10th World Congress on Structural and Multidisciplinary Optimization
May 19 -24, 2013, Orlando, Florida, USA

2D Isogeometric Shape Optimization considering both control point positions and weights as design variables

Yeo-Ul Song[1], JunYoung Hur[2] and Sung-Kie Youn[3*]

[1] Dept. of ME, KAIST, Guseong-dong, Yuseong-gu, Daejeon, Korea, find3969@kaist.ac.kr
[2] Dept. of ME, KAIST, Guseong-dong, Yuseong-gu, Daejeon, Korea, junyoung@kaist.ac.kr
[3] Dept. Of ME, KAIST, Guseong-dong, Yuseong-gu, Daejeon, Korea, skyoun@kaist.ac.kr

1. Abstract

NURBS (Non-uniform Rational B-spline) has been widely used as a standard shape representation technique for shape optimization due to its accuracy and efficiency in the integration of CAD and CAE. Although NURBS represents geometry by positions of control points and their weights, until now most of shape optimization studies use only control point positions as design variables. In some shape optimization processes control points come closer to each other. This deteriorates the mesh quality and hampers the convergence. Control point weights enable conical shape representation such as circle and ellipse and more flexible curve representation. If weights are considered as additional design variables, refined shape control could be expected.

In this work, a new 2D isogeometric shape optimization based on spline finite element method is proposed. Both positions and weights of NURBS control points are used as design variables and shape optimization algorithm is composed of position optimization step and weight optimization step. In order to prevent the disadvantage for location of control points in the limited space cases, a new shape optimization algorithm starts with control point positions as design variables. If the closest distance of two neighboring control points is less than the threshold value during the position optimization step, weights become the design variables. The proposed NURBS based shape optimization is applied to some benchmarking problems. It is shown that a new shape optimization algorithm has advantages in conical shape representation and in the treatment for location of control points in the limited space cases.

2. Keywords: Shape optimization, Isogeometric analysis, NURBS, control point, weight

3. Introduction

In the product design and development, Computer Aided Design (CAD) and Computer Aided Engineering (CAE) play increasingly important roles. The importance of CAE has been increased due to the high standard of consumers and to increase the effect and success rate in new products.

The communicative processes between CAD and CAE are frequently required. Whilst CAD uses spline basis for shape optimization; CAE, uses for numerical analysis and design optimization, employs finite elements and polynomial basis which results in a gap between mathematical languages in CAD and CAE. Therefore, many ongoing researches focus to overcome this issue.

4. NURBS (Non-Uniform Rational B-spline)

4.1. B-spline

A B-spline curve is defined as Eq.(1). The equation includes linear combination of control points and B-spine basis functions of degree p.

$$\mathbf{C}(s) = \sum_{i=1}^{n} N_{i,p}(s)\mathbf{P}_i \tag{1}$$

B-spline basis functions are generated with knot vector (Eq.(2)), which is non-decreasing sequence of parameter. And multiplicity (p+1) of the first knot and last knot makes both ends clamped.

$$\mathbf{s} = \left[s_1, s_2, \cdots, s_{n+p+1} \right] \tag{2}$$

B-spline basis functions are recursively obtained based on knot vector.

$$
\begin{aligned}
N_{i,0}(s) &= \begin{cases} 1, & s_i \le s < s_{i+1} \\ 0, & \text{otherwise} \end{cases} \\
N_{i,p}(s) &= \frac{s - s_i}{s_{i+p} - s_i} N_{i,p-1}(s) + \frac{s_{i+p+1} - s}{s_{i+p+1} - s_{i+1}} N_{i+1,p-1}(s)
\end{aligned} \tag{3}
$$

These basis functions have some characteristics. These are non-negative function and satisfy the partition of unity. Desired continuity of B-spline curve is obtained because the basis is $\mathbf{C}^{p-k}$ continuous at a knot of multiplicity k. Also B-spline curve can be locally modified due to the compact support characteristics, as follows

$$N_{i,p}(s) = 0\,,\ \ s \notin [s_i, s_{i+p+1}] \tag{4}$$

On the two dimensional case, B-spline surface is defined as Eq.(5) which is based on two knot vectors.

