N.B.Petrovskaya

The Impact of Grid Cell Geometry on the Least-Squares Gradient Reconstruction

Moscow
N.B. Petrovskaya

The Impact of Grid Cell Geometry on the Least-Squares Gradient Reconstruction.

Abstract

We consider the problem of the least-squares approximation on two-dimensional unstructured grids with "bad" cells. We discuss how the accuracy of the least-squares approximation depends on the cell geometry. We analyze a simple geometry and demonstrate that introducing weight coefficients into the problem may help to essentially improve the accuracy of the least-squares approximation. Based on the results of our analysis, a heuristic choice of the weights in a general least-squares procedure is suggested. Our approach is illustrated by numerical tests.
Introduction. 1

We discuss the issue of a least-squares approximation on two-dimensional unstructured grids with “bad” cell geometry. The issues related to approximation and interpolation are covered in many books (e.g. see [2]). Nevertheless, to our best knowledge, the approximation over grid cells, which are almost degenerate, is not widely addressed in the literature. A general discussion of the topic can be found in [1], [5]. The impact of the cell geometry on the quality of approximation has been studied in [3, 4, 7].

Generally, a least-squares reconstruction is difficult for analysis, since a data set of an arbitrary dimension is used for the approximation. Thus we first discuss a simple geometry and demonstrate that data measuring at remote points which lie beyond an actual domain of interest, may considerably worsen the approximation. As a result of our analysis, we elaborate the weighting coefficients which correct the impact of “bad” geometry (i.e. remote points) in a general least-squares procedure. Our approach is illustrated by numerical tests.

1. Problem statement.

Consider a data set \( U = (U_1, U_2, ..., U_N) \) where the data \( U_i \) represent a continuous function \( U(x, y) \) at points \( P_i = (x_i, y_i), \ i = 1, ..., N \). We have to fit the data \( U \) to the function

\[
u(x, y) = \sum_{k=1}^{M} u_k \phi_k(x, y), \quad M \leq N,
\]

where \((u_1, u_2, ..., u_M)\) are fitting parameters, and \( \phi_k(x, y), \ k = 1, ..., M \) are basis functions. We define the merit function \( F^2 \) as follows (e.g. see [6])

\[
F^2 = \sum_{i=1}^{N} \left[ \frac{U_i - \sum_{k=1}^{M} u_k \phi_k(P_i)}{\sigma_i} \right]^2.
\]

Below we loosely refer to parameter \( \sigma_i \) as the weight of the \( i \)-th data point.

A least-squares approach considers the vector \( u = (u_1, u_2, ..., u_M) \) as the best fit to a given data set, if \( u \) minimizes the function (2). Thus, the parameters \( u_k \) can be found from the \( M \) conditions

\[
\frac{\partial F^2}{\partial u_k} = 0, \quad k = 1, ..., M,
\]

\[1\]This work was supported by The Boeing Company under contract No 104AE and Russian Foundation for Basic Research, grant No 03-01-00063.
which are called the normal equations of the least-squares problem. Taking into account the definition (2), we obtain the normal equations in the following form

$$\sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[ U_i - \sum_{j=1}^{M} u_j \phi_j(P_i) \right] \phi_k(P_i) = 0, \ k = 1, ..., M.$$  

We introduce the weighted data $b$ and the design matrix $[A]$ as follows

$$b_i = U_i/\sigma_i, \ A_{ij} = \phi_j(P_i)/\sigma_i, \ i = 1, ..., N, \ j = 1, ..., M.$$  

The normal equations can be written as $[\alpha] \ u = [\beta]$, where the matrix $[\alpha] = [A]^T [A]$, and the right-hand side $\beta = [A]^T \ b$. They are to be solved for the vector of parameters $u = (u_1, ..., u_M)$,

$$u = [\alpha]^{-1}[\beta]. \quad (3)$$

Below we consider the approximation (1) with linear basis functions,

$$u(x, y) = u_0 + u_1(x - x_0) + u_2(y - y_0), \quad (4)$$

where the origin $P_0 = (x_0, y_0)$ is chosen to serve the needs of the problem under consideration. Our main purpose is to understand how the fitting parameters $u$ in (4) depend on the geometry $\{P\}$. The problem can be illustrated by the following example. Consider the simplest geometry, $P_1 = (-\Delta x, 0), \ P_2 = (0, \Delta y), \ P_3 = (\Delta x, 0)$, and $P_4 = (0, -\Delta y)$. We reconstruct $U(x, y)$ at $P_0 = (0, 0)$, assuming $\sigma_i = 1, \ \forall i = 1, ..., 4$. The matrix $[\alpha]^{-1}$ is diagonal,

$$[\alpha]^{-1} = \begin{bmatrix} \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{2\Delta x^2} & 0 \\ 0 & 0 & \frac{1}{2\Delta y^2} \end{bmatrix},$$

and the reconstruction (4) is

$$u(x, y) = \frac{U_1 + U_2 + U_3 + U_4}{4} + \frac{(U_3 - U_1)}{2\Delta x} x + \frac{(U_2 - U_4)}{2\Delta y} y.$$ 

