Externalized biperiodicization

Antonio Stanesic, April 2008.

1. Biperiodicization

In this document biperiodicization in ALADIN is described. Also some coupling aspects which are related to biperiodicization are mentioned. Document is mostly retyped from Dragunalescu (1995) internal memo, with some updates on computations and code structure.

1.1 Introduction

In hydrostatic numerical weather prediction models, the horizontal and vertical dimensions are well separated. The use of spectral technique for treating the horizontal part of the equations has the advantage of a more accurate computation of the horizontal derivatives in comparison with the finite difference technique. For the same reason, in spectral models fewer degrees of freedom are necessary to obtain the same accuracy. The integration area for limited area models is usually rectangular so double Fourier representation can be used and fast Fourier transform can be preformed in both directions. The problem is to formulate boundary conditions according to the basis of the expansion functions. To use Fourier series, a cyclic domain has to be assumed. The truncation used in ALADIN to obtain isotropic representation over rectangular domain is elliptical.

For ALADIN, with bi-Fourier expansion, the fields are transformed in the following way:

$$Q(x,y)=\sum_{m=-M}^{M}\sum_{n=-N}^{N} Q_m^n e^{i\cdot m\left(\frac{2p}{Lx}\right)x} e^{i\cdot n\left(\frac{2p}{Ly}\right)y}$$

In discrete formulation, the coefficients are:

$$Q_m^n=\frac{1}{JK}\sum_{j=0}^{J-1}\sum_{k=0}^{K-1} Q(x_j,x_k)\, e^{-i\cdot m\left(\frac{2\pi}{Lx}\right)x_j} e^{i\cdot n\left(\frac{2\pi}{Ly}\right)y_k}$$

where
J, K – number of points along x,y directions respectively
Lx, Ly – the periods in x,y directions respectively;
The elliptical truncation is defined by the relation:

$$\left(\frac{m}{M}\right)^2+\left(\frac{n}{N}\right)^2\leqslant 1$$

where M and N are the maximum wave numbers in the two directions.

Horizontal domain of ALADIN can be divided in three zones (Figure 1.). The first one is central zone noted **C**, witch represents the region of meteorological interest, where forecast is fully adapted to small scale conditions. Any limited area model needs information about the state of the atmosphere outside of its integration domain. In order to solve this difficulty, one defines an intermediate zone **I** – the coupling zone – where large scale solution computed with the global model ARPEGE is mixed with solution resulting from ALADIN integration.

The use of spectral methods imposes for the fields similar boundary conditions to those characterizing the basis of functions used for expansion, i.e. the fields must be periodic in the x and in the y directions (bi-periodic). An artificial zone **E** is defined like an outer belt only for this previously-mentioned mathematical reason. It size is chosen so to avoid too sharp slopes at the boundaries of the domain.

