

ISOGEOMETRIC STRUCTURAL SHAPE OPTIMIZATION USING PARTICLE SWARM ALGORITHM

S. Shojaee^{*,†}, N. Valizadeh and M. Arjomand

Department of Civil Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

ABSTRACT

One primary problem in shape optimization of structures is making a robust link between design model (geometric description) and analysis model. This paper investigates the potential of Isogeometric Analysis (IGA) for solving this problem. The generic framework of shape optimization of structures is presented based on Isogeometric analysis. By discretization of domain via NURBS functions, the analysis model will precisely demonstrate the geometry of structure. In this study Particle Swarm Optimization (PSO) is used for Isogeometric shape optimization. The option of selecting the position and weight of control points as design variables, needless to sensitivity analysis relationships, is the superiority of the proposed method over gradient-based methods. The other advantages of this method are its straightforward implementation and acceptable accuracy. The numerical examples verify the authenticity of this method.

Received: 8 September 2011; Accepted: 27 December 2011

KEY WORDS: Isogeometric Analysis; NURBS; exact geometry; shape optimization; particle swarm algorithm

1. INTRODUCTION

The goal of shape optimization of structures is finding the best geometric shape for structures boundary, so that the objective function is minimized and constraints are satisfied. In these types of problems the objective function is generally conserved energy, weight, stress or natural frequency of structure. The constraints can be behavioral constraints like stress and displacement restrictions or geometric constraints such as limiting the volume of structure.

*Corresponding author: S. Shojaee, Department of Civil Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

† E-mail address: saeed.shojaee@uk.ac.ir

In the primitive researches of shape optimization of structures, the boundary nodes in finite elements discretization were considered as design variables, however, later this approach was faced with serious problems. The independent displacement of nodes in this method would result in zigzag, irregular and non-real shapes. High dependence of optimal solution on meshing form, excessive number of design variables and difficulties of preservation of a suitable mesh during optimization process, are some other drawbacks of selecting finite elements nodes as design variables

Imposing a series of geometric constraints was necessary for overcoming these drawbacks, but this treatment would result in enormous computational costs, and was not easily feasible. Therefore appropriate conforming between analysis model (finite elements model) and design model was avoided. Afterward, researchers focused on methods in which analysis model and design model were considered separately.

These methods benefit from novel geometry developments for description of geometry of structure in design model. Splines, NURBS basis functions and B-splines are used for modelling of boundaries in these methods. These functions, particularly NURBS, provide the effective control of geometry with continuous boundaries. In fact, NURBS basis functions are references of CAD (Computer Aided Design) systems and have been widely used for defining geometry modelling and design variables in shape optimization problems. In these methods by considering the control points as design variables, the boundaries of domain are changed in each stage of optimization, and then the finite element model needs to be updated again. Specific Algorithms are used for finite element meshing (e. g. adaptive meshing or remeshing strategies). Despite their efficiency, these strategies are costly and time consuming. Moreover, linking between analysis model and design model remains one of the difficulties in these methods.

The IGA that introduced by Hughes et al. [1], has been recently applied in researches which deal with shape optimization of structures [2-5]. In IGA the NURBS basis functions are employed for modelling geometry of structure and with respect to isoparametric concept, these functions are also used in approximation of structural response. By using the presented Isogeometric method in shape optimization, the analysis model will be naturally merged by design model, which eliminates the desired link between the design and analysis models in optimization process. In all researches related with Isogeometric optimization of structures [2-5], gradient methods are used in optimization process. Gradient-based methods require data from sensitivity analysis, which is not easily applicable in various problems.

On the other hand, there are gradient-free methods for optimization which do not need sensitivity analysis, and their implementation is achievable in different problems. This study has adopted the particle swarm algorithm, which is one the most recent gradient-free optimization algorithms. The advantage of this method is that the position and weight of control points can be considered as design variables, needless to sensitivity analysis. The examples that are discussed here are limited to two-dimensional elastic problems, but this method can be easily developed for three-dimensional or other physical problems.