One can see that the least-squares procedure provides a second order approximation to the function $U(x, y)$ and gradient $(\partial U(x, y)/\partial x, \partial U(x, y)/\partial y)|_{P_0}$. However, this result should be essentially attributed to the geometry of the problem. Now let us have an arbitrary geometry, $P_i = (\Delta x_i, \Delta y_i), \ i = 1, ..., 4$. The entries of the matrix $[\alpha]^{-1}$ are not "decoupled" anymore, as each of them
depends now on each of deviations \((\Delta x_i, \Delta y_i)\). Hence, each fitting parameter \(u_k\) can be presented as

\[
u_k = \sum_{i=1}^{N} C_{kl}(\{P\}) U_i,
\]

where coefficients \(C_{kl}(\{P\})\) are defined by the geometry \(\{P\}\).

We are going to study how far the geometric coefficients \(C_{kl}(\{P\})\) are responsible for the quality of the least-squares approximation. In particular, we are interested in the gradient reconstruction at a given point \(P_0\), in which case \(u_0 = U(x_0, y_0)\) and \(\mathbf{u} = (u_1, u_2)\). The matrix \([\alpha]^{-1}\) used for the gradient reconstruction takes the form

\[
[\alpha]^{-1} = \begin{bmatrix}
\sum_{i=1}^{N} (y_i - y_0)^2/\sigma_i^2 & - \sum_{i=1}^{N} (x_i - x_0)(y_i - y_0)/\sigma_i^2 \\
\Delta & \Delta \\
- \sum_{i=1}^{N} (x_i - x_0)(y_i - y_0)/\sigma_i^2 & \sum_{i=1}^{N} (x_i - x_0)^2/\sigma_i^2 \\
\Delta & \Delta
\end{bmatrix},
\]

(5)

where

\[
\Delta = \sum_{i=1}^{N} (x_i - x_0)^2/\sigma_i^2 \sum_{i=1}^{N} (y_i - y_0)^2/\sigma_i^2 - \left( \sum_{i=1}^{N} (x_i - x_0)(y_i - y_0)/\sigma_i^2 \right)^2.
\]

The gradient error is

\[
e_{\nabla}(P_0) = ||\nabla U(x, y) - \nabla u(x, y)||_{|P_0} = \sqrt{\left( (\partial U(x, y)/\partial x)|_{P_0} - u_1 \right)^2 + \left( (\partial U(x, y)/\partial y)|_{P_0} - u_2 \right)^2}
\]

where \(\nabla\) is a formal notation for the gradient vector, \(\nabla = (\partial/\partial x, \partial/\partial y)\).

We begin our consideration with the following simple configuration. Let \(P_1 = (-H, h_1)\), \(P_2 = (0, h_0)\), and \(P_3 = (H, h_1)\), where \(H \gg h_0\) (see configuration I in fig.1). We define the function \(U(x, y)\) as

\[
U(x, y) = ax^2 + y,
\]

(6)

where the parameter \(a = -0.001\), and reconstruct the gradient \((u_1, u_2)\) at the origin \(P_0 = (0, 0)\). The analytic gradient is \(\partial U/\partial x = 2ax\), \(\partial U/\partial y = 1\), so that \(\nabla U(P_0) = (0, 1)\).

An unweighted least-squares approach (i.e. \(\sigma_i = 1, \forall i = 1, ..., N\)) gives us the matrix \([\alpha]^{-1}\) and the right-hand side \(\beta\) as follows
\[ P_1 = (-H, h_1) \]
\[ P_2 = (0, h_0) \]
\[ P_3 = (H, h_1) \]
\[ P_0 = (0, 0) \]

Figure 1: The geometry for the least-squares approximation. Configuration I has outliers in the data set. Configurations II, III have no outliers.

\[ (\alpha)^{-1} = \begin{bmatrix} \dfrac{1}{2H^2} & 0 \\ 0 & 1 \end{bmatrix}, \quad \beta = \begin{bmatrix} (U_3 - U_1)H \\ U_2 h_0 + (U_3 + U_1) h_1 \end{bmatrix}, \]

where \( U_i = U(P_i) \). According to (3), the gradient \((u_1, u_2)\) is reconstructed as

\[ u_1 = \frac{U_3 - U_1}{2H}, \quad u_2 = \frac{h_0}{h_0^2 + 2h_1^2} + \frac{h_1}{h_0^2 + 2h_1^2}. \]  