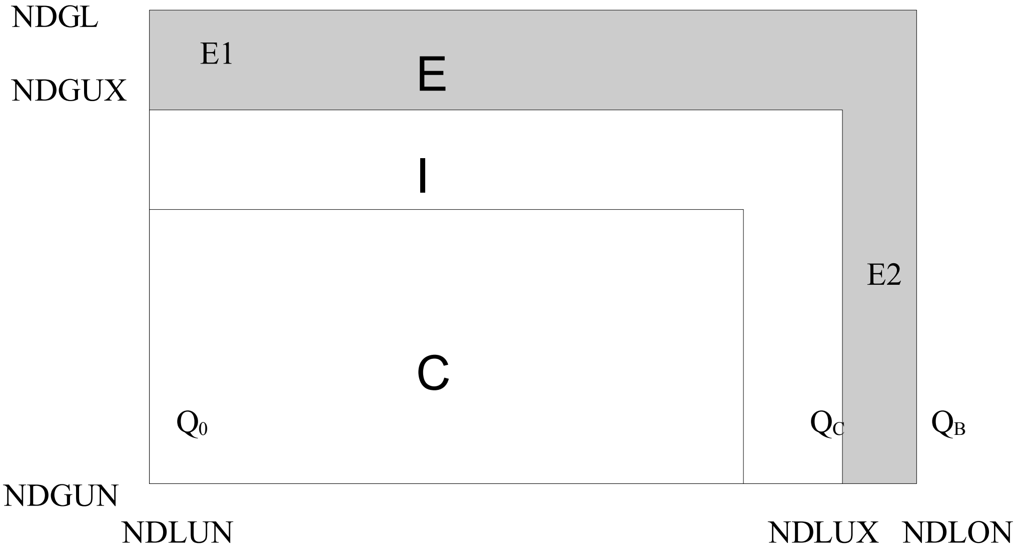

Figure 1. Horizontal representation of ALADIN domain: C – the central zone, I – the coupling zone, E – the extension zone; NDLUN,NDLUX – the lower and upper bound for the x dimension of C+I zone; NDLON – the upper bound for the x dimension of C+I+E zone; NDGUN,NDGUX – the lower and upper bound for the y dimension of C+I zone; NDGL - the upper bound for the y dimension of C+I+E zone.

1.2. Double-periodicization

When using the bi-Fourier representation, a cyclic domain is requested, but boundary values of the fields and their slopes are different at the opposite boundaries and this fact would generate discontinuities if the domain would be made biperiodic (periodic in both directions). For this reason it is necessary to find a function to extend the grid-point fields such to obtain a smooth transition between opposite boundaries. Following this, there is a need of continuity not only for fields at the borders but also for their derivatives. The conditions are:

$$Q(x_0, y)=Q(x_B, y)$$

$$Q(x, y_0)=Q(x, y_B)$$

$$\frac{\partial Q}{\partial x}(x_0, y)=\frac{\partial Q}{\partial x}(x_B, y)$$

$$\frac{\partial Q}{\partial y}(x, y_0)=\frac{\partial Q}{\partial y}(x, y_B)$$

where subscripts $[.]_0$ and $[.]_B$ denote the lower, respectively upper boundary values.

After biperiodicization in one direction (x), the domain could be seen as becoming a cylinder and after second one (y) as a torus obtained by joining opposite borders.

The algorithm implemented in ALADIN has been coded and described by R. Bubnova (1992). A biperiodic field has to satisfy the conditions:

Q(1,JY) = Q(NDLON+1,JY)
Q(JX,1) = Q(JX,NDGL+1),
with JX=1,...,NDLON; JY=1,...,NDGL.

After different tests with some operators, M. Batka developed an isotropic one that uses first cubic spline function acting row by row followed by transversal smoothing.
Biperiodicization can be divided in two steps:

- When applying the spline function, the values of a variable Q in part E_2 of the extension zone (see figure 1) will be determined row by row with the formula:

$Q_{i,j}=Q_{C,j}+N_i\cdot\Delta$ with

$$\Delta=\frac{Q_{0,j}-Q_{C,j}}{N}-\left[2\cdot\frac{2Z_1-\frac{N}{N+1}\cdot Z_2}{4-\left(\frac{N}{N+1}\right)^2}+\frac{2Z_2-\frac{N}{N+1}\cdot Z_1}{4-\left(\frac{N}{N+1}\right)^2}\right]\cdot\frac{N}{6}+\Delta_1$$

$$\Delta_1=\frac{N_i}{2}\cdot\frac{2Z_1-\frac{N}{N+1}Z_2}{4-\left(\frac{N}{N+1}\right)^2}+\frac{N_i^{\,2}}{6\mathrm{N}}\cdot\left[\frac{2Z_2-\frac{N}{N+1}Z_1}{4-\left(\frac{N}{N+1}\right)^2}-\frac{2Z_1-\frac{N}{N+1}Z_2}{4-\left(\frac{N}{N+1}\right)^2}\right]$$

and

$$Z_1=\left[\frac{Q_{0,j}-Q_{C,j}}{N}-Q_{C,j}+Q_{C-1,j}\right]\cdot\frac{6}{N+1}-\frac{\alpha}{N+1}\cdot(Q_{C,j}-2\mathrm{Q}_{C-1,j}+Q_{C-2,j})$$

$$Z_2=\left[Q_{0+1,j}-Q_{0,j}-\frac{Q_{0,j}-Q_{C,j}}{N}\right]\cdot\frac{6}{N+1}-\frac{\alpha}{N+1}\cdot(Q_{0+2,j}-2\mathrm{Q}_{0+1,j}+Q_{0,j})$$

The following notation were used:

α - boundary condition of spline :

0. - natural spline (used in both ETIBIHIE and FPBIPERE)
1. - boundary condition computed differentially (additional option)

N = NDLON-NDLUX+1,
i = NDLUX+1,...,NDLON,
j = NDGUN,...,NDGUX
N_i = i – NDLUX,
$Q_{0,j}$ = Q(NDLUN,j),
$Q_{C,j}$ = Q(NDLUX,j).
$Q_{B,j}$ = Q(NDLON,j).

