KSME

Springer

Journal of Mechanical Science and Technology 38 (7) 2024

Original Article
DOI 10.1007/s12206-024-0625-8

Keywords:
· MQ quasi-interpolation
· Structural topology optimization
· Level set method
· Level set band method

Correspondence to:
Jian-Hu Feng
jhfeng@chd.edu.cn

Citation:
Yang, C.-D., Feng, J.-H., Ren, J., Shen, Y.-D. (2024). MQ quasi-interpolation-based level set method for structural topology optimization. Journal of Mechanical Science and Technology 38 (7) (2024) 3521~3532.
http://doi.org/10.1007/s12206-024-0625-8

Received October 30th, 2023
Revised February 21st, 2024
Accepted March 15th, 2024

† Recommended by Editor Hyun-Gyu Kim

MQ quasi-interpolation-based level set method for structural topology optimization

Chen-Dong Yang[1,2], Jian-Hu Feng[1], Jiong Ren[2] and Ya-Dong Shen[3]

[1]School of Science, Chang'an University, Xi'an 710064, China, [2]School of Science, Xi'an Aeronautical Institute, Xi'an 710077, China, [3]School of Civil Engineering and Architecture, Nanyang Normal University, Nanyang 473061, China

Abstract Parametric level set method (PLSM) using interpolation method, such as radial basis function (RBF) interpolation, exposes high computational cost and poor stability when solving structural topology optimization (STO) problems with large-scale nodes. However, the quasi-interpolation method can approximate the level set function (LSF) and its generalized functions without solving any system of linear equations. With this good property, this paper utilizes multiquadric (MQ) quasi-interpolation to parameterize the LSF and innovatively introduces it into the STO problem. Moreover, the MQ quasi-interpolation is utilized to compute the element density, which makes the level set band method (LSBM) more rigorous. The proposed methods were compared with the PLSM based on compactly supported radial basis functions (CSRBFs). The results show that the approximation accuracy, computational efficiency and stability of the evolution process of the proposed methods are better than those of CSRBFs when the shape parameter takes a suitable small value.

1. Introduction

The concept of structural topology optimization (STO) was first introduced by Michell [1] in 1904, mainly for solving the problem of minimizing the mass of a truss under stress constraints. So far, many topology optimization models have been developed, such as the homogenization-based method [2], the solid isotropic material with penalty (SIMP) method [3], the evolutionary structural optimization (ESO) method [4, 5], and the level set method (LSM) [6-8]. To compensate for the disadvantages of the different methods, scholars have successively proposed different improvement methods which seek for different trade-offs between design results and computational efficiency [9]. At the same time, the various methods have gradually borrowed from each other, and the proposal of combined methods has become a trend.

SIMP and ESO are often referred to as density-based methods, which directly discretize the design domain into a number of material units, each of which consists of a relative density as a design variable, and this discretization can be consistent with finite elements, where all material units are pooled together to wholly reflect the structural topology. They have been widely used due to the advantages of simplicity, fast convergence, and the ability to incorporate well-established optimization methods [10-12], but there are some drawbacks [13], such as checkerboard phenomenon and mesh dependence.

Level set method can be viewed as a boundary-based method. It represents the interface implicitly as a zero level set of a higher dimensional function called level set function (LSF) to efficiently represent the structural interface. It was initially applied to STO by Sethian [6]. Wang et al. [7] established a foundational mathematical programming framework that describes a STO as a sequence of motions of the implicit boundaries converging to an optimum solution and satisfying specified constraints. Compared with density-based methods, the main advantages of LSM in dealing with STO problems are always (1) obtaining exact boundary information without fuzzy boundaries due to intermediate densities, and (2) merging holes and dealing with topology changes do not require complex computations [14].

The parametric level set method (PLSM) decouples the Hamilton-Jacobi partial differential equation (H-J PDE) in time and space by interpolating the scalar data field at the nodes. While inheriting the superior performance of the LSM, the PLSM also exhibits unique advantages [15-17]. However, the basic models of PLSM still have many shortcomings, such as their results are heavily dependent on initialization design and meshing, high computational cost, and unstable optimization process, which greatly hinder their application to real engineering problems [18].

To overcome the shortcomings of PLSM, many improvement methods have been proposed. For instance, Dunning and Kim [19] propose sequential linear programming LSM that can handle multiple constraints and simultaneously optimize non-level-set design variables. Hyun and Kim [20] develop a level-set thermal eigenvalue topology optimization method to efficiently optimize the overall response time of heat transfer systems. Li et al. [21] incorporated a maximum length scale constraint into the reaction-diffusion equation-based LSM to generate shapes with rich features. Liu et al. [22] proposed an ordinary differential equation (ODE) driven level-set density topology optimization method, which combines the LSM and the density method and thereby drives the optimization process to converge stably and efficiently to a solid space solution. In our previous paper [23], we proposed a step size adaptive method based on the first-order Eulerian method. This method can use large time steps to accelerate the evolution process while maintaining stability.

Interpolation methods, such as radial basis function (RBF) interpolation, are commonly used to parameterize LSF. They typically require solving systems of linear equations, which is computationally inefficient and memory intensive. To improve the efficiency of interpolation, Wang et al. [24] utilizes the velocity field level-set (VFLS) method to adaptively change the arrangement of velocity knots (design variables), which significantly reduced the number of design variables. Our previous paper [25] transformed the interpolation matrix based on compactly supported radial basis functions (CSRBFs) into a triangular matrix, which effectively improves the efficiency of solving systems of linear equations.

Unlike interpolation methods, the quasi-interpolation method can obtain a well approximation of the original function without solving any system of linear equations [26]. Therefore, it has become one of the important methods in approximation field in recent years. The two schemes of quasi-interpolation that have been most widely studied are spline quasi-interpolation [27-29] and multiquadric (MQ) quasi-interpolation [30-33]. In this paper, we propose an MQ quasi-interpolation based level set method (MQLSM) to solve the STO problem, mainly based on the good approximation property of MQ quasi-interpolation to LSF and its generalized functions.

The paper is organized as follows: Sec. 2 discusses LSM for the compliance minimization problem. Sec. 3 briefly describes MQ quasi-interpolation. In Sec. 4, two MQLSM methods using the Ersatz material method and MQ quasi-interpolation for finite element analysis (FEA) respectively are presented. The discussions are given in Sec. 5. Finally, some general conclusions are given in Sec. 6.

2. Level set method for the compliance minimization problem

For the LSM, the specific value of Lipschitz continuous high-dimensional scalar function Φ represents the boundary. Considering a reference domain $D \subset R^d$, d = 2 or 3, the LSF Φ, which includes all allowable shapes $\Omega \subset D$, holds as

$$\Phi(\mathbf{x})\begin{cases} >0 \ , if \ \ \mathbf{x}\in \Omega \setminus \partial\Omega \\ =0 \ , if \ \ \mathbf{x}\in \partial\Omega \\ <0 \ , if \ \ \mathbf{x}\in D \setminus \Omega \end{cases}. \tag{1}$$

The H-J PDE is the most common driving equation for LSF [34], which is defined as

$$\frac{\partial \Phi}{\partial t} - \mathrm{v}_n \left|\nabla \Phi\right| = 0, \ \Phi(\mathbf{x}, \mathrm{t}_0) = \Phi_0(\mathbf{x}). \tag{2}$$

where v_n is the velocity component in the normal direction, which is defined as [35]

$$\mathrm{v}_n = \mathbf{v}\cdot\mathbf{n} = \mathbf{v}\cdot\left(-\frac{\nabla\Phi}{|\nabla\Phi|}\right). \tag{3}$$

where $\mathbf{n} = -\nabla\Phi/|\nabla\Phi|$ is the unit vector in the normal direction.

The H-J PDE needs to be discretized in space and pseudo-time, and the resulting discrete system of equations is often numerically solved by the finite difference upwind scheme [7, 36].

When the LSF is evolved explicitly, pseudo time step Δt needs to comply with the Courant–Friedrichs–Lewy (CFL) condition [7, 15]:

$$\Delta t \max\left(\left|\mathrm{v}_n\right|\right) \le h \tag{4}$$

where h denotes the minimum mesh size.