$$\mathbf{S}(s,t) = \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(s) M_{j,q}(t) \mathbf{P}_{ij} \tag{5}$$

4.2. NURBS curve and surface

NURBS (Non-uniform Rational B-spline) is the generalized form of B-spline. A NURBS curve $\mathbf{C}(s)$ and surface $\mathrm{S}(s,t)$ of degree p is defined as follows

$$\mathbf{C}(s) = \sum_{i=1}^{n} \frac{w_i N_{i,p}(s)}{\sum_{j=1}^{n} w_j N_{j,p}(s)} \mathbf{P_i} = \sum_{i=1}^{n} R_{i,p}(s) \mathbf{P_i} \tag{6}$$

$$\mathbf{S}(s,t) = \sum_{i=1}^{n}\sum_{j=1}^{m} \frac{N_{i,p}(s) M_{j,q}(t) w_{i,j}}{\sum_{k=1}^{n}\sum_{l=1}^{m} N_{k,p}(s) M_{l,q}(t) w_{k,l}} \mathbf{P_{i,j}} = \sum_{i=1}^{n}\sum_{j=1}^{m} R_{i,j}(s,t) \mathbf{P_{i,j}} \tag{7}$$

Unlike B-spline, NURBS has rational form and additional variable weight. Each control point has corresponding weight value. As a result, NURBS retains all properties of B-spline with some extra properties. Because of the rational form and weight, NURBS can exactly represent the conic curves like circles, ellipse etc. as follows:

$$C(s) = \frac{(1-s^2) w_0 P_0 + 2s(1-s) w_1 P_1 + s^2 w_2 P_2}{(1-s^2) w_0 + 2s(1-s) w_1 + s^2 w_2} \tag{8}$$

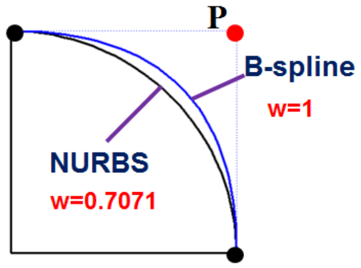

Figure 1 : Conic curve representation

In Fig.1, NURBS curve with 0.7071 weight value is exactly same with a quarter circle, but B-spline curve cannot represent the conic curve. NURBS represents geometry by changing the positions of the control points and their weights. The NURBS curve is included in the convex hull of corresponding control points. When control point position is changed, new control net is obtained. The new control net enables rough geometric modification. On the other hand, when weight is changed, convex hull remains intact and the corresponding local NURBS curve changes in the space within the convex hull. For these characteristics, weight can be considered as design variable in the shape optimization to modify the NURBS curve more exquisitely and flexibly.

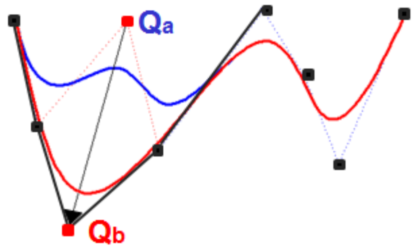

Figure 2 : Control point changed from Q_a to Q_b

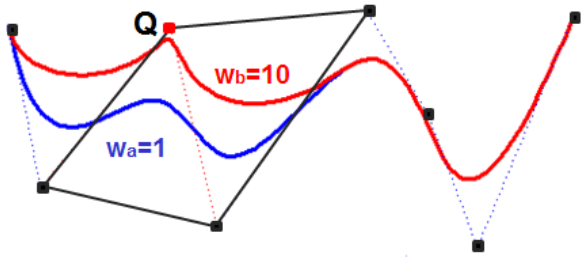

Figure 3 : Convex hull property, Weight of the control point Q changed from w_a to w_b

5. Spline finite element method

Spline finite element method was developed to increase accuracy and efficiency in integration of structure modeling of CAD and numerical analysis of CAE. In 2005 [4], a method of using spline information of CAD to apply directly to analysis was proposed and named "Isogeometric analysis". Existing finite element method used geometric information of CAD as a basis to construct an additional finite element model through a shape approximation process, which proved to be costly and cumbersome. However spline finite element method has an advantage in using geometric information from CAD modeling directly for analysis. The process of spline finite element method is shown in Fig.4.

