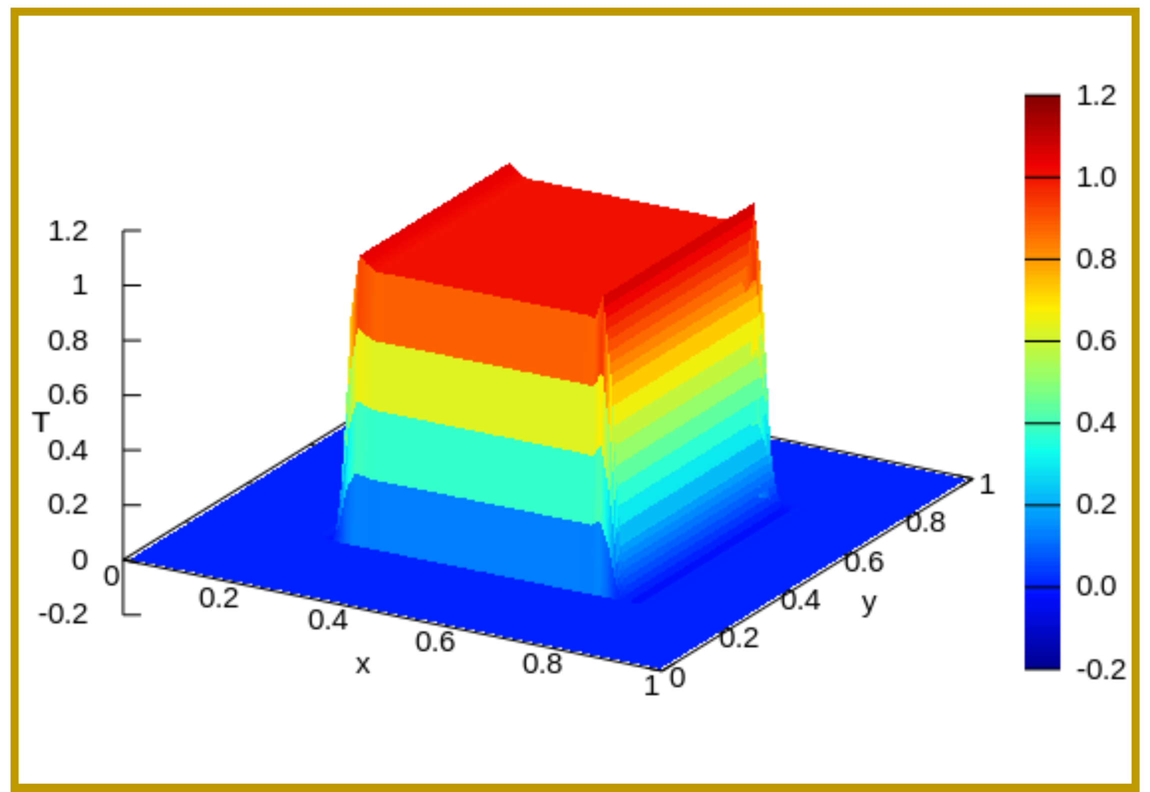
The Overshoot Problem in Mantle Convection Models

Ted H. Studley[1,3], Elbridge G. Puckett[3], Rajesh Kommu[1], Eric M. Heien[1], Louise H. Kellogg[1,2]

Introduction and Background

The overshoot problem in computational modeling is an effect which may occur when attempting to model certain behaviors of a field with a sharp or discontinuous gradient.



Overshoot in a 2D advection equation causes the data to exceed its maximum T=1 and minimum T=0 along the direction of advection towards the positive x direction, as can be seen in this figure.

The overshoot problem is charac-terized by oscillat-ory behavior along sharp gradients, causing the data to exceed its maximum and minimum values.

For the purposes of this report, we will consider the overshoot problem in the mantle convection simulation code ASPECT (Advanced Solver for Problems in Earth's ConvecTion) as well as in a small test code written explicitly to experiment with the overshoot phenomenon. Both codes are available upon request.

Overshoot in Mantle Convection Codes

Overshoot may occur in mantle convection simulation in a number of different situations. For example, models of subducting slabs or <other exmples here> may result in the sharp gradients which cause overshoot.