The compliance minimization formulas of a linear elastic structure with static load under volume constraints are as follows:

$$\begin{aligned} &Minimize: J(\mathbf{u},\Phi) = \int_D \left(\mathbf{e}(\mathbf{u}):\mathbf{C}:\mathbf{e}(\mathbf{u})\right) H(\Phi)\, d\Omega \\ &s.t \begin{cases} a(\mathbf{u},\ \mathbf{v},\Phi) = l(\mathbf{v},\Phi), \forall \mathbf{v}\in U \\ G(\Phi) = \int_D H(\Phi)\, d\Omega - V_{\max} \le 0 \\ \mathbf{u} = \mathbf{u}_0, in\ \Gamma_D \\ \mathbf{C}:\mathbf{e}(\mathbf{u})\cdot\mathbf{n} = \boldsymbol{\tau}, in\ \Gamma_\tau \end{cases} \end{aligned} \tag{5}$$

where $a(\mathbf{u},\mathbf{v},\Phi)$ and $l(\mathbf{v},\Phi)$ are defined as

$$a(\mathbf{u},\mathbf{v},\Phi) = \int_D \left(\mathbf{e}(\mathbf{u}):\mathbf{C}:\mathbf{e}(\mathbf{u})\right) H(\Phi)\, d\Omega$$

$$l(\mathbf{v},\Phi)=\int_{\Gamma_\tau}\boldsymbol{\tau}\cdot\mathbf{v}d\Gamma+\int_D\mathbf{b}\cdot\mathbf{v}H(\Phi)d\Omega\,.$$

In Eq. (5), $J(\mathbf{u},\Phi)$ is the objective function, $\mathbf{e}$ is the linearized strain tensor, $\mathbf{C}$ is the Hook elasticity, $\mathbf{u}$ and $\mathbf{v}$ are the displacement and the virtual displacement. The $V_{\max}$ is the target volume, $G(\Phi)$ is the constrain introduced to restrict the use of material. The specific displacement $\mathbf{u}_0$ is given on Dirichlet boundary Γ_D, $\boldsymbol{\tau}$ is the traction on traction boundary Γ_τ, and $\mathbf{b}$ is body force, $a(\mathbf{u},\mathbf{v},\Phi)$ and $l(\mathbf{v},\Phi)$ are written in energy bilinear form and load linear respectively. The Heaviside function H is defined as

$$H(\Phi)=\begin{cases}1, if\ \ \Phi\geq 0\\ 0, if\ \ \Phi<0\end{cases}. \tag{6}$$

The normal velocitys along the moving free boundary can be simply determined by utilizing the strain energy density and the Lagrange multiplier [36, 37].

$$\mathrm{v}_n=\mathbf{e}(\mathbf{u}):\mathbf{C}:\mathbf{e}(\mathbf{u})-\lambda, \tag{7}$$

where λ is updated by the following scheme [37]:

$$\lambda^{k+1}=\begin{cases}\mu G^k, k\leq n_R\\ \lambda^k+\gamma^k G, k>n_R\end{cases} \tag{8}$$

where μ and n_R are the optimization parameters, γ and G are updated as follows

$$G^k=\int_D H(\Phi)d\Omega-\left[V_0-(V_0-V_{\max})\frac{k}{n_R}\right], k\leq n_R, \tag{9}$$

$$\gamma^{k+1}=\min\left(\gamma^k+\Delta\gamma,\gamma_{\max}\right), k>n_R, \tag{10}$$

where $\gamma_{\max}$ and $\Delta\gamma$ are the upper limit and increment of the parameter γ.

3. MQ quasi-interpolation

3.1 Univariate MQ quasi-interpolation

Assume that a given set of interpolation nodes $\{\mathrm{x}_i\}_{i=0}^n$ of a finite interval $[a,b]$ satisfies:

$$a=x_0<x_1<\cdots<x_n=b\,. \tag{11}$$

Considering the interpolation data $\{x_i,f_i\}_{i=0}^n$, the general form of the quasi-interpolation [30] is:

$$(\mathcal{L}f)(x)=\sum_{i=0}^n f_i\Psi(x-x_i) \tag{12}$$

where Ψ is a kernel function.

Wu and Schaback [31] constructed a classical quasi-interpolation operator $\mathcal{L}_D$ that preserve linearity and convexity.

$$\begin{aligned}(\mathcal{L}_Df)(x)=&f_0\alpha_0(x)+f_1\alpha_1(x)+\sum_{i=2}^{n-2}f_i\alpha_i(x)\\&+f_{n-1}\alpha_{n-1}(x)+f_n\alpha_n(x)\end{aligned} \tag{13}$$

where

$$\alpha_0(x)=\frac{1}{2}+\frac{\phi_1(x)-(x-x_0)}{2(x_1-x_0)}$$

$$\alpha_1(x)=\frac{\phi_2(x)-\phi_1(x)}{2(x_2-x_1)}-\frac{\phi_1(x)-(x-x_0)}{2(x_1-x_0)}$$

$$\alpha_i(x)=\frac{\phi_{i+1}(x)-\phi_i(x)}{2(x_{i+1}-x_i)}-\frac{\phi_i(x)-\phi_{i-1}(x)}{2(x_i-x_{i-1})},2\leq i\leq n-2$$

$$\alpha_{n-1}(x)=\frac{x_n-x-\phi_{n-1}(x)}{2(x_n-x_{n-1})}-\frac{\phi_{n-1}(x)-\phi_{n-2}(x)}{2(x_{n-1}-x_{n-2})}$$

$$\alpha_n(x)=\frac{1}{2}+\frac{\phi_{n-1}(x)-(x_n-x)}{2(x_n-x_{n-1})}$$

and $\phi_i(x)=\sqrt{(x-x_i)^2+c^2}$ is a one-dimensional MQ basis functions.

In addition, $\mathcal{L}_D$ can also be expressed in the following form:

$$\begin{aligned}(\mathcal{L}_Df)(x)=&\frac{f_0+f_n}{2}+\frac{f_1-f_0}{2(x_1-x_0)}(x-x_0)\\&+\frac{1}{2}\sum_{i=1}^{n-1}\left(\frac{f_{i+1}-f_i}{x_{i+1}-x_i}-\frac{f_i-f_{i-1}}{x_i-x_{i-1}}\right)\phi_i(x)\\&-\frac{f_n-f_{n-1}}{2(x_n-x_{n-1})}(x_n-x)\,.\end{aligned} \tag{14}$$

For Eq. (14), assume that $\phi_r(x)=\sqrt{(x-x_r)^2+c^2}$, including $1\leq r\leq n-1$, $\phi_n(x)\equiv 1$ and $\phi_{n+1}(x)\equiv x$ are reorganized as coupled polynomials. Then,

$$(\mathcal{L}_Df)(x,y)=\sum_{r=1}^{n+1}\left[\left(T_{\alpha\to\phi}\right)F\right]_r\phi_r(x)=\sum_{r=1}^{n+1}\lambda_r\phi_r(x) \tag{15}$$

where $F=[f_0,f_1,\cdots f_n]^{\mathrm{T}}$ is column vector of length $n+1$. The upper triangular ribbon principal diagonal matrix $T_{\alpha\to\phi}$ of size $(n+1)\times(n+1)$ is:

$$T_{\alpha\to\phi}=\begin{bmatrix}D_1 & E_2 & D_2 & 0 & \cdots & \cdots & \cdots & 0\\ 0 & D_2 & E_3 & D_3 & 0 & \cdots & \cdots & 0\\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots\\ \vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots\\ \vdots & & & 0 & D_{n-2} & E_{n-1} & D_{n-1} & 0\\ 0 & \cdots & \cdots & \cdots & 0 & D_{n-1} & E_n & D_n\\ 1+x_0D_1 & -x_0D_1 & 0 & \cdots & \cdots & 0 & x_nD_n & 1-x_nD_n\\ -D_1 & D_1 & 0 & & & 0 & -D_n & D_n\end{bmatrix} \tag{16}$$

where $D_i = 1/(x_i - x_{i-1})$, $1 \le i \le n$, $E_i = -D_i - D_{i-1}$, $2 \le i \le n$.

