Adaptive discrete thin plate spline smoother

L. Fang[1] L. Stals[2]

(Received 5 November 2020; revised 27 August 2021)

Abstract

The discrete thin plate spline smoother fits smooth surfaces to large data sets efficiently. It combines the favourable properties of the finite element surface fitting and thin plate splines. The efficiency of its finite element grid is improved by adaptive refinement, which adapts the precision of the solution. It reduces computational costs by refining only in sensitive regions, which are identified using error indicators. While many error indicators have been developed for the finite element method, they may not work for the discrete smoother. In this article we show three error indicators adapted from the finite element method for the discrete smoother. A numerical experiment is provided to evaluate their performance in producing efficient finite element grids.

Contents

DOI:10.21914/anziamj.v62i0.15979, Published 2021-11-05, as part of the Proceedings of the 19th Biennial Computational Techniques and Applications Conference. ISSN 1445-8810. (Print two pages per sheet of paper.)

1 Introduction

Data fitting and smoothing is an important tool to model and analyse data in scientific and engineering communities [7, 9]. The thin plate spline is a type of data fitting technique that is insensitive to noise in data. However, it becomes computationally expensive as the data size increases. The discrete smoother was developed to fit large data sets efficiently while retaining the smoothing properties of the thin plate spline [8].

Adaptive refinement further improves the discrete smoother's efficiency. It adapts the precision of the solution iteratively within sensitive regions indicated by an error indicator and reduces the number of nodes required to achieve a certain accuracy. Traditional error indicators for approximating partial differential equations will not always work for the discrete smoother as its equations are formed using the observed data that may be perturbed by noise.

In this article, we show three error indicators adapted for the discrete smoother. Their performance is compared through a numerical experiment with two-dimensional model problems. We give more background information on the discrete smoother in Section 2. Section 3 presents the iterative adaptive refinement process of the discrete smoother. Section 4 provides details of the three error indicators. Section 5 displays and analyses the results of the numerical experiment for these error indicators.

2 Discrete thin plate spline smoother

The discrete smoother was developed as a first-order finite element approximation of the thin plate spline [8]. More details of the thin plate spline are provided by Bookstein [1]. Suppose the observed data consists of n data points $\{(\mathbf{x}_{(i)}, y_{(i)}) : i = 1, 2, \ldots, n\}$, where $\mathbf{x}_{(i)} \in \mathbb{R}^d$ is a predictor value and $y_{(i)} \in \mathbb{R}$ is a response value. The discrete smoother $s(\mathbf{x})$ is represented as a combination of piecewise linear basis functions $s(\mathbf{x}) = \mathbf{b}(\mathbf{x})^T\mathbf{c}$, where $\mathbf{b}(\mathbf{x})^T = [b_1(\mathbf{x}), \ldots, b_m(\mathbf{x})]$ are m basis functions, $\mathbf{c}^T = [c_1, \ldots, c_m]$ are the corresponding coefficients and m is the number of nodes in the finite element grid. The resulting system of equations is sparse and the size depends only on the number of nodes in the grid [11].

Since piecewise linear basis functions are not defined for the minimiser of the thin plate spline, auxiliary functions $\mathbf{u}(\mathbf{x}) = [\mathbf{b}(\mathbf{x})^T\mathbf{g}_1, \ldots, \mathbf{b}(\mathbf{x})^T\mathbf{g}_d]^T$ are introduced to represent the gradient of s, where $\mathbf{g}_1, \ldots, \mathbf{g}_d$ are coefficients of the gradient approximation. The discrete smoother s also needs to satisfy an additional constraint

$$\int_\Omega \nabla s(\mathbf{x}) \cdot \nabla b_j(\mathbf{x})\, d\mathbf{x} = \int_\Omega u(\mathbf{x}) \cdot \nabla b_j(\mathbf{x})\, d\mathbf{x}$$

so ∇s and $\mathbf{u}$ are equivalent in a weak sense.

