Exercise - Topography

PALM group

Institute of Meteorology and Climatology, Leibniz Universität Hannover

last update: 21st September 2015

Exercise

Please carry out **two runs** with following conditions.

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo
- Neutral boundary layer in a channel

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo
- Neutral boundary layer in a channel
- Constant bulk velocity

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo
- Neutral boundary layer in a channel
- Constant bulk velocity
- No Coriolis force

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo
- Neutral boundary layer in a channel
- Constant bulk velocity
- No Coriolis force
- Simulation features:
  - domain size: $(80\ \text{m})^3$ (x/y/z)
  - grid size: 2 m equidistant
  - cube: size $(40\ \text{m})^3$, location centered in the domain center
  - simulated time: 7200 s
  - initial velocity: $u = 1,\ v = 0$ m/s

Exercise

Please carry out **two runs** with following conditions.

- Single cube
  1.) First run "generic" using `topography = 'single_building'`
  2.) Second run "raster" using `topography = 'read_from_file'` with ASCII file ..._topo
- Neutral boundary layer in a channel
- Constant bulk velocity
- No Coriolis force
- Simulation features:
  - domain size: $(80\text{ m})^3$ (x/y/z)
  - grid size: 2 m equidistant
  - cube: size $(40\text{ m})^3$, location centered in the domain center
  - simulated time: 7200 s
  - initial velocity: $u = 1$, $v = 0$ m/s

Please use the same building (size, location) for both runs!

Questions to be Answered

Questions to be Answered

1. Can you identify any interesting flow patterns around the cube and what do they tell us?
   - What kind of output do you need to answer this?

Questions to be Answered

1. Can you identify any interesting flow patterns around the cube and what do they tell us?
    - What kind of output do you need to answer this?
2. How do the horizontally and temporally averaged velocity and momentum flux profiles look like?
    - How long should the averaging time interval be?

Questions to be Answered

1. Can you identify any interesting flow patterns around the cube and what do they tell us?
   - What kind of output do you need to answer this?
2. How do the horizontally and temporally averaged velocity and momentum flux profiles look like?
   - How long should the averaging time interval be?
3. Is it really a fully developed large-eddy simulation?
   - Are the subgrid-scale fluxes much smaller than the resolved-scale fluxes?
   - How do the total kinetic energy and the maximum velocity components change with time?

Questions to be Answered

1. Can you identify any interesting flow patterns around the cube and what do they tell us?
   - What kind of output do you need to answer this?
2. How do the horizontally and temporally averaged velocity and momentum flux profiles look like?
   - How long should the averaging time interval be?
3. Is it really a fully developed large-eddy simulation?
   - Are the subgrid-scale fluxes much smaller than the resolved-scale fluxes?
   - How do the total kinetic energy and the maximum velocity components change with time?
4. **Final question:** Do the results of both runs agree?

Hints (I)

Hints (I)

- **Domain size**
  - Is controlled by grid size (**dx**, **dy**, **dz**) and number of grid points (**nx**, **ny**, **nz**). Since the first grid point along one of the directions has index 0, the total number of grid points used are **nx**+1, **ny**+1, **nz**+1. The total domain size in case of cyclic horizontal boundary conditions is (**nx**+1)·**dx**, (**ny**+1)·**dy**.

Hints (I)

- **Domain size**
  - Is controlled by grid size (**dx**, **dy**, **dz**) and number of grid points (**nx**, **ny**, **nz**). Since the first grid point along one of the directions has index 0, the total number of grid points used are **nx**+1, **ny**+1, **nz**+1. The total domain size in case of cyclic horizontal boundary conditions is (**nx**+1)·**dx**, (**ny**+1)·**dy**.
- **Initial profiles**
  - Constant with height. See parameter **initializing_actions** for available initialization methods. See **ug_surface**, **vg_surface** for initial values of velocity.