Afterwards the values of the variable Q in the part E_1 of the extension zone are computed column by column with a similar formula.

- The smoothing consists in changing the grid-point values already determined with an average value obtained with following expression:

$$P_{j,i}=(4\mathrm{P}_{j,i}+2(P_{j+1,i}+P_{j-1,i}+P_{j,i+1}+P_{j,i-1})+P_{j+1,i+1}+P_{j-1,i+1}+P_{j+1,i-1}+P_{j-1,i-1})\cdot\frac{1}{16}$$

One imposes the conditions:
Q(i, NDGUN-1) = Q(i, NDGL)
Q(i, NDGL+1) =Q(i, NDGUN)

At first, the values from Q_{C+1} to Q_B are computed. The computations are repeated afterwards from Q_{C+2} to Q_{B-1} and so on, the last value obtained being in the middle of the extension zone. All calculations are made row by row and similar operations are preformed on the columns.

After spline calculation conditions for borders are fulfilled so indeed: Q(NDLUN,JY) = Q(NDLON+1,JY) and Q(JX,NDGUN) = Q(JX,NDGL+1). But after the smoothing values on bottom border in E zone are changed (for Q(i,NDGUN) i=NDLUX,...,NDLON) and also values in E zone on left border (for Q(NDLUN,j) j=NDGUX,...,NDGL) so above condition is no longer valid for this points. Nevertheless, goal of biperiodicization is accomplished – to make a smooth transition between the two ends of the domain so we minimize Gibbs effect at Fourier transform.

The initial and coupling fields are available from ARPEGE, stored in spectral form. In order to be used in the ALADIN area, these fields are transformed in real space and interpolated on the ALADIN grid. The biperiodicization is applied just after the horizontal interpolation in order to reduce steady noise that occurs when fine-mesh detailed fields are passing the spectral fits. The vertical part of interpolations acts afterwards on the whole domain C+I+E.

After biperiodicization the grid point fields are transformed in spectral space and than back to real space (spectral fitting). The weakness of the $\boldsymbol{B}$ biperiodicizaton operator is that it has not the property of projector with respect to the linear operator $\boldsymbol{L}=\boldsymbol{R}\circ\boldsymbol{F}^{-1}\circ\boldsymbol{T}\circ\boldsymbol{F}\circ\boldsymbol{B}$ and therefore the final fields are not exactly the same as initial ones. Here $\boldsymbol{F}$ and $\boldsymbol{F}^{-1}$ are the direct and inverse Fourier transforms, $\boldsymbol{T}$ is truncation operator and $\boldsymbol{R}$ is the reduction on C+I domain. The $\boldsymbol{B}$ operator was chosen to provide a solution still in reasonable range of 'projectibility'.

The width of extension zone must be chosen such as the fields over the entire domain are well represented with th current truncation. The number of points in E zone has to be less than number of degrees of freedom defined by elliptical truncation used in ALADIN in order to avoid Gibbs waves in E zone. It has not to be too small to avoid a too sharp transition in this area. In practice, a number of 12 points seems reasonable and fulfill the two above conditions.

1.3. Externalized biperiodicization

To externalize biperidocization following steps are done. First the structure of directories is created:
programs – for test programs
external – interface subroutines for calling biperidocization
modules – modules for making and smoothing
interface – interface for external programs.
build – make file and some libraries for building test program

In ALADIN, biperidocization is done in two subroutines: esplin.F90 (makes the cubic spline extension) and esmooth.F90 (smooth the fields over extension zone).
Instead of this subroutines two internal module subroutines are created: espline_mod.F90 and esmoothe_mod.F90. The externally callable routines are etibihie.F90 and fpbipere.F90. To call them explicit interface block is needed. Both subroutines preform same task (biperiodicization), only difference is layout of input/output field.

Test program is provided for testing biperiodicization by call to external subroutines. Test is not general but rather specific for this spline computation. As input it takes analytical temperature field (generated by external subroutine HORIZ_FIELD) and than transforms field in appropriate layout for ETIBIHIE and FPBIPERE. In external subroutines and module for computing spline new integer variable IADD is introduced with purpose to test correctness of spline. IADD takes two values: 1 – test, 0 – no test. If test is done then one more step in computation of spline is done (in both x and y direction) so the values for Q(NDLON+1,JY) and Q(JX,NDGL+1) are calculated. Then difference between left and right Q(1,JY)-Q(NDLON+1,JY) and top and bottom Q(JX,NDGL+1)-Q(JX,1) are calculated and written to files, for both external subroutines.

