Prague 18.5.–14.6.2003

Reformulation of Bottom Boundary Condition for Term $\dfrac{\partial \tilde{p}}{\partial \pi}$

(report from stay)

Scientific supervisor: Radmila Brožková

Ján Mašek

SHMÚ, Jeséniova 17
833 15 Bratislava
Slovak Republic

E-mail: jan.masek@shmu.sk

Typeset by LaTeX

Contents

1 Introduction

Precision of numerical scheme depends not only on used discretizations, it can be influenced also by choice of computational boundary conditions. Aim of this stay was to test impact of one such choice in NH ALADIN. It is believed that inconsistent or imprecise bottom boundary conditons (BBC) might contribute to so called "chimney effect" in semi-lagrangian scheme.

2 Reformulation of BBC for term $\dfrac{\partial \tilde{p}}{\partial \pi}$

Term $\frac{\partial \tilde{p}}{\partial \pi}$ (where $\tilde{p} \equiv p - \pi$ is non-hydrostatic pressure departure) occurs in prognostic equations for pseudovertical divergence d and horizontal wind $\boldsymbol{v}$. In discretized system computational boundary conditions for this term are needed at model bottom and top. Bottom boundary condition will be studied here.

Warning:

In the following text it is assumed that map factor equals to one. As a consequence, metric terms are zero. In general case they would appear in expression for $\frac{\mathrm{d}}{\mathrm{d}t}\nabla\phi_S$.

2.1 Continuous case

Momentum equations written in η coordinate together with free slip boundary condition imposed at surface take the form (source terms $\boldsymbol{\mathcal{V}}$, $\mathcal{W}$ contain Coriolis acceleration):

$$\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = -\frac{RT}{p}\nabla p - \left(\frac{\partial \tilde{p}}{\partial \pi} + 1\right)\nabla\phi + \boldsymbol{\mathcal{V}} \tag{1}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}(gw) = g^2\frac{\partial \tilde{p}}{\partial \pi} + g\mathcal{W} \tag{2}$$

$$gw_S = \boldsymbol{v}_S \cdot \nabla\phi_S \tag{3}$$

Equations (1)–(3) enable to eliminate time evolution and express vertical derivative $\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S$ using only *surface* quantities $\boldsymbol{v}_S$, T_S, p_S, ϕ_S, $\boldsymbol{\mathcal{V}}_S$ and $\mathcal{W}_S$:

$$\begin{aligned}
g^2\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S &= \frac{\mathrm{d}}{\mathrm{d}t}(gw_S) - g\mathcal{W}_S = \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{v}_S \cdot \nabla\phi_S) - g\mathcal{W}_S = \\
&= \frac{\mathrm{d}\boldsymbol{v}_S}{\mathrm{d}t} \cdot \nabla\phi_S + \boldsymbol{v}_S \cdot \frac{\mathrm{d}}{\mathrm{d}t}(\nabla\phi_S) - g\mathcal{W}_S = \\
&= \left[-\frac{RT}{p}\nabla p - \left(\frac{\partial \tilde{p}}{\partial \pi} + 1\right)\nabla\phi + \boldsymbol{\mathcal{V}}\right]_S \cdot \nabla\phi_S + \underbrace{\boldsymbol{v}_S \cdot [(\boldsymbol{v}_S \cdot \nabla)\nabla\phi_S]}_{J_S} - g\mathcal{W}_S = \\
&= -\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S (\nabla\phi_S)^2 + \left[-\frac{RT}{p}\nabla p - \nabla\phi + \boldsymbol{\mathcal{V}}\right]_S \cdot \nabla\phi_S + J_S - g\mathcal{W}_S
\end{aligned}$$

$$\Downarrow$$

$$[g^2 + (\nabla\phi_S)^2]\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S = \left[-\frac{RT}{p}\nabla p - \nabla\phi + \boldsymbol{\mathcal{V}}\right]_S \cdot \nabla\phi_S + J_S - g\mathcal{W}_S$$

$$\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S = \frac{\left[-\frac{RT}{p}\nabla p - \nabla\phi + \boldsymbol{\mathcal{V}}\right]_S \cdot \nabla\phi_S + J_S - g\mathcal{W}_S}{g^2 + (\nabla\phi_S)^2} \tag{4}$$

$$\left(J_S = \frac{\partial^2\phi_S}{\partial x^2}{u_S}^2 + 2\frac{\partial^2\phi_S}{\partial x\partial y}u_S v_S + \frac{\partial^2\phi_S}{\partial y^2}{v_S}^2\right)$$

Formula (4) clearly indicates that term $\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_S$ cannot be prescribed arbitrarily. It must be consistent with other surface quantities.

