Transactions of JASCOME Vol. 23 (Nov. 2023), paper No. 16-231124 JASCOME

A fast skeletonisation of 2D objects using a generalised double-layer potential and the $\mathcal{H}$-matrix method

Hiroshi ISAKARI[1)]

1) Keio University (〒 223-0061 3-14-1 Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa E-mail: isakari@sd.keio.ac.jp)

This paper proposes a fast numerical method to calculate the so-called topological skeleton of two-dimensional objects. The proposed method is based on a generalised double-layer potential that provides a smoothed singed distance field for an object given in a surface format and is thus considered CAD-friendly. To efficiently evaluate the skeleton, we here incorporate the $\mathcal{H}$-matrix method to calculate the potential. Numerical experiments show that, although sometimes the $\mathcal{H}$-matrix method might become unstable, the skeleton can be calculated with sufficient accuracy within acceptable computational time in many practical cases.

Key Words: Generalised Layer Potential, Topological Skeleton, $\mathcal{H}$-matrix method, Geometric Feature Extraction, Image Processing

1. Introduction

Topology optimisation [1, 2] has long been studied in academic and industrial communities. Beyond its original aim in the field of structural mechanics to provide light and stiff mechanical members, it is now applied to a variety of engineering fields such as acoustics [3, 4], fluid dynamics [5], electromagnetics [6], etc. Some emergence of useful commercial CAE software equipped with the topology optimisation capability [7] has also helped enhance its practical use in product developments in engineering industries. The scope of the topology optimisation application remains, however, limited to the conceptual design phase in most industries. Some trial and error modifications on topology-optimised design are still necessary before finalising the detailed design. This is partly because the current topology optimisation technologies may not provide a print-ready design that is manufacturable as is. For example, the topology optimisation maximising the stiffness sometimes gives a geometrically complicated design that includes extremely thin members. In such a case, the members need to be fleshed out to give a durable engineering product. In order for topology optimisation to truly revolutionise product development, the print-ready design needs to be addressed.

Some effort on this aspect has already been made by some researchers. The so-called overhang constraints for 3D printing and thickness control are especially addressed by many researchers, see e.g. [8, 9, 10] for early contributions. It may, however, be more convenient to handle various geometrical constraints (such as the overhang and closed cavity exclusion constraints for 3D printing, die-cutting capabilities, minimum or maximum local thickness constraints, etc) in a unified manner than to develop a separate method for each possible constraint. Some studies in this direction are found in literature including Allaire et al [11] and Yamada et al [12, 13, 14, 15, 16], the latter of which is particularly remarkable. All of these perform the topology optimisation while controlling the geometric feature. The Yamada method uses partial-differential equations (PDEs) whose solutions extend the normal vector field. Various geometric features can then be extracted from the field.

The normal vector extension can also be characterised as the gradient of the signed distance field. Various numerical methods for calculating the distance function have been studied for a long time, mainly in the field of image processing. Typical ones include those solving the eikonal equation [17, 18], some other PDEs [19], and variational problems [20]. All of these methods require recognising the target objects as images or characteristic functions and thus are essentially domain-based methods. On the other hand, especially in the three-dimensional case, the shape data is often given as surface data such as in the STL format. It may thus be preferable to use a method that converts the input surface data directly to the corresponding signed distance

15 Oct. 2023 received, 2 Nov. 2023 accept

field. For this purpose, some researchers used a kind of layer potential to calculate the signed distance [21, 22]. Belyaev et al proposed the signed L_p-distance field based on the generalised double-layer potential and established a methodology to calculate a highly accurate smoothed approximation for the signed distance function.

This paper extends the double-layer potential method [22] in two ways. One is to calculate the topological skeleton, which is one of the typical geometric features of an object. The other is an acceleration of the method. Belyaev et al proposed a method to naively calculate the potential. The original method thus requires $O(MN)$ arithmetic operations to evaluate the potential at M field points for an object whose surface is characterised by N segments. This study combines the $\mathcal{H}$-matrix method to accelerate the computation.