While moderate (≤9%) overshoot may not be an issue for many codes, certain postprocessing techniques, such as melt calculations or <another example here> rely nonlinearly upon certain values which may cause drastic differences in the exact and approximate calculated values.

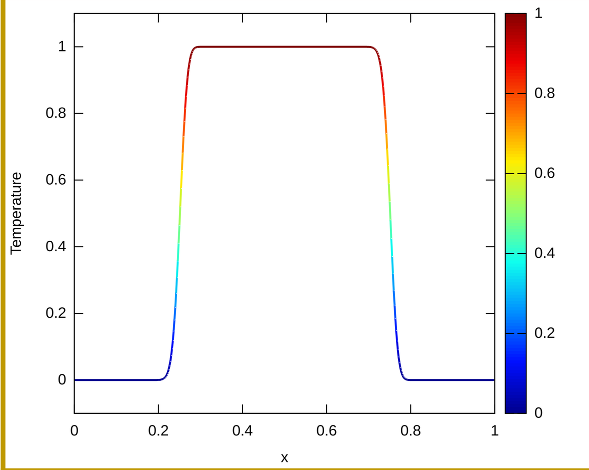
These effects have been noted in the finite element mantle convection codes Citcom, ASPECT and CONMAN, and can be expected in any finite element mantle convection code. The overshoot problem is a general problem of higher-order finite element methods which, to our knowledge, has not be solved by any code within the mantle convection community.

Overshoot in Finite Difference Codes

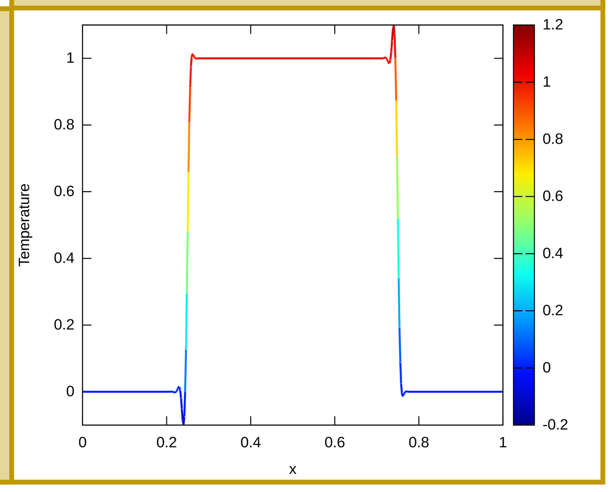
For the purposes of this problem, we developed a simple finite difference advection/diffusion model to study the effects of various numerical methods on the overshoot problem. To this end, we attempt to solve the advection-diffusion equation $u_t + \nabla(v \cdot u + \beta \nabla u) = 0$. Splitting this up into seperate advection and diffusion codes allows us to determine the solution for the overshoot problem in each problem seperately.

For the advection equation, $u_t + v \cdot \nabla u = 0$, with temperature data given by u_j^n where n is the timestep $0 \le n \le T$ and j is the space step, $0 \le j \le N$. we may discretize the equation using the simple first-order upwind method to obtain the solution $u_j^{n+1} = u_j^n + \frac{v\Delta t}{\Delta x}\left(u_{j-1}^n - u_j^n\right)$, or using the second-order Fromm's method to obtain the solution

$$u_j^{n+1} = u_j^n + \frac{v\Delta t}{\Delta x}\left(u_{j-1}^n - u_j^n\right) + \left(\frac{v\Delta t}{2\Delta x} - \frac{v^2\Delta t^2}{4\Delta x^2}\right)\left(u_{j+1}^n - u_j^n - u_{j-1}^n + u_{j+2}^n\right).$$