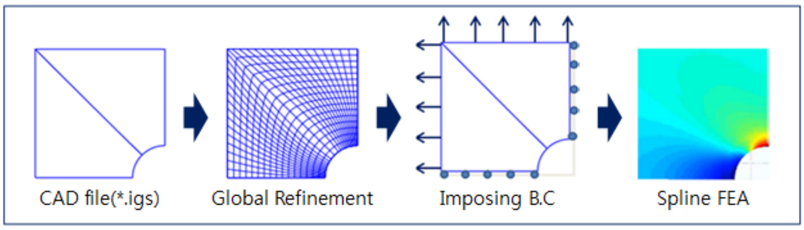

Figure 4 : Spline finite element method flow chart

First, modeling information from CAD system is received as IGS file. The input file includes spline information such as coordinates of control point positions, knot vectors and control point weights. In general, since coarse mesh information comes from CAD, global refinement is needed to achieve the desired accuracy. In element refinement, the spline finite element method uses knot insertion which is simple and straightforward thus considerable time and effort can be saved. Then, the boundary and loading conditions of the problem are enforced and the same spline basis functions used in CAD model are employed for spline finite element analysis.

6. 2D Shape optimization

6.1. Problem definition

The general mathematical formulation of 2D shape optimization problem is stated as follows

$$\begin{aligned}
&\text{Minimize} && \Psi\left(\mathbf{b},\mathbf{u}(\mathbf{b})\right) && \\
&\text{subject to} && h_j\left(\mathbf{b},\mathbf{u}(\mathbf{b})\right)=0 && j=1,\cdots,\text{Number of equality constraint}\\
& && g_k\left(\mathbf{b},\mathbf{u}(\mathbf{b})\right)\le 0 && k=1,\cdots,\text{Number of inequality constraint}\\
& && \underline{b}_i \le b_i \le \overline{b}_i && i=1,\cdots,\text{Number of design variables}
\end{aligned} \tag{9}$$

Ψ represents objective function and state variable $\mathbf{u}$ is the function of design variable $\mathbf{b}$. The constraints are represented as equality and inequality constraints h_j, g_k. Most of existing shape optimization uses only the location of the control point as the design variable, but this paper set the weight as an additional design variable to represent the optimal shape flexibly. In recent work in [5], both positions and weights are considered as design variables and both parameters are used during the every iteration of shape optimization simultaneously. Because control point positions and their weights depend on each other, this paper separately uses two parameters during the optimization steps.

6.2. Shape sensitivity formulation

Shape sensitivity analysis is necessary for shape optimization. Sensitivity on objective function is obtained from differentiation of objective function (Eq.(10)). In the sensitivity formulation, adjoint variable method is used and $\mathbf{v}$ represents solution of the adjoint equation (Eq.(11)).

$$\frac{d\Psi\left(\mathbf{b},\mathbf{u}(\mathbf{b})\right)}{d\mathbf{b}}=\frac{\partial\Psi}{\partial\mathbf{b}}+\frac{\partial\Psi}{\partial\mathbf{u}}\frac{\partial\mathbf{u}}{\partial\mathbf{b}}=\frac{\partial\Psi}{\partial\mathbf{b}}+\frac{\partial\Psi}{\partial\mathbf{u}}\mathbf{K}^{-1}\left(\frac{\partial\mathbf{F}}{\partial\mathbf{b}}-\frac{\partial\mathbf{K}}{\partial\mathbf{b}}\mathbf{u}\right)=\frac{\partial\Psi}{\partial\mathbf{b}}+\mathbf{v}^{\mathrm{T}}\left(\frac{\partial\mathbf{F}}{\partial\mathbf{b}}-\frac{\partial\mathbf{K}}{\partial\mathbf{b}}\mathbf{u}\right) \tag{10}$$

$$\mathbf{K}\mathbf{v}=\left(\frac{\partial\Psi}{\partial\mathbf{u}}\right)^{\mathrm{T}} \tag{11}$$

