Implementation of momentum budget into DALES

Arnold Moene (WUR), David Pino (UPC)

January 19, 2010

Abstract

This report discusses the details of implementing the momentum budget in the Dutch Atmospheric LES (DALES) code version 3.2 following the formulation proposed by Gao et al. (1994).

1 User information

In the DALES code V3.2 an additional subroutine (*modstress.f90*) has been included. Moreover, a section called NAMSTRESS has been created in the *namoptions.exp* file which include a switch (lstress) to calculate all the terms of the stress tensor. It's important to notice that the subroutine calculates and writes, in ASCII or NetCDF, the file *stressbudget.exp* the budget for the elements u'^2, $u'v'$, $u'w'$, v'^2, $v'w'$, and w'^2 of the stress tensor. Finally, at the end of the file the sum of the elements of the diagonal divided by 2 is written. This corresponds to the TKE budget calculated in subroutine *modbudget.f90.*

2 Momentum budget

The prognostic equation for the turbulent fluctuations, u'_i, reads (Stull, 1988, chapter 4, eq. 4.1.1):

$$\frac{\partial u'_i}{\partial t}+\overline{u}_k\frac{\partial u'_i}{\partial x_k}+u'_k\frac{\partial \overline{u}_i}{\partial x_k}+u'_k\frac{\partial u'_i}{\partial x_k}=\delta_{i3}\left(\frac{\theta'_v}{\overline{\theta}_v}\right)g+f_c\epsilon_{ik3}u'_k-\frac{1}{\overline{\rho}}\frac{\partial p'}{\partial x_i}+\left(\nu\frac{\partial^2 u'_i}{\partial x_k^2}\right)'+\frac{\partial(\overline{u'_i u'_k})}{\partial x_k}.$$

By using the condition of incompressibility, $\partial u'_k/\partial x_k=0$, this equation can be written:

$$\frac{\partial u'_i}{\partial t}+\frac{\partial \overline{u}_k u'_i}{\partial x_k}+\frac{\partial u'_k\overline{u}_i}{\partial x_k}+\frac{\partial u'_k u'_i}{\partial x_k}=\delta_{i3}\left(\frac{\theta'_v}{\overline{\theta}_v}\right)g+f_c\epsilon_{ik3}u'_k-\frac{1}{\overline{\rho}}\frac{\partial p'}{\partial x_i}+\left(\nu\frac{\partial^2 u'_i}{\partial x_k^2}\right)'+\frac{\partial(\overline{u'_i u'_k})}{\partial x_k}. \tag{1}$$

Since we are using an LES model that uses a subgrid model that is expressed in terms of a subgrid-scale viscosity (Heus et al., 2010), the ν in the above equation should be interpreted as the subgrid-scale viscosity (with the molecular viscosity being negligible).

From this equation the different stresses can be built up. The next step is to multiply 1 by the fluctuation u'_j and perform Reynolds averaging. Note that the last term in 1 dissapears because $\overline{u'_i} = 0$ and $\partial/\partial x_k = 0$ for $k = 1, 2$:

$$\begin{aligned}&\underbrace{\overline{u'_j \frac{\partial u'_i}{\partial t}}}_{a1} + \underbrace{\overline{u'_j \frac{\partial \overline{u}_k u'_i}{\partial x_k}}}_{b1} + \underbrace{\overline{u'_j \frac{\partial u'_k \overline{u}_i}{\partial x_k}}}_{c1} + \underbrace{\overline{u'_j \frac{\partial u'_k u'_i}{\partial x_k}}}_{d1} = \\ &\qquad \underbrace{\overline{u'_j \delta_{i3} \left(\frac{\theta'_v}{\overline{\theta}_v}\right) g}}_{e1} + \underbrace{\overline{u'_j f_c \epsilon_{ik3} u'_k}}_{f1} - \underbrace{\overline{u'_j \frac{1}{\overline{\rho}} \frac{\partial p'}{\partial x_i}}}_{g1} + \underbrace{\overline{u'_j \nu \left(\frac{\partial^2 u'_i}{\partial x_k^2}\right)'}}_{h1}\end{aligned} \tag{2}$$