2. Formulations

2.1. Point in polygon and the double-layer potential

We first see that the double-layer potential for the Laplace equation with constant density is somehow related to a geometric feature of a given object. Here, we consider two-dimensional cases. As an example, let us consider an N-sided polygon Ω (Fig. 1). The surface of Ω is oriented, and its outward normal on the surface is denoted as n. The j^{th} edge of the polygon is denoted as s_j for $j = 1, \cdots, N$. In

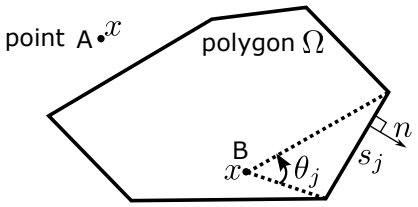

Fig. 1 Given 2D object Ω (a polygon in this example) and points A and B.

order to check if a given point $x \in \mathbb{R}^2$ is inside or outside Ω, we may use the following sum of the steradian:

$$\varphi_0(x) := \sum_{j=1}^{N} \theta_j(x), \tag{1}$$

where $\theta_j(x)$ is the steradian of s_j from the point x. $\varphi_0(x)$ returns the value of 2π if x is located in the polygon, and 0 if outside. It is easy to see that θ_j is nothing but the following double-layer potential of the two-dimensional Laplace equation with the constant density of 1:

$$\theta_j(x) = \int_{y \in s_j} \frac{(y-x)\cdot n}{|x-y|^2} \mathrm{d}s. \tag{2}$$

The double-layer potential

$$\int_{\partial\Omega} \frac{(y-x)\cdot n}{|x-y|^2} \mathrm{d}s \tag{3}$$

thus gives the characteristic function of the domain Ω.

2.2. Generalised double-layer potential and the singed L_p-distance field

Now, our goal is to generalise the double-layer potential (3) to provide an approximation of the signed-distance function defined as

$$\Psi(x) := \begin{cases} d(x, \partial\Omega) & x \in \Omega \\ -d(x, \partial\Omega) & x \notin \Omega \end{cases}, \tag{4}$$

with $d(x, \partial\Omega) := \min_{y\in\partial\Omega} |x - y|$. To this end, Belyaev et al proposed the L_p-distance field, which is briefly reviewed in this section. The detailed and mathematically rigorous discussion can be found in the original paper [22]. To derive the L_p-distance field, the following generalised double-layer potential (with unit density) is first introduced:

$$\varphi_p(x) := \int_{y\in\partial\Omega} \frac{(y-x)\cdot n}{|x-y|^{2+p}} \mathrm{d}s \tag{5}$$

where p is a positive integer. By introducing an appropriate coordinate system, in the case of $x \in \Omega$, the asymptotic behaviour of φ_p as $p \to \infty$ reads

$$\varphi_p(x) = \int_0^{2\pi} \frac{\mathrm{d}\theta}{|x-y|^p} \sim \frac{1}{\left(\min\limits_{y\in\partial\Omega} |x-y|\right)^p \sqrt{p}} \tag{6}$$

with the help of the Laplace method, provided that y is twice differentiable with respect to θ. In (6), θ represents the argument of $x - y$. Note that, in the case of $x \notin \Omega$, the mid-equation in (6) needs to be adjusted, but the final expression is valid also for this case. The p^{th} root of the reciprocal of the generalised double-layer potential (5) thus approaches the distance function as $p \to \infty$. With some more careful observation, one may find that

$$\Psi_p(x) := \left(\frac{c_p}{\varphi_p(x)}\right)^{\frac{1}{p}} \tag{7}$$

with the following sequence:

$$c_0 = \pi,\ c_1 = 2,\ c_{p+2} = \frac{p+1}{p+2} c_p \tag{8}$$

gives

$$\Psi_p(x) = d(x, \partial\Omega) + O\left(\frac{1}{p}\right) \tag{9}$$

as $p \to \infty$, which has a better convergence than $(1/\varphi_p(x))^{1/p}$ to the distance function.