For the given geometry, a second order approximation to the gradient would be

\[ u_1 = \frac{U_3 - U_1}{2H}, \quad u_2 = \frac{U_2}{h_0}. \]  

It can be seen from (7) that the least-squares procedure results in a correct reconstruction of the \(x\)-component of the gradient, while the value \(u_2\) depends on both \(h_0\) and \(h_1\). Let \( H = 100, h_1 = 10, \) and \( h_0 = 1 \). The relations (6), (7) give us \( u_1 = 0.0, u_2 = 4.97512 \cdot 10^{-3} \), so that the error in the gradient norm is \( e_\nabla = 9.95025 \cdot 10^{-1} \).

Let us estimate which value \( h_1 \) would be appropriate for the accurate reconstruction (8) of the gradient. Substituting \( u_2 \) into (7) yields

\[ \frac{1}{h_0^2 + 2h_1^2} [U_2 h_0 + (U_1 + U_3) h_1] - \frac{U_2}{h_0} = 0. \]

Taking into account that \( U_1 = U_3 = aH^2 + h_1 \) and \( U_2 = h_0 \), we find that \( h_1 = 0 \). This trivial solution means that our domain of interest, where the
data $U$ are to be defined, has only two characteristic lengths, one of which is $h_0$ and another one is $H$.

Suppose that we carry out a physical experiment and measure by mistake our function $U$ at some points which lie beyond the characteristic domain. Such measurements (called outliers in the statistics), included into the data set, may seriously affect the results of the least-squares procedure used to treat the experimental data. For the configuration $I$, the length $h_1 \neq 0$ indicates that the geometry of the domain associated with the function $U(x, y)$ has not been correctly defined. Thus, we may suggest outliers in our problem, i.e. the data at the points which lie beyond the actual characteristic domain.

What are outliers in our data set? An evident answer is to recognize the remote points $P_1$ and $P_3$ as the outliers. The configuration $II$ shown in the figure provides $h_1 = 0$ and gives us the gradient (8) at the point $P_0$. Another, less evident conclusion, is to admit that it is not correct to measure the data at the point $P_0$. An appropriate choice would be to put the point $P_0$ in the vicinity of the midpoint of the edge $P_1 - P_3$ (see configuration $III$ in the figure).

2. The choice of weights in the least-squares procedure.

Now let us look how we can improve the gradient estimate (7) by using weights in the least-squares procedure. Since the configuration is symmetric relative to the $y$-axis, the equal weights $\sigma_1 = \sigma_3 \equiv \sigma$ are assumed for the data at the points $P_1$ and $P_3$. We denote the weight of $U_2$ as $\delta$.

For the weighted least-squares approximation, the matrix $[\alpha]^{-1}$ and the right-hand side $\beta$ are given by

$$[\alpha]^{-1} = \begin{bmatrix} \frac{\sigma^2}{2H^2} & 0 \\ 0 & \frac{\sigma^2 \delta^2}{2h_0^2 \delta^2 + \sigma^2 h_0^2} \end{bmatrix}, \quad \beta = \begin{bmatrix} \frac{(U_3 - U_1) H}{\sigma^2} \\ \frac{(U_3 + U_1) h_1}{\sigma^2} + \frac{h_0 \delta}{\delta^2} \end{bmatrix}. \quad (9)$$

The gradient is

$$u_1 = (U_3 - U_1) \frac{1}{2H},$$

$$u_2 = \frac{\sigma^2 h_0^2}{\sigma^2 h_0^2 + \delta^2 2h_1^2} \frac{U_2}{h_0} + \frac{\delta^2 h_1}{\sigma^2 h_0^2 + \delta^2 2h_1^2} (U_1 + U_3) =$$

$$A_w(\sigma, \delta) \frac{U_2}{h_0} + B_w(\sigma, \delta) (U_1 + U_3),$$

where

$$A_w(\sigma, \delta) = \frac{\sigma^2 h_0^2}{\sigma^2 h_0^2 + \delta^2 2h_1^2}, \quad B_w(\sigma, \delta) = \frac{\delta^2 h_1}{\sigma^2 h_0^2 + \delta^2 2h_1^2}.$$
It can be seen from the expression above that the values $A_w(\sigma, \delta) = 1$, $B_w(\sigma, \delta) = 0$ are required to get the consistent approximation (8). Solving these equations, we obtain $\delta = 0$, $\sigma \in \mathbb{R}$, which solution is irrelevant to the least-squares problem.