The discrete formulation of the minimiser becomes

$$J_\alpha(\mathbf{c}, \mathbf{g}_1, .., \mathbf{g}_d) = \mathbf{c}^T A\mathbf{c} - 2\mathbf{d}^T\mathbf{c} + y^T y/n + \alpha \sum_{k=1}^{d} \mathbf{g}_k^T L \mathbf{g}_k\,, \tag{1}$$

subject to Constraint

$$L\mathbf{c} = \sum_{k=1}^{d} G_k \mathbf{g}_k\,, \tag{2}$$

for $A = \frac{1}{n}\sum_{i=1}^{n} \mathbf{b}(\mathbf{x}_{(i)})\mathbf{b}(\mathbf{x}_{(i)})^T$, $\mathbf{d} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{b}(\mathbf{x}_{(i)})y_{(i)}$, $y = [y_{(1)}, \ldots, y_{(n)}]^T$, L a discrete approximation to the negative Laplacian, G_k a discrete approximation to the gradient operator and α a smoothing parameter. Stals [11]

provides a derivation of the discrete smoother from the thin plate spline. We solve Minimiser (1) using Lagrange multipliers and the size of the resulting system of equations depends only on the number of nodes in the grid. This allows the discrete smoother to interpolate large data sets efficiently.

3 Adaptive refinement

The accuracy of the discrete smoother depends on several factors, including the mesh size h of its finite element grid [8]. The solution is improved by a finer grid with smaller h, although this leads to a larger system of equations with increasing computational cost and memory requirement. Refinement is the process of iteratively resolving the problem of interest with finer grids to increase the accuracy of the solution. Adaptive refinement is one approach that concentrates refinement in certain regions to achieve the required accuracy [6]. It allows the discrete smoother to obtain satisfactory accuracy with fewer nodes than a uniform grid.

The iterative adaptive refinement process begins by building the discrete smoother on an initial coarse grid. The error indicator evaluates and indicates regions with large errors for adaptive refinement. This process is executed iteratively until the discrete smoother reaches a satisfactory error tolerance. As the optimal value of the smoothing parameter α in Minimiser (1) may vary with the mesh size, the optimal value must be updated each time the grid is refined. We calculate α using a stochastic approximation of the generalised cross-validation method [5, 8].

The stopping criteria of this iterative process are based on the rate of change of the discrete smoother's root mean square error (RMSE). This differs from the approach taken by traditional error indicators as we are fitting a smoothing spline as opposed to approximating partial differential equations. The RMSE may not be reduced to a required error tolerance value when the data is perturbed by noise [3]. In our implementation, the iterative process terminates when the RMSE value is reduced less than 10% by one iteration of refinement

for two consecutive iterations. The finite element grid generated before these two iterations is chosen to be the optimal grid.

4 Error indicators

The error indicator marks elements with large errors during the iterative adaptive refinement process. Traditional error indicators were developed for approximating partial differential equations and they face new challenges when being applied to the discrete smoother. The error convergence of the discrete smoother depends on several factors, including the smoothing parameter α, the maximum distance between data points d_x and mesh size h [8]. A smaller h value does not necessarily reduce the error of the discrete smoother [4].

In this article, we focus on two-dimensional grids with triangular elements and refine elements using the newest node bisection [6]. Therefore, the error indicators indicate the error on triangle pairs to determine whether they are refined. We describe three error indicators adapted for the discrete smoother, which are the auxiliary problem error indicator, recovery-based error indicator and norm-based error indicator. The auxiliary problem error indicator and norm-based error indicator were studied previously by Fang [4]. Additional modifications have since been made to improve the accuracy of the auxiliary problem error indicator.

The *auxiliary problem error indicator* computes the error indicator value by comparing the discrete smoother with a locally more accurate approximation. It is built by solving an auxiliary problem that minimises

$$J_\alpha(\widehat{c}, \widehat{g}_1, \ldots, \widehat{g}_d) = \frac{1}{\widehat{n}} \sum_{i=1}^{\widehat{n}} \left(\widehat{s}(x_{(i)}) - y_{(i)}\right)^2 + \widehat{\alpha} \int_{\widehat{\Omega}} \sum_{|\nu|=2} (D^\nu \widehat{s}(x))^2 \, dx,$$