To augment the distance function (9) by the $\pm$ sign to obtain the signed distance field (4), we may use the "point in polygon" algorithm presented in Section 2.1.

In evaluating the signed L_p-distance field, it is essentially important to accurately evaluate the generalised double-layer potential (5). Note that, since its integrand can be nearly

singular when $x \in \Omega$ is close to the boundary $\partial\Omega$, naive numerical quadratures cannot be used to evaluate the potential. Instead, we here derive the analytical expression for the boundary integral (5). It suffices to consider the following potential generated by an oriented line segment:

$$\int_0^\gamma \frac{\mathrm{d}\theta}{|x - y(\theta)|^p}, \tag{10}$$

at a given point $x \in \Omega$, where γ is the steradian at x of the segment, and y is a point on the segment and is a function of θ. It can easily be found that, when p is odd, the integral (10) can be expressed as

$$\varphi_{2n-1}(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{C_{ij}^n}{(ab)^{j-1}} \left(\frac{1}{a^{2n+1-j}} + \frac{1}{b^{2n+1-j}} \right) t^{2i-1} \tag{11}$$

for $p = 2n-1$, where t is defined as $t := \tan(\gamma/2)$, a (resp. b) is the distance between x and the starting (resp. ending) point of the target line segment, and $C_{ij}^n \in \mathbb{Q}$ is the coefficient. In our implementation, the coefficients $C^n \in \mathbb{R}^{n\times n}$ (for $n = 1, 2, \cdots$) are precomputed by Maple. C^5, for example, is given as

$$C^5 = \begin{pmatrix} 1/5 & 1/5 & 1/5 & 1/5 & 1/5 \\ 2/15 & 2/5 & 3/5 & 11/15 & 4/5 \\ 2/35 & 2/7 & 23/35 & 1 & 6/5 \\ 1/70 & 1/10 & 11/35 & 3/5 & 4/5 \\ 1/630 & 1/70 & 2/35 & 2/15 & 1/5 \end{pmatrix}. \tag{12}$$

To illustrate the singed L_p-distance fields, we here calculate $\Psi_p(x)$ for a unit circle. Fig. 2 shows $\Psi_5(x)$, $\Psi_{15}(x)$, and $\Psi_{25}(x)$ for $\{x \mid -2 < x_1 < 2,\ x_2 = 0\}$ as well as the exact signed distance function $\Psi(x) = 1 - |x_1|$. For the computation, the surface of the circle is approximated by an inscribed polygon of it with 2000 edges, and the signed L_p-distance fields are evaluated at 500 points of equal intervals on the given line. From the figure, one finds that Ψ_p

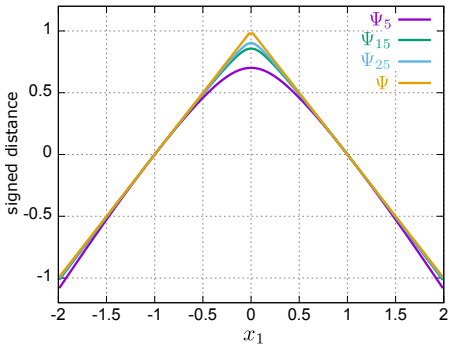

Fig. 2 Signed L_p-distance field for a unit circle.

approaches the exact signed distance Ψ as p becomes large. Although a relatively large difference between Ψ_p and Ψ is observed away from the boundary $|x_1| = 1$, Ψ_p agrees well with Ψ even when p is small in the vicinity of the boundary. To see the convergence, we also show the relative ℓ_1-error of Ψ_p on the 500 points against the exact signed distance vs p in Fig. 3. As expected from the estimate (9), the relative error scales with $1/p$.