Hints (I)

- **Domain size**
  - Is controlled by grid size (**dx**, **dy**, **dz**) and number of grid points (**nx**, **ny**, **nz**). Since the first grid point along one of the directions has index 0, the total number of grid points used are **nx**+1, **ny**+1, **nz**+1. The total domain size in case of cyclic horizontal boundary conditions is (**nx**+1)·**dx**, (**ny**+1)·**dy**.
- **Initial profiles**
  - Constant with height. See parameter **initializing_actions** for available initialization methods. See **ug_surface**, **vg_surface** for initial values of velocity.
- **Boundary conditions**
  - For channel boundary condition, see **bc_uv_t**.

Hints (I)

- **Domain size**
  - Is controlled by grid size (**dx**, **dy**, **dz**) and number of grid points (**nx**, **ny**, **nz**). Since the first grid point along one of the directions has index 0, the total number of grid points used are **nx**+1, **ny**+1, **nz**+1. The total domain size in case of cyclic horizontal boundary conditions is (**nx**+1)·**dx**, (**ny**+1)·**dy**.
- **Initial profiles**
  - Constant with height. See parameter **initializing_actions** for available initialization methods. See **ug_surface**, **vg_surface** for initial values of velocity.
- **Boundary conditions**
  - For channel boundary condition, see **bc_uv_t**.
- **Forcing**
  - For constant bulk velocity, see **conserve_volume_flow**.
  - For Coriolis force, see **omega**.
  - For neutral flow, see **neutral**.

Hints (I)

- **Domain size**
  - Is controlled by grid size (**dx**, **dy**, **dz**) and number of grid points (**nx**, **ny**, **nz**). Since the first grid point along one of the directions has index 0, the total number of grid points used are **nx**+1, **ny**+1, **nz**+1. The total domain size in case of cyclic horizontal boundary conditions is (**nx**+1)·**dx**, (**ny**+1)·**dy**.
- **Initial profiles**
  - Constant with height. See parameter **initializing_actions** for available initialization methods. See **ug_surface**, **vg_surface** for initial values of velocity.
- **Boundary conditions**
  - For channel boundary condition, see **bc_uv_t**.
- **Forcing**
  - For constant bulk velocity, see **conserve_volume_flow**.
  - For Coriolis force, see **omega**.
  - For neutral flow, see **neutral**.
- **Topography**
  - For generic topography, see **building_height**, **building_length_x** and **building_length_y**.
  - For raster topography, please use a text editor to manually create an ASCII "raster_topo" file that contains the same building.

Hints (II)

Hints (II)

- **Simulation time**
  - See parameter **end_time**.

Hints (II)

- **Simulation time**
  - See parameter **end_time**.
- **Variables**
  - Output variables are chosen with parameters **data_output** (3d-data or 2d-cross-sections) and **data_output_pr** (profiles).
  - Time series are activated using **dt_dots**.

Hints (II)

- **Simulation time**
  - See parameter **end_time**.
- **Variables**
  - Output variables are chosen with parameters **data_output** (3d-data or 2d-cross-sections) and **data_output_pr** (profiles).
  - Time series are activated using **dt_dots**.
- **Output intervals**
  - Output intervals are set with parameter **dt_data_output**. This parameter affects all output (cross-sections, profiles, etc.). Individual temporal intervals for the different output quantities can be assigned using parameters **dt_do3d**, **dt_do2d_xy**, **dt_do2d_xz**, **dt_do2d_yz**, **dt_dopr**, etc.

Hints (II)

- **Simulation time**
  - See parameter **end_time**.
- **Variables**
  - Output variables are chosen with parameters **data_output** (3d-data or 2d-cross-sections) and **data_output_pr** (profiles).
  - Time series are activated using **dt_dots**.
- **Output intervals**
  - Output intervals are set with parameter **dt_data_output**. This parameter affects all output (cross-sections, profiles, etc.). Individual temporal intervals for the different output quantities can be assigned using parameters **dt_do3d**, **dt_do2d_xy**, **dt_do2d_xz**, **dt_do2d_yz**, **dt_dopr**, etc.
- **Time averaging**
  - Time averaging is controlled with parameters **averaging_interval**, **averaging_interval_pr**, **dt_averaging_input**, **dt_averaging_input_pr**.