$$\widehat{s} = s \quad \text{on } \partial\widehat{\Omega}_D,$$

where $\widehat{s}$ is the local approximation with coefficients $\widehat{c}, \widehat{g}_1, \ldots, \widehat{g}_d$ on a local domain $\widehat{\Omega}$, $\widehat{X} = \{(x_{(i)}, y_{(i)}) : i = 1, 2, \ldots, \widehat{n}\}$ is the data inside $\widehat{\Omega}$ and $\partial\widehat{\Omega}_D$ is

the Dirichlet boundary. The Dirichlet boundary conditions are defined using the approximations of the discrete smoother s, including $\mathbf{c}$ and $\mathbf{g}_1, \ldots, \mathbf{g}_d$ values. We choose Dirichlet boundaries as they incorporate effects from the discrete smoother to smooth noise and produce more stable local approximations. The local domain $\widehat{\Omega}$ is a union of the triangle pair and its neighbouring elements and we improve the accuracy of $\widehat{\mathsf{s}}$ by refining the triangle pair. While a local domain $\widehat{\Omega}$ with more elements or further refinement may improve the accuracy of $\widehat{\mathsf{s}}$, we choose to avoid it for a more efficient error indicator. The error indicator value is calculated using an energy norm of the difference between s and $\widehat{\mathsf{s}}$.

The local smoothing parameter $\widehat{\alpha}$ is another factor that affects the accuracy of $\widehat{\mathsf{s}}$. In previous work, Fang [4] used the smoothing parameter α of s for $\widehat{\mathsf{s}}$ for all auxiliary problems. However, the α value may decrease as the grid gets finer during the iterative process and a small $\widehat{\alpha}$ makes $\widehat{\mathsf{s}}$ more sensitive to noise in data. Moreover, the effects of noise increase when the number of data points decreases and the range of the response values decreases. We may calculate the $\widehat{\alpha}$ values using the generalised cross-validation method for each $\widehat{\mathsf{s}}$ but it is too computationally expensive and unstable. Considering that calculated $\widehat{\alpha}$ values of auxiliary problems often distribute in a small range, we choose the median of these values as the optimal $\widehat{\alpha}$ for all auxiliary problems. When the error indicator needs to indicate errors for a large number of auxiliary problems, we will randomly select ten auxiliary problems to obtain the median of optimal $\widehat{\alpha}$ values with a lower computational cost.

The *recovery-based error indicator* post-processes the discontinuous gradients $\mathsf{D}^1\mathsf{s}$ of the discrete smoother to estimate the error [12]. The discrete smoother is represented as a combination of piecewise linear basis functions and discontinuities of gradients occur at the boundary of elements. The true error is estimated by comparing the the post-processed and current gradient of the discrete smoother. Zienkiewicz and Zhu [12] post-processed

the gradient $\widehat{D}^1 s_i$ on the jth node by solving a system of equations

$$\sum_{j=1}^{m} \int_\Omega b_k b_j \widehat{D}^1 s_j \, d\Omega = \int_\Omega b_k D^1 s \, d\Omega, \quad k = 1, \ldots, m.$$

The post-processed gradient $\widehat{D}^1 s$ is assembled using the post-processed gradients $\widehat{D}^1 s_j$ in each node and the error indicator value is estimated by

$$\|e\|_E^2 \approx \int_\Omega (\widehat{D}^1 s - D^1 s)^2 \, d\Omega.$$

The *norm-based error indicator* is based on an error bound on the L_∞ norm of the discrete smoother [10]. The accuracy of the discrete smoother is limited by the approximation capacity of the approximating space. Sewell [10] suggested that the error of a two-dimensional finite element solution with piecewise linear basis functions is bounded by

$$\int_{t_i} D^2_{\max} u(x_1, x_2) \, dx = \int_{t_i} \max_{i+j=2} |\partial^2 u(x_1, x_2)/\partial x_1^i x_2^j| \, dx,$$

where t_i is ith triangle and u is the model function of a smooth problem. Therefore, for a near-optimal grid, the integral value should be approximately equal for all the elements. This error indicator indicates elements with large integral values as large errors to ensure the integral value is evenly distributed to all elements. It identifies regions where the solution changes rapidly and uses finer elements to achieve the required accuracy in those regions. Even though $D^2_{\max} u$ cannot be calculated exactly without u, we can approximate it using the discrete smoother [4].

