A NEW ALGORITHM FOR ALADIN-NH DYNAMICS USING SPLINES AND GREEN FUNCTIONS

Carlos Geijo, AEMET

Motivation

The SI scheme in the ALADIN-NH dynamical kernel is used to represent gravity waves and non-hydrostatic dynamics in a convenient manner for operational weather forecasting. In particular it permits to avoid the stringent CFL limitations due to fast propagating waves. Several successful formulations to solve numerically this scheme have already been presented and some of them have even been implemented. However, recent developments with FE seem to have found obstacles with the issue of the “constraints” at the time of finding a representation for the integral and differential operators that appear in this scheme. A more detailed study of the issue shows that there are also some problems with the FD scheme, like the X-term problem, stability problems that require unrealistic reference temperatures for the base state or the relegation of the GEOGW model due to, also, problems with the formulation of the operators. For all this, this new algorithm, which we could call GF scheme, may be of interest.

The main ideas

$$D+(kH)^2(N\Delta t)^2\left(-G^*[T]+G^*[P]-P-\pi_s\right)=D^\bullet \qquad D=D'\,\Delta t;\quad N^2=\frac{g}{H};\quad H=\frac{RT^*}{g}$$

$$T+\left(\frac{R}{c_v}\right)(D+d)=T^\bullet \qquad d=d'\,\Delta t$$

$$P-\left(S^*[D]-\left(\frac{c_p}{c_v}\right)(D+d)\right)=P^\bullet \qquad T=\frac{T'}{T^*}$$

$$\pi_s+N^*[D]=\pi_s^\bullet \qquad P=P'$$

$$d+(N\Delta t)^2L^*[P]=d^\bullet \qquad \pi_s=\frac{\pi_s'}{\pi_s^*}$$

On the right we have the SI system in its VDPD form. The unknowns correspond to the variables in the next time step, and the “dot terms” on the r.h.s include the dynamical, physical and SI corrections. In this form the system has been discretized in time but not yet in the vertical direction. If we carry forward the substitution process, after simple but lengthy manipulations the system is reduced to a single linear second order differential equation for the vertical divergence (Eq 1). We see that these manipulations involve derivatives only up to second order in the dot terms. Therefore C^2 smoothness is sufficient to get to this differential equation and this suggests that a discretization in the vertical coordinate by means of cubic splines may be of interest. The problem becomes how to solve this equation in this base, and the Green Function (GF) of $(-\lambda+L^*)$ is the right answer.

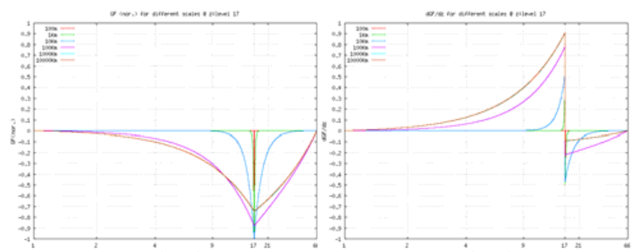
$$\left(\frac{c_p}{c_v}\right)(-\lambda+L^*)[d]=\frac{-d^\bullet}{(N\Delta t)^2}\left(1+\frac{c_p}{c_v}\mathbb{k}^2\right)+\left(\partial D^\bullet-\frac{c_p}{c_v}L^*\left[D^\bullet\right]\right) \qquad Eq1$$

$$+\,\mathbb{k}^2\left(\left(\frac{c_p}{c_v}-1\right)T^\bullet+\frac{c_p}{c_v}\partial T^\bullet\right)+L^*\left[P^\bullet\right]+\mathbb{k}^2\left(\left(1-\frac{c_p}{c_v}\right)(\partial+1)\left[P^\bullet\right]\right);\qquad \mathbb{k}^2=(kH)^2(N\Delta t)^2$$