This paper is organized as follows. In Section 2 the B-spline and NURBS basis functions are introduced. The typical Isogeometric analysis framework is briefly presented in Section 3. In Section 4 the optimization problem of a structural system is generally discussed. The fundamentals of particle swarm algorithm are reviewed in Section 5. The efficiency of the

proposed method is confirmed through numerical examples in Section 6. Finally, the conclusions of this method are summarized in Section 7.

2. INTRODUCTION OF NURBS AND NURBS SURFACE

Non-Uniform Rational B-Splines (NURBS) are a generalization of piecewise polynomial curves, which are derived from B-splines basis functions. These functions are defined on knot vectors in a parametric space. A knot vector is a non-descending sequence of real numbers which is defined as,

$$\Xi = \{x_1, x_2, \dots, x_{n+p+1}\} \quad (1)$$

Where x_i is the i th knot vector, i is the knot number ($i=1,2,\dots,n+p+1$), p is the order of B-spline and n is the number of basis functions. The $[x_i, x_{i+1})$ interval is called the i th "knot span" which might have a zero length, as the knots can be repeated more than once. The $[x_i, x_{i+p+1}]$ interval is called a "patch". In Isogeometric analysis the open knot vectors are used. A knot vector is said to be open if its first and last knots appear $p+1$ times. For a specific knot vector, the basis B-spline functions are defined recursively by,

$$N_{i,0}(x) = \begin{cases} 1 & \text{if } x_i \leq x \leq x_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$N_{i,p}(x) = \frac{x - x_i}{x_{i+p} - x_i} N_{i,p-1}(x) + \frac{x_{i+p+1} - x}{x_{i+p+1} - x_{i+1}} N_{i+1,p-1}(x), \quad p = 1, 2, 3, \dots \quad (3)$$

Where $i=1,2,\dots,n+p+1$. B-spline basis functions have considerable features, for instance:

1. The summation of basis functions values for each $0 \leq x \leq 1$ is equal to one,

$$\sum_{i=1}^n N_{i,p}(x) = 1 \quad (4)$$

2. Being non-negative,

$$\forall x, \sum_{i=1}^n N_{i,p}(x) \geq 0 \quad (5)$$

3. The basis functions of order p have $p-m_i$ continuous derivatives at knot x_i , where m_i is the number that x_i is repeated in knot vector.
4. Local support, which means $N_{i,p}(x)$ basis function has the non-zero value only in $[x_i, x_{i+p+1})$ interval and is equal to zero at other points.

A p -order B-spline curve is given by,

$$C(x) = \sum_{i=1}^n N_{i,p}(x)P_i. \quad (6)$$

Where $N_{i,p}(x)$ is the i th B-spline of order p and P is the vector of control points. The B-splines composed of open knot vectors, merely interpolate the start and end points. The second-order B-spline basis functions are illustrated for $\Xi = \{0, 0, 0, 0.2, 0.4, 0.4, 0.6, 0.8, 1, 1, 1\}$ open knot vector in Figure 1.

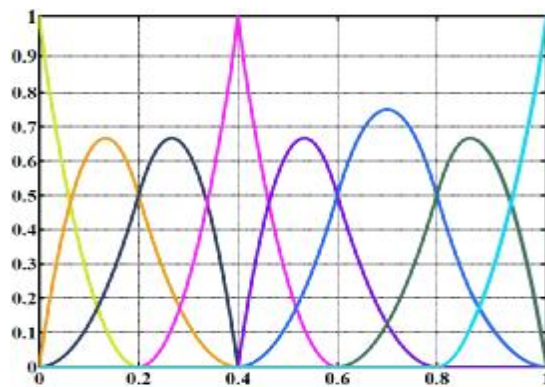


Figure 1. B-spline functions of order two

Now we can define the NURBS curve of order p as follows,

$$C(x) = \sum_{i=1}^n R_{i,p}(x)P_i \quad (7)$$

$$R_{i,p}(x) = \frac{N_{i,p}(x)w_i}{\sum_{i=1}^n N_{i,p}(x)w_i} \quad (8)$$

Where $R_{i,p}(x)$ is the NURBS basis function of order p , P_i is the i th control point and w_i is the weight of i th control point. In the two-dimensional parametric space of $[0,1]^2$, the NURBS surfaces are obtained by tensor product of $\Xi = \{x_1, x_2, \dots, x_{n+p+1}\}$ and $\Psi = \{y_1, y_2, \dots, y_{m+q+1}\}$ knot vectors,