For practical purposes, we need to evaluate $\sigma$ and $\delta$ in magnitude in order to obtain asymptotic estimates $A_w(\sigma, \delta) \to 1$, $B_w(\sigma, \delta) \to 0$. To get such estimates we use the following approach. Let us consider the unweighted least-squares procedure in which new basis functions are exploited instead of the basis $\phi_1 = x$, $\phi_2 = y$. The new basis functions are

$$\tilde{\phi}_1 = f(x, y)\phi_1, \quad \tilde{\phi}_2 = g(x, y)\phi_2,$$

where the functions $f(x, y)$ and $g(x, y)$ must take the required values $1/\sigma$ and $1/\delta$ at points $P_i$. This approach is equivalent to the weighting procedure in which the data vector $U(P_i)$ remains unweighted. The matrix $[\alpha]^{-1}$ and the right-hand side $\beta$ are now given by

$$[\alpha]^{-1} = \begin{bmatrix} \frac{\sigma^2}{2H^2} & 0 \\ 0 & \frac{\sigma^2\delta^2}{2h_1^2\delta^2 + \sigma^2 h_0^2} \end{bmatrix}, \quad \beta = \begin{bmatrix} (U_3 - U_1) \frac{H}{\sigma} \\ (U_3 + U_1) \frac{h_1}{\sigma} + U_2 \frac{h_0}{\delta} \end{bmatrix}. \quad (10)$$

The fitting parameters are

$$u_1 = (U_3 - U_1) \frac{\sigma}{2H}, \quad (11)$$

and

$$u_2 = \frac{\sigma^2 \delta^2}{\delta^2 2h_1^2 + \sigma^2 h_0^2} \left[ U_2 \frac{h_0}{\delta} + (U_1 + U_3) \frac{h_1}{\sigma} \right]. \quad (12)$$

Again, we seek for the weights $(\sigma, \delta)$ which provide the second order approximation (8) to the gradient. Comparing (11) and (8), we get $\sigma = 1$. Substituting (8) and $\sigma = 1$ into (12) yields

$$U_2 h_0 \delta + 2(U_1 + U_3) h_1 \delta^2 = 2h_1^2 \delta^2 + h_0^2.$$

Solving the above equation for $\delta$, we require that the discriminant $D = U_2^2 h_0^2 + 8h_0^2 h_1(U_1 + U_3 - h_1) > 0$. Taking into account the explicit form of the function (6), we obtain the following estimate

$$h_1 < h_1^{max} = 125 \frac{h_0^2}{H^2}, \quad (13)$$

i.e. $h_1 < 1/80$ in case that $h_0 = 1$, $H = 100$. Only if the points $P_1$ and $P_3$ lie in the narrow band between 0 and $h_1^{max}$, the weights will help to get the accurate gradient.
Since our geometry does not meet the requirement (13), a formal conclusion should be that the accurate reconstruction of both gradient components is impossible. However, we essentially used our knowledge of the function $U(x, y)$ in our analysis. The symmetry of the function $U(x, y)$ allows to get the required value $u_1 = 0$ along the $x$-axis. Thus, below we analyze the $y$-component of the gradient, assuming an arbitrary weight in the expression (11).

Let us rewrite (12) as

$$u_2 = A(\sigma, \delta) \frac{U_2}{h_0} + B(\sigma, \delta) (U_1 + U_3),$$

where

$$A(\sigma, \delta) = \frac{\sigma^2 \delta h_0^2}{\sigma^2 h_0^2 + \delta^2 2h_1^2}, \quad B(\sigma, \delta) = \frac{\sigma \delta^2 h_1}{\sigma^2 h_0^2 + \delta^2 2h_1^2}.$$ 

First we define the parameter $\sigma$. The equation $A(\sigma, \delta) = 1$ yields

$$\sigma^* = \frac{h_1}{h_0} \sqrt{\frac{2\delta^2}{\delta - 1}}. \quad (14)$$

Substituting $\sigma^*$ into the equation $B(\sigma^*, \delta^*) = 0$, we obtain

$$\frac{\sqrt{2} h_1 \delta \sqrt{\delta - 1}}{2h_0^2 + 2h_0h_1(\delta - 1)} = 0.$$ 

The two roots of the equation are $\delta_1^* = 0$ and $\delta_2^* = 1$. The choice of $\delta_1^* = 0$ is irrelevant to the least-squares problem. The value $\delta_2^* = 1$ gives us $\sigma^* = \infty$ with a consistent asymptotic estimate $u_2 \to 1$, as $\sigma \to \infty$. Hence, for practical purposes one should take $\delta = 1$ and choose the weights (14) for the data at the remote points $P_1$ and $P_3$ as big as possible to obtain the gradient reconstruction $u_2$ with the desired accuracy.