5 Numerical experiment

In this section, we evaluate the performance of the three error indicators on two-dimensional model problems. The numerical results on more model

problems are presented by Fang [3]. The response values of two model problems, exponential and Franke functions, are modelled by

$$y = e^{-50(x_1-0.5)^2} e^{-50(x_2-0.5)^2}$$

and

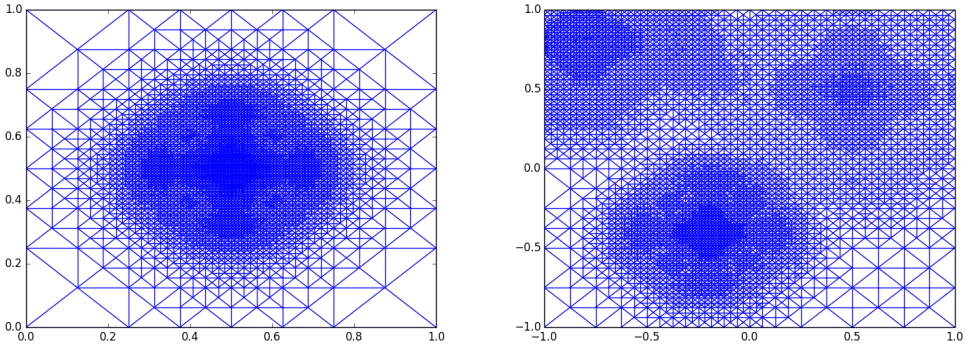
$$\begin{aligned} y = {} & e^{-0.1(x_1^2+x_2^2)} + e^{-5((x_1-0.5)^2+(x_2-0.5)^2)} \\ & + e^{-15((x_1+0.2)^2+(x_2+0.4)^2)} + e^{-9((x_1+0.8)^2+(x_2-0.8)^2)}, \end{aligned}$$

respectively. We test the error indicators using data sets with or without noise. In the case of noise, it is modeled as a normally distributed random variable with mean zero and variance 0.05. The data sets of these two model problems consist of 1×10^6 data points that are uniformly distributed on $[0.05, 0.95]^2$ and $[-0.9, 0.9]^2$, respectively. The maximum distance between data points d_x of these two data sets are about 0.9×10^3 and 1.8×10^3, respectively.

The error ϵ of the discrete smoother is estimated using the RMSE, which is popular among data fitting techniques [2]. This experiment compares the performance using the discrete smoother RMSE against the number of nodes m in the finite element grid. We also measure the efficiency of grids using $c = \epsilon m$, which is shown in the legend of error convergence plots. A small c value suggests a given error is achieved using a small number of nodes. An adaptively refined grid is considered more efficient if it reaches a certain accuracy with fewer nodes than other grids. The iterative adaptive refinement process begins with an initial uniform grid with 25 nodes and is refined for eight iterations or until the stopping criteria are satisfied.

Two adaptively refined grids generated using the norm-based error indicator are shown in Figure 1. The exponential function has a steep peak at the centre of the domain and has been used to test the performance of adaptive refinement [4]. The Franke function has a more oscillatory surface and is nonzero at some of its boundary, which is more complicated to model.

The optimal smoothing parameter α is small if the observed data is not perturbed by noise [3]. When both d_x and α are small, h becomes the

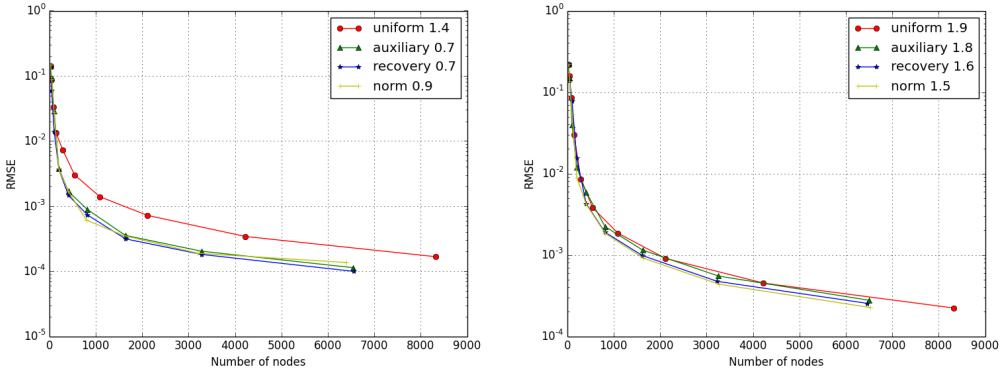


(a) 6400 nodes (b) 6528 nodes

Figure 1: Adaptively refined grids using the norm-based error indicator for (a) the exponential function; and (b) the Franke function.