$$S(x,y) = \sum_{i=1}^n \sum_{j=1}^m \frac{N_{i,p}(x)M_{j,q}(y)w_{i,j}}{W_{i,j}(x,y)} P_{i,j} \quad (9)$$

$$W_{i,j}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n \sum_{j=1}^m N_{i,p}(\mathbf{x}) M_{j,q}(\mathbf{y}) w_{i,j} \quad (10)$$

Where $P_{i,j}$ are $n \times m$ control points which form a control mesh. $N_{i,p}(\mathbf{x})$ and $M_{j,q}(\mathbf{y})$ are B-spline basis functions of p and q orders respectively. Therefore the two-dimensional NURBS basis functions are defined as,

$$R_{i,j}^{p,q}(\mathbf{x}, \mathbf{y}) = \frac{N_{i,p}(\mathbf{x}) M_{j,q}(\mathbf{y}) w_{i,j}}{W_{i,j}(\mathbf{x}, \mathbf{y})} \quad (11)$$

The $[y_1, y_{n+p+1}] \times [x_1, x_{m+q+1}]$ interval is called a patch and $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$ is called the knot span.

Without loss of generality, here we consider a NURBS surface in a knot span which is defined by an arrangement of $n_{en} = (p+1) \times (q+1)$ control points.

It should be noted that the NURBS basis functions have the local support (control) property, which means in defined knot span and there are only $(p+1) \times (q+1)$ non-zero basis functions. Thus, the number of control points corresponding to each knot span is $n_{en} = (p+1) \times (q+1)$. In Isogeometric method the domain of problem is divided into a number of patches and each patch is divided into some knot spans (elements). Patches are similar to sub-domains in classic finite elements method and in each patch the element type and material properties are uniform [6]. However many of complex domains can be modeled by just a single patch.

3. FORMULATION OF ISOGOMETRIC ANALYSIS BASED ON NURBS BASIS FUNCTIONS

Here the formulation of Isogeometric analysis is presented for tow-dimensional linear elastic problems. The strong form of equations for linear elastic problems is given by,

$$\left\{ \begin{array}{l} \nabla \cdot \mathbf{s}_x + b_x = 0 \quad \text{and} \quad \nabla \cdot \mathbf{s}_y + b_y = 0 \quad \text{on } \Omega \\ \mathbf{s} = \mathbf{D} \nabla_s \mathbf{u} \\ \mathbf{s}_x \cdot \mathbf{n} = t_x \quad \text{and} \quad \mathbf{s}_y \cdot \mathbf{n} = t_y \quad \text{on } \Gamma_t \\ u = \bar{u} \quad \text{on } \Gamma_u \end{array} \right. \quad (12)$$

Where Ω is the domain of problem, Γ_t is the part of boundary which the surface loads are applied and Γ_u is the part of boundary where the displacements are known. After discretization of domain, the basic equilibrium equation is defined as,

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (13)$$

Where \mathbf{K} is the global stiffness matrix, \mathbf{u} and \mathbf{f} are displacement vector and external force vector, respectively. The stiffness matrix, \mathbf{K} , is assembled using element stiffness matrices (\mathbf{K}_e) and the force vector, \mathbf{f} , is obtained by insertion of elements force vectors (\mathbf{f}_e). The stiffness matrix of each element is calculated by following Equation,

$$\mathbf{K}_e = t_e \int_{\hat{\Omega}} \mathbf{B}^T \mathbf{D} \mathbf{B} |\mathbf{J}| d\hat{\Omega} \quad (14)$$