3.2 Multivariate MQ quasi-interpolation

Consider the two-dimensional example, the MQ basis function for the other one-dimensional variable y is:

$$\psi_j(y) = \sqrt{(y - y_j)^2 + c^2}. \tag{17}$$

Its interpolation kernel coefficients $\beta_j(y)$ $(0 \le j \le m)$ are similar to those defined in Eq. (13).

The dimensional-splitting MQ basis function (DSMQ) proposed by Ling [32] is used as the two-dimensional basis function:

$$\sqrt{(x - x_i)^2 + c^2} \cdot \sqrt{(y - y_j)^2 + c^2}. \tag{18}$$

Based on the interpolated data $\{x_i, y_j, f_{ij}\}$ $i = 0,1,\cdots,n; j = 0,1,\cdots,m$, the coefficients of the DSMQ basis function are the result of the multiplication of $\alpha_i(x)$ and $\beta_j(y)$, so the two-dimensional quasi-interpolation formula is:

$$(\mathcal{L}_D f)(x,y) = \sum_{i=0}^{n}\sum_{j=0}^{m} f_{ij}\alpha_i(x)\beta_j(y). \tag{19}$$

Ling [32] proposed a two-dimensional formula to transform from $(\alpha \otimes \beta)$ to $(\phi \otimes \psi)$, transforming the Eq. (19) into:

$$\begin{aligned}(\mathcal{L}_D f)(x,y) &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\left[\left(T_{\alpha\to\phi}\right)F\left(T_{\beta\to\psi}\right)^{\mathrm{T}}\right]_{rs}\phi_r(x)\psi_s(y)\\ &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\lambda_{rs}\phi_r(x)\psi_s(y)\end{aligned} \tag{20}$$

where F is the matrix that stores the function values. Similar to $T_{\alpha\to\phi}$, $T_{\beta\to\psi}$ is a square matrix of size $(m+1)\times(m+1)$.

By converting $T_{\alpha\to\phi}$ and $T_{\beta\to\psi}$ into sparse matrices or as filters, Eq. (20) needs about $11N^2$ floating-point calculations once, where N is the total number of nodes. By using the fast multipole method [38] to calculate MQ base functions, the total calculation amount can be reduced to about $2N^2$ [32].

4. Level set method based on MQ quasi-interpolation

4.1 Parameterization of level set function using MQ quasi-interpolation

According to Eq. (20), the quasi-interpolation result of the LSF can be obtained by using the function value at the interpolation node.

$$\begin{aligned}(\mathcal{L}_D\Phi)(x,y) &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\left[\left(T_{\alpha\to\phi}\right)\boldsymbol{\Phi}\left(T_{\beta\to\psi}\right)^{\mathrm{T}}\right]_{rs}\phi_r(x)\psi_s(y)\\ &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\lambda_{rs}\phi_r(x)\psi_s(y)\end{aligned} \tag{21}$$

where $\boldsymbol{\Phi}$ is the matrix that stores the function values of the interpolating nodes.

According to the property that quasi-interpolation is a good approximation to the derivative of the original function, the approximation to x and y partial derivatives can be derived as:

$$\begin{aligned}(\mathcal{L}_D\Phi)'_x(x,y) &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\lambda_{rs}\phi'_r(x)\psi_s(y)\\ (\mathcal{L}_D\Phi)'_y(x,y) &= \sum_{r=1}^{n+1}\sum_{s=1}^{m+1}\lambda_{rs}\phi_r(x)\psi'_s(y).\end{aligned} \tag{22}$$

Thus, the corresponding gradient approximation is

$$|\nabla\Phi| \approx \sqrt{\left((\mathcal{L}_D\Phi)'_x\right)^2 + \left((\mathcal{L}_D\Phi)'_y\right)^2} \tag{23}$$

4.2 Finite element analysis using MQ quasi-interpolation

The traditional LSM generally uses the Ersatz material method to calculate the element density, and the density of each element near the zero level set is a two-point distribution {0, 1}, which will cause numerical instability in the evolution process with the formation or disappearance of holes [39]. The level set banding method (LSBM) [39] effectively improves the topology optimization ability of the LSM by introducing intermediate densities around the zero level set. In this method, the density of each element of the structure depends on its node value of the LSF. For the LSBM, the density of each element of the structure depends on the node values of its LSF.

$$\rho_e = H(\Phi_e) \tag{24}$$

where ρ_e is the density of the e-th element and Φ_e is the value of the LSF in the middle of the element using interpolation, $H(\Phi)$ is the approximation scheme of Heaviside function as Ref. [7]:

$$H(\Phi) = \begin{cases} 1, & \Phi > \Delta \\ \dfrac{3}{4\Delta}\left(\Phi - \dfrac{\Phi^3}{3\Delta^2}\right) + \dfrac{1+\zeta}{2}, & -\Delta \le \Phi \le \Delta \\ \zeta, & \Phi < -\Delta \end{cases} \tag{25}$$

where ζ is the smaller positive number, Δ is the corresponding transition range between the LSF and the material distribution, and 2Δ is the width of the level set band.

Obviously, the Ref. [39] is not rigorous in calculating elemental densities using the approximate substitution of the value of the midpoint of the element.

In fact, the element density can be calculated exactly by the following formula [22]:

$$\rho_e = \frac{\int_{D_e} H(\Phi) dV}{\int_{D_e} dV}. \tag{26}$$

In this paper, we use MQ quasi-interpolation to rigorously calculate the integral of $H(\Phi)$ over an element, which makes the calculation of element density more reasonable. For rectangular mesh, the e-th density of element $[x_p, x_{p+1}]\times[y_q, y_{q+1}]$ is

$$\begin{aligned}\rho_e &= \frac{\int_{x_p}^{x_{p+1}}\int_{y_q}^{y_{q+1}} \mathcal{L}_D\left(H(\Phi)\right) dxdy}{\int_{x_p}^{x_{p+1}}\int_{y_q}^{y_{q+1}} dxdy} \\ &= \frac{\sum_{r=1}^{n+1}\sum_{s=1}^{m+1} \tilde{\lambda}_{rs}\int_{x_p}^{x_{p+1}} \phi_r(x)dx \int_{y_q}^{y_{q+1}} \psi_s(y)dy}{\left(x_{p+1}-x_p\right)\left(y_{q+1}-y_q\right)}\end{aligned} \tag{27}$$

where $\tilde{\lambda} = \left[\left(T_{\alpha\to\phi}\right) H(\Phi) \left(T_{\beta\to\psi}\right)^{\mathrm{T}}\right]$. The analytical solutions of $\int_{x_p}^{x_{p+1}} \phi_r(x)dx$ and $\int_{y_q}^{y_{q+1}} \psi_s(y)dy$ are easily derived.

After determining the element density ρ_e, the Young's modulus E_e can be calculated as [22, 37]

$$E_e(\rho_e) = E_{\min} + (\rho_e)^{pen}\left(E_0 - E_{\min}\right) \tag{28}$$

where $E_{\min}$ is the small positive value that avoids the singularity of the global stiffness matrix, E_0 is the Young's module of the full material element and pen is the penalty, which is taken as 1 in this paper. The physical FEA model of the structure is established after calculating the element stiffness and assembly overall stiffness matrices.

Obviously, the proposed method is more reasonable and accurate than the Ref. [39] where the values of element centers are taken as densities. The FEA of calculating element density using Eq. (27) is named as multiquadric quasi-interpolation finite element analysis (MQFEA).

4.3 Optimization scheme

An approximation function $\delta(\Phi)$ was introduced to limit the unbounded growth of Φ in evolutionary scheme [7, 37].

$$\delta(\Phi) = \begin{cases} 0, & \Phi > \tilde{\Delta} \\ \dfrac{3}{4\tilde{\Delta}}\left(1 - \dfrac{\Phi^2}{\tilde{\Delta}^2}\right), & -\tilde{\Delta} \le \Phi \le \tilde{\Delta} \\ 0, & \Phi < -\tilde{\Delta} \end{cases} \tag{29}$$

where $\tilde{\Delta}$ is the defined upper limit of Φ.