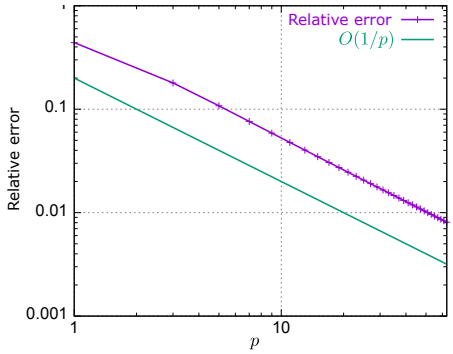

Fig. 3 Relative error of Ψ_p against the exact signed distance vs p.

2.3. Skeleton computation

One may observe from Fig. 2 that the signed L_p-distance field is smoothened at the centre of the circle $x_1 = 0$ at which the gradient of Ψ does not exist. We may thus evaluate the gradient of Ψ_p everywhere. Since Ψ_p approximates the signed distance, it is expected that $|\nabla\Psi_p| = 1$ holds* except where $\nabla\Psi$ does not exist. Indeed, Fig 4 showing $1 - |\nabla\Psi_p|$ vs x_1 for the unit circle case indicates that $|\nabla\Psi_p| \sim 1$ holds except in the vicinity of $x_1 = 0$. On the other hand, it is

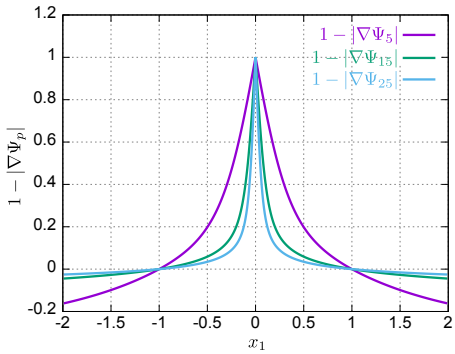

Fig. 4 $1 - |\nabla\Psi_p|$ for a unit circle.

known that the set of points where $\nabla\Psi$ does not exist corresponds to the topological skeleton that is defined as a set of points x such that there exist at least 2 points y satisfying $\min_{y\in\partial\Omega} |x - y|$. It is thus inferred that the topological skeleton can be recovered as e.g. $\{x \mid 1 - |\nabla\Psi_p(x)| > \varepsilon\}$, where ε is a given positive parameter. Note that there exist several definitions of the skeleton. See e.g. Allaire et al [11]

*Recall that the gradient of the signed distance field on the boundary agrees with the unit outward normal.

and the references therein for other definitions and further discussions.

Note also that we may use the automatic differentiation to evaluate the gradient of the signed L_p-distance. In our implementation, we used a Fortran 95 library for the automatic differentiation of the forward mode [23] for this purpose.

2.4. $\mathcal{H}$-matrix-based acceleration

Evaluating the signed L_p-distance field (7) and its gradient (for skeleton computation) with the generalised double-layer potential (5) at M field points for a given N-sided polygon requires $O(MN)$ arithmetic operations. Since this computation can be viewed as matrix-vector multiplication of an $\mathbb{R}^{M\times N}$ matrix and $\mathbb{R}^N$ vector of ones, and the matrix has a hierarchical block structure with rank-deficient blocks, we may use the $\mathcal{H}$-matrix method [24] or the fast-multipole method [25] to accelerate this computation. We here adopt the former one for its simple implementation. In the standard $\mathcal{H}$-matrix method, the target matrix is hierarchically subdivided into subblocks in a binary tree format until its row and column size is less than a given parameter $n_{\min}$. The submatrix of the target matrix is then low-rank approximated if the submatrix satisfies the following so-called admissible condition:

$$\min\{\mathrm{diam}(t), \mathrm{diam}(s)\} \le \eta\, \mathrm{dist}(t,s), \tag{13}$$