Further Hints

Please see under
http://palm.muk.uni-hannover.de/trac/wiki/doc/app/netcdf

where the complete PALM netCDF-data-output and the respective steering parameters are described.

For topography, see
http://palm.muk.uni-hannover.de/trac/wiki/doc/app/inipar#topo

and especially for raster topography, see also
http://palm.muk.uni-hannover.de/trac/wiki/doc/app/iofiles#TOPOGRAPHY_DATA

as well as the presentation "Using topography (I)".

Proceeding

Please proceed as follows:

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.
2. Check your results to answer all questions – except the final question.

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.
2. Check your results to answer all questions – except the final question.
3. After this run has finished, use ncview, ncdump etc. to check the precise location of the building (look at 2D array *zusi* that is contained in 2D xy cross-sections and 3D volume data).

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.
2. Check your results to answer all questions – except the final question.
3. After this run has finished, use ncview, ncdump etc. to check the precise location of the building (look at 2D array *zusi* that is contained in 2D xy cross-sections and 3D volume data).
4. Use this information to manually create the "raster_topo" file.

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.
2. Check your results to answer all questions – except the final question.
3. After this run has finished, use ncview, ncdump etc. to check the precise location of the building (look at 2D array *zusi* that is contained in 2D xy cross-sections and 3D volume data).
4. Use this information to manually create the "raster_topo" file.
5. Run the "raster" topography case.

Proceeding

Please proceed as follows:

1. Please run with the "generic" topography case first.
2. Check your results to answer all questions – except the final question.
3. After this run has finished, use ncview, ncdump etc. to check the precise location of the building (look at 2D array *zusi* that is contained in 2D xy cross-sections and 3D volume data).
4. Use this information to manually create the "raster_topo" file.
5. Run the "raster" topography case.
6. Compare both simulation results to answer the final question.

How to Start?

How to Start?

- Create two **INPUT** directories for both new runs:

```
cd ~/palm/current_version
mkdir -p JOBS/generic/INPUT
mkdir -p JOBS/raster/INPUT
```

How to Start?

- Create two **INPUT** directories for both new runs:

```
cd ~/palm/current_version
mkdir -p JOBS/generic/INPUT
mkdir -p JOBS/raster/INPUT
```

- Create the parameter files and `raster_topo` file and set the required parameters in

```
JOBS/generic/INPUT/generic_p3d
JOBS/raster/INPUT/raster_p3d
```

How to Start?

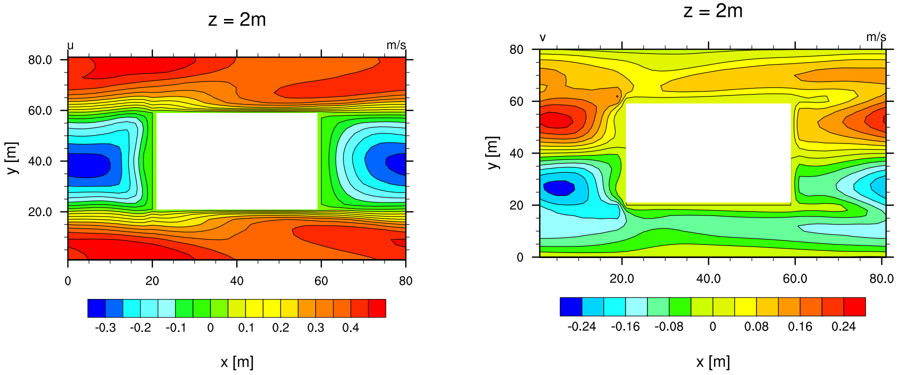
- Create two **INPUT** directories for both new runs:
  ```
  cd ~/palm/current_version
  mkdir -p JOBS/generic/INPUT
  mkdir -p JOBS/raster/INPUT
  ```
- Create the parameter files and `raster_topo` file and set the required parameters in
  ```
  JOBS/generic/INPUT/generic_p3d
  JOBS/raster/INPUT/raster_p3d
  ```
- Start the runs one by one with mrun-commands
  ```
  mrun -d generic -K parallel ...
  mrun -d raster -K parallel ...
  ```

How to Start?