dominating factor of the error. The error convergence of the exponential function and Franke function without noise is shown in Figures 2(a) and 2(b), respectively. The error convergence rates of adaptive grids using the three error indicators are similar and are significantly higher than that of the uniform grid for the exponential function. Adaptive refinement concentrates its refinement on the peak, which requires finer elements to model, as shown in Figure 1(a). The resulting adaptive grids are mostly twice as efficient as the uniform grid. The error convergence for the Franke function is similar to the adaptive grids and have higher error convergence rates than the uniform grid. However, the error convergence rates of adaptive grids are only slightly higher than the uniform grid since the Franke function's surface contains several peaks and requires refinement on more regions of the domain, as shown in Figure 1(b).

When data is perturbed by noise, the iterative refinement process often terminates earlier than eight iterations. Figure 3 shows the error convergence rates of uniform and adaptive grids two iterations before the termination. Figure 3(a) shows the error convergence for the exponential function perturbed

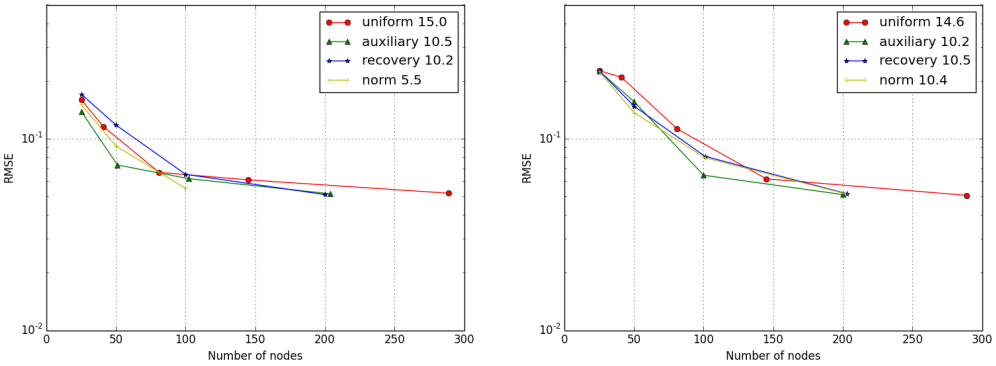


(a) (b)

Figure 2: RMSE of uniform and adaptive grids for data sets without noise of (a) the exponential function; and (b) the Franke function. Numbers in the legend are efficiency values $\mathbf{c}$ of final grids.

by noise. The error of uniform and adaptive grids decrease rapidly in the first few iterations and then slows down and stabilises at about 0.05. While further refinement may further reduce the RMSE of the discrete smoother, the efficiency of the grid will deteriorate. Both the uniform and adaptive grids terminate with a similar RMSE; however, the adaptive grids are all more efficient than the uniform grid. Figure 3(b) shows similar error convergence results for the Franke function perturbed by noise. The error convergence of adaptive grids is slightly higher than that of the uniform grid with close final efficiency values.

The error convergence rates of all adaptive grids are higher than that of the uniform grid in this numerical experiment. Adaptive refinement produces more efficient grids by focusing on refining elements in regions that are complex to model and it performs well for the two model problems considered here. The performances of the three error indicators are similar in the numerical experiment. However, their efficiency may also be affected by other factors



(a) (b)

Figure 3: RMSE of uniform and adaptive grids for data sets perturbed by noise of (a) the exponential function; and (b) the Franke function. Numbers in the legend are efficiency values $\mathbf{c}$ of final grids.

not explored here, such as the smoothing parameter α and the maximum distance between data points $\mathbf{d_x}$. Fang [4] provided more details.