where t and s respectively denote the subset of the field points and the boundary elements, $\mathrm{diam}(a)$ indicates the size of the bounding box of a, and $\mathrm{dist}(s,t)$ the distance between t and s, and $\eta > 0$ is a parameter. Using larger η means that more submatrices shall be low-rank approximated. Note that the larger p brings the more severe singularity in the double-layer potential (5). The smaller η in (13) would, therefore, be necessary for the larger p. Also note that the sole use of the admissible condition (13) often ends up with a small number (say 4 or 8) of huge subblocks to be low-rank approximated, which may degrade the parallel computing performance. To avoid such a situation, we augment the admissible condition by the following ones:

$$|t| < n_{\max} \text{ and } |s| < n_{\max}, \tag{14}$$

where $n_{\max} > n_{\min}$ is a given parameter, and $|a|$ is the number of elements in a.

For the low-rank approximation, we use the standard adaptive cross approximation (ACA).

3. Numerical experiments

3.1. Demonstration

We first demonstrate an illustrative numerical example of extracting the topological skeleton of a given 2D object with complicated geometry. Fig. 5 (a) shows the given object. Here, the boundary data similar to the STL one (or the boundary element mesh) of the black object is given as the input. The surface of the object is parameterised by 37999 line segments. For this example, we calculated the signed L_p-distance field (7) for $p = 15$ at the 120000 ($= 300 \times 400$) lattice points distributed in $\{x \mid |x_1| < 0.75,\ |x_2| < 1.00\}$ by the proposed method accelerated by the $\mathcal{H}$-matrix method. The parameters for the $\mathcal{H}$-matrix method is empirically set as $n_{\min} = 64$, $n_{\max} = 1024$, $\eta = 0.516$, $\varepsilon_{\mathrm{ACA}} = 10^{-4}$, where $\varepsilon_{\mathrm{ACA}}$ is the given tolerance for the adaptive cross approximation. Fig. 5 (b), (c), and (d) show Ψ_{15}, $\nabla\Psi_{15}$, and $1-|\nabla\Psi_{15}|$, respectively. From the figure, we may confirm at least qualitatively that we can extract the geometric features of the given object. We then discuss the timing. The elapsed

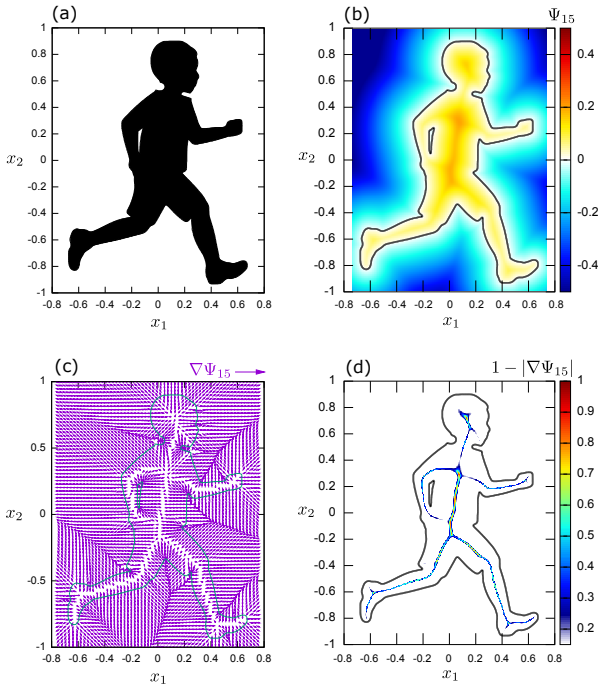

Fig. 5 (a) Given 2D object, (b) signed L_p-distance field of 15^{th} degree, (c) its gradient, and (d) the skeleton recovered by $1-|\nabla\Psi_{15}| > 0.2$.

time to compute Ψ_{15} and its gradient was 462 sec with the $\mathcal{H}$-matrix method, while 830 sec without it. The computation was carried out on a desktop PC with Xeon Platinum 8360Y (36 cores and 72 threads). We may, therefore, conclude that the proposed method does accelerate the Belyaev method [22] by the $\mathcal{H}$-matrix method.