Where t_e is the thickness of element and $\hat{\Omega}$ is the parametric space. \mathbf{B} and \mathbf{D} are strain-displacement matrix and stress-strain matrix respectively. $|\mathbf{J}|$ is the determinant of Jacobian matrix. The integral of Equation (14) can be computed through Gauss-Legendre method by estimation of integration value at Gaussian points in each element. The strain-displacement matrix (\mathbf{B}) is calculated as follows,

$$\mathbf{B} = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & \dots & \frac{\partial N_{n_{en}}}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & \dots & 0 & \frac{\partial N_{n_{en}}}{\partial y} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \dots & \frac{\partial N_{n_{en}}}{\partial y} & \frac{\partial N_{n_{en}}}{\partial x} \end{bmatrix} \quad (15)$$

Where N_i is the NURBS basis function. For plane stress problems the stress-strain matrix (\mathbf{D}) is given by,

$$\mathbf{D} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (16)$$

Where E is Young modulus and ν is the Poisson ratio. The Jacobian matrix, which maps the points from parametric space into physical space is defined as,

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \quad (17)$$

The force vector of an element is given by,

$$\mathbf{f}_e = \int_{\hat{\Omega}_e} \mathbf{N}^T \mathbf{b} |\mathbf{J}| t_e d\hat{\Omega} + \int_{\hat{\Gamma}_e} \mathbf{N}^T \mathbf{t} |\mathbf{J}_b| d\hat{\Gamma}_t \quad (18)$$

Where \mathbf{b} is the body force, \mathbf{t} is the surface force on boundary, \mathbf{J}_b and $\hat{\Gamma}_t$ are, respectively, the

Jacobian and the parametric space of the part of boundary which is subjected to surface forces.

4. SHAPE OPTIMIZATION OF STRUCTURES

The mathematical formulation of an optimization problem is generally given by,

$$\left\{ \begin{array}{ll} \min_{\mathbf{a}_s} & f(\mathbf{u}(\mathbf{a}), \mathbf{a}) \\ \text{s.t.} & h_i(\mathbf{u}(\mathbf{a}), \mathbf{a}) = 0, \quad i = 1, \dots, n_h \\ & g_j(\mathbf{u}(\mathbf{a}), \mathbf{a}) \leq 0, \quad j = 1, \dots, n_g \\ & \mathbf{a}_{\min} \leq \mathbf{a}_s \leq \mathbf{a}_{\max}, \quad s = 1, \dots, n_{eq} \end{array} \right. \quad (19)$$

Where f is the objective function, \mathbf{u} is the state variable that can be displacement, \mathbf{a} is the design variable, n_h is the number of equal constraints, n_g is the number of unequal constraints and n_{eq} is the number of design variables.

The optimization problem, equation (19), can be solved through several mathematical programming algorithms [7]. In general, the methods for solving a nonlinear optimization problem are classified as gradient-based methods and gradient-free methods. In gradient-free methods for finding optimal solution only the link between the analysis model, objective function and constraints is needed. But in gradient methods besides this link the sensitivity analysis data is also required. The sensitivity analysis can be carried out by numerical, semi-analytical or analytical methods. The numerical gradient methods, which employ finite difference approaches, are easily applicable but they result in considerable computational costs. In contrast, semi-analytical and analytical gradient methods are more complex for implementation but they are much efficient in reducing computational costs. The superiority of gradient-free methods remains in their ability to find the optimal solution needles to sensitivity analysis. However the gradient methods may find the local optimal solution but the sensitivity analysis can be complicated in different problems.

The principle concentration of this paper is using a free-gradient method; in considering this the particle swarm algorithm is applied. This algorithm is one of the most recent random search methods and is presented for optimization in both of continuous and discrete states. In this study with respect to continuous entity of considered variables in examples, the continuous particle swarm algorithm is adopted.

5. PARTICLE SWARM ALGORITHM

The Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [8], is an optimization technique based on probabilistic rules. PSO was first intended for simulating social behavior as a stylized representation of the movement of organisms, for example in a bird flock. A swarm of particles is considered, each particle represents a bird in search-space. The algorithm promotes the swarm to optimal solution by updating the position of particles

based on their fitness. The algorithm initiates with a candidate group of random solutions, then by updating the position and velocity of particles, the algorithm searches for optimal solution in the space of problem. Each particle is characterized by X and V values which denote the position and velocity respectively. The position of the particles is the desired answer of our problem and their velocity implies the rate of their position variations. The larger velocity values suggest that the current position is not favorable and it has a noticeable distance to optimal position. In each movement of the swarm, position and velocity of each particle is updated based on local and global values.