In order to keep Φ at the boundary as a signed distance function, which is $|\nabla\Phi|$ = 1, an approximate reinitialization scheme was proposed in Ref. [37].

Algorithm 1. The algorithm implementation of MQLSM with Ersatz or MQFEA.

Algorithm: MQLSM

1: Initialize the LSF and the values of parameters.
2: **Repeat**
3: Using the Ersatz material method (Ersatz) or MQ quasi-interpolation (MQ) to calculate the element densities.
4: Calculate normal velocity v_n using Eq. (7).
5: Update the LSF Φ using Eq. (31).
6: The coordinates of the nodes around the zero level set are calculated, and the LSF Φ is re-initialized by the Eq. (30).
7: **Until** the termination condition is met.

$$\tilde{\Phi} = \Phi / \mathrm{M} \tag{30}$$

where $\mathrm{M} = mean\left(|\nabla\Phi_1|, |\nabla\Phi_2|, \ldots |\nabla\Phi_l|\right)$, $\nabla\Phi_l$ is the gradient value of the l-th point around the zero level set, which can be obtained by Eq. (23).

Unlike the Ref. [37], which calculates the gradient values of all nodes, this paper determines the coordinates of the nodes needed around the zero level set based on the FEA, and then calculates the gradient values based on Eq. (23), which can save unnecessary computational cost.

The first-order Euler forward method is used to evolve the level set equation, and the updated formula is as follows:

$$\Phi^{k+1}(x,y) = \tilde{\Phi}^k(x,y) + \Delta t B^k(x,y) \tag{31}$$

where k is the number of iterations, Δt is the time step and $B^k(x,y) = \mathrm{v}_n^k(x,y)\delta\left(\tilde{\Phi}^k(x,y)\right)$.

This method is named as the multiquadric quasi-interpolation based level set method (MQLSM), and its algorithm implementation is as Algorithm 1.

5. Numerical experiments and discussion

The symbols are illustrated as follows: MQLSM-Ersatz (or MQLSM) denotes the MQLSM using the Ersatz material method, MQLSM-MQFEA denotes the MQLSM using MQFEA, CSRBFs denotes the PLSM based on compactly supported radial basis functions (CSRBFs) in Ref. [38].

In this section, the material properties are chosen as that Young's modulus E_0 = 1 for solid material, $E_{\min}$ = 10^9 for void material, Poisson's ratio v = 0.3. The parameters μ, γ, $\Delta\gamma$, $\gamma_{\max}$, n_R and $\tilde{\Delta}$ are assigned the values of 20, 0.05, 0.05, 5, 30 and 10, respectively. The termination criterion is that the relative errors of the objective function values $J(\mathbf{u},\Phi)$ in Eq. (5) are less than 10^{-3} in consecutive 10 iterations and the relative errors between the consecutive volume fraction are less than 10^{-3}.

When using explicit Euler forward method evolution, the time step and velocity have a very important impact on the numerical stability and evolution efficiency of the PLSM, and they still need to satisfy the CFL condition [15]. However, since many

Refs. [17, 38] use larger time steps for acceleration, for comparison purposes, the step with stable convergence process is selected according to 0.5, 0.25, 0.1, 0.05, 0.025 and 0.01 for each case in descending order.

When the scale of the mesh is large, the CSRBFs need to consume a large amount of memory, which cannot be run on a personal computer, so the CSRBFs in Secs. 5.3 and 5.4 are run on a workstation with the following parameters: Phytium 2000 (2.20 GHz) CPU, 512G of RAM, Linux operating system, Matlab development environment. Other codes are run on a personal computer with the following parameters: Intel i5 (2.20 GHz) CPU, 12G RAM, Windows 10 (64-bit) operating system, Matlab development environment.

5.1 Influence of parameter c on approximation accuracy

The test LSFs are defined as

$$P=\cos\left[\frac{(1+h_x)\pi}{n_x}\left(x-\frac{n_x}{2}\right)\right]+\cos\left[\frac{(1+h_y)\pi}{n_y}\left(y-\frac{n_y}{2}\right)\right]$$

$$\Phi 1=1-P,\left(h_x=1,h_y=1\right)$$

$$\Phi 2=1-|P|,\left(h_x=5,h_y=3\right)$$

where n_x (n_y) and h_x (h_y) denote the length of the region and the number of holes along the x (y) direction, respectively.

The images of functions $\Phi 1$, $\Phi 2$ and their corresponding structures are shown in Fig. 1. The function $\Phi 1$ is smooth and its partial derivatives are continuous in the reference domain, while the partial derivatives of $\Phi 2$ are discontinuous at $\Phi 2=1$.

Set n_x = 60, n_y = 30. To compare the approximation accuracy of MQLSM and CSRBFs, and 61×31 nodes are uniformly selected as interpolating nodes according to the interval h = 1, and 601×301 nodes are uniformly selected as test nodes according to the interval h_t = 0.1. The shape parameter c of the MQ basis functions is taken as 0.01 and 1, respectively.

The mean of the absolute values of the differences between the results and the test function values is used as a measure of error.

$$e_{\mathrm{m}}=mean\left(\left|\Phi(x,y)-\hat{\Phi}(x,y)\right|\right) \tag{32}$$

where $\Phi(x,y)$ denotes the test function and $\hat{\Phi}(x,y)$ denotes the approximation function. The partial derivatives Φ_x, Φ_y and the modulus of the gradient $|\nabla\Phi|$ are also measured using this error measure, and the results are shown in Table 1.

In Table 1, when the shape parameter c = 0.01, the approximation accuracy of the MQLSM for the original function ($\Phi 1$ and $\Phi 2$) and its generalized functions is higher than that of the CSRBFs. In particular, the approximation accuracy of MQLSM is about one order of magnitude higher than that of the CSRBFs for the smooth function $\Phi 1$. When the shape parameter c = 1, the approximation accuracy of the MQLSM for the smooth function $\Phi 1$ is still higher than that of the CSRBFs. But the approximation accuracy of MQLSM for non-smooth functions $\Phi 2$ is lower than that of CSRBFs.

This indicates that better approximation accuracy is generally obtained when the parameter c takes a reasonably small value. It is worth noting that the parameter c as a shape parameter of the RBF, gives a higher order smooth approximation function when it takes larger values [33]. Meanwhile, maintaining the smoothness of the LSF is also beneficial for the MQLSM to maintain a high approximation accuracy.

Can a fixed parameter c maintain a high approximation accuracy for the LSF and its generalized functions as the LSF is varied in topology optimisation? This is verified in Sec. 5.2.

5.2 Performance of MQLSM-Ersatz

The design domain is shown in Fig. 2, and the dimension ratio of the short cantilever beam is 2×1. Its left boundary is fixed, and the middle point of the right boundary is subjected to a force F = -1 in the vertical direction. The expected volume fraction is set to 50 %. The function $\Phi 2$ is used as the initial value of the LSF, and the corresponding initial design is shown in Fig. 1.

In order to compare the performance of CSRBFs and

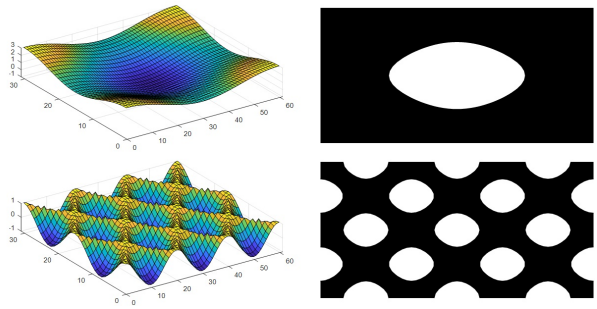

Fig. 1. Different LSFs (left column) and their corresponding structures (right column) under mesh 60×30.