In order to compute shape sensitivities on compliance and stress with respect to a design variable, sensitivities on an element stiffness matrix (Eq.(12)) and strain-displacement matrix should be calculated.

$$\frac{\partial\mathbf{k_e}}{\partial\mathbf{b}}=\sum_{k=1}^{NINT}\left[\frac{\partial\mathbf{B}_k^{\mathbf{T}}}{\partial\mathbf{b}}\mathbf{D}\mathbf{B}_k\left|\mathbf{J}^k\right|+\mathbf{B}_k^{\mathbf{T}}\mathbf{D}\frac{\partial\mathbf{B}_k}{\partial\mathbf{b}}\left|\mathbf{J}^k\right|+\mathbf{B}_k^{\mathbf{T}}\mathbf{D}\mathbf{B}_k\frac{\partial\left|\mathbf{J}^k\right|}{\partial\mathbf{b}}\right]W^k \tag{12}$$

Also to calculate sensitivities on strain-displacement matrix and Jacobian matrix, the sensitivity on the element stiffness matrix needs to be calculated. Unlike the case that the design variables are control point positions, when the weights are design variables basis functions are also the functions of weight. The matrices Γ and G are also functions of weight. Eq. (13) represents B matrix, Eq. (14) represents sensitivity of B matrix with respect to control point position. And Eq. (15) shows sensitivity of B matrix with respect to weight.

$$\mathbf{B}=\mathbf{M}\boldsymbol{\Gamma}\mathbf{G}:\quad\left\{\mathbf{M}=\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&1&1&0\end{bmatrix},\quad\boldsymbol{\Gamma}=\begin{bmatrix}\mathbf{J}_{\mathbf{S}}^{-1}&0\\0&\mathbf{J}_{\mathbf{S}}^{-1}\end{bmatrix},\quad\mathbf{G}=\begin{bmatrix}R_{1,s}&0&\cdots&R_{n,s}&0\\R_{1,t}&0&\cdots&R_{n,t}&0\\0&R_{1,s}&\cdots&0&R_{n,s}\\0&R_{1,t}&\cdots&0&R_{n,t}\end{bmatrix}\right\} \tag{13}$$

$$\frac{\partial \mathbf{B}}{\partial \mathbf{x}} = \mathbf{M}\frac{\partial \mathbf{\Gamma}}{\partial \mathbf{x}}\mathbf{G} \; : \qquad \left\{ \frac{\partial \mathbf{\Gamma}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial \mathbf{x}} & 0 \\ 0 & \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial \mathbf{x}} \end{bmatrix}, \quad \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial \mathbf{x}} = -\mathbf{J}_{\mathbf{S}}^{-1}\frac{\partial \mathbf{J}_{\mathbf{S}}}{\partial \mathbf{x}}\mathbf{J}_{\mathbf{S}}^{-1} \right\} \tag{14}$$

$$\frac{\partial \mathbf{B}}{\partial w_A} = \mathbf{M}\frac{\partial \mathbf{\Gamma}}{\partial w_A}\mathbf{G} + \mathbf{M}\mathbf{\Gamma}\frac{\partial \mathbf{G}}{\partial w_A} \; :$$

$$\left\{ \frac{\partial \mathbf{\Gamma}}{\partial w_A} = \begin{bmatrix} \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial w_A} & 0 \\ 0 & \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial w_A} \end{bmatrix}, \quad \frac{\partial \mathbf{J}_{\mathbf{S}}^{-1}}{\partial w_A} = -\mathbf{J}_{\mathbf{S}}^{-1}\frac{\partial \mathbf{J}_{\mathbf{S}}}{\partial w_A}\mathbf{J}_{\mathbf{S}}^{-1}, \quad \frac{\partial \mathbf{G}}{\partial w_A} = \begin{bmatrix} R_{1,sw} & 0 & \cdots & R_{n,sw} & 0 \\ R_{1,sw} & 0 & \cdots & R_{n,sw} & 0 \\ 0 & R_{1,tw} & \cdots & 0 & R_{n,tw} \\ 0 & R_{1,tw} & \cdots & 0 & R_{n,tw} \end{bmatrix} \right\} \tag{15}$$