The above equation can also be rewritten with the i and j indices interchanged, giving:

$$\begin{aligned}&\underbrace{\overline{u'_i \frac{\partial u'_j}{\partial t}}}_{a2} + \underbrace{\overline{u'_i \frac{\partial \overline{u}_k u'_j}{\partial x_k}}}_{b2} + \underbrace{\overline{u'_i \frac{\partial u'_k \overline{u}_j}{\partial x_k}}}_{c2} + \underbrace{\overline{u'_i \frac{\partial u'_k u'_j}{\partial x_k}}}_{d2} = \\ &\qquad \underbrace{\overline{u'_i \delta_{j3} \left(\frac{\theta'_v}{\overline{\theta}_v}\right) g}}_{e2} + \underbrace{\overline{u'_i f_c \epsilon_{jk3} u'_k}}_{f2} - \underbrace{\overline{u'_i \frac{1}{\overline{\rho}} \frac{\partial p'}{\partial x_j}}}_{g2} + \underbrace{\overline{u'_i \left(\nu \frac{\partial^2 u'_j}{\partial x_k^2}\right)'}}_{h2}\end{aligned} \tag{3}$$

Then by adding 2 and 3 these can be combined, and by using the product rule, the equation for the tendency of the momentum flux $\overline{u'_i u'_j}$ is obtained. The equation reads:

$$\begin{aligned}&\underbrace{\frac{\partial \overline{u'_i u'_j}}{\partial t}}_{a} + \underbrace{\overline{u}_k \frac{\partial \overline{u'_j u'_i}}{\partial x_k}}_{b} + \underbrace{\overline{u'_k u'_j} \frac{\partial \overline{u}_i}{\partial x_k}}_{c} + \underbrace{\overline{u'_k u'_i} \frac{\partial \overline{u}_j}{\partial x_k}}_{c} + \underbrace{\frac{\partial \overline{u'_k u'_j u'_i}}{\partial x_k}}_{d} = \\ &\qquad \underbrace{\left(\frac{g}{\overline{\theta}_v}\right)\left(\overline{u'_j \theta'_v \delta_{i3}} + \overline{u'_i \theta'_v \delta_{j3}}\right)}_{e} + \underbrace{f_c \left(\overline{\epsilon_{ik3} u'_j u'_k} + \overline{\epsilon_{jk3} u'_i u'_k}\right)}_{f} - \\ &\qquad \underbrace{\overline{u'_j \frac{1}{\overline{\rho}} \frac{\partial p'}{\partial x_i}} + \overline{u'_i \frac{1}{\overline{\rho}} \frac{\partial p'}{\partial x_j}}}_{g} + \underbrace{\overline{u'_j \nu \left(\frac{\partial^2 u'_i}{\partial x_k^2}\right)'} + \overline{u'_i \nu \left(\frac{\partial^2 u'_j}{\partial x_k^2}\right)'}}_{h}\end{aligned} \tag{4}$$

The various terms in equation 4 are:

a tendency/storage

b advection

c shear production

d transport turbulent diffusion

e buoyancy production/destruction

f Coriolis production/destruction

g pressure-gradient velocity interaction

h viscous dissipation

The essential point in the method of Gao et al. (1994) is that the discretization of 4 will give different results than discretization of the sum of 2 and 3.

3 Discretization

In order to discretize the sum of 2 and 3 we have to take into account the discretization of the terms in the momentum budget equation (4). The rate equation for u_i' should be discretized in the same way as that of the instantaneous velocity. Therefore, 2 and 3 are the starting point for the discretization. Since, equations 2 and 3 are identical in form, we will only discuss 2:

$$\underbrace{\overline{u_j'\frac{\partial u_i'}{\partial t}}}_{a1} + \underbrace{\overline{u_j'\frac{\partial \overline{u}_k u_i'}{\partial x_k}}}_{b1} + \underbrace{\overline{u_j'\frac{\partial u_k'\overline{u}_i}{\partial x_k}}}_{c1} + \underbrace{\overline{u_j'\frac{\partial u_k' u_i'}{\partial x_k}}}_{d1} =$$

$$\underbrace{\overline{\delta_{i3}\left(\frac{u_j'\theta_v'}{\overline{\theta}_v}\right)g}}_{e1} + \underbrace{\overline{f_c\epsilon_{ik3}u_j'u_k'}}_{f1} - \underbrace{\overline{u_j'\frac{1}{\overline{\rho}}\frac{\partial p'}{\partial x_i}}}_{g1} + \underbrace{\overline{u_j'\nu\frac{\partial^2 u_i'}{\partial x_k^2}}}_{h1}$$

The DALES uses a finite volume discretization with a staggered grid Harlow and Welch (1965) (see figure 1).