Table 1. Comparison of approximation accuracies of different methods under mesh 60×30.

		CSRBFs	MQLSM	
			$c=0.01$	$c=1$
$\Phi 1$	$\Phi 1$	4.02×10^{-2}	2.40×10^{-3}	3.02×10^{-2}
	$\Phi 1_x$	7.27×10^{-2}	1.42×10^{-3}	1.25×10^{-3}
	$\Phi 1_y$	7.87×10^{-2}	5.80×10^{-3}	7.47×10^{-3}
	$\lvert\nabla\Phi 1\rvert$	7.69×10^{-2}	4.36×10^{-3}	6.44×10^{-3}
$\Phi 2$	$\Phi 2$	2.09×10^{-2}	1.83×10^{-2}	1.30×10^{-1}
	$\Phi 2_x$	4.69×10^{-2}	3.24×10^{-2}	7.51×10^{-2}
	$\Phi 2_y$	5.32×10^{-2}	4.86×10^{-2}	1.06×10^{-1}
	$\lvert\nabla\Phi 2\rvert$	5.24×10^{-2}	4.34×10^{-2}	1.24×10^{-1}

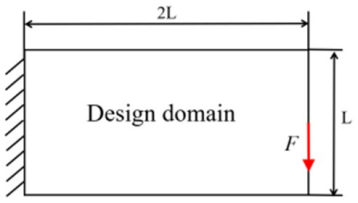

Fig. 2. Design domain of cantilever beam.

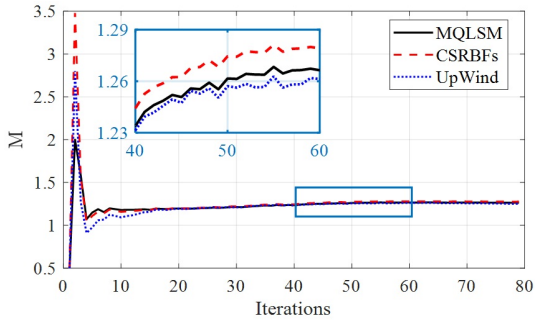

Fig. 3. Comparison of calculation results of M by MQLSM, CSRBFs and upwind scheme.

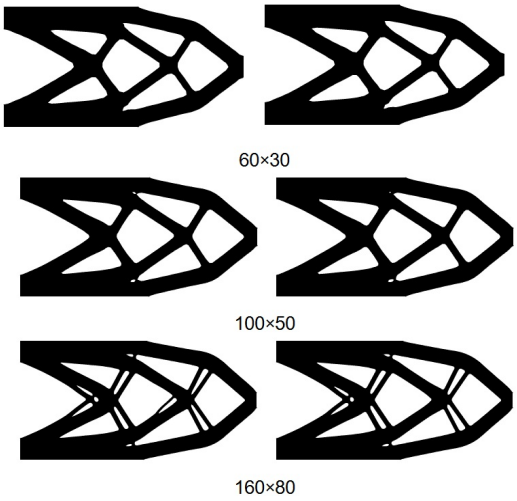

Fig. 4. Optimal design of CSRBFs (left column) and MQLSM (right column).

MQLSM, comparative experiments are conducted on them under mesh divisions of 60×30 (n_x = 60, n_y = 30), 100×50 (n_x = 100, n_y = 50) and 160×80 (n_x = 160, n_y = 80), respectively. The FEA of the MQLSM uses the Ersatz material method to calculate the element density with the parameter c = 0.01. The time step of both methods is 0.5 in this section.

When the mesh is divided into 60×30 and the same node is selected to calculate the gradient in each iteration, the comparison of the calculation results of $\mathrm{M} = mean\left(\left|\nabla\Phi_1\right|, \left|\nabla\Phi_2\right|, \ldots \left|\nabla\Phi_I\right|\right)$ by MQLSM, CSRBFs [37] and upwind scheme [7, 36] is shown in Fig. 3. As easily found, the results of the three methods are very similar, which indicates that MQLSM can maintain a high approximation accuracy for the LSF and its generalized functions when the parameter c is chosen as a fixed value.

Table 2. Comparison of the computational performance of the two methods, the e_m is approximation error of initial design, the O represents objective function value, the T represents the single iteration time and the N represents the number of iterations.

		e_m	O	T/s	N
60×30	CSRBFs	2.09×10^{-2}	59.81	0.29	80
	MQLSM	1.83×10^{-2}	59.81	0.06	80
100×50	CSRBFs	1.96×10^{-2}	60.21	1.63	83
	MQLSM	6.94×10^{-3}	60.22	0.19	83
160×80	CSRBFs	1.94×10^{-2}	60.41	18.57	63
	MQLSM	2.67×10^{-3}	60.41	0.65	69

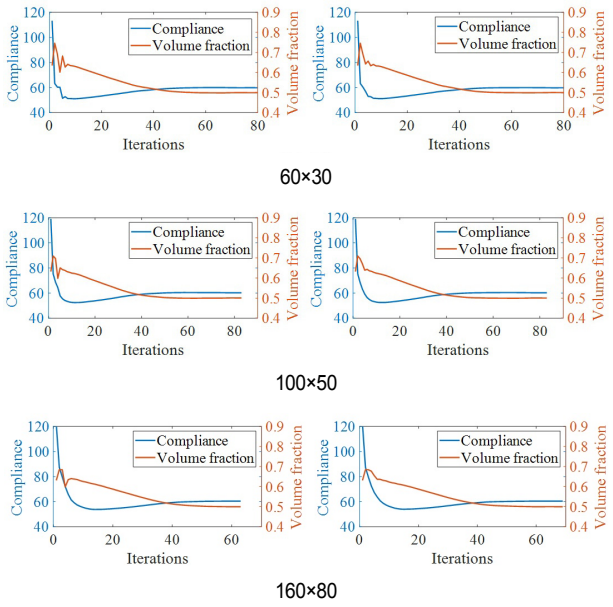

Fig. 5. Convergence curves of CSRBFs (left column) and MQLSM (right column).

Fig. 4 shows the results of the optimal design of MQLSM and CSRBFs, Fig. 5 illustrates their convergence curves, and Table 2 compares their performances.

In Fig. 4, the optimal structures of the two methods under different meshes have small differences, and both of them have mesh dependence. In Fig. 5, the variation patterns of the convergence curves of them are similar. Moreover, the fluctuation amplitude of MQLSM is smaller compared with the CSRBFs. Obviously, in Table 2, their number of iterations and objective function values are very close to each other. Those indicate that MQLSM has a similar convergence process with the CSRBFs. Since they have different approximation accuracies, their convergence processes cannot be identical. In Table 2, the approximation accuracy of MQLSM is higher for different meshes, which may be the reason why its convergence curve fluctuates less than CSRBFs.

In addition, their approximation accuracies both increase with the number of meshes, but the growth of MQLSM is more

significant. Moreover, the conditional number of the interpolation matrix of the CSRBFs increases with the influence domain which may lead to numerical instability. On the contrary, the MQLSM is more stable for numerical computation because it does not have to solve a system of linear equations.

In Table 2, the single iteration time of CSRBFs is about 2.5, 8.6 and 28.6 times that of MQLSM, respectively. This indicates that MQLSM has a very high computational efficiency.

5.3 Evolutionary stability of MQLSM-MQFEA

As shown in Fig. 6, consider the MBB beam subjected to a vertical force F = -1 at the middle point of the upper boundary. The design domain is discretized into a mesh of 400 × 100 = 40000 and the expected volume fraction is set to 50 %. The time step of each method in this section is taken as 0.5.

Fig. 7 shows the structural topology evolution of different methods, Fig. 8 shows the comparison of their convergence curves, and Table 3 demonstrates their computational efficiency. The values of Δ here are taken using both fixed and decreasing schemes, where Δ_0 denotes the initial value, $d\Delta$ denotes the decrease in each iteration, and $\min\Delta$ denotes the minimum value.