Following three formulations Eq. (16) represents represent Jacobian matrix, Eq. (17) represents sensitivity of Jacobian matrix with respect to control point position and Eq. (18) shows sensitivity of J matrix with respect to weight.

$$\mathbf{J} = \mathbf{J}_{\mathbf{P}_{\mathbf{R}}}\mathbf{J}_{\mathbf{S}} = \begin{bmatrix} \frac{1}{2}(s_{\max} - s_{\min}) & 0 \\ 0 & \frac{1}{2}(t_{\max} - t_{\min}) \end{bmatrix} \begin{bmatrix} \sum_i^n R_{i,s}x_i & \sum_i^n R_{i,s}y_i \\ \sum_i^n R_{i,t}x_i & \sum_i^n R_{i,t}y_i \end{bmatrix} \tag{16}$$

$$\begin{aligned} \frac{\partial|\mathbf{J}|}{\partial x_j} &= |\mathbf{J}_{\mathbf{P}_{\mathbf{R}}}|\frac{\partial|\mathbf{J}_{\mathbf{S}}|}{\partial x_j} : \left\{ \frac{\partial|\mathbf{J}_{\mathbf{S}}|}{\partial x_j} = R_{j,s}(\sum R_{i,t}y_i) - R_{j,t}(\sum R_{i,s}y_i) \right\} \\ \frac{\partial|\mathbf{J}|}{\partial y_j} &= |\mathbf{J}_{\mathbf{P}_{\mathbf{R}}}|\frac{\partial|\mathbf{J}_{\mathbf{S}}|}{\partial y_j} : \left\{ \frac{\partial|\mathbf{J}_{\mathbf{S}}|}{\partial y_j} = R_{j,t}(\sum R_{i,s}x_i) - R_{j,s}(\sum R_{i,t}x_i) \right\} \end{aligned} \tag{17}$$

$$\begin{aligned} &\frac{\partial|\mathbf{J}|}{\partial w_A} = |\mathbf{J}_{\mathbf{P}_{\mathbf{R}}}|\frac{\partial|\mathbf{J}_{\mathbf{S}}|}{\partial w_A} : \\ &\left\{ \frac{\partial|J_S|}{\partial w_A} = \sum\frac{\partial R}{\partial w_A\partial s}x_i\sum\frac{\partial R}{\partial t}y_i + \sum\frac{\partial R}{\partial s}x_i\sum\frac{\partial R}{\partial w_A\partial t}y_i - \sum\frac{\partial R}{\partial w_A\partial s}y_i\sum\frac{\partial R}{\partial t}x_i - \sum\frac{\partial R}{\partial s}y_i\sum\frac{\partial R}{\partial w_A\partial t}x_i \right\} \end{aligned} \tag{18}$$

Shape sensitivity on compliance (Eq. (19)) and stress (Eq. (20)) with respect to a design variable can be calculated with above sensitivities of matrix.

$$\frac{d\Psi(\mathbf{b},\mathbf{u}(\mathbf{b}))}{d\mathbf{b}} = \mathbf{F}^{\mathrm{T}}\frac{\partial \mathbf{u}}{\partial \mathbf{b}} = -\mathbf{F}^{\mathrm{T}}\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \mathbf{b}}\mathbf{u} = -\mathbf{u}^{\mathrm{T}}\frac{\partial \mathbf{K}}{\partial \mathbf{b}}\mathbf{u} = \sum_{e=1}^{NECS} -\mathbf{u}_e^{\mathrm{T}}\frac{\partial \mathbf{k}_{\mathbf{e}}}{\partial \mathbf{x}}\mathbf{u}_e \tag{19}$$

$$\boldsymbol{\sigma} = [\sigma_{xx}\ \sigma_{yy}\ \sigma_{xy}]^T = \mathbf{DBd} = \mathbf{DBAu}$$