FPBIPERE – Full-POS interface for double periodicisation

Purpose
To bi-periodicise the post-processed fields.

Interface
CALL FPBIPERE(KDLUX,KDGUX,KDLON,KDGL,KNUBI,KD1,PGPBI,KDADD,LDZON)

Explicit arguments

KDLUX	- upper bound for the x (or longitude) dimension of C U I
KDGUX	- upper bound for the y (or latitude) dimension of C U I
KDLON	- upper bound for the x (or longitude) dimension of the gridpoint array on C U I U E
KDGL	- upper bound for the y (or latitude) dimension of the gridpoint array on C U I U E
KNUBI	- number of horizontal fields to doubly-periodicise.
KD1	- dimension of input/output array
PGPBI	- input/output gridpoint array on C U I U E.
LDZON	- optional argument, .TRUE. if input grid on C U I U E (.FALSE. if C U I)
KDADD	- 1 to test biperiodiz.

ETIBIHIE – Doubly-periodicisation

Purpose
KNUBI horizontal fields which are known on C U I, are extended over E, in order to obtain doubly-periodic fields. IF LDBIX is equal .TRUE. , then the fields are periodicise in the x (or longitude)

direction. If it is not the case, KDLUX must be equal to KDLON. IF LDBIY is equal .TRUE. , then the fields are periodicise in the y (or latitude) direction. If it is not the case, KDGUX must be equal to KDGL.

Interface
CALL ETIBIHIE(KDLON,KDGL,KNUBI,KDLUX,KDGUX,KSTART,KDLSM,PGPBI,
LDBIX,LDBIY,KDADD)

Explicit arguments

KDLON	- upper bound for the x (or longitude) dimension of the gridpoint array on C U I U E
KDGL	- upper bound for the y (or latitude) dimension of the gridpoint array on C U I U E
KNUBI	- number of horizontal fields to doubly-periodicise.
KDLUX	- upper bound for the x (or longitude) dimension of C U I.
KDGUX	- upper bound for the y (or latitude) dimension of C U I.
KSTART	- first dimension in x direction of g-p array
KDLSM	- second dimension in x direction of g-p array
PGPBI	- gridpoint array on C U I U E.
LDBIX	- logical to periodicize or not in the x (or longitude) direction.
LDBIY	- logical to periodicize or not in the y (or latitude) direction.
KDADD	- 1 to test biperiodiz.

2. Coupling

Description of coupling can be divided in two parts:

- organization within the model timestep – more in Davies (1976) and Radnoti (1995)
- geometrical (analytical) definition of the relaxation function – more in following text

2.1. Relaxation coefficients

The relaxation coefficient α_0 is a function depending of both x and y directions and of the type of field Q. It increases from 0 in the ***C*** region to 1 in the ***E*** region. Because of it constant value 1 in ***E*** region ARPEGE values are used in that area.
The relaxation coefficient α_0 is a function of argument z, the function and its argument taking both their values in the [0,1] interval. In ALADIN it was implemented by V. Ducrocq (1992) in the following forms:

$$\alpha_0(z)=(p+1)z^p-pz^{p+1}$$ ($\alpha_0 \rightarrow$ FEZBP(PZ, PEPA), p $\rightarrow$ REPA(JFLD))

when p>2, it corresponds to a shift of the function slope towards the outer side of inner zone, and:

$$\alpha_0(z)=1-(p+1)(1-z)^p+p(1-z)^{p+1}$$ ($\alpha_0 \rightarrow$ FEZBM(PZ, PEPA), p $\rightarrow$ REPA(JFLD))

when p< -2, it corresponds to a shift of the function slope towards the inner side of ***I*** zone.

Except in ***I*** corner areas, z is distance between the point and central ***C*** zone (Figure 2), i.e. z=y or z=x. In the corner areas z is given by:

$$z^e=x^e+y^e$$

where:

$$e = \frac{1}{\beta \cdot z^n (1-z)^m} \qquad \beta = \frac{(n+m)^{n+m}}{\mu \cdot n^n m^m}$$

with n, m, p, μ = degrees of freedom of relaxation coefficient formulation.