First of all, the topology optimization process (Fig. 7) and convergence curves (Fig. 8) of MQLSM-Ersatz are very similar to them of CSRBFs. This reaffirms the conclusion in Sec. 5.2 that MQLSM-Ersatz has a similar convergence process as CSRBFs. Due to the different approximation accuracies of the two parameterization methods, it is impossible to keep their structural topologies exactly the same with the dynamic changes of the evolution process [24].

In Fig. 7, CSRBFs and MQLSM-Ersatz are boundary-based methods that are more likely to produce finer components in the initial stage, and achieve the disappearance of finer components in the later stage using hole merging. On the contrary, MQLSM-MQFEA is similar to the density-based method in that the component sizes appear according to a pattern from coarse to fine, and this topological evolution seems to be more stable.

In Fig. 8, the convergence curve of MQLSM-MQFEA does not dramatically fluctuate in the initial stage, which indicates that this method is more stable and reduces the topology mutation.

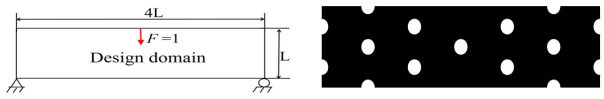

Fig. 6. MBB beam design domain (left) and initial design (right).

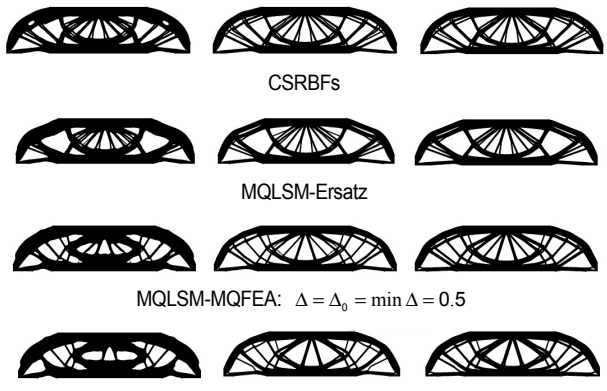

Fig. 7. Comparison of topology optimization process for 25 (left), 45 iterations (middle) and final design (right) of different methods.

5.4 Computational efficiency of MQLSM-MQFEA

Finally, a Michell structure with vertical force F = -1 at the bottom center is considered. Its design domain and initial design are shown in Fig. 9. The design domain is divided into a 300 × 150 = 45000 mesh with an expected volume fraction of 45 %. The time step of each method is taken as 0.01 in this section.

Fig. 10 shows the final design results of the different methods, Fig. 11 shows the comparison of their convergence curves, and Table 4 demonstrates their optimization performance.

In Fig. 10, the final designs of the various methods are similar. In Fig. 11, the various methods have very similar evolutionary processes at the initial stage.

In Tables 3 and 4, the single iteration time of MQLSM is much lower than that of CSRBFs. This indicates that MQLSM has very high computational efficiency. Unlike CSRBFs, which

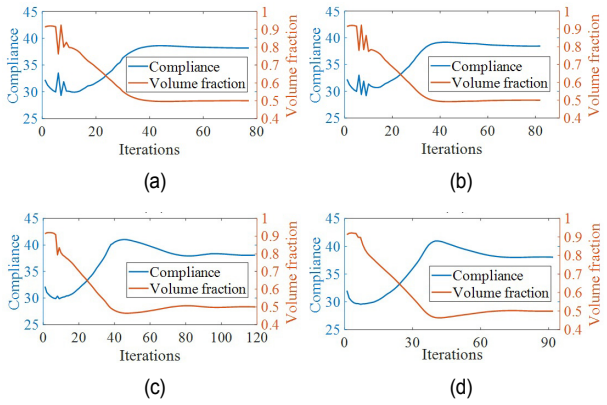

Fig. 8. Comparison of convergence curves of (a) CSRBFs; (b) MQLSM-Ersatz; (c) MQLSM-MQFEA: $\Delta=\Delta_0=\min\Delta=0.5$; (d) MQLSM-MQFEA: $\Delta_0=2, d\Delta=-0.01, \min\Delta=0.5$.

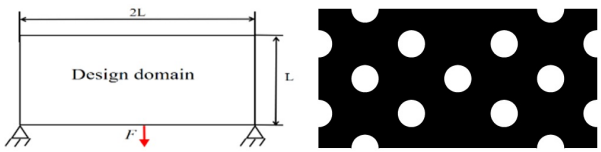

Fig. 9. Design domain (left) and initial design (right) of Michell structure.

Table 3. Comparison of computational performance of different methods.

	O	T/s	N
CSRBFs	38.19	189.87	77
MQLSM-Ersatz	38.45	2.54	82
MQLSM-MQFEA: $\Delta = \Delta_0 = \min \Delta = 0.5$	38.08	3.65	119
MQLSM-MQFEA: $\Delta_0 = 2,\ d\Delta = -0.01, \min \Delta = 0.5$	38.03	3.74	92

Table 4. Comparison of computational performance of different methods.

	O	T/s	N
CSRBFs	11.91	249.34	663
MQLSM-Ersatz	11.94	2.27	534
MQLSM-MQFEA: $\Delta = \Delta_0 = \min \Delta = 0.5$	12.07	3.11	359
MQLSM-MQFEA: $\Delta_0 = 2,\ d\Delta = -0.01, \min \Delta = 0.1$	11.99	3.19	667

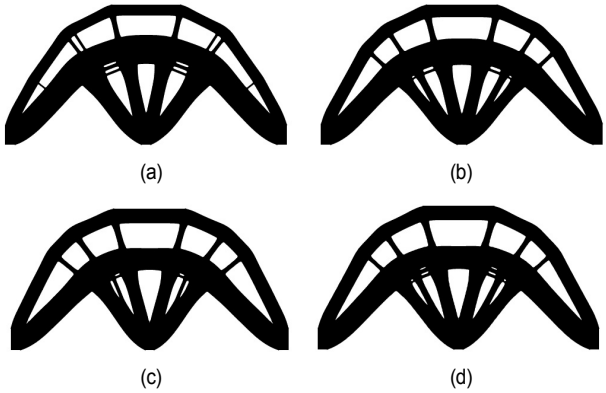

Fig. 10. Comparison of the final design of (a) CSRBFs; (b) MQLSM-Ersatz; (c) MQLSM-MQFEA: $\Delta = \Delta_0 = \min \Delta = 0.5$; (d) MQLSM-MQFEA: $\Delta_0 = 2$, $d\Delta = -0.01, \min \Delta = 0.1$.

need to be run on workstations with large memory footprint, the MQLSM can be run on ordinary personal computers, which greatly reduces the requirement of device memory.

Wei [39] argued that the LSBM can be regarded as a zero level set (or boundary-based evolution) model, when Δ is very small. When the value of Δ is large, the topology evolution is mainly driven by the density change, and it can be regarded as a density-based model. According to narrow-band theory [40], the method maps the LSF on the narrow-band around the zero level set (or boundary) to the element density. It is fully equivalent to the LSM only if width of the narrow-band (Δ) is very small.

In Tables 3 and 4, the single iteration time of MQFEA is slightly higher than that of the Ersatz material method, partly due to the fact that the computational cost of MQFEA is slightly higher than that of the linear interpolation method, and the fact that MQFEA is computed on a narrow band near the boundary instead of at the boundary points, which extends the computational range of intermediate densities.

Table 5. Comparison of computational performance of MQLSM-MQFEA for 3D cantilever beam.

	O	T/s	N
$\Delta = \Delta_0 = \min \Delta = 0.5$	13.25	7.42	133
$\Delta_0 = 2,\ d\Delta = -0.01,\ \min \Delta = 0.1$	13.27	7.50	142

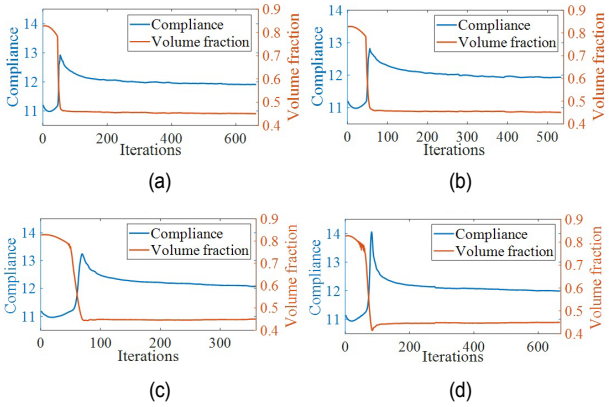

Fig. 11. Comparison of convergence curves of (a) CSRBFs; (b) MQLSM-Ersatz; (c) MQLSM-MQFEA: $\Delta = \Delta_0 = \min \Delta = 0.5$; (d) MQLSM-MQFEA: $\Delta_0 = 2, d\Delta = -0.01, \min \Delta = 0.1$.

