Giving this another try. Starting with the simpler constraint given in the paper,

(1)

The idea is to begin with the requirement that

(2)

where depends on the usual diffusion and source terms, and the right-hand-side is the function specified in Eq. 1 but with replacing . The value of is arbitrary, since the intention is that it will go away by the time we’re done. If we solve the above equation for and take a Taylor expansion of Eq. 1 about , we find that really does cancel out, giving

(3)

where is meant to remind us that this is the result of a Taylor expansion. By contrast, the original algorithm (in the paper) is very similar to this,

(4)

Some notes about this:

1. The “source” term in Eq. 3 is so close to the one in Eq. 4 that I think probably they actually are, I just forgot the factor of somewhere along the way. In any case, in our simulations, so in practice these terms are identical.
2. The big difference between Eqs. 3 and 4 is . In numerical simulations, I’ve found that omission of this term altogether (i.e., use of Eq. 3) leads to unmitigated growth, i.e., no steady state.
3. It appears that the term used in the code (DoverdeltaX2) is too big by a factor on the order of 1000 or so.

The bottom line is, when DoverdeltaX2 is reduced by a factor of 1000 or so. That’s embarrassing, but the good news is, we get steady states (see Fig. 1).

|  |  |
| --- | --- |
|  |  |
|  |  |
| **Figure 1**. Steady states using the original formulation for (i.e., Eq. 4), with multiplied by a factor of 0.001 (left) and 0.002 (right). Each exhibits a diffusive slowdown of about 2%. | |