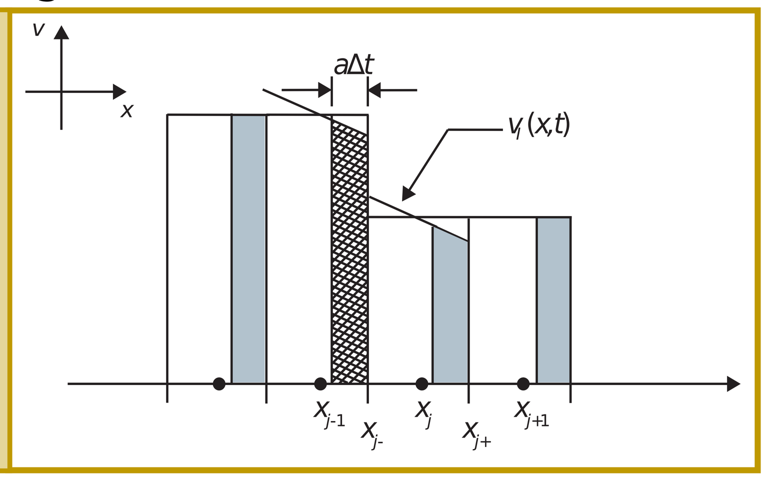


Upwind method does not overshoot, but has a much lower rate of convergence, as evidenced by the rounded corners of the advected square wave. This causes the simulation to require a much finer spatial discretization to achieve the same level of accuracy as a higher-order method, usually at a severe cost to computation time.



Fromm's method has a much higher rate of convergence, resulting in the sharper corners visible in this figure, but overshoots the maximum value T=1.0 and undershoots the minimum value T=1.0. In both plots, the initial conditions were a square wave with 256 subdivisions advection in the positive x direction for one cycle.

The cause for this overshoot is directly related to the usage of second-order methods. In a first-order method, the flux is approximated to be constant across the width of each cell, whereas in a second-order method, the flux is refined using an additional linear term across the width of the cell. This linear term is the cause for the overshoot in the case of steep gradient, as it may over or under-estimate the flux along a cell containing a steep gradient.

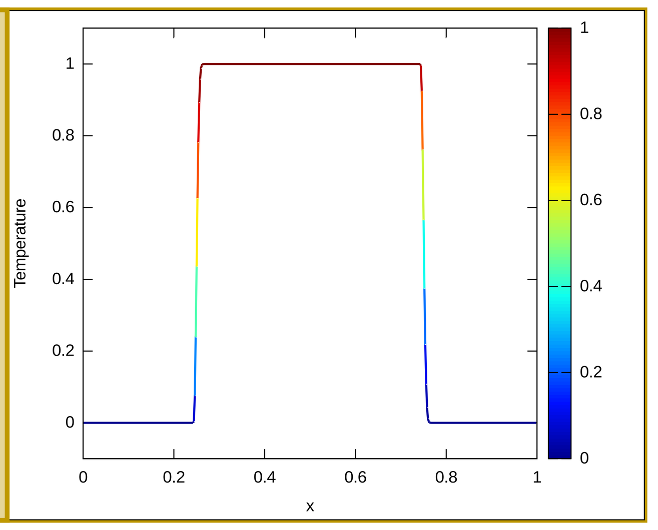


The use of a piecewise linear flux approximation may lead to non-physical overshoots and undershoots along sharp gradients, due to the tendency of these approximations to over or under-estimate the flux across a sharp edge

The solution to the overshoot problem in the finite-difference case is to use a flux-limiter which will force the method to use Upwind fluxes along sharp gradients and Fromm fluxes along smoother portions, as Upwind is better suited to working with sharp gradients and Fromm's method is best suited to working along smooth initial data. To this extent, we arrive at a new solution to the problem, given by the equation

$$u_j^{n+1} = u_j^n + \frac{v\Delta t}{\Delta x}\left(u_{j-1}^n - u_j^n\right) + \left[\left(\frac{v\Delta t}{2\Delta x} - \frac{v^2\Delta t^2}{4\Delta x^2}\right)\left(u_{j+1}^n - u_j^n - u_{j-1}^n + u_{j+2}^n\right)\right]\phi(\theta_j)$$