3.2. Validation

In this subsection, to quantitatively evaluate the performance of the proposed method, we show the test results for a simple geometry. As a test object, we here use a rectangular-shaped object $\{x \mid |x_1| < 0.75,\ |x_2| < 0.15\}$ and

evaluate the singed L_p-distance fields at the equally spaced 40000 ($= 200 \times 200$) points allocated in $\{x \mid |x_{1,2}| < 1.00\}$. $n_{\min} = 64$ and $n_{\max} = 1024$ are again used for this example. The other parameters for the $\mathcal{H}$-matrix method are empirically set as $\eta = 2/p^{1/4}$, $\varepsilon_{\rm ACA} = \min(10^{-4},\ 0.1^{\sqrt{p}})$. We

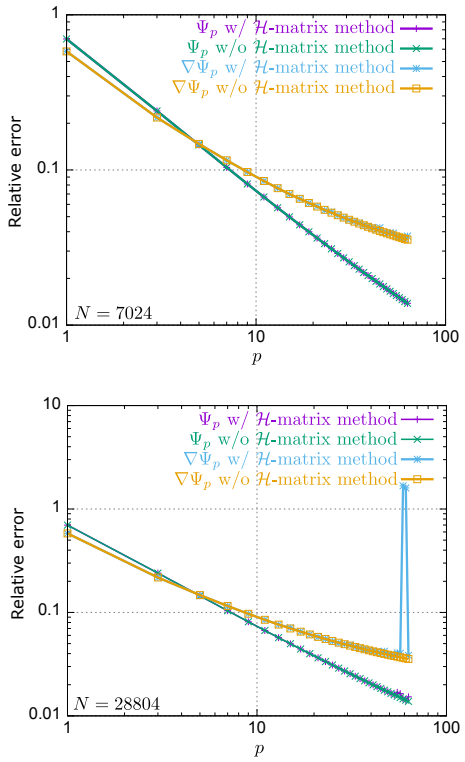

Fig. 6 Relative error in signed L_p-distance and its gradient to the exact ones for the case of $N = 7024$ boundary elements (top) and $N = 28804$ (bottom).

first discuss the accuracy. Fig. 6 shows the relative ℓ_1-error in L_p-distance field and its gradient to their exact distance counterparts evaluated at the lattice points against p for the case of $N = 7024$ and $N = 28804$, where N is the number of boundary elements for the rectangular object. Note that the generalised double-layer potential for this case can accurately be evaluated with $N = 4$, i.e. a single boundary element on each rectangular edge gives the exact potential (3) except for the rounding error. We here intentionally use, however, excessively large N to evaluate the performance of the $\mathcal{H}$-matrix method. Although it is expected that the accuracy may degrade for this example with corners in the object surface (see the discussion in Section 2.2), the relative error for the distance field almost scales as $O(1/p)$. The accuracy of the gradient is somewhat worse than that of the distance field itself**. Nonetheless, the relative error for the gradient is less than 10% if $p > 10$ is used.

One also observes that when p and N are large, the $\mathcal{H}$-matrix method fails to give the accurate Ψ_p and $\nabla\Psi_p$. This tendency is more severely observed in $\nabla\Psi_p$ than in Ψ_p itself. This is obviously caused by the ill-conditioned kernel for large p (see (5)). We might not use the ACA to evaluate such an ill-conditioned potential for large p. We would like, however, to emphasise that in practical use $p \sim 10$ would be enough in skeletonising an object (see Section 3.1). We thus may conclude that the proposed method with the $\mathcal{H}$-matrix method is still promising in skeletonising 2D objects in practical cases.