$$\frac{d\sigma_{\max}}{d\mathrm{b}} = \frac{\partial \sigma_{\max}}{\partial \mathrm{b}} + \frac{\partial \sigma_{\max}}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathrm{b}} = \mathbf{D}\frac{\partial \mathbf{B}}{\partial \mathbf{b}}\mathbf{Au} + \mathbf{DBA}\frac{\partial \mathbf{u}}{\partial \mathbf{b}} = \mathbf{D}\frac{\partial \mathbf{B}}{\partial \mathbf{b}}\mathbf{Au} - \mathbf{DBAK}^{-1}\frac{\partial \mathbf{K}}{\partial \mathbf{b}}\mathbf{u} = \mathbf{D}\frac{\partial \mathbf{B}}{\partial \mathbf{b}}\mathbf{Au} - \mathbf{v}^{\mathrm{T}}\frac{\partial \mathbf{K}}{\partial \mathbf{b}}\mathbf{u} \qquad (20)$$

$$\mathbf{Kv} = (\mathbf{DBA})^{\mathrm{T}}$$

6.3. Two dimensional shape optimization flow chart

Fig.5 shows the flow chart of new shape optimization algorithm. When only control point positions are considered as design variables during the shape optimization, control points can be locally gathered during the optimization process which hampers the convergence and restricts their movement. As a result, the suggested shape optimization process considers both control point position and weight as design variables and separates the optimization steps. The suggested shape optimization is composed of position optimization (1^{st} step) and weight optimization (2^{nd} step). At first, control point position optimization proceeds and if the result is converged, it is the final optimal result. But if the closest distance among control points is less than the threshold value during the position optimization, weights become the design variables to prevent locally crowding of control points.

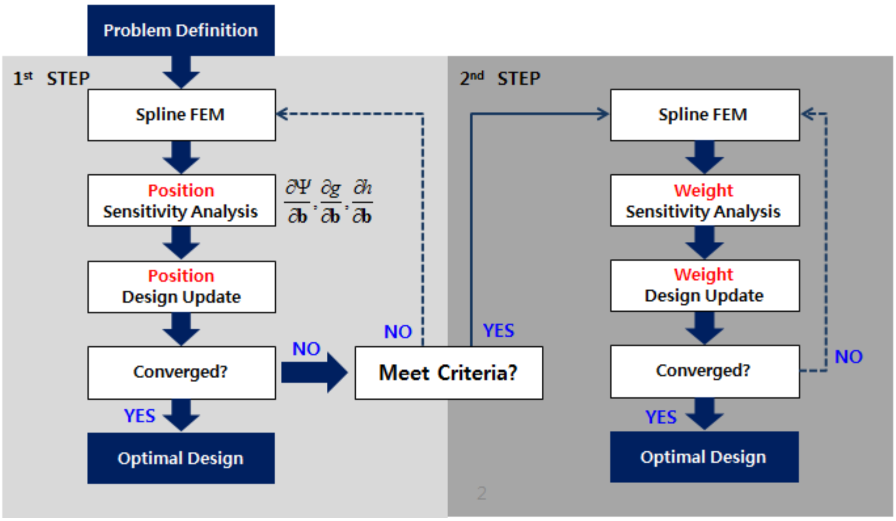

Figure 5 : New shape optimization algorithm

6.4. Shape optimization of cantilever beam: displacement minimization

New shape optimization algorithm is applied to a short cantilever beam (Fig.6). The beam is subjected to concentrated force at right bottom vertex. Minimization of y displacement of point A with volume constraint is employed as Eq. (21).

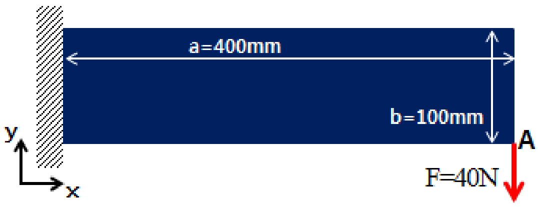

Figure 6 : Problem definition, cantilever beam

$$\begin{aligned} &\text{Minimize} \quad u_{A_y} \\ &\text{subject to} \quad V \le 0.6V^* \end{aligned} \qquad (21)$$

Fig.7 shows control points of cantilever beam the top edge is set to design boundary with control point positions and weights. If the closest distance of any two neighboring control points is less than 25 percent of the initial closest distance during the position optimization step, weights become the design variables.