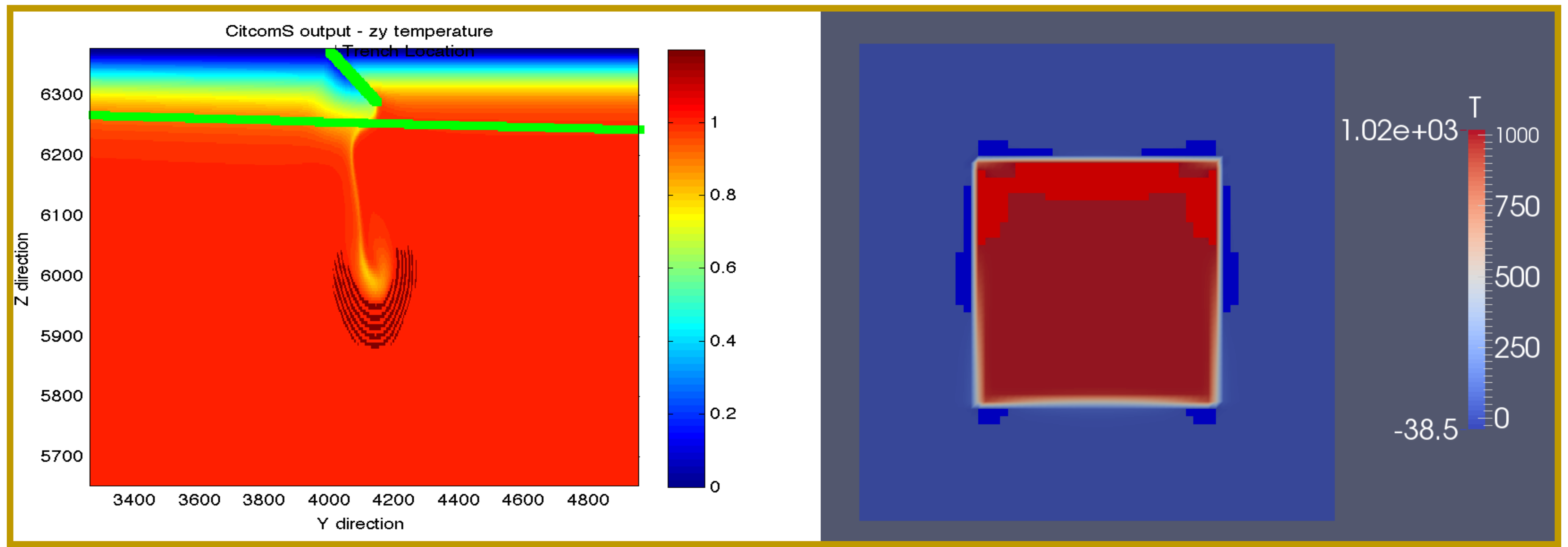
Where θ_j is an approximation to the "smoothness" of the data at the point x_j and ϕ is a function which varies between 0 and 1 depending upon the value of θ_j. This yields an entire family flux-limiter methods depending upon the choice of θ_j and ϕ. This approach does not lend itself well to a finite element implementation, as the introduction of the non-linear term ϕ works poorly with matrix-based solvers.



Fromm's method with Van-Leer flux limiters yields almost-second-order accuracy with discontinuous initial data, while preserving sharp corners much more accurately than upwind method. The initial conditions for this plot were the same as those shown previously.

Results and Conclusions

While the introduction of a nonlinear flux-limiting term is suficient to fix the overshoot problem in finite-difference codes, initial inquiries into a similar solution for finite element codes have been unsuccessful. To this end, we hope to approach the same resolution of the overshoot problem by using discontinuous Galerkin elements to prevent the same sort of over-estimation along sharp gradients that we have seen in the finite-difference case. For more information, see "Addressing the Overshoot Phenomenon in Geodynamic Codes" (DI31A-2199) by Rajesh Kommu.



Examples of overshoot from CitcomS and ASPECT, two finite element mantle convection codes. Overshoot can be seen below a falling plume in a subducting slab simulation in CitcomS (left), causing a nonphysical oscillation which appears to "push" the colder falling material back upwards. The initial temperature was strictly between 0 and 1, and physically should remain so. in ASPECT (right), a rising hot blob has overshoot trailing the leading edge and undershoot along all sides of the box. The initial temperature was strictly between 0 and 1000, and should also remain between these values.