We lastly discuss the computational efficiency of the proposed method. Fig. 7 shows the elapsed time for computing Ψ_p and its gradient for various p and $N = 7024$. The proposed method is always faster than the conventional method without $\mathcal{H}$-matrix acceleration among tested $p = 1,\ 3, \cdots, 63$. The plot for the proposed method is steeper than that for the conventional method for large p, though. This is also caused by the slow convergence in ACA due to the ill-conditioned kernel.

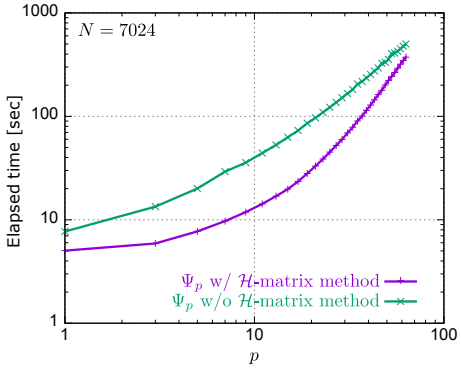

Fig. 7 Elapsed time for computing Ψ_p vs the degree p of the L_p-distance.

From the above observations, we conclude that the proposed method works well at least when the degree of L_p-distance field is moderate, say $p \sim 10$.

4. Concluding remarks

In this paper, we proposed an extension of the generalised double-layer potential method for the signed L_p-distance field evaluation [22] to skeletonise 2D objects. The proposed method only relies on the boundary data of given objects and is thus able to convert directly from, e.g., STL data to

**The distance function appears to be calculated with good accuracy, but in the lower right corner of Fig. 7 one may find a slight jump in the purple line (representing the error in Ψ_p evaluated with the $\mathcal{H}$-matrix method). Although the increase in the error is small, it is considerably large compared to the given tolerance for ACA (e.g. $\varepsilon_{\rm ACA} \simeq 1.8 \times 10^{-8}$ for $p = 60$) and cannot be ignored.

the approximation of its skeleton. We found that the proposed method can provide the topological skeleton even with a small p. We also addressed its acceleration by the $\mathcal{H}$-matrix method. We confirmed that the $\mathcal{H}$-matrix method can accelerate the skeleton computation with the condition that p is small added. On the other hand, the stable and fast L_p-distance computation for large p is left for future works. To address this, we may try the fast-multipole method instead of the $\mathcal{H}$-matrix method. It may also be possible to improve the ACA algorithm for the current kernel. To this end, we might exploit the fact that the generalised double-layer potential (5) can have a tiny value for the case that both p and $|x - y|$ are large. We might also, instead of using the automatic differentiation, manually differentiate the double-layer potential and take appropriate measures to avoid the loss of significant digits to improve the accuracy in $\nabla\Psi_p$. It may also be an interesting future direction to combine the proposed method with the topology and shape optimisation technologies to optimise engineering designs with geometric constraints.

References

(1) M-P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, Vol. 71, No. 2, pp. 197–224, 1988.

(2) T. Yamada, K. Izui, S. Nishiwaki, and A. Takezawa. A topology optimization method based on the level set method incorporating a fictitious interface energy. *Computer Methods in Applied Mechanics and Engineering*, Vol. 199, No. 45, pp. 2876–2891, 2010.

(3) E. Wadbro and M. Berggren. Topology optimization of an acoustic horn. *Computer Methods in Applied Mechanics and Engineering*, Vol. 196, No. 1, pp. 420–436, 2006.

(4) H. Isakari, K. Kuriyama, S. Harada, T. Yamada, T. Takahashi, and T. Matsumoto. A topology optimisation for three-dimensional acoustics with the level set method and the fast multipole boundary element method. *Mechanical Engineering Journal*, Vol. 1, No. 4, CM0039, 2014.

(5) K. Yaji, T. Yamada, M. Yoshino, T. Matsumoto, K. Izui, and S. Nishiwaki. Topology optimization using the lattice Boltzmann method incorporating level set boundary expressions. *Journal of Computational Physics*, Vol. 274, pp. 158–181, 2014.