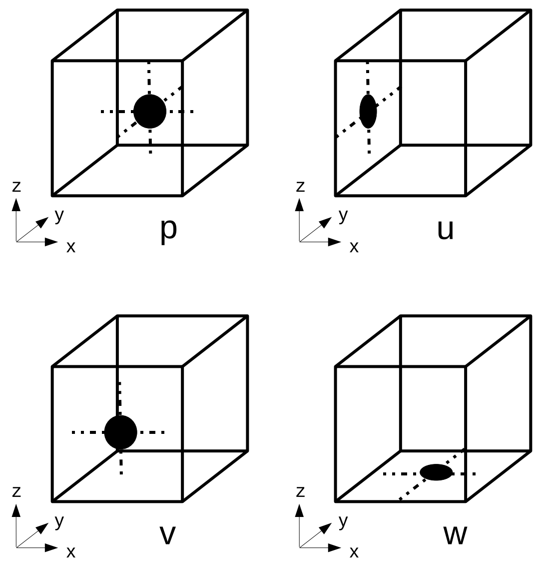

Figure 1: Arrangement of variables on a staggered grid: pressure (and other scalars like potential temperature, humidity, ...) are located in the center of the cell, velocity components are displaced in the upstream direction of the component.

The turbulence stresses in 2 are defined at the edges of the grid box where the faces of the u_i and u_j velocity intersect. Below, on the the terms in 2 are dealt with, but the total budget term will consist of a term originating from 2 and one from 3. See figure 2 to understand how the off-diagonal stresses are interpolated to cell edges.

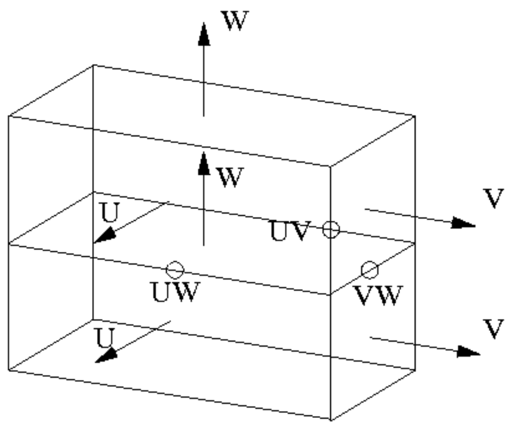

Figure 2: Interpolation of off-diagonal stresses to cell edges.

For all the terms of the stress budget, the codification at DALES of $\overline{u_1' u_1'}$,

$\overline{u_1'u_2'}$ and $\overline{u_1'u_3'}$ stress elements are written as:

```
dumfield=2.*u0_dev*u_term

dumfield(2:i1,2:j1,:) =
0.25*(u0_dev(2:i1,2:j1,:)+u0_dev(2:i1,1:jmax,:))*&
     (v_term(2:i1,2:j1,:)+v_term(1:imax,2:j1,:))+&
0.25*(u_term(2:i1,2:j1,:)+u_term(2:i1,1:jmax,:))*&
     (v0_dev(2:i1,2:j1,:)+v0_dev(1:imax,2:j1,:))

dumfield(2:i1,2:j1,2:k1) =
0.25*(u0_dev(2:i1,2:j1,2:k1)+u0_dev(2:i1,2:j1,1:kmax))* &
     (w_term(2:i1,2:j1,2:k1)+w_term(1:imax,2:j1,2:k1))+ &
0.25*(u_term(2:i1,2:j1,2:k1)+u_term(2:i1,2:j1,1:kmax))* &
     (w0_dev(2:i1,2:j1,2:k1)+w0_dev(1:imax,2:j1,2:k1)),
```

where `u0_dev`$=u_1'$, `v0_dev`$=u_2'$, `w0_dev`$=u_3'$, $i1 = imax + 1$, $j1 = jmax + 1$, and $k1 = kmax + 1$. `u_term`, `v_term`, and `w_term` are calcualted with the advection subroutines of the model and are different for each particular term of the stress budget.

