, Simpson RN, Evans JA, Lipton S, Bordas SPA, Hughes TJR, et al. Isogeometric boundary element analysis using unstructured T-splines. Comput Methods Appl Mech Eng 2013;254:197–221.
[39] Peake MJ, Trevelyan J, Coates G. Extended isogeometric boundary element method (XIBEM) for two-dimensional Helmholtz problems. Comput Methods Appl Mech Eng 2013;259:93–102.
[40] Wang Y, Benson DJ, Nagy AP. A multi-patch nonsingular isogeometric boundary element method using trimmed elements. Comput Mech 2015;56(1):173–91.
[41] Feischl M, Gantner G, Haberl A, Praetorius D. Adaptive 2D IGA boundary element methods. Eng Anal Bound Elem 2016;62:141–53.
[42] Hassani B, Khanzadi M, Mehdi Tavakkoli S. An isogeometrical approach to structural topology optimization by optimality criteria. Struct Multidisc Optim 2012;45(2):223–33.
[43] Shojaee S, Mohamadian M, Valizadeh N. Composition of isogeometric analysis with level set method for structural topology optimization. Int J Optim Civil Eng 2012;2(1):47–70.
[44] Dedè L, Borden MJ, Hughes TJR. Isogeometric analysis for topology optimization with a phase field model. Arch Comput Methods Eng 2012;19:427–65.
[45] Tavakkoli SM, Hassani B. Isogeometric topology optimization by using optimality criteria and implicit function. Int J Optim Civil Eng 2014;4(2):151–63.
[46] Tavakkoli SM, Hassani B, Ghasemnejad H. Isogeometric topology optimization of structures by using MMA. Int J Optim Civil Eng 2013;3(2):313–26.
[47] Seo YD, Kim HJ, Youn SK. Isogeometric topology optimization using trimmed spline surfaces. Comput Methods Appl Mech Eng 2010;199:3270–96.
[48] Qian XP. Topology optimization in B-spline space. Comput Methods Appl Mech Eng 2013;265:15–35.
[49] Espath LFR, Linn RV, Awruch AM. Shape optimization of shell structures based on NURBS description using automatic differentiation. Int J Numer Meth Eng 2011;88(7):613–36.
[50] Koo B, Yoon M, Cho S. Isogeometric shape design sensitivity analysis using transformed basis functions for Kronecker delta property. Comput Methods Appl Mech Eng 2013;253:505–16.
[51] Park BU, Seo YD, Sigmund O, Youn SK. Shape optimization of the stokes flow problem based on isogeometric analysis. Struct Multidisc Optim 2013;48(5):965–77.
[52] Blanchard L, Duvigneau R, Vuong AV, Simeon B. Shape gradient for isogeometric structural design. J Optim Theory Appl 2014;161(2):361–7.
[53] Kiendl J, Schmidt R, Wüchner R, Bletzinger KU. Isogeometric shape optimization of shells using semi-analytical sensitivity analysis and sensitivity weighting. Comput Methods Appl Mech Eng 2014;274:148–67.
[54] Ding CS, Cui XY, Li GY. Accurate analysis and thickness optimization of tailor rolled blanks based on isogeometric analysis. Struct Multidisc Optim 2016;54(4):871–87.
[55] Nagy AP, Ijsselmuiden ST, Abdalla MM. Isogeometric design of anisotropic shells: optimal form and material distribution. Comput Methods Appl Mech Eng 2013;264:145–62.
[56] Le-Manh T, Lee J. Stacking sequence optimization for maximum strengths of laminated composite plates using genetic algorithm and isogeometric analysis. Compos Struct 2014;116:357–63.
[57] Herath MT, Natarajan S, Gangadhara Prusty B, John NS. Isogeometric analysis and genetic algorithm for shape-adaptive composite marine propellers. Comput Methods Appl Mech Eng 2015;284:835–60.
[58] Lian H, P Kerfriden, Bordas SPA. Shape optimization directly from CAD: an isogeometric boundary element approach using T-splines. Comput Methods Appl Mech Eng 2017;317:1–41.
[59] Kennedy J, Eberhart RC. Particle swarm optimization. In: Proceedings of the 1995 IEEE Conference on Neural Networks, IV. Piscataway; 1995. p. 1942–8.
[60] Kennedy J. The particle swarm: social adaptation of knowledge. In: Proceedings of the 2002 IEEE International Conference on Evolutionary Computation; 2002. p. 303–8.
[61] Perez RE, Behdinan K. Particle swarm approach for structural design optimization. Comput Struct 2007;85:1579–88.
[62] Shojaee S, Valizadeh N, Arjomand M. Isogeometric structural shape optimization using particle swarm algorithm. Int J Optim Civil Eng 2011;4:633–45.
[63] Becker AA. The boundary element method in engineering – a complete course. McGraw-Hill; 1992.
[64] Telles JCF. A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary integrals. Int J Numer Methods Eng 1987;24:959–73.
[65] Trelea IC. The particle swarm optimization algorithm: convergence analysis and parameter selection. Inf Process Lett 2003;85(6):317–25.
[66] Goldberg DE. Genetic algorithms in search, optimization and machine learning. Addison-Wesley: Reading, MA Publisher; 1989.
[67] Parsopoulos KE, Vrahatis MN. Particle swarm optimizer in noisy and continuously changing environments. In: Proceeding of the IASTED international conference on artificial intelligence and soft computing. LASTED/ACTS Press; 2001. p. 289–94.
[68] Van den Bergh F. An analysis of particle swarm optimizers. South Africa: University of Pretoria; 2002.
[69] Kennedy J, Spears WM. Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. In: Proceeding of the 1998 IEEE international conference on evolutionary computation. IEEE World Congress on Computational Intelligence. IEEE; 1998. p. 78–83.
[70] Shi YH, Eberhart R. Experimental study of particle swarm optimization. In: Proceeding of the 2000 SCI conference, Orlando; 2000.
[71] Singh SK, Singh IV, Mishra BK, Bhardwaj G, Bui TQ. A simple, efficient and accurate Bézier extraction based T-spline XIGA for crack simulations. Theor. Appl. Fract. Mech. 2017;88:74–96.
[72] Johannessen KA, Kvamsdal T, Dokken T. Isogeometric analysis using LR B-splines. Comput Methods Appl Mech Eng 2014;269:471–514.