In the code: n → NNAL(JFLD)
m → NMAL(JFLD)
μ → NEAL(JFLD)

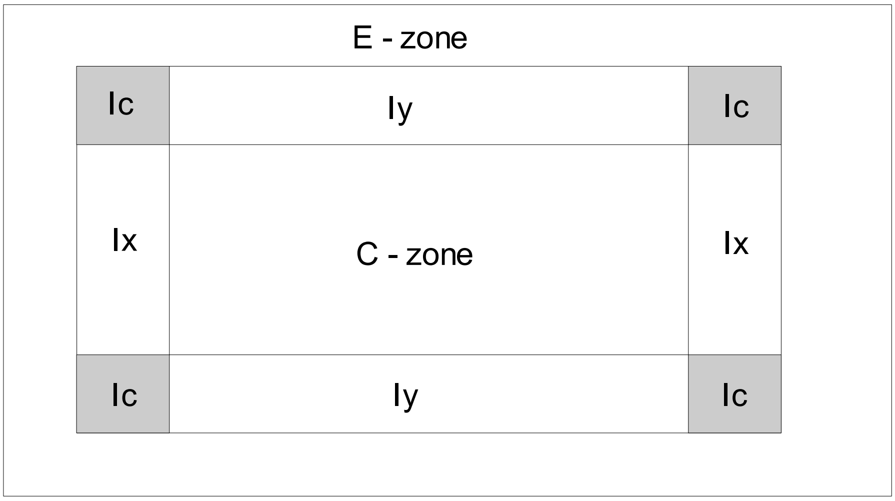

Figure 1. Definition of argument z of the coupling function: in the ***Ix*** zones z=x, in the ***Iy*** zones z=y and in the corners ***Ic*** $z=(x^e+y^e)^{1/e}$.

Different shapes of coupling functions corresponding to different sets of degrees of freedom are presented by V. Ducrocq (1992). In Figure 3., only one case is presented: the case of bidimensional coupling function corresponding to following values: m=1, n=3, μ=2, p=2.

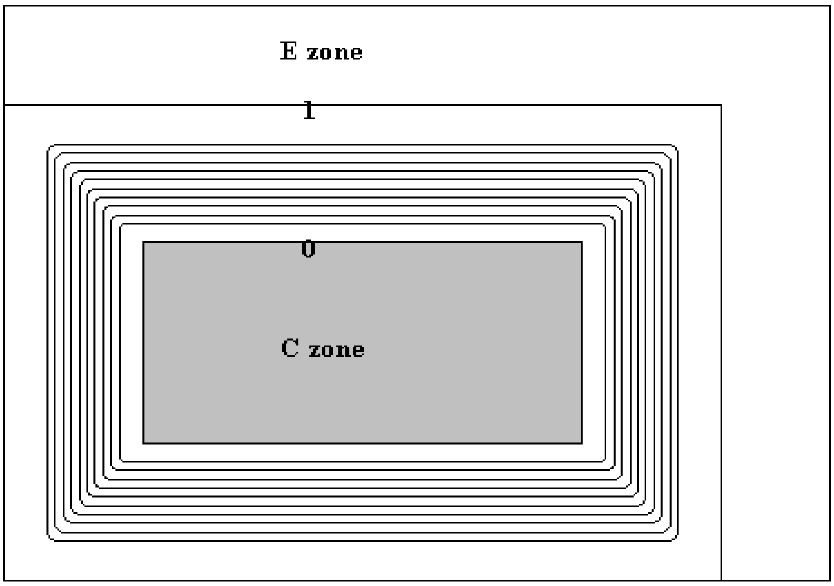

Figure 3. The coupling function $\alpha_Q(x,y)$ with m=1, n=3, μ=2, p=2.

2.2. Algorithmic aspects

The routines involved in the coupling algorithm are shown in Figure 4.