2.2 Discrete case

In discrete case relation (4) cannot be applied directly, since model variables like $\boldsymbol{v}$, p and T are available only on full levels l. Their values on half levels $\tilde{l}$ must be determined by interpolation, in case of top level $\tilde{0}$ and bottom level $\tilde{L}$ by extrapolation. The aim is to use as few extrapolation rules as possible. One such rule is used for surface wind $\boldsymbol{v}_{\tilde{L}}$, which appears in free slip boundary condition (3):

$$\boldsymbol{v}_{\tilde{L}} = \boldsymbol{v}_L \tag{5}$$

Extrapolation rule (5) is sufficient for getting discrete formula analogical to (4). First steps of derivation are similar to continuous case, giving:

$$g^2\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_{\tilde{L}} = -\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_L \nabla\phi_L \cdot \nabla\phi_{\tilde{L}} + \left[-\frac{RT}{p}\nabla p - \nabla\phi + \boldsymbol{\mathcal{V}}\right]_L \cdot \nabla\phi_{\tilde{L}} + J_L - g\mathcal{W}_{\tilde{L}} \tag{6}$$

$$\left(J_L = \frac{\partial^2\phi_{\tilde{L}}}{\partial x^2}{u_L}^2 + 2\frac{\partial^2\phi_{\tilde{L}}}{\partial x\partial y}u_L v_L + \frac{\partial^2\phi_{\tilde{L}}}{\partial y^2}{v_L}^2\right)$$

First term on right hand side of equation (6) is problematic, since it contains derivative $\frac{\partial \tilde{p}}{\partial \pi}$ needed at full level L. But pressure departure $\tilde{p}$ is a full level quantity. Vertical derivative of full level quantity X cannot be evaluated directly on full level l, since difference formula requires half level values $X_{\tilde{l}}$ and $X_{\tilde{l}-1}$. Possible solution is to employ vertical averaging, e.g. linear interpolations. It can be introduced in two ways:

1. Interpolate X into half levels $\tilde{l}$ and $\tilde{l}-1$, then compute derivative at full level l.
2. Compute derivatives of X at half levels $\tilde{l}$ and $\tilde{l}-1$, then interpolate them into full level l.

Interpolation weights may be based on any vertical coordinate (z, π, η, ...). If the levels are spaced equidistantly in chosen coordinate, both approaches are equivalent. They become different for irregular level spacing. It was shown by C. Smith that second approach has generally smaller leading error term.

None of these approaches can be used directly for lowest full level L. First one requires extrapolation rule for $\tilde{p}_{\tilde{L}}$, while second one requires boundary value $\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_{\tilde{L}}$:

$$\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_L = \varepsilon_L\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_{\tilde{L}} + (1-\varepsilon_L)\left(\frac{\partial \tilde{p}}{\partial \pi}\right)_{\tilde{L}-1} \tag{7}$$

However, second approach combined with equation (6) can provide formula for term $\left(\frac{\partial\tilde{p}}{\partial\pi}\right)_{\tilde{L}}$, without introducing any other extrapolation rule. Inserting (7) into equation (6) gives:

$$
\begin{aligned}
(g^2+\varepsilon_L\nabla\phi_L\cdot\nabla\phi_{\tilde{L}})\left(\frac{\partial\tilde{p}}{\partial\pi}\right)_{\tilde{L}} = -(1-\varepsilon_L)\left(\frac{\partial\tilde{p}}{\partial\pi}\right)_{\tilde{L}-1}\nabla\phi_L\cdot\nabla\phi_{\tilde{L}} + \\
+\left[-\frac{RT}{p}\nabla p-\nabla\phi+\boldsymbol{\mathcal{V}}\right]_L\cdot\nabla\phi_{\tilde{L}}+J_L-g\mathcal{W}_{\tilde{L}}
\end{aligned}
$$

$$
\left(\frac{\partial\tilde{p}}{\partial\pi}\right)_{\tilde{L}} = \frac{-(1-\varepsilon_L)\left(\frac{\partial\tilde{p}}{\partial\pi}\right)_{\tilde{L}-1}\nabla\phi_L\cdot\nabla\phi_{\tilde{L}}+\left[-\frac{RT}{p}\nabla p-\nabla\phi+\boldsymbol{\mathcal{V}}\right]_L\cdot\nabla\phi_{\tilde{L}}+J_L-g\mathcal{W}_{\tilde{L}}}{(g^2+\varepsilon_L\nabla\phi_L\cdot\nabla\phi_{\tilde{L}})} \tag{8}
$$