The best local value, denoted as P_{best} , is the solution that has the most fitness, and is obtained individually for each particle. The best global position is the best value that is achieved among the whole particles and is denoted as G_{best} . The new velocity and position of the i th particle in the k th iteration are updated as follows,

$$V_i^{k+1} = w^k V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k) \quad (20)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (21)$$

Where V_i^k is the velocity vector in the k th iteration, r_1 and r_2 are two random numbers between one and ten, P_i^k stands for best position of the i th particle and P_g^k is the position of the best particle up to the k th iteration. c_1 and c_2 are personal and social learning factors, that are also called acceleration coefficients. c_1 and c_2 take the values between 1.5 and 2, but the best value for these two parameters is 2 [9-10, 12]. w is the weight inertia parameter. For large values of w the velocity increases and the steps become larger, and as the w decreases the steps become smaller. This would be helpful for convergence to optimal solution in last steps. Therefore the constant value of w is replaced by following relationship,

$$w_{k+1} = w_{max} - \frac{w_{max} - w_{min}}{k_{max}} k \quad (22)$$

Where k_{max} is the maximum number of iterations, w_{max} and w_{min} are equal to 0.9 and 0.4 respectively [11].

6. NUMERICAL EXAMPLES

6.1. A plate with a circular hole

The first example is one of the classic problems in optimization. The shape optimization is considered for a plate with a circular hole under the x-direction traction. The objective of problem is minimizing the conserved energy of the plate, and the constraint is limiting the net volume to 96% of the initial material volume (without considering the hole). For a large infinite plate the analytical solution leads to a circle subjected to symmetric loads or an ellipse subjected to asymmetric loads.

A quarter of plate is modeled with respect to the symmetry of plate and loading. The initial layout of model including dimensions and loads is illustrated in Figure 2. This problem is modeled by second-order NURBS surfaces in both directions with 4×3 control points. At the upper left corner of model, two control points are located at the same position. The knot vectors in ξ and ψ directions are $\{0,0,0,0.5,1,1,1\}$ and $\{0,0,0,1,1,1\}$ respectively, which result in 2×1 knot spans (elements). The plate has a thickness of 1 and the Young modulus and Poisson ratio are 1×10^5 and 0.3 respectively.