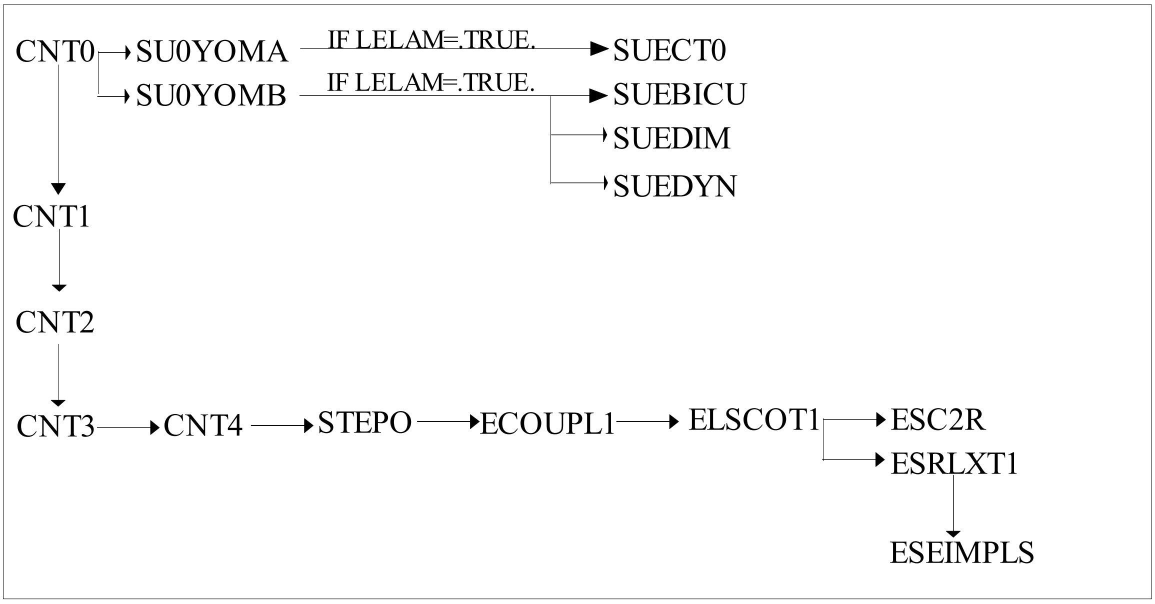

Figure 4. Flowchart of coupling routines.

CNT4 – controls the integration job at level 4; calls STEPO
STEPO – controls integration job at lowest level; calls ECOUPL1
ECOUPL1 – control routine for t1 coupling; calls ELSCOT1
ELSCOT1 – relaxation and doubly-biperiodicization at time level t1; calls ESC2R, ESRLXT1
ESC2R – reads in large scale grid-point array from I/O file of buffer
ESRLXT1 – coupling of variables at time level t1
ESEIPMLS – computes semi-implicit correction terms for large scale fields.

The coupling function α is computed on the whole domain by the subroutine SUEBICU. Afterwards the coefficients are spectrally fitted by means of routines ESPEREE and EREESPE. The parameters and switches involved in the coupling are initialized in comedecks YEMBICU, YEMDIM and YEMCT0 in routines SUEBICU, SUEDIM and SUECT0. Their values by default are:
(μ) NEAL=2, (n) NNAL=3, (m) NMAL=1, (p) ZEPA=2.16_JPRB, NBZONG=8, NBZONL=8, NECRIPL=1, TEFRCL=10800, NBICOx=1.

NBZONG, NBZONL – number of points in the coupling zone in y respective x direction
TEFRCL – time interval in seconds for coupling
NECRIPL – switch for coupling mode: 1 – t1 coupling, 0 – to coupling and 2 – both.
NBICOU – switch for coupling the dynamical variables
NBICOT – switch for coupling the temperature
NBICOP – switch for coupling the specific humidity
NBICOL – switch for coupling the surface pressure
NBICOx = 0 means no coupling and 2 means other method than default one.

Bibliography

Bubnova, R: Biperiodicization in ALADIN: Review and some explenation. December 1992. Internal memo

Davies, H.C., 1976: A lateral boundary formulation for multi-level prediction models. Quart. J. Roy. Meteorol. Soc., 102, 405-418.

Ducrocq, V: Relaxation coefficient for ARPAGE/ALADIN. August 1992. Internal memo.

Dragunalescu,L. : ARPAGE/ALADIN:Double-periodicization and coupling, February 1995. Internal memo.

Haugen, J., Machenhauer, B.: A spectral limited-area formulation with time-dependent boundary conditions applied to shallow water equations. Monthly weather review. 121. 2618-2630. 1993.

Joly, A: ARPAGE/ALADIN: adiabatic model equations and algorithm. March 1992. Internal memo.

Radnóti,G., 1995: Comments on "A spectral limited-area formulation with time-dependent boundary conditions applied to the shallow-water equations". Mon. Wea. Rev., 123, pp. 3122-3123