It should be noted that in Figs. 8 and 11 the number of iterations of the methods using MQFEA in the early iteration is slightly higher than the Ersatz material method. Since the topological evolution is more drastic early in the iteration, slow iteration is beneficial to keep the evolution process stable. But, in Tables 3 and 4, the total number of iterations of MQFEA is not significantly higher than that of Ersatz. If Δ be taken to an appropriate value, such as $\Delta = \Delta_0 = \min \Delta = 0.5$ in Table 4, MQFEA can consume fewer iterations.

Indeed, choosing different step sizes at different evolutionary stages, such as gradually increasing the step size from early to late stages, can reduce the number of iterations while maintaining evolutionary stability. Interested readers can consult the Ref. [25].

In summary, MQLSM-MQFEA greatly improves computational efficiency without significantly increasing the number of iterations.

5.5 3D cantilever beam

The optimization process of a 3D cantilever beam problem is illustrated in Fig. 12. The left side of the beam is fixed and a vertical force F = -1 is applied at the midpoint of the bottom line of the right side. The structure is discretised by 60 × 30 × 10 elements, and the expected volume fraction is set to 35 %.The time step of each method is taken as 0.1 in this section. The parameters μ , $\gamma_{\max}$ and n_R are assigned the values of 80, 50 and 2.5, respectively. Fig. 13 illustrates convergence curves of MQLSM-MQFEA using both fixed and decreasing schemes, and Table 5 compares their performances.

Obviously, the MQLSM-MQFEA method is effective for 3D

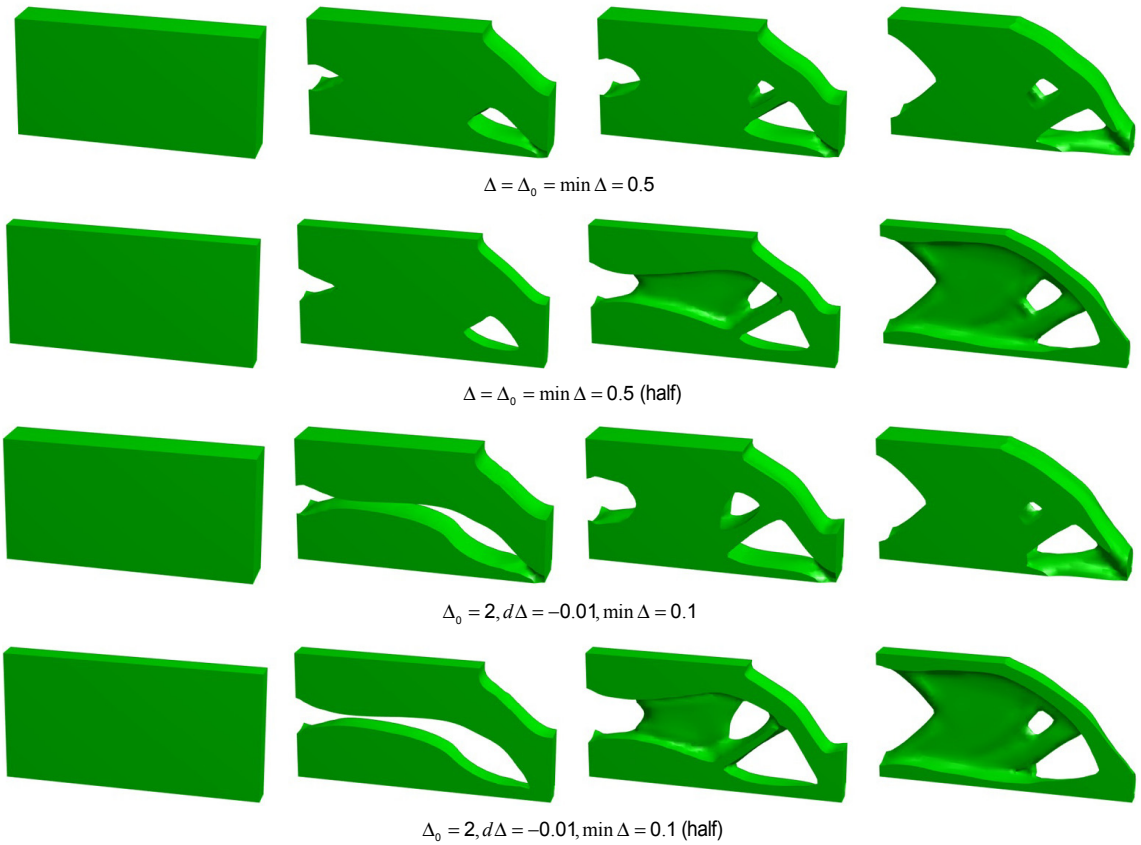

Fig. 12. Comparison of topology optimization process for 1, 15, 30 iterations and final design of MQLSM-MQFEA.

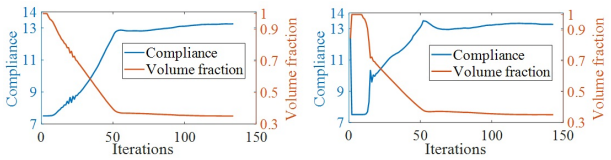

Fig. 13. Comparison of convergence curves of MQLSM-MQFEA: (a) $\Delta = \Delta_0 = \min \Delta = 0.5$; (b) $\Delta_0 = 2, d\Delta = -0.01, \min \Delta = 0.1$.

cases. In addition, no matter Δ is updated with a fixed or descending scheme, a reasonable topology structure can be obtained within an acceptable time and iterations, which indicates that MQLSM-MQFEA method can still maintain a high computational efficiency for 3D cases.

6. Conclusions

Based on the good approximation property of multivariate MQ quasi-interpolation to the original function and its generalized function, this paper proposes two approaches, MQLSM-Ersatz and MQLSM-MQFEA, the main conclusions are as follows:

1) The approximation accuracy of MQ quasi-interpolation method is higher than that of CSRBFs when the shape parameter takes a suitable small value.

2) The proposed methods occupy little memory and has high computational efficiency, which is suitable for large-scale mesh structural topology optimization problems.

3) Compared with CSRBFs, the MQLSM-MQFEA has higher evolutionary stability.

This paper does not test other spline functions such as B-splines, which will be the focus of the next research of the authors.

Acknowledgments

The authors gratefully acknowledge the financial support provided by the Science Foundation of Xi 'an Aeronautical Institute (Grant No. 2023KY1202) and the equipment support sponsored by the Big Data Laboratory of Xi 'an Aeronautical Institute.

References

[1] A. G. M. Michell, LVIII. The limits of economy of material in frame-structures, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 8 (47) (1904) 589-597.