To evaluate the order of magnitude for $\sigma$, we analyze the matrix (5) in the presence of outliers. Let points $P_1, P_2, \ldots, P_{N-1}$ be $(x_i, y_i)$, where $x_i \sim l$, $y_i \sim l$, and the point $P_N$ be a remote point, $x_N \sim L$, $y_N \sim L$, where we assume $L \gg Nl$. Consider the diagonal entries of the matrix $[\alpha]^{-1}$ for the unweighted least-squares approximation,

$$\alpha_{11}^{-1} = \frac{\sum_{i=1}^{N} y_i^2}{\sum_{i=1}^{N} x_i^2 \sum_{i=1}^{N} y_i^2 - \left( \sum_{i=1}^{N} x_i y_i \right)^2}, \quad (15)$$

where the origin $(x_0, y_0) = (0, 0)$.  

9
Each sum in (15) can be rearranged as

\[
\alpha_{11}^{-1} \sim \frac{\sum_{i=1}^{N-1} y_i^2 + L^2}{(\sum_{i=1}^{N-1} x_i^2 + L^2)(\sum_{i=1}^{N-1} y_i^2 + L^2) - \left(\sum_{i=1}^{N-1} x_i y_i + L^2\right)^2} = \frac{\sum_{i=1}^{N-1} y_i^2 + L^2}{\sum_{i=1}^{N-1} x_i^2 + L^2},
\]

or, taking into account \[\sum_{i=1}^{N-1} x_i^2 \sim \sum_{i=1}^{N-1} y_i^2 \sim (N-1)l^2,\]

\[
\alpha_{11}^{-1} \sim \frac{L^2}{\sum_{i=1}^{N-1} x_i L^2 + \sum_{i=1}^{N-1} y_i L^2 - 2 \sum_{i=1}^{N-1} x_i y_i L^2 - L^4} = \frac{1}{\sum_{i=1}^{N-1} (x_i - y_i)^2}.
\]

A similar estimate holds for \(\alpha_{22}^{-1}\).

Since we assume \(x_i \sim l, y_i \sim l\), their difference can be very small, \(x_i - y_i \sim \epsilon \to 0\). (Note the singularity in the estimate above which arise in degenerated case that all of the points \(P_i\) are placed at the same straight line.) The entries of \(\alpha^{-1}\) will grow with grid refinement and affect the gradient reconstruction (3), unless the outlier is suppressed by a weight \(\sigma_L, L/\sigma_L \to 0, \) as \(L \to \infty\).

For the configuration \(I\), we have \(L \sim \sqrt{H^2 + h^2} \approx H\). Thus, we choose \(\sigma = H^2\) in which case the weighted least-squares procedure yields \(u_2 = 0.999998, e_\nabla = 2 \cdot 10^{-6}\) for values \(h_0 = 1, h_1 = 10, H = 100\).


The analysis made in the previous section concerns a particular configuration and function \(U(x, y)\). Nevertheless, it allows us to elaborate heuristic weights in the least-squares procedure. A general recommendation is that once outliers have been detected in the problem, those data must have small values \(\rho_i = (1/\sigma_i)\). Based on the analysis above we suggest that the weights \(\sigma_i\) can be defined as

\[
\sigma_i = r_i^2 = (x_i - x_0)^2 + (y_i - y_0)^2,
\]

where \((x_0, y_0)\) is the point where the gradient is reconstructed. Formally, the weights can be scaled as

\[
\delta_i = \sigma_i / r_{\text{min}}^2, \quad r_{\text{min}} = \min_{i=1,\ldots,N} \{r_i\}.
\]
so that the point closest to the origin \( P_0 \) has \( \tilde{\sigma}_{\text{min}} = 1 \).

Below we consider numerical tests which illustrate the choice of the weight coefficients. We define the geometry for our test cases as follows. Let points \( P_1 = (-H, 0) \), \( P_2 = (0.02, h) \), \( P_3 = (H, -0.01) \), and \( P_4 = (-0.07, -h) \), where \( h \) and \( H \) are the controlling parameters for the configuration. Let \( H = 1 \), \( h = 1 \), so the points \( P_i \) lie very close to a unit circle. We are going to imitate a refinement procedure by halving \( h \) at each ”refinement step”, the parameter \( H \) being fixed. We reconstruct the gradient \((u_1, u_2)\) at \( P_0 = (0, 0) \) and compare the error \( e_{\nabla}(h) \) for the unweighted and weighted least-squares approximation. We are also interested in the diagonal elements of the matrix \([\alpha]^{-1}\), which, in our opinion, may help to detect the outliers.

