Nik Leuenberger
Deal.ii *Finite Volume Method test problem*
June 19, 2024

Deal.ii

1. One dimensional example, finite volume method

1.1. Governing equations:

$$\frac{\partial}{\partial x}\left(\kappa \frac{\partial \phi}{\partial x}\right) = 0, \quad 0 \le x \le 1, \quad \kappa = \text{const.} \tag{1}$$

The reason why I use partial derivatives, and write the second order derivative like this, is because the notation is closer to the more complicated system I would like to solve later on.

1.2. Boundary conditions:

- $x = 0 \quad \phi = 0$
- $x = 1 \quad$ constant flux I_{app} (i.e. $\kappa \frac{\partial \phi}{\partial x} = I_{app}$)

1.3. Solution. The analytical solution is given by

$$\phi(x) = \frac{I_{app}}{\kappa} x \tag{2}$$

1.4. Integral Formulation. We formulate equation (1) in integral formulation for a constant κ

$$\kappa \int_V \frac{\partial}{\partial x}\left(\frac{\partial \phi}{\partial x}\right) dV = 0, \tag{3}$$

Using Gauss' divergence theorem in 1D, we replace the volume integration with a surface integration

$$\kappa \int_S \left(\frac{\partial \phi}{\partial x}\right) dS = 0 \tag{4}$$

1.5. Discretization. We want to use n_x cell-centered finite volumes of constant width $h = \Delta x = \Delta y = \Delta z = \frac{1}{n_x}$, cell interfaces located at $f_i = ih, \quad 0 \le i \le n_x$ and cell centeres located at $x_i = (i - 1/2)h = \left\{\frac{h}{2}, \frac{3h}{2}, \ldots, 1 - \frac{h}{2}\right\}, \quad 1 \le i \le n_x$. Figure 1 shows the setup of the finite volume method for $n_x = 7$ volumes. The discrete unknowns ϕ_i will be an approximation to the solution at $x = x_i$. Therefore, our system has n_x unknowns.

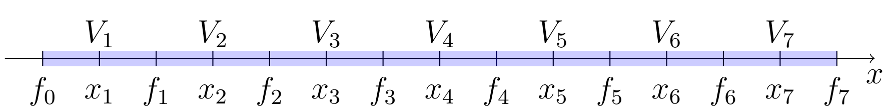

Figure 1. 1D Finite Volume setup

For each control volume, we then approximate equation (4) to obtain n_x equations for the n_x unknowns. For example for control volume V_5 in Figure 1, this would read:

$$\kappa \left[\int_{f_4} \frac{\partial \phi}{\partial x} dS + \int_{f_5} \frac{\partial \phi}{\partial x} dS\right] \tag{5}$$

$$= \kappa \left[-\left(\frac{\phi_5 - \phi_4}{h}\right) + \left(\frac{\phi_6 - \phi_5}{h}\right)\right] h^2 = 0, \tag{6}$$

where we have chosen an outward pointing normal vector (i.e. positive for f_5 and negative for f_4). The equations for all controle volumes $2, \ldots, n_x - 1$ will yield similar equations of the form.

$$\kappa \left[\left(\frac{\phi_{i-1} - \phi_i}{h}\right) + \left(\frac{\phi_{i+1} - \phi_i}{h}\right)\right] h^2 = 0, \quad 2 \le i \le n_x - 1 \tag{7}$$

For the two boundary conditions, the boundary fluxes lead to the following equations:

$$\kappa\left[\left(\frac{\phi_{\text{left}}-\phi_1}{h/2}\right)+\left(\frac{\phi_2-\phi_1}{h}\right)\right]h^2=0 \quad i=1 \tag{8}$$

$$\kappa\left(\frac{\phi_{n_x-1}-\phi_{n_x}}{h}\right)h^2-I_{app}h^2=0 \quad i=n_x \tag{9}$$

Residual and Jacobian formulation

Because ultimately, we would like to solve a time-dependent, non-linear problem, we will denote the linear system to be solved $J\delta=-R$, where $R\in\mathbb{R}^{n_x\times 1}$ is the nonlinear residual, and $J\in\mathbb{R}^{n_x\times n_x}$ is the Jacobian matrix. The solution $\delta\in\mathbb{R}^{n_x\times 1}$ will be the incremental update in a Newton-Raphson scheme, but for this *linear* problem, the solution will be given by $\phi_i=\delta_i$.

2. Assembly function

In our assembly function, we need to fill the matrix J and right-hand size $-R$ with the right entries. Similar to the `step-12/12b` tutorials, we can split the task into three distinct tasks for cells, boundaries and faces:

2.1. **Cell.** nothing to be done, later, add time-derivative term here

2.2. **Boundary.** Below is the assembly part for the boundary faces

Algorithm 1 boundary faces at $x=0$

for each face f in BoundaryFaces $x=0$ **do**
  $x_i \leftarrow$ CellCenters[right_cell(f)]
  Residuals[i]$+=\kappa\left(\frac{\phi_{\text{left}}-\phi_i}{h/2}\right)h^2$
  Jacobian[i,i]$+=-\kappa\frac{2}{h/2}h^2$
end for

Algorithm 2 boundary faces at $x=1$

for each face f in BoundaryFaces $x=1$ **do**
  $x_i \leftarrow$ CellCenters[left_cell(f)]
  Residuals[i]$+=-I_{app}h^2$
end for

2.3. **Interior Faces.** Below is the assembly part for the interior faces

Algorithm 3 interior face (visit each face once)

for each face f in BoundaryFaces $x=1$ **do**
  $x_l \leftarrow$ CellCenters[left_cell(f)]
  $x_r \leftarrow$ CellCenters[right_cell(f)]
  flux $=\kappa\left(\frac{\phi_r-\phi_l}{h}\right)h^2$
  Residuals[l]$+=$flux
  Residuals[r]$+=$-flux
  Jacobian[l,l]$+=-\kappa\frac{1}{h}h^2$
  Jacobian[r,r]$+=-\kappa\frac{1}{h}h^2$
  Jacobian[l,r]$+=\kappa\frac{1}{h}h^2$
  Jacobian[r,l]$+=\kappa\frac{1}{h}h^2$
end for

3. Plan on how to implement in deal.ii

- use elements of type `FE_DGQ`
- for a start (maybe even forever), the degree of the element can be zero (i.e. piecewise constant), like in a piecewise constant finite volume scheme.
- build on `step-12` or `step-12b` tutorial and mainly change the three assembly functions as defined above as well as the boundary conditions.
- I hard-coded these functions in the appended nik-step12.cc file that produces the correct matrix and solution for the special case of $\kappa = h = 1$ and $I_{app} = -1$.

4. Questions

I think starting from the `step-12` or `step-12b` tutorials seems to be a suitable choice since they already use discontinous elements.

My concrete questions for the forum are the following.

- How should I best write the three functions for the matrix assembly with as less hard-coding as possible?
- How can I best access the values of the current cell and neighbor cell (i.e. ϕ_{me} and $\phi_{neighbor}$) in (7)?

When I was trying to find resources for the derivation of a weak form using the discontinous Galerkin method for (1), there were many different methods in chapter 1 of Rivière [1], some with penalty terms etc., which I think might be more complicated to implement. I really like the simplicity of the finite volume method but would like to use the provided DG tools in deal.ii

References

1. Rivière, B. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations* eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898717440. https://epubs.siam.org/doi/abs/10.1137/1.9780898717440 (Society for Industrial and Applied Mathematics, 2008).