3.1 Advection (b1)

The advection term, $\overline{u_j' \frac{\partial \overline{u_k} u_i'}{\partial x_k}}$, is identical zero if the mean vertical velocity ($\overline{w}$)is zero, and the flow is horizontally homogeneous.

In this case `u_term`$=\frac{\partial \overline{u_k} u_1'}{\partial x_k}$, and `v_term`$=\frac{\partial \overline{u_k} u_2'}{\partial x_k}$, `w_term`$=\frac{\partial \overline{u_k} u_3'}{\partial x_k}$.

3.2 Shear production (c1)

In the term $\overline{u_j' \frac{\partial u_k' \overline{u_i}}{\partial x_k}}$, the term $\frac{\partial \overline{u}_k u_i'}{\partial x_k}$ is evaluated at the u_i-point and needs to be interpolated to the ij edge by averaging in the j-direction. u_j' is available at the u_j-point and is interpolated to the ij edge by averaging in the i-direction.

For this term, `u_term`$=\frac{\partial u_k' \overline{u_1}}{\partial x_k}$, `v_term`$=\frac{\partial u_k' \overline{u_2}}{\partial x_k}$, `w_term`$=\frac{\partial u_k' \overline{u_3}}{\partial x_k}$.

3.3 Turbulent diffusion (d1)

In the term $\overline{u_j' \frac{\partial u_k' u_i'}{\partial x_k}}$, the term $\frac{\partial u_k' u_i'}{\partial x_k}$ is available at the u_i-point and needs to be interpolated to the ij edge by averaging in the j-direction. Again, u_j' is available at the u_j-point and is interpolated to the ij edge by averaging in the i-direction.

The codification of this term implies `u_term`$=\frac{\partial u_k' u_1'}{\partial x_k}$, `v_term`$=\frac{\partial u_k' u_2'}{\partial x_k}$, `w_term`$=\frac{\partial u_k' u_3'}{\partial x_k}$.

3.4 Buoyancy production/destruction (e1)

In the term $\overline{u_j' \delta_{i3} \left(\frac{\theta_v'}{\overline{\theta}_v}\right) g}$, the term $\delta_{i3} \left(\frac{\theta_v'}{\overline{\theta}_v}\right) g$ is available at the u_i-point and is interpolated to the correct edge by averaging in the j-direction. u_j' is available at the u_j-point and is interpolated to the ij edge by averaging in the i-direction.

This term has only component in the z-direction. The `w_term` is codified as:

```
do k=1,kmax
thv0h_avl(k) =  sum(thv0h(2:i1,2:j1,k))/rslabs
enddo

call MPI_ALLREDUCE(thv0h_avl, thv0h_av, k1,  MY_REAL, &
                              MPI_SUM, comm3d,mpierr)

do k=1,k1
w_term(:,:,k) = (grav/thvs) * (thv0h(2-ih:i1+ih,2-jh:j1+jh,k) - thv0h_av(k))
enddo

call cyclicx(w_term)
```

where $k1 = kmax + 1$, grav is the acceleration of gravity and thvs is mean virtual potential temperature in the boundary layer defined in *modsurface.f90.*

3.5 Coriolis production/destruction (f1)

In the term $\overline{u'_j f_c \epsilon_{ik3} u'_k}$, $f_c \epsilon_{ik3} u'_k$ is defined at the u_i-point and is interpolated to the correct edge by averaging in the j-direction. u'_j is available at the u_j-point and is interpolated to the ij edge by averaging in the i-direction.

In this case:

```
u_term(2:i1,2:j1,1:kmax) =   &
+(v0_dev(2:i1,2:j1,1:kmax)+v0_dev(2:i1,3:j2,1:kmax) &
+ v0_dev(1:imax,2:j1,1:kmax)+v0_dev(1:imax,3:j2,1:kmax))*om23*0.25 &
-(w0_dev(2:i1,2:j1,1:kmax)+w0_dev(2:i1,2:j1,2:k1) &
+w0_dev(1:imax,2:j1,2:k1)+w0_dev(1:imax,2:j1,1:kmax))*om22*0.25

v_term(2:i1,2:j1,:) =  &
-(u0_dev(2:i1,2:j1,:)+u0_dev(2:i1,1:j1-1,:)&
+ u0_dev(3:i2,1:j1-1,:)+u0_dev(3:i2,2:j1,:))*om23*0.25

do k=2,k1
w_term(2:i1,2:j1,k) =  &
om22 * 0.25*((dzf(k-1) * (u0_dev(2:i1,2:j1,k-1)  + u0_dev(3:i2,2:j1,k-1))&
+ dzf(k) * (u0_dev(2:i1,2:j1,k) + u0_dev(3:i2,2:j1,k))) / dzh(k))
enddo
```

where $i2 = imax+2$, $j2 = jmax+2$ and $om22 = 2\Omega\cos\phi$, and $om23 = 2\Omega\sin\phi$ are already defined in the subroutine *modglobal.f90.*