We begin our consideration with a function

\[
U(x) = \frac{1}{2}((x - A)^2 + (y - A)^2),
\]

where the parameter \( A = 2 \). The gradient is \( \partial U/\partial x = x - A, \partial U/\partial y = y - A \), \( \nabla U(P_0) = (-2, -2) \).

First, we reconstruct the gradient by using an unweighted least-squares approach. The error \( e_{\nabla}(h) \) and the diagonal entries \( \alpha_{ii}^{-1}, i = 1, 2 \) are shown in Table 1.

<table>
<thead>
<tr>
<th>( h )</th>
<th>( e_{\nabla}(h) )</th>
<th>( \alpha_{11}^{-1} )</th>
<th>( \alpha_{22}^{-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>0.0152469</td>
<td>0.4993633</td>
<td>0.5005867</td>
</tr>
<tr>
<td>0.5000</td>
<td>0.01270758</td>
<td>0.4992012</td>
<td>2.001396</td>
</tr>
<tr>
<td>0.2500</td>
<td>0.04471311</td>
<td>0.498952</td>
<td>7.996793</td>
</tr>
<tr>
<td>0.1250</td>
<td>0.1690752</td>
<td>0.4987531</td>
<td>31.89793</td>
</tr>
<tr>
<td>0.06250</td>
<td>0.652037</td>
<td>0.4995403</td>
<td>126.5818</td>
</tr>
<tr>
<td>0.03125</td>
<td>2.506628</td>
<td>0.5056627</td>
<td>493.8101</td>
</tr>
<tr>
<td>0.01562</td>
<td>9.159516</td>
<td>0.5333749</td>
<td>1817.866</td>
</tr>
<tr>
<td>0.007813</td>
<td>28.16445</td>
<td>0.6214184</td>
<td>5610.583</td>
</tr>
<tr>
<td>0.003907</td>
<td>59.85515</td>
<td>0.7776305</td>
<td>11945.89</td>
</tr>
<tr>
<td>0.001953</td>
<td>84.41483</td>
<td>0.9052282</td>
<td>16863.26</td>
</tr>
<tr>
<td>0.0009766</td>
<td>94.74845</td>
<td>0.9624739</td>
<td>18936.42</td>
</tr>
<tr>
<td>0.0004883</td>
<td>98.10343</td>
<td>0.9827952</td>
<td>19611.53</td>
</tr>
</tbody>
</table>

It can be seen from the table that the error in the gradient grows, as \( h \) is getting smaller. The error in the gradient norm is mostly due to the \( y \)-component of the gradient. Actually, at the last step of the refinement we have \( e_x = |\partial U/\partial x|_{P_0} - u_1| = 0.48743 \) and \( e_y = |\partial U/\partial y|_{P_0} - u_2| = 98.103 \).
Let us evaluate the characteristic lengths $l_x$ and $l_y$ from the relation

$$
\frac{U(l_x,0) - U(0,0)}{l_x} \sim 1, \quad \frac{U(0,l_y) - U(0,0)}{l_y} \sim 1.
$$

For "homogeneous" function (16), the characteristic domain is $l_x = l_y$. Our "anisotropic" refinement procedure makes the points $P_1$ and $P_3$ lie outside the characteristic domain. The presence of the outliers in the problem is evidenced by different orders of magnitude in $\alpha_{11}^{-1}$ and $\alpha_{22}^{-1}$, as one may expect $\alpha_{11}^{-1} \sim \alpha_{22}^{-1}$, if the characteristic domain is chosen adequate to the function (16). Thus, the weighting procedure is required to eliminate the outliers.

**Table 2.**

The weighted least-squares procedure for the function (16).