Formula (8) prescribes bottom boundary condition for term $\frac{\partial\tilde{p}}{\partial\pi}$ consistently with dynamical equations. It was derived using only extrapolation rule (5). When model levels L and $\tilde{L}$ coincide, weight ε_L equals to one and formula (8) reduces to (4).

Outlined approach cannot be used for obtaining top boundary condition analogical to (8). This is due to the fact that in model formulation top geopotential $\phi_{\tilde{0}}$ depends on time. Free slip boundary condition at model top therefore takes more general form than (3):

$$
gw_T = \frac{\partial\phi_T}{\partial t}+\boldsymbol{v}_T\cdot\nabla\phi_T \tag{9}
$$

Presence of extra term $\frac{\partial\phi_T}{\partial t}$ disables elimination of time evolution from the system (1), (2), (9).

2.3 Discretization of X and Z terms

Described approaches to discretization of term $\frac{\partial\tilde{p}}{\partial\pi}$ apply also to other terms, namely X and Z:

$$
\begin{aligned}
X &= -\frac{\partial\boldsymbol{v}}{\partial\phi}\cdot\nabla\phi \\
Z &= -\frac{\partial\boldsymbol{v}}{\partial\phi}\cdot\nabla(gw)
\end{aligned}
$$

X term is part of 3-dimensional divergence D_3, while Z term occurs in prognostic equation for pseudovertical divergence d (exact form of Z depends on used d variable, given one is valid for $d=d_3$). Both terms can be symbolically written as:

$$
-\frac{\partial\boldsymbol{v}}{\partial\phi}\cdot\nabla\psi \tag{10}
$$

Velocity $\boldsymbol{v}$ is full level quantity, variable ψ is generally half level quantity and geopotential ϕ is available on both full and half levels. Term (10) is needed on full levels. Currently it is discretized using per partes rule combined with first approach:

$$
\begin{aligned}
\left(-\frac{\partial\boldsymbol{v}}{\partial\phi}\cdot\nabla\psi\right)_l &= \left[-\frac{\partial}{\partial\phi}(\boldsymbol{v}\cdot\nabla\psi)+\boldsymbol{v}\cdot\frac{\partial}{\partial\phi}\nabla\psi\right]_l = \\
&= -\frac{\boldsymbol{v}_{\tilde{l}}\cdot\nabla\psi_{\tilde{l}}-\boldsymbol{v}_{\tilde{l}-1}\cdot\nabla\psi_{\tilde{l}-1}}{\phi_{\tilde{l}}-\phi_{\tilde{l}-1}}+\boldsymbol{v}_l\cdot\frac{\nabla\psi_{\tilde{l}}-\nabla\psi_{\tilde{l}-1}}{\phi_{\tilde{l}}-\phi_{\tilde{l}-1}} = \\
&= \frac{(\boldsymbol{v}_l-\boldsymbol{v}_{\tilde{l}})\cdot\nabla\psi_{\tilde{l}}+(\boldsymbol{v}_{\tilde{l}-1}-\boldsymbol{v}_l)\cdot\nabla\psi_{\tilde{l}-1}}{\phi_{\tilde{l}}-\phi_{\tilde{l}-1}}
\end{aligned} \tag{11}
$$

Second approach leads to expression:

$$\left(-\frac{\partial \boldsymbol{v}}{\partial \phi} \cdot \nabla\psi\right)_l = \varepsilon_l \left(-\frac{\partial \boldsymbol{v}}{\partial \phi} \cdot \nabla\psi\right)_{\tilde{l}} + (1-\varepsilon_l)\left(-\frac{\partial \boldsymbol{v}}{\partial \phi} \cdot \nabla\psi\right)_{\tilde{l}-1} =$$
$$= -\varepsilon_l \frac{\boldsymbol{v}_{l+1} - \boldsymbol{v}_l}{\phi_{l+1} - \phi_l} \cdot \nabla\psi_{\tilde{l}} - (1-\varepsilon_l)\frac{\boldsymbol{v}_l - \boldsymbol{v}_{l-1}}{\phi_l - \phi_{l-1}} \cdot \nabla\psi_{\tilde{l}-1} \tag{12}$$