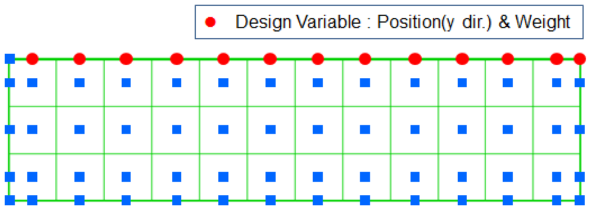

Figure 7 : Definition of design variables, cantilever beam

The result from shape optimization with control point position only and from the shape optimization considering both control point position and weight are shown in Fig.8. In the comparison of optimization results, width of position & weight optimization is larger than position-only case. In position-only case, the control points of right edge come closer each other. In order to prevent this phenomenon, if the closest distance of control points is less than the criteria value, control points positions are fixed and the process is switched to weight optimization. Fig.9 represents convergence history. It shows position & weight shape optimization has lower objective function.

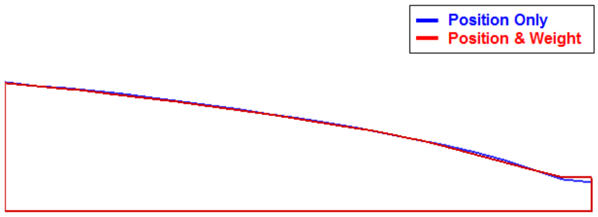

Figure 8 : Comparison of beam optimization result

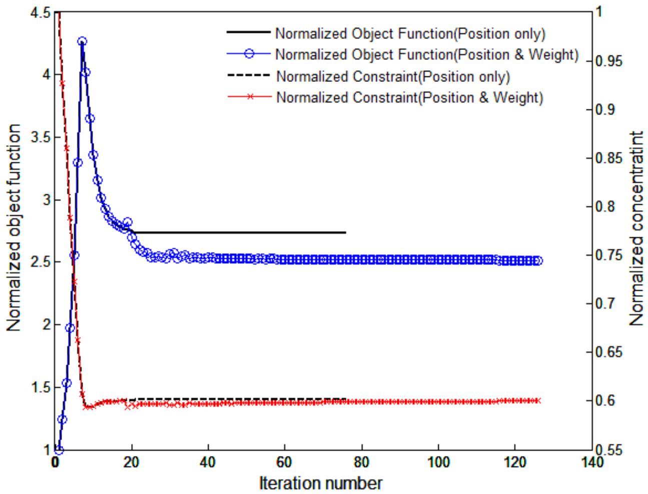

Figure 9 : Histories of objective function and constraint, Beam

7. Conclusion

In the present work, suggested shape optimization algorithm considers both control point positions and weights as design variables. New shape optimization algorithm has advantages in flexible shape representation and in the treatment for location of control points in the limited case by separating the control position optimization step and weight optimization step.

8. Acknowledgements

This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No.2012-0000976) and the NRF grant funded by the Korea government (MEST) (No.2011-0015469).

9. References

[1] L. Piegl and W. Tiller, *The NURBS Book. 2nd ed*. Spinger, New York, 1997
[2] Y. D. Seo, H. J. Kim, and S. K. Youn, Shape optimization and its extension to topological design based on isogeometric analysis, *International Journal of Solids and Structures*, 47, 1618-1640, 2010

[3] S. Cho and S. H. Ha, Isogeometric shape design optimization: exact geometry and enhanced sensitivity, *Structural and Multidisciplinary Optimization*, 38, 53-70, 2009
[4] T. J. R. Hughes, J. A. Cottrell and Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Structural Engineering and Mechanics*, 30, 225-245, 2008
[5] X. Qian, Full analytical sensitivities in NURBS based isogeometric shape optimization, *Computer Methods in Applied Mechanics and Engineering*, 197, 2059-2071, 2010