<table>
<thead>
<tr>
<th>$h$</th>
<th>$e_{\nabla}(h)$</th>
<th>$\alpha_{11}^{-1}$</th>
<th>$\alpha_{22}^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>0.01489822</td>
<td>0.4994162</td>
<td>0.5030751</td>
</tr>
<tr>
<td>0.5000</td>
<td>0.05869409</td>
<td>0.4927092</td>
<td>0.1304421</td>
</tr>
<tr>
<td>0.2500</td>
<td>0.1923858</td>
<td>0.4118147</td>
<td>0.0430285</td>
</tr>
<tr>
<td>0.1250</td>
<td>0.2295866</td>
<td>0.1353229</td>
<td>0.01894057</td>
</tr>
<tr>
<td>0.0625</td>
<td>0.1119552</td>
<td>0.024711</td>
<td>0.006007634</td>
</tr>
<tr>
<td>0.03125</td>
<td>0.06687997</td>
<td>0.009721229</td>
<td>0.002569299</td>
</tr>
<tr>
<td>0.01562</td>
<td>0.06012273</td>
<td>0.007275371</td>
<td>0.003624347</td>
</tr>
<tr>
<td>0.007812</td>
<td>0.07669123</td>
<td>0.00675025</td>
<td>0.01162409</td>
</tr>
<tr>
<td>0.003906</td>
<td>0.127259</td>
<td>0.006623483</td>
<td>0.04456423</td>
</tr>
<tr>
<td>0.001953</td>
<td>0.2398984</td>
<td>0.006590117</td>
<td>0.1765079</td>
</tr>
<tr>
<td>0.0009766</td>
<td>0.4704899</td>
<td>0.006577596</td>
<td>0.703892</td>
</tr>
<tr>
<td>0.0004883</td>
<td>0.9289341</td>
<td>0.00656544</td>
<td>2.809573</td>
</tr>
</tbody>
</table>

The results of the weighted least-squares approximation are shown in Table 2. Comparing the unweighted and weighted gradient reconstruction, one can see that weighting reduces the gradient error. However, the weighting procedure is efficient only in the presence of outliers. At initial steps of the refinement, the error is smaller for the unweighted least-squares approach. Also, it can be seen from Table 2 that at last steps of the refinement the geometry distortion is so strong that the error begins to grow again.
Table 3.
The unweighted least-squares procedure for the function (17).

<table>
<thead>
<tr>
<th>$h$</th>
<th>$e_{\nabla}(h)$</th>
<th>$\alpha^{-1}_{11}$</th>
<th>$\alpha^{-1}_{22}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>2.153319</td>
<td>0.4993633</td>
<td>0.5005867</td>
</tr>
<tr>
<td>0.5000</td>
<td>2.153135</td>
<td>0.4992012</td>
<td>2.001396</td>
</tr>
<tr>
<td>0.2500</td>
<td>2.18501</td>
<td>0.498952</td>
<td>7.996793</td>
</tr>
<tr>
<td>0.1250</td>
<td>2.429651</td>
<td>0.4987531</td>
<td>31.89793</td>
</tr>
<tr>
<td>0.0625</td>
<td>4.095294</td>
<td>0.4995403</td>
<td>126.5818</td>
</tr>
<tr>
<td>0.03125</td>
<td>11.7667</td>
<td>0.5056627</td>
<td>493.8101</td>
</tr>
<tr>
<td>0.01562</td>
<td>38.88393</td>
<td>0.5333749</td>
<td>1817.866</td>
</tr>
<tr>
<td>0.007812</td>
<td>114.0223</td>
<td>0.6214184</td>
<td>5610.583</td>
</tr>
<tr>
<td>0.003906</td>
<td>236.5301</td>
<td>0.7776305</td>
<td>11945.89</td>
</tr>
<tr>
<td>0.001953</td>
<td>329.5013</td>
<td>0.9052282</td>
<td>16863.26</td>
</tr>
<tr>
<td>0.0009766</td>
<td>367.5444</td>
<td>0.9624739</td>
<td>18936.42</td>
</tr>
<tr>
<td>0.0004883</td>
<td>379.3713</td>
<td>0.9827952</td>
<td>19611.53</td>
</tr>
</tbody>
</table>

Table 4.
The weighted least-squares procedure for the function (17).

<table>
<thead>
<tr>
<th>$h$</th>
<th>$e_{\nabla}(h)$</th>
<th>$\alpha^{-1}_{11}$</th>
<th>$\alpha^{-1}_{22}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>2.153138</td>
<td>0.4994162</td>
<td>0.5030751</td>
</tr>
<tr>
<td>0.5000</td>
<td>2.171339</td>
<td>0.4927092</td>
<td>0.1304421</td>
</tr>
<tr>
<td>0.2500</td>
<td>2.021146</td>
<td>0.4118147</td>
<td>0.0430285</td>
</tr>
<tr>
<td>0.1250</td>
<td>1.099674</td>
<td>0.1353229</td>
<td>0.01894057</td>
</tr>
<tr>
<td>0.0625</td>
<td>0.6035814</td>
<td>0.024711</td>
<td>0.006007634</td>
</tr>
<tr>
<td>0.03125</td>
<td>0.5099836</td>
<td>0.009721229</td>
<td>0.002569299</td>
</tr>
<tr>
<td>0.0156</td>
<td>0.4908921</td>
<td>0.00725371</td>
<td>0.003624347</td>
</tr>
<tr>
<td>0.007812</td>
<td>0.4869887</td>
<td>0.00675025</td>
<td>0.01162409</td>
</tr>
<tr>
<td>0.003906</td>
<td>0.4880201</td>
<td>0.006623483</td>
<td>0.04456423</td>
</tr>
<tr>
<td>0.001953</td>
<td>0.4951019</td>
<td>0.006590117</td>
<td>0.1765079</td>
</tr>
<tr>
<td>0.0009766</td>
<td>0.5184101</td>
<td>0.006577596</td>
<td>0.703892</td>
</tr>
<tr>
<td>0.0004883</td>
<td>0.5765298</td>
<td>0.00656544</td>
<td>2.809573</td>
</tr>
</tbody>
</table>