Formula (11) requires interpolation of velocity $\boldsymbol{v}$ into half levels together with boundary conditions for $\boldsymbol{v}_{\tilde{0}}$ and $\boldsymbol{v}_{\tilde{L}}$. They are usually chosen as:

$$\boldsymbol{v}_{\tilde{0}} = \boldsymbol{v}_1 \qquad \boldsymbol{v}_{\tilde{L}} = \boldsymbol{v}_L \tag{13}$$

Formula (12) requires boundary conditions for derivatives $\left(\frac{\partial \boldsymbol{v}}{\partial \phi}\right)_{\tilde{0}}$ and $\left(\frac{\partial \boldsymbol{v}}{\partial \phi}\right)_{\tilde{L}}$. They can be set to zero, as analogy to conditions (13).

C. Smith showed that formula (12) is generally more precise than (11). It would therefore be desirable to explore its impact on model accuracy.

2.4 Code implementation

Reformulated BBC for term $\frac{\partial \tilde{p}}{\partial \pi}$ is calculated in subroutine `GNHGRP`, just before final computation of pressure gradient term. It is then transferred as argument `PDTILP` into subroutine `GNHPDVD`, where it is used in prognostic equation for pseudovertical divergence d. Transfer must be done via level of `CPG`:

```
CPG
  |
  CPG_GP
  | |
  | GNHGRP
  |
  CPG_DYN
    |
    GNHDYN/LACDYN
      |
      GNHPDVD
```

Interpolation weights ε_l needed in equations (7), (8) are based on logarithmic pressure thicknesses α_l and δ_l:

$$\varepsilon_l = 1 - \frac{\alpha_l}{\delta_l}$$

3 Experiments

All experiments were done using 2D vertical plane model. Non-linear non-hydrostatic (NLNH) orographic flow was chosen as test case, since chimney problem occurs in this regime.

3.1 Common settings

- Initial state:
  - temperature profile with constant Brunt-Väisälä frequency $N = 0.01\,\mathrm{s}^{-1}$ up to tropopause at height 21 km, isothermal above tropopause
  - sea level temperature 293 K
  - tropopause temperature 133 K
  - constant wind profile with $V = 10\,\mathrm{ms}^{-1}$
  - sea level pressure 101 325 Pa
- Orography: Bell shaped mountain.

  height: $h = 1000\,\mathrm{m}$
  half-width: $a = 1000\,\mathrm{m}$
- Dimensionless flow parameters:

  $$C_L = \frac{Nh}{V} = 1.0 \quad (C_L \ll 1 \Rightarrow \text{linear flow})$$
  $$C_H = \frac{V}{Na} = 1.0 \quad (C_H \ll 1 \Rightarrow \text{hydrostatic flow})$$
- Geometry:

Δx	[m]	200	($a = 5\Delta x$)
Δz	[m]	≈300	(regular z-levels)
NDGUX		128	(C+I zone)
NDGL		128	(no E zone)
NBZONG		14	(I zone)
NSMAX		42	(quadratic grid)
NFLEVG		100	(30 levels above tropopause)

- Vertical coordinate: σ
- Coupling files: Identical with initial file (time constant LBC).
- Integration settings:

NPDVAR		2
NVDVAR		3
SIPR	[Pa]	90000.
REPONBT	[m]	20000.
REPONTP	[m]	29500.
VESL		0.0
XIDT		0.0

3.2 Scheme dependent settings

	euler	sl3tl	sl2tl
Δt [s]	2.5	5.0	10.0
NSTOP	2000	1000	500
REPONTAU [s]	100.	100.	50.
RCMSLP0	0.0	0.0	1.0
SITR [K]	220.	220.	300.
SITRA [K]	220.	220.	50.
LPC	OLD	OLD	FULL NESC
NSITER	1	1	3

Remark:

Due to the bug in `SUPONG`, sponge applied in 3 time level scheme is two times stonger than in 2 time level scheme (using the same absorption timescale `REPONTAU`). That is the reason why different value of `REPONTAU` was used with 2 time level scheme.

3.3 Experimental results

During preparation of clean reference experiment it was found that turning on horizontal diffusion can create chimney in vertical velocity field w even with eulerian advection. This was bit surprising, since up to now it was more or less believed that chimney problem is specific to semi-lagrangian advection.