[2] M. P. Bendsøe and N. Kikuchi, Generating optimal topologies in structural design using a homogenization method, *Computer Methods in Applied Mechanics and Engineering*, 71 (2) (1988)

197-224.
[3] M. P. Bendsøe, Optimal shape design as a material distribution problem, *Structural Optimization*, 1 (4) (1989) 193-202.
[4] D. J. Munk, G. A. Vio and G. P. Steven, Topology and shape optimization methods using evolutionary algorithms: a review, *Structural and Multidisciplinary Optimization*, 52 (3) (2015) 613-631.
[5] L. Xia, Q. Xia and X. Huang, Bi-directional evolutionary structural optimization on advanced structures and materials: a comprehensive review, *Archives of Computational Methods in Engineering*, 25 (2) (2018) 437-478.
[6] J. A. Sethian and A. Wiegmann, Structural boundary design via level set and immersed interface methods, *Journal of Computational Physics*, 163 (2) (2000) 489-528.
[7] M. Y. Wang, X. Wang and D. Guo, A level set method for structural topology optimization, *Computer Methods in Applied Mechanics and Engineering*, 192 (1-2) (2003) 227-246.
[8] T. Belytschko, S. P. Xiao and C. Parimi, Topology optimization with implicit functions and regularization, *International Journal for Numerical Methods in Engineering*, 57 (8) (2003) 1177-1196.
[9] S. Mukherjee, D. Lu and B. Raghavan, Accelerating largescale topology optimization: state-of-the-art and challenges, *Archives of Computational Methods in Engineering*, 28 (7) (2021) 1-23.
[10] S. Xu, J. Liu and B. Zou, Stress constrained multi-material topology optimization with the ordered SIMP method, *Computer Methods in Applied Mechanics Engineering*, 373 (2021) 113453.
[11] J. Yan, Q. Sui and Z. Fan, Multi-material and multiscale topology design optimization of thermoelastic lattice structures, *CMES-Computer Modeling in Engineering Sciences*, 130 (2) (2022) 967-986.
[12] T. Ben, P. Zhang and L. Chen, Vibration reduction of reactor using global magneto-structural topology optimization, *IEEE Transactions on Applied Superconductivity*, 32 (6) (2022) 1-5.
[13] O. Sigmund and K. Maute, Topology optimization approaches, *Structural Multidisciplinary Optimization*, 48 (6) (2013) 1031-1055.
[14] F. Gibou, R. Fedkiw and S. Osher, A review of level-set methods and some recent applications, *Journal of Computational Physics*, 353 (2018) 82-109.
[15] S. Wang and M. Y. Wang, Radial basis functions and level set method for structural topology optimization, *International Journal for Numerical Methods in Engineering*, 65 (12) (2006) 2060-2090.
[16] Z. Luo, M. Y. Wang and S. Wang, A level set-based parameterization method for structural shape and topology optimization, *International Journal for Numerical Methods in Engineering*, 76 (1) (2008) 1-26.
[17] H. Li, Z. Luo and L. Gao, An improved parametric level set method for structural frequency response optimization problems, *Advances in Engineering Software*, 126 (2018) 75-89.
[18] N. P. van Dijk, K. Maute and M. Langelaar, Level-set methods for structural topology optimization: a review, *Structural and Multidisciplinary Optimization*, 48 (3) (2013) 437-472.
[19] P. D. Dunning and H. A. Kim, Introducing the sequential linear programming level-set method for topology optimization, *Structural and Multidisciplinary Optimization*, 51 (3) (2015) 631-643.
[20] J. Hyun and H. A. Kim, Level-set topology optimization for effective control of transient conductive heat response using eigenvalue, *International Journal of Heat and Mass Transfer*, 176 (2021) 121374.
[21] H. Li, M. Yu and P. Jolivet, Reaction-diffusion equation driven topology optimization of high-resolution and feature-rich structures using unstructured meshes, *Advances in Engineering Software*, 180 (2023) 103457.
[22] Y. Liu, C. Yang and P. Wei, An ODE-driven level-set density method for topology optimization, *Computer Methods in Applied Mechanics and Engineering*, 387 (2021) 114159.
[23] C. D. Yang, J. H. Feng and Y. D. Shen, Step-size adaptive parametric level set method for structural topology optimization, *Journal of Mechanical Science and Technology*, 36 (10) (2022) 5153-5164.
[24] Y. Wang, Z. Kang and P. Liu, Velocity field level-set method for topological shape optimization using freely distributed design variables, *International Journal for Numerical Methods in Engineering*, 120 (13) (2019) 1411-1427.
[25] C. D. Yang, J. H. Feng and Y. D. Shen, Two interpolation matrix triangularization methods for parametric level set-based structural topology optimization, *International Journal of Computational Methods*, 19 (10) (2022) 2250024.
[26] Z. Sun and Y. Gao, A meshless quasi-interpolation method for solving hyperbolic conservation laws based on the essentially non-oscillatory reconstruction, *International Journal of Computer Mathematics*, 100 (6) (2023) 1303-1320.
[27] R. C. Mittal, S. Kumar and R. Jiwari, A cubic B-spline quasi-interpolation method for solving two-dimensional unsteady advection diffusion equations, *International Journal of Numerical Methods for Heat & Fluid Flow*, 30 (9) (2020) 4281-4306.
[28] R. C. Mittal, S. Kumar and R. Jiwari, A comparative study of cubic B-spline-based quasi-interpolation and differential quadrature methods for solving fourth-order parabolic PDEs, *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, 91 (3) (2021) 461-474.
[29] R. C. Mittal, S. Kumar and R. Jiwari, A cubic B-spline quasi-interpolation algorithm to capture the pattern formation of coupled reaction-diffusion models, *Engineering with Computers*, 38 (S2) (2022) 1375-1391.
[30] R. K. Beatson and M. J. D. Powell, Univariate multiquadric approximation: quasi-interpolation to scattered data, *Constructive Approximation*, 8 (3) (1992) 275-288.
[31] Z. M. Wu and R. Schaback, Shape preserving properties and convergence of univariate multiquadric quasi-interpolation, *Acta Mathematicae Applicatae Sinica*, 10 (4) (1994) 441-446.
[32] L. Ling, Multivariate quasi-interpolation schemes for dimension-splitting multiquadric, *Applied Mathematics and Computation*, 161 (1) (2005) 195-209.
[33] L. M. Ma and Z. M. Wu, Approximation to the *k*-th derivatives by multiquadric quasi-interpolation method, *Journal of Computational and Applied Mathematics*, 231 (2) (2009) 925-932.

[34] R. Tsai and S. Osher, Level set methods and their applications in image science, *Communications in Mathematical Sciences*, 1 (4) (2003) 623-656.
[35] H. Li, P. Li and L. Gao, A level set method for topological shape optimization of 3D structures with extrusion constraints, *Computer Methods in Applied Mechanics and Engineering*, 283 (2015) 615-635.
[36] G. Allaire, F. Jouve and A. M. Toader, Structural optimization using sensitivity analysis and a level-set method, *Journal of Computational Physics*, 194 (1) (2004) 363-393.
[37] P. Wei, Z. Li and X. Li, An 88-line matlab code for the parameterized level set method based topology optimization using radial basis functions, *Structural and Multidisciplinary Optimization*, 58 (2) (2018) 831-849.
[38] R. Beatson and L. Greengard, A short course on fast multipole methods, *Wavelets, Multilevel Methods and Elliptic PDEs*, 1 (1997) 1-37.
[39] P. Wei, W. Wang and Y. Yang, Level set band method: A combination of density-based and level set methods for the topology optimization of continuums, *Frontiers of Mechanical Engineering*, 15 (3) (2020) 390-405.
[40] J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, New York, USA (1999).

Chen-Dong Yang is currently a doctoral student of the School of Science, Chang'an University, Xi'an, China, and a lecturer of the School of Science, Aeronautical Institute, Xi'an, China. His research interests include optimum structural design and mathematical optimization.

Jian-Hu Feng is a Professor of the School of Science, Chang'an University, Xi'an, China. He received his Ph.D. in Aero Engine of Northwestern Polytechnical University Xi'an, China. His research interests include optimum structural design, computational fluid dynamics and high-performance computing technology for scientific and engineering problems.

Jiong Ren is a Lecturer of the School of Science, Aeronautical Institute, Xi'an, China. She received her Ph.D. in School of Aeronautics of Northwestern Polytechnical University Xi'an, China. Her research interests include optimum structural design, computational fluid dynamics and high-performance computing technology for scientific and engineering problems.

Ya-Dong Shen is a Lecturer in the School of Civil and Architectural Engineering, Nanyang Normal University, Nanyang, China. He received Ph.D. in School of Science, Chang'an University, Xi'an, China. His research interests include optimum structural design and foundation treatment.