3.6 Pressure-gradient velocity interaction (g1)

In the term $\overline{u'_j \frac{1}{\rho}\frac{\partial p'}{\partial x_i}}$, $\frac{1}{\rho}\frac{\partial p'}{\partial x_i}$ is defined at the u_i-point and is interpolated to the correct edge by averaging in the j-direction. u'_j is available at the u_j-point and is interpolated to the ij edge be averaging in the i-direction.

For this case:

```
u_term(2:i1,2:j1,1:k1) = -(p(2:i1,2:j1,1:k1)-p(1:imax,2:j1,1:k1))*dxi
v_term(2:i1,2:j1,1:k1) = -(p(2:i1,2:j1,1:k1)-p(2:i1,1:jmax,1:k1))*dyi

do k=2,k1
w_term(2:i1,2:j1,k) = -(p(2:i1,2:j1,k)-p(2:i1,2:j1,k-1))/dzh(k)
enddo
```

where $dxi = 1/\Delta x$ and $dyi = 1/\Delta y$.

3.7 Viscous dissipation (h1)

In the term $\overline{u'_j \nu \frac{\partial^2 u'_i}{\partial x_k^2}}$, $\nu\frac{\partial^2 u'_i}{\partial x_k^2}$ is defined at the u_i-point and is interpolated to the correct edge by averaging in the j-direction. u'_j is available at the u_j-point and is interpolated to the ij edge by averaging in the i-direction. Note that in the LES model, ν includes (or in fact: *only* includes) the subgrid viscosity.

The calculation of the `u_term, v_term and w_term` is as follows:

```
call diffu(u_term)
call diffv(v_term)
call diffw(w_term)

call cyclicx(u_term)
call cyclicx(v_term)
call cyclicx(w_term)

do k=1,k1
   uterm_avl(k) = sum(u_term(2:i1,2:j1,k))/rslabs
   vterm_avl(k) = sum(v_term(2:i1,2:j1,k))/rslabs
   wterm_avl(k) = sum(w_term(2:i1,2:j1,k))/rslabs
enddo

call MPI_ALLREDUCE(uterm_avl, uterm_av, k1,  MY_REAL, &
                         MPI_SUM, comm3d, mpierr)
call MPI_ALLREDUCE(vterm_avl, vterm_av, k1,  MY_REAL, &
                         MPI_SUM, comm3d, mpierr)
call MPI_ALLREDUCE(wterm_avl, wterm_av, k1,  MY_REAL, &
                         MPI_SUM, comm3d, mpierr)
```

```
do k=1,k1
    u_term(:,:,k) = u_term(:,:,k) - uterm_av(k)
    v_term(:,:,k) = v_term(:,:,k) - vterm_av(k)
    w_term(:,:,k) = w_term(:,:,k) - wterm_av(k)
enddo
```

References

Gao, S., Z. Yang, and P. Voke: 1994, 'Balance Equations in Finite-Volume Large-Eddy Simulations'. Departmental Working Paper ME-FD/94.27, Department of Mechanical Engineering, Surrey University.

Harlow, F. and J. Welch: 1965, 'Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface'. *Phys. Fluids* **8**(12), 2182–2189.

Heus, T., van Heerwaarden C.C., H. Jonker, A. Siebesma, S. Axelsen, K. van den Dries, O. Geoffroy, A. Moene, D. Pino, S. de Roode, and J. Vil-Guerau de Arellano: 2010, 'Formulation and numerical studies by the Dutch Atmospheric Large-Eddy Simulation (DALES)'. *Geosci. Model Dev.*

Stull, R.: 1988, *An introduction to boundary layer meteorology.* Dordrecht, Boston and London: Kluwer Academic Publishers.