On figures 1, 2 and 3 there is w field plotted after 2000 timesteps of eulerian integration. When horizontal diffusion is not used, field is noisy (figure 1). Using weak diffusion (`HDIRDIV=HDIRVD=25.`, `HDIRT=125.`) reduces the noise, but there are first indications of chimney formation (figure 2). With stronger diffusion (`HDIRDIV=HDIRVD=5.`, `HDIRT=25.`) fields are smooth, but chimney is fully developed (figure 3).

Several tests with eulerian scheme were performed in order to explore circumstances of chimney formation. It was recognized that:

- Chimney evolves very quickly, it can be identified after several tens of timesteps.
- Chimney is not sensitive to choice of pseudovertical divergence d.
- Chimney is not sensitive to choice of vertical coordinate (σ/η).
- Turning off the sponge does not affect chimney formation. Hovewer, integration becomes unstable without sponge.
- Chimney evolves regardless temperature is diffused or not. It is sufficient to diffuse divergence for chimney to occur.
- Stronger diffusion causes stronger chimney.

Instability occured in some tests. One possible source could be horizontal diffusion of temperature field. Diffusion is not applied directly on temperature, but on temperature reduced to constant altitude. Reduction is done approximately, assuming standard atmosphere, which is too far from stratification used in experiments. It was therefore decided not to diffuse temperature (`HDIRT=0.`).

Another situation in which instability occured is when strength of diffusion applied to horizontal and vertical divergence differs too much.

Impact of reformulated BBC is shown on figures 5–16. Vertical velocity field w after 5000 s integration is displayed. Impact is generally very weak, with almost no influence on chimney. If there is an influence, then it slightly amplifies chimney.

4 Unfinished work

Too many things remained unfinished:

- Reformulated BBC for term $\frac{\partial \tilde{p}}{\partial \pi}$ is coded assuming zero metric terms and zero Coriolis parameter. This means that code can be used only for 2D academic tests.
- I was not able to debug code with alternative discretization of X and Z terms, even if modification itself takes only few lines. There is some problem with correct array allocations needed for transfer of additional $t - \Delta t$ quantities from `CPG_GP` via `CPG` to `GNHPDVP`.

5 Conclusions

Understanding of chimney mechanism is very difficult task, since it is connected with non-linear regimes. It can be summarized that:

- Impact of reformulated BBC on model results is very weak. Influence on chimney effect is neutral or slightly amplifying. Therefore, proposed BBC reformulation seems to be useless.
- Chimney effect occurs in NLNH regimes with semi-lagrangian advection. However, when horizontal diffusion is used it can occur even with eulerian advection. It is therefore highly probable that diffusive properties of semi-lagrangian interpolators are at least partially responsible for chimney formation when semi-lagrangian advection is used.
- For the moment there are two known ways how to suppress chimneys:

    1. advection of vertical velocity w instead of pseudovertical divergence d (implemented by C. Smith)
    2. advective BBC treatment for term $\frac{\partial \tilde{p}}{\partial \pi}$ (implemented by P. Smolíková)

    Both solutions are implemented only for semi-lagrangian advection scheme. First one indicates that chimneys might be caused by inconsistency between prognostic equation for d and free slip boundary conditions, while second one indicates that they might be generated by imprecise BBC treatment of some terms. Definite answer is not known for the moment.

6 Code info

Modifications were done on top of cycle `25t2`. Two versions of code were used:

`00` = reference version, including bugfix from J. Vívoda

`01` = `00` + reformulated BBC for term $\frac{\partial \tilde{p}}{\partial \pi}$

Modified sources (`voodoo`):

```
~mma157/utemp/cycle_25t2/mod_00_ald/
                         mod_00_arp/
                         mod_01d00_ald/
                         mod_01d00_arp/
```

Sources + dependencies for compilation (`voodoo`):

```
~mma157/utemp/cycle_25t2/dep_00_ald/
                         dep_00_arp/
                         dep_01_ald/
                         dep_01_arp/
```

Loading scripts (`voodoo`):

```
~mma157/utemp/cycle_25t2/load/load_00_sx4
                              load_01_sx4
```

Executables (`archiv`):

```
~mma157/bin/master_al25t2_00_sx4
            master_al25t2_01_sx4
```

Integration scripts (`sx6`):

```
~mma157/m2d/exp/script_03/
```