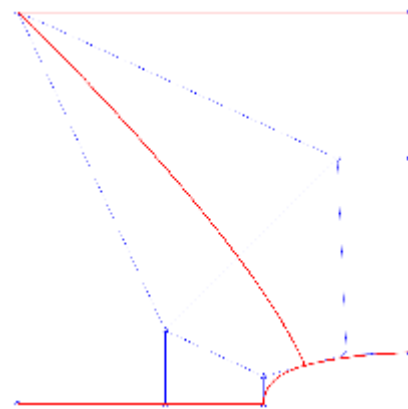
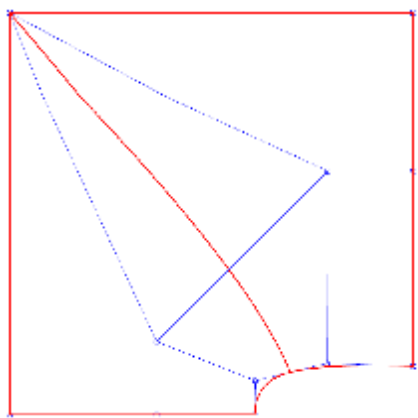
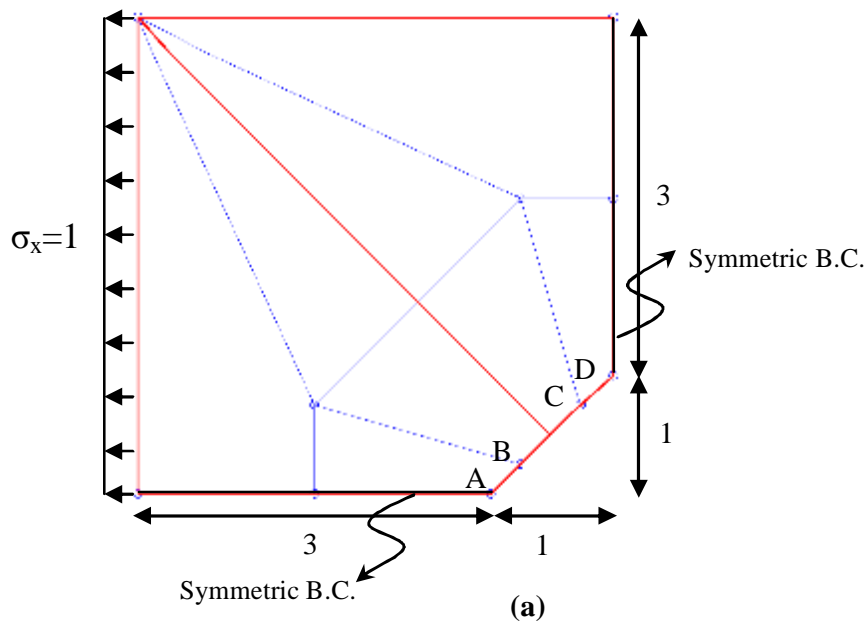


Figure 2. A quarter plate with a quarter circular hole, (a) initial scheme of model, (b) optimal design without considering the weight of control points (c) optimal design without considering the weight of control points. The control mesh is shown with blue color and physical mesh (elements) is shown with red color

Two scenarios are considered for solving the problem: 1- the positions of control points are

assumed as the only design variables; 2- in addition to the positions of control points of first scenario, the weights of some control points are also assumed as design variables. In first scenario, the design variables are the x coordinate of control points A, B and C and the y coordinate of control points B, C and D. In the second scenario the weight of points C and D are added to the design variables of first scenario. As shown in Figure 3, the optimal shape is attained after 30 iterations. As it was anticipated, the optimal shape is an ellipse, which is consistent with analytical solution. In Figure 3 the history of objective function is shown for both scenarios, and the values of both objective functions are tabulated in Table 1. The results imply that by considering the weight as design variable, the value of objective function is decreased, and consequently the solution becomes more optimized in comparison with the first scenario.

Table 1. the values of objective functions for two scenarios of plate problem

| The plate with circular hole | 1 st scenario | 2 nd scenario |
|------------------------------|--------------------------|--------------------------|
| Objective function | 1.7124×10^{-4} | 1.7119×10^{-4} |
| Volume | 15.358 | 15.3194 |
| Number of design variables | 6 | 8, $w_B=1$, $w_C=0.947$ |
| Number of iterations | 30 | 30 |

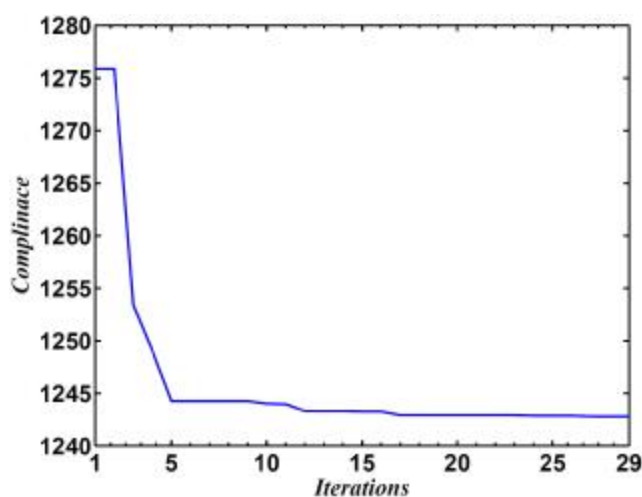


Figure 3. History of objective function

6.2. Spanner problem

Finally one of the standard optimization problems [2-5] is considered. The target of this problem is designing the outer shape of a spanner. The objective function is to

minimize $\frac{1}{2}(u_{F_A} - u_{F_B})$, the displacements in two loading cases, with u_{F_A} and u_{F_B} as vertical displacements at the point of loads F_A and F_B respectively. The volume of optimal design is limited to 30% of the given material property.