A GF is the inverse of a differential operator. It allows to express the solution of a linear differential equation: L [u] = f and also the derivatives of this solution (up to order n-1, where n is the order of L) by means of direct integrations. For instance, if L is second order,

$$u(z)=\int_a^b GF(z,\eta)\,f(\eta)\qquad;\qquad u'(z)=\int_a^b GF'(z,\eta)\,f(\eta)$$

and u’’ is read off the defining equation. The specification of the BC for the problem is incorporated in the construction of the GF. Just on the right we have an example of the GF and GF’ for the operator in Eq 1 with homogeneous boundary conditions (BC) on the function. The example corresponds to the GF and GF’ required to compute the solution at a given point inside the domain of interest. The dependence on the horizontal scale (which comes in through the parameter λ) is apparent. The smaller the scale, the shortest the distance over which the solution picks up contributions.



The numerical approximation

Performing integrations on splines is an easy matter, but to solve the system we need to undo all the substitutions. This can be cumbersome and render the approach unpractical because the representation of the solutions obtained from these quadratures can be inefficient to handle. However there is a remarkable simple approximation to these calculations that make the algorithm very convenient. This approximation basically provides solutions that are also piecewise cubic polynomials. This point is central to the whole algorithm and it is still not fully understood. It may be possible that it is a necessary consequence of the way in which the numerical tests have been done.

Just for reference, these approximations for the function and its derivative are (with sincere apologies for not having space to explain the notation, see article in the next ALADIN-HIRLAM NewsLetters)

$$u(z)\cong\frac{-1}{\Delta}\left[Q^+_{j(z)}(B)-Q^-_{j(z)}(B)\right]+\left(-\Omega_2(z)\right)\left[Q^+_1(0)-Q^-_1(0)\right]+e^{\frac{\bar{\xi}}{2}}\Omega_1(z)\left[Q^+_{j(\bar{\xi})}(1)-Q^-_{j(\bar{\xi})}(1)\right];\quad z=\xi_{j(z)}+\Delta\xi_{j(z)}B$$

$$u'(z)\cong\frac{-1}{\Delta}\left[a^-Q^+_{j(z)}(B)-a^+Q^-_{j(z)}(B)\right]+\left(-\Omega_2'(z)\right)\left[Q^+_1(0)-Q^-_1(0)\right]+e^{\frac{\bar{\xi}}{2}}\Omega_1'(z)\left[Q^+_{j(\bar{\xi})}(1)-Q^-_{j(\bar{\xi})}(1)\right]$$

Scale	HD	VD	PD	Tm	π_s	HD	VD	PD	Tm	HD	VD	PD	Tm
100m	8	8	7	5	9	10	6	8	8	10	7	9	10
500m	8	10	8	6	9	11	8	9	10	10	8	11	10
1Km	8	10	8	6	10	10	8	9	11	10	8	10	11
10Km	7	11	9	7	12	11	10	06	11	12	10	12	10
100Km	7	8	8	7	12	8	8	06	10	11	10	11	12
1000Km	6	7	7	5	12	8	8	8	9	10	8	9	10
10000Km	7	6	6	5	12	9	7	8	8	10	8	9	10

The numerical tests

The algorithm presented here has been tested in order to assess its performance. The tests consist of a “forward calculation” where, starting from a set of arbitrary functions, the “dot terms” for the r.h.s of the NH-VDPD system are calculated and a “backward calculation” where the system is solved using this algorithm. The solutions obtained in this way are then compared with the known original functions and their accuracy is determined. The forward calculation proceeds as follows. We first project the set of arbitrary functions onto the cubic spline base. This is the set of C^2 solutions used as reference in the tests. We operate on this set with the differential and integral operators to get the “dot terms”. The results of the tests are very satisfactory. Each cell in the table shows the significant digit up to which reference and calculated value coincide for one of the tests carried out. In double precission arithmetic maximum significant digit is 14.The first set of five columns is for the 0th order derivative, the second and third sets of four columns are for the 1st and 2nd derivative respectively.