(6) F. Lucchini, R. Torchio, V. Cirimele, P. Alotto, and P. Bettini. Topology optimization for electromagnetics: A survey. *IEEE Access*, Vol. 10, pp. 98593–98611, 2022.

(7) Topology Optimization (SOLIDWORKS). `https://www.solidworks.com/media/topology-optimization`.

(8) L. Harzheim and G. Graf. A review of optimization of cast parts using topology optimization. *Structural and Multidisciplinary Optimization*, Vol. 31, No. 5, pp. 388–399, 2006.

(9) A-T. Gaynor, N-A. Meisel, C-B. Williams, and J-K. Guest. Topology optimization for additive manufacturing: Considering maximum overhang constraint. In *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2014.

(10) S. Galjaard, S. Hofman, and S. Ren. New Opportunities to optimize structural designs in metal by using additive manufacturing. In *Advances in Architectural Geometry 2014*, pp. 79–93, 2015.

(11) G. Allaire, F. Jouve, and G. Michailidis. Thickness control in structural optimization via a level set method. *Structural and Multidisciplinary Optimization*, Vol. 53, No. 6, pp. 1349–1382, 2016.

(12) T. Yamada. Geometric shape features extraction using a steady state partial differential equation system. *Journal of Computational Design and Engineering*, Vol. 6, No. 4, pp. 647–656, 2019.

(13) T. Yamada and Y. Noguchi. Topology optimization with a closed cavity exclusion constraint for additive manufacturing based on the fictitious physical model approach. *Additive Manufacturing*, Vol. 52, 102630, 2022.

(14) T. Miki and T. Yamada. Topology optimization considering the distortion in additive manufacturing. *Finite Elements in Analysis and Design*, Vol. 193, 103558, 2021.

(15) D-Y. Hur, Y. Sato, T. Yamada, K. Izui, and S. Nishiwaki. Level-set based topology optimization considering milling directions via fictitious physical model. *Mechanical Engineering Journal*, Vol. 7, No. 6, 20–00226, 2020.

(16) Y. Sato, T. Yamada, K. Izui, and S. Nishiwaki. Manufacturability evaluation for molded parts using fictitious physical models, and its application in topology optimization. *The International Journal of Advanced Manufacturing Technology*, Vol. 92, No. 1, pp. 1391–1409, 2017.

(17) J-A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the Na-*

tional Academy of Sciences, Vol. 93, No. 4, pp. 1591–1595, 1996.

(18) J. Qian, Y‑T. Zhang, H‑K. Zhao. Fast sweeping methods for eikonal equations on triangular meshes. *SIAM Journal on Numerical Analysis*, Vol. 45, No. 1, pp. 83–107, 2007.

(19) K. Crane, C. Weischedel, and M. Wardetzky. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Trans. Graph.*, Vol. 32, No. 5, 2013.

(20) A. Belyaev and P-A. Fayolle. An ADMM-based scheme for distance function approximation. *Numerical Algorithms*, Vol. 84, No. 3, pp. 983–996, 2020.

(21) N. Ahuja and J-H. Chuang. Shape representation using a generalized potential field model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 2, pp. 169–176, 1997.

(22) A. Belyaev, P-A Fayolle, and A. Pasko. Signed L_p-distance fields. *Solid and Physical Modeling 2012*, Vol. 45, No. 2, pp. 523–528, 2013.

(23) G.M. von Hippel. TaylUR 3, a multivariate arbitrary-order automatic differentiation package for Fortran 95. *Computer Physics Communications*, Vol. 181, No. 3, pp. 705–706, 2010.

(24) J. Ostrowski, Z. Andjelic, M. Bebendorf, B. Cranganu-Cretu, and J. Smajic. Fast BEM-solution of Laplace problems with H-matrices and ACA. *IEEE Transactions on Magnetics*, Vol. 42, No. 4, pp. 627–630, 2006.

(25) V Rokhlin. Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, Vol. 60, No. 2, pp. 187–207, 1985.