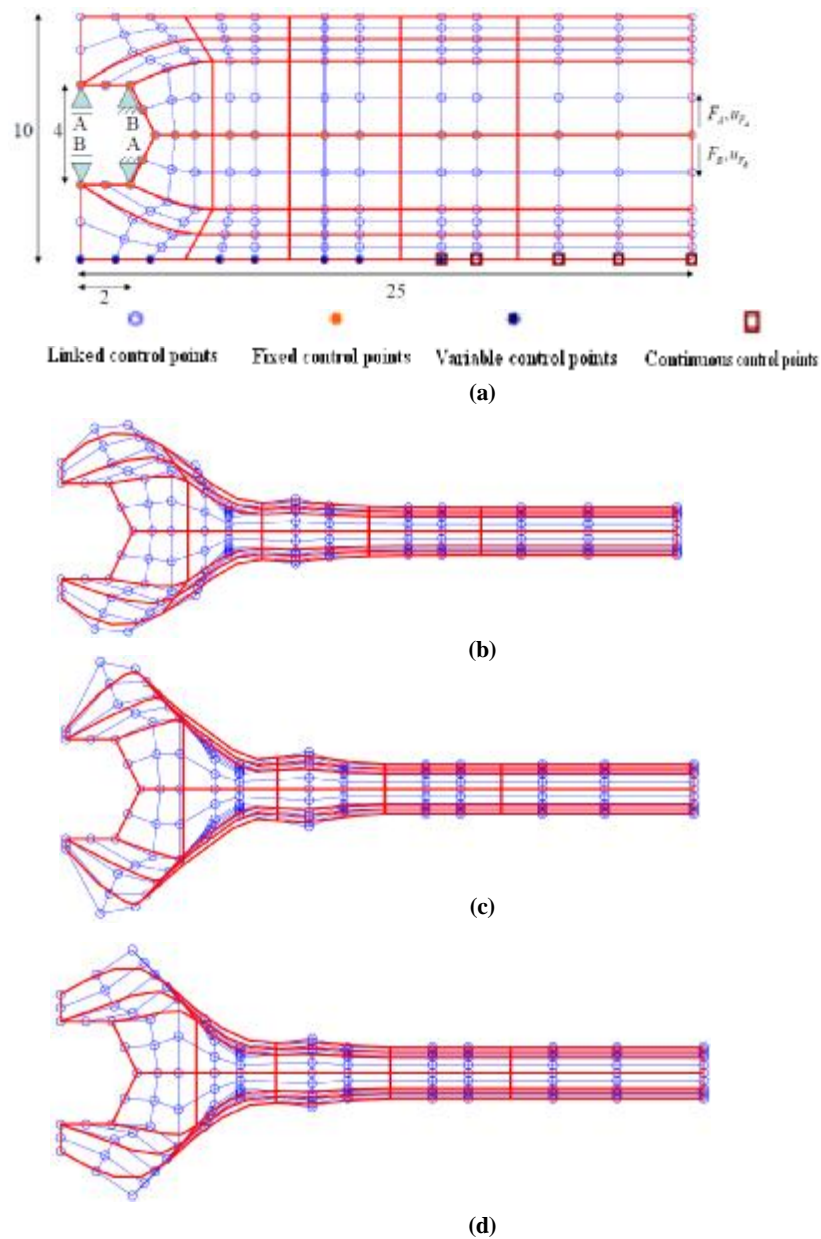


Figure 4. Spanner problem, (a) Initial scheme of spanner, (b) optimal design after 15 iterations (c) optimal design with considering with considering the weights of control points, after 20 iterations, (d) the optimal design without considering the weights after 20 iterations. The control mesh is shown with blue color and physical mesh (elements) is shown with red color

