Weighted least squares Poisson solver

Version: June 26, 2020

This solver for the Poisson equation

$$\nabla^2 f(y,z) = s(y,z) \tag{1}$$

on unstructured meshes is a 2D implementation of the 3D algorithm described in detail by

> *Grid free method for solving the Poisson equation* by J. Kuhnert and S. Tiwari (2001), Berichte des Fraunhofer ITWM, Nr. 25.

We want to solve (1) in a 2D domain (y,z), where $s(y,z)$ is a potentially spatially varying source term that does not depend on f. Boundary conditions are either Dirichlet or Neumann. Dirichlet boundaries are passed as additional points at which the solution is not found but instead given, which allows for an accurate description of the boundary, that can significantly affect the magnitude of f. The Neumann boundary condition is applied to the subset of the regular points that forms the respective boundary. Periodic boundaries can be prescribed along y for a fixed distance by duplicating relevant points in the domain.

For each point p in the domain, we find the smallest possible number of neighbour points $i \in \{1,2,\dots,k\}$ and consider the Taylor series

$$f_i = f_p + \Delta y_i \left.\frac{\partial f}{\partial y}\right|_p + \Delta z_i \left.\frac{\partial f}{\partial z}\right|_p + \frac{(\Delta y_i)^2}{2} \left.\frac{\partial^2 f}{\partial y^2}\right|_p + \frac{(\Delta z_i)^2}{2} \left.\frac{\partial^2 f}{\partial z^2}\right|_p + \Delta y_i \Delta z_i \left.\frac{\partial^2 f}{\partial y \partial z}\right|_p + e_i, \tag{2}$$

where e_i is an error term and $\Delta y_i = y_i - y_p$, $\Delta z_i = z_i - z_p$. Further,

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_p + \left.\frac{\partial^2 f}{\partial z^2}\right|_p = s_p \tag{3}$$

or potentially instead

$$\left.\frac{\partial f}{\partial n}\right|_p = b, \tag{4}$$

if p is on a boundary with normal vector n that is scaled to satisfy $|n| = 1$. We combine the system into

$$\begin{bmatrix} s_p \\ b \\ f_1 \\ f_2 \\ \vdots \\ f_k \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \{0,1\} & \{0,1\} & 0 \\ 0 & \{0,n_y\} & \{0,n_z\} & 0 & 0 & 0 \\ 1 & \Delta y_1 & \Delta z_1 & (\Delta y_1)^2/2 & (\Delta z_1)^2/2 & \Delta y_1 \Delta z_1 \\ 1 & \Delta y_2 & \Delta z_2 & (\Delta y_2)^2/2 & (\Delta z_2)^2/2 & \Delta y_2 \Delta z_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \Delta y_k & \Delta z_k & (\Delta y_k)^2/2 & (\Delta z_k)^2/2 & \Delta y_k \Delta z_k \end{bmatrix} \begin{bmatrix} f_p \\ \partial f/\partial y|_p \\ \partial f/\partial z|_p \\ \partial^2 f/\partial y^2|_p \\ \partial^2 f/\partial z^2|_p \\ \partial^2 f/(\partial y \partial z)|_p \end{bmatrix} + \begin{bmatrix} e_{\text{rhs}} \\ e_b \\ e_1 \\ e_2 \\ \vdots \\ e_i \end{bmatrix} \tag{5}$$

or $K = AR + E$. For instance, a zero-gradient boundary $\partial f/\partial z = 0$ at the top of the domain is prescribed via $b = 0$, $n_y = 0$, $n_z = -1$ with only zeros in the first row of A and $s_p = 0$. At other points p, the second row of A contains only zeros as the first row accounts for (3). We want to minimise the sum of the weighted squared errors $J = E^T W E$, where the $(k+2) \times (k+2)$ weight matrix is

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & w_1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & w_k \end{bmatrix}. \tag{6}$$

The weights on the diagonal $w_i = (\alpha + 1)/\left(\alpha + (\Delta_i/\Delta_1)^2\right)$ are inverse proportional to the distance between p and the neighbour point, $\Delta_i = \sqrt{\Delta y_i^2 + \Delta z_i^2}$. Weights are assigned relative to that of the closest point (denoted here by the index $i = 1$), so that $w_1 = 1$ and $w_{i>1} \in [0,1]$ are similar to the first two weights on the diagonal in (6) regardless of the mesh dimensions. The weighting parameter α can be used to adjust spatial averaging. For $\alpha = 0$, only the closest points are effectively used to find the value at p, as the weights are given by the inverse of the squared distance, relative to the that of the closest neighbour point. The weights of neighbours farther from p increase with α and for $\alpha \rightarrow \infty$, all neighbour points are weighted equally, which promotes spatial averaging. We chose $\alpha = 0.1$ to include all points necessary to solve for f_p, while at the same time avoiding spatial averaging as much as possible. By minimising $J = (AR - K)^T W (AR - K)$, we find

$$R = \begin{bmatrix} f_p \\ \partial f/\partial y|_p \\ \partial f/\partial z|_p \\ \partial^2 f/\partial y^2|_p \\ \partial^2 f/\partial z^2|_p \\ \partial^2 f/(\partial y \partial z)|_p \end{bmatrix} = \left[\left(A^T W A\right)^{-1} A^T W\right] K \qquad (7)$$

at each point p, update the field values $f_p^{\tau+1} = r R_1^\tau + (1-r) f_p^\tau$ and iterate $\tau = 1, 2, \ldots$ following the Jacobi method with optional relaxation $r > 0$ until the field f is converged. For convergence, we require the relative change of the maximum norm $||f||_\infty$ between iterations to fall below e.g. $r \times 10^{-12}$ and then verify that e_{rhs} and e_b from (5) are small. Note that the term in square brackets on the right-hand side of (7) need only be calculated once for each point p of a given mesh. The ideal number of neighbours $k \gtrsim 6$ used at a given point p is the smallest that nevertheless makes $\left(A^T W A\right)$ invertible.