- Create two **INPUT** directories for both new runs:
  ```
  cd ~/palm/current_version
  mkdir -p JOBS/generic/INPUT
  mkdir -p JOBS/raster/INPUT
  ```
- Create the parameter files and `raster_topo` file and set the required parameters in
  ```
  JOBS/generic/INPUT/generic_p3d
  JOBS/raster/INPUT/raster_p3d
  ```
- Start the runs one by one with mrun-commands
  ```
  mrun -d generic -K parallel ...
  mrun -d raster -K parallel ...
  ```
- and analyze the output files in
  ```
  JOBS/generic/OUTPUT
  JOBS/raster/OUTPUT
  ```

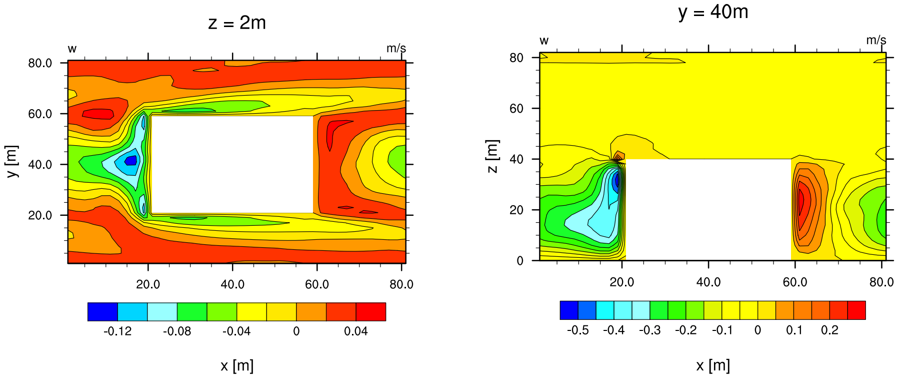
Question 1: Flow patterns (I)

Horizontal cross sections of 1-h averaged velocity components u and v



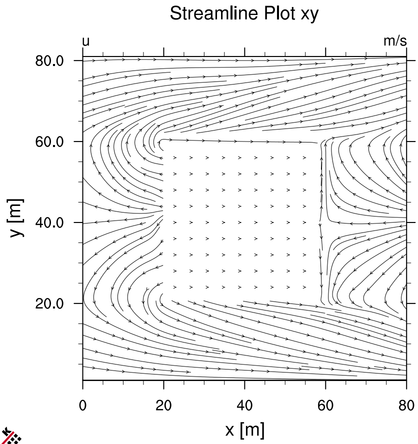
Question 1: Flow patterns (II)

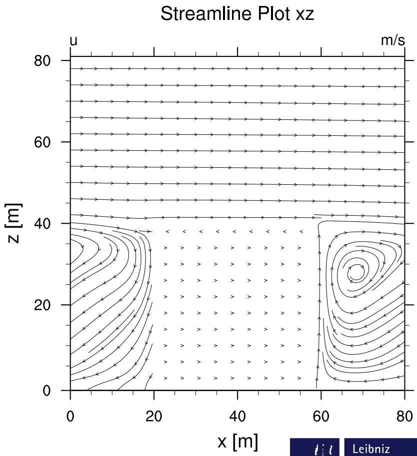
Horizontal and streamwise vertical cross sections of 1-h averaged velocity component w



Question 1: Flow patterns (III)