The initial scheme, dimensions and restraints for loading cases A and B are shown in Figure 4. The load value is equal to 10 in both cases. For discretization of the problem domain, a NURBS patch of order 3 is used in horizontal direction and another patch of degree 2 is used in vertical direction. The control mesh consists of 12×13 control points. The symmetry of problem is considered in numerical modeling and two strategies are used. In first strategy the weights of all points are set to 1 and the vertical positions of control points at the bottom side of model are assumed as design variables. To ensure that the final shape would be applicable, the positions of 5 control points on the upper and lower right side are assumed to be equal, and the end width of the tool is limited to 2. Therefore 8 design variables exist in first strategy. The optimal solution is obtained after 15 iterations. As shown in Figure 4 the shape of boundaries are smooth and the optimal shape illustrates a standard spanner. This shape conforms to results of previous studies [3-5]. In the second strategy in addition to the design variables of first strategy, the weight of control points, at the lower left corner, are taken as design variables. The results of two strategies are summarized in Table 2. In this strategy by considering the weight, the value of objective function is reduced and the volume constraint is satisfied easily.

Table 2. Results of spanner problem

| The Spanner problem | 1 st strategy | 1 st strategy | 2 nd strategy |
|----------------------------|--------------------------|--------------------------|--|
| Objective function | 0.3684 | 0.3684 | 0.2697 |
| Volume | 72.6821 | 72.6821 | 72.1484 |
| Number of design variables | 8 | 8 | 13, $w_1=0.4177$, $w_2=0.2153$ $w_3=1$, $w_4=0.1$, $w_5=0.837$ |
| Number of iterations | 15 | 15 | 20 |

7. CONCLUSIONS

This paper investigates a hybrid isogeometric shape optimization of structures through particle swarm algorithm. The results show that using isogeometric analysis, the geometry is represented precisely and the shape of boundaries after optimization are smooth. Eliminating the finite elements meshing stage and solution stability during optimization process, are the other advantages of this method. In this study the position of control points were taken as design variables, but the weight of control points can also be considered as design variables. However, in the numerical examples the rate of convergence was favorable, but it seems that for more complex examples using metamodels would be beneficial for increasing the convergence rate and decreasing the computational costs.

REFERENCES

1. 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.
2. Cho S, Ha SH. Isogeometric shape design optimization: exact geometry and enhanced sensitivity, *Struct Multidiscip Optim* 2009; **38**:53–70.
 3. Wall WA, Frenzel MA, Cyron C. Isogeometric structural shape optimization, *Comput Methods Appl Mech Eng* 2008; **197**:2976–2988.
 4. Nagy AP, Abdalla MM, Gurdal Z. Isogeometric sizing and shape optimization of beam structures, *Comput Methods Appl Mech Eng* 2010; **199**:1216–30.
 5. Qian X. Full analytical sensitivities in NURBS based Isogeometric shape optimization, *Comput Methods Appl Mech Eng* 2010; **199**:2059-71.
 6. Cottrell JA, Hughes TJR, Bazilevs Y. Isogeometric Analysis: Toward Integration of CAD and FEA, John Wiley & Sons, 2009.
 7. Fletcher R. *Practical Methods of Optimization*, John Wiley & Sons, 2003.
 8. Kennedy J, Eberhart R. Particle swarm optimization, *Proceeding of IEEE International Conference on Neural Networks*, 1995; **4**:1942–48.
 9. Perez RE, Behdinan K. Particle swarm approach for structural design optimization, *Comput Struct* 2007; **85**:1579-88.
 10. Shi Y, Eberhart R. A modified particle swarm optimizer, *IEEE International Conference on Evolutionary Computation*, IEEE Press, Piscataway, NJ, 1998, 69–73.
 11. Li LJ, Huang ZB, Liu F, Wu QH. A heuristic particle swarm optimizer for optimization of pin connected structures, *Comput Struct* 2007; **85**:340-49.
 12. Fourie P, Groenwold A. The particle swarm optimization algorithm in size and shape optimization, *Struct Multidiscip Optim* 2002; **23**:259-67.