6 Conclusion

A new iterative adaptive refinement process and three error indicators of the discrete smoother are presented in this article. The new process utilises the generalised cross-validation and new stopping criteria to produce optimal finite element grids. Two model problems were chosen to evaluate the error indicators' performance in producing efficient adaptive grids. The resulting error convergence rates and grid efficiency of the three error indicators show significant improvement over uniform refinement. When data is not perturbed by noise, the adaptive grids reach low errors with fewer nodes than the uniform grid. In the presence of noise, the error indicators still produce adaptive grids that reach the required accuracy with coarser grids. Current results are

obtained using data sets with a large number of data points that are uniformly distributed. The performance of the error indicators may be affected if the uniformly distributed data points assumption no longer holds, especially in the presence of noise. Further work is being done to measure the performance when this assumption does not hold.

References

[1] F. L. Bookstein. "Principal warps: Thin-plate splines and the decomposition of deformations". In: *IEEE Trans. Pat. Anal. Mach. Int.* 11.6 (1989), pp. 567–585. DOI: 10.1109/34.24792 (cit. on p. C47).

[2] C. Chen and Y. Li. "A robust method of thin plate spline and its application to DEM construction". In: *Comput. Geosci.* 48 (2012), pp. 9–16. DOI: 10.1016/j.cageo.2012.05.018 (cit. on p. C52).

[3] L. Fang. "Error estimation and adaptive refinement of finite element thin plate spline". PhD thesis. The Australian National University. URL: http://hdl.handle.net/1885/237742 (cit. on pp. C48, C52).

[4] L. Fang. "Error indicators and adaptive refinement of the discrete thin plate spline smoother". In: *ANZIAM J.* 60 (2018), pp. 33–51. DOI: 10.21914/anziamj.v60i0.14061 (cit. on pp. C49, C50, C51, C52, C55).

[5] M. F. Hutchinson. "A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines". In: *Commun. Stat. Simul. Comput.* 19.2 (1990), pp. 433–450. DOI: 10.1080/03610919008812866 (cit. on p. C48).

[6] W. F. Mitchell. "A comparison of adaptive refinement techniques for elliptic problems". In: *ACM Trans. Math. Soft.* 15.4 (1989), pp. 326–347. DOI: 10.1145/76909.76912 (cit. on pp. C48, C49).

[7] R. F. Reiniger and C. K. Ross. "A method of interpolation with application to oceanographic data". In: *Deep Sea Res. Oceanographic Abs.* 15.2 (1968), pp. 185–193. DOI: 10.1016/0011-7471(68)90040-5 (cit. on p. C46).

[8] S. Roberts, M. Hegland, and I. Altas. "Approximation of a thin plate spline smoother using continuous piecewise polynomial functions". In: *SIAM J. Numer. Anal.* 41.1 (2003), pp. 208–234. DOI: 10.1137/S0036142901383296 (cit. on pp. C46, C47, C48, C49).

[9] D. Ruprecht and H. Muller. "Image warping with scattered data interpolation". In: *IEEE Comput. Graphics Appl.* 15.2 (1995), pp. 37–43. DOI: 10.1109/38.365004 (cit. on p. C46).

[10] E. G. Sewell. *Analysis of a finite element method*. Springer, 2012. DOI: 10.1007/978-1-4684-6331-6 (cit. on p. C51).

[11] L. Stals. "Efficient solution techniques for a finite element thin plate spline formulation". In: *J. Sci. Comput.* 63.2 (2015), pp. 374–409. DOI: 10.1007/s10915-014-9898-x (cit. on p. C47).

[12] O. C. Zienkiewicz and J. Z. Zhu. "A simple error estimator and adaptive procedure for practical engineerng analysis". In: *International Journal for Numerical Methods in Engineering* 24.2 (1987), pp. 337–357. DOI: 10.1002/nme.1620240206 (cit. on p. C50).

Author addresses

1. **L. Fang**, Mathematical Sciences Institute, Australian National University, ACT 0200, AUSTRALIA.
mailto:Lishan.Fang@anu.edu.au
orcid:0000-0002-1914-7817

2. **L. Stals**, Mathematical Sciences Institute, Australian National University, ACT 0200, AUSTRALIA.
mailto:Linda.Stals@anu.edu.au
orcid:0000-0003-1557-8393