Another solution to Laplace’s equation is

$$U(x, y) = A \log \left( \sqrt{R_1(x, y)/R_2(x, y)} \right), \quad (17)$$

where $R_1(x, y) = (x - a)^2 + y^2$, $R_2(x, y) = (x - 4a)^2 + y^2$. In our numerical calculations, $a = 1.0$, $A = 1.0$. The gradient is $\partial U / \partial x = A((x - a)/R_1(x, y) - (x - 4a)/R_2(x, y))$, $\partial U / \partial y = A(y/R_1(x, y) - y/R_2(x, y))$, $\nabla U(P_0) = (-0.25, 0)$. 

13
The results for the unweighted and weighted least-squares approach are shown in Table 3 and Table 4, respectively. The results are similar to those for the function (16). Again, the outliers in the problem are evidenced by different orders of magnitude in the diagonal elements of $[\alpha]^{-1}$.

Our next test is to consider the function

$$U(x, y) = x + \exp(Ay), \quad (18)$$

where the parameter $A = 5$. The gradient is $\partial U/\partial x = 1$, $\partial U/\partial y = A \exp(Ay)$, $\nabla U(P_0) = (1, 5)$.

**Table 5.**
The least-squares procedure (LS) for the function (18). The gradient error for the unweighted and weighted LS approach.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$\epsilon_{\nabla}(h)$ (unweighted LS)</th>
<th>$\epsilon_{\nabla}(h)$ (weighted LS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>69.28902</td>
<td>69.63832</td>
</tr>
<tr>
<td>0.5000</td>
<td>7.106995</td>
<td>7.835687</td>
</tr>
<tr>
<td>0.2500</td>
<td>1.408095</td>
<td>5.731806</td>
</tr>
<tr>
<td>0.1250</td>
<td>0.3305086</td>
<td>5.29683</td>
</tr>
<tr>
<td>0.0625</td>
<td>0.07890232</td>
<td>1.911436</td>
</tr>
<tr>
<td>0.03125</td>
<td>0.01458822</td>
<td>0.6712025</td>
</tr>
<tr>
<td>0.01562</td>
<td>0.009495008</td>
<td>0.2829979</td>
</tr>
<tr>
<td>0.007812</td>
<td>0.03703674</td>
<td>0.133406</td>
</tr>
<tr>
<td>0.003906</td>
<td>0.07644436</td>
<td>0.06547207</td>
</tr>
<tr>
<td>0.0019531</td>
<td>0.1057604</td>
<td>0.03250343</td>
</tr>
<tr>
<td>0.0009766</td>
<td>0.1176681</td>
<td>0.0161402</td>
</tr>
<tr>
<td>0.0004883</td>
<td>0.1213422</td>
<td>0.007900425</td>
</tr>
</tbody>
</table>

The gradient error for the unweighted and weighted least-squares approach is shown in Table 5. For the function (18), one may expect the characteristic length $l_y \ll l_x$, so that the data at the points $P_2$ and $P_4$ will be outliers at the first steps of the refinement. It can be seen from the table that the weighting procedure does not recognize the outliers for the function (18), as a uniform initial geometry provides almost equal weights for the points of the configuration. On the other hand, the outliers, which exist on the initial grid, can be eliminated by means of the refinement. The unweighted least-squares procedure is appropriate for the problem, as the gradient error reduces with the grid refinement. Only that the grid parameter $h \ll l_y$ (i.e. at last steps of the refinement), the unweighted approach fails to reconstruct the gradient, so that weighting is of use.
Conclusions.

- We have obtained preliminary results concerning the problem of the least-squares approximation on bad grids. It has been shown that if data used for the approximation are measured at the points which lie beyond an actual characteristic domain, the least-squares procedure will give us a reconstruction with poor accuracy.

- We have suggested a heuristic choice of the weights in the least-squares approximation. It has been demonstrated that introducing weight coefficients into the problem may help to eliminate outliers and improve the accuracy of the least-squares procedure. The issue of weighting requires a further thorough study and has to be considered together with the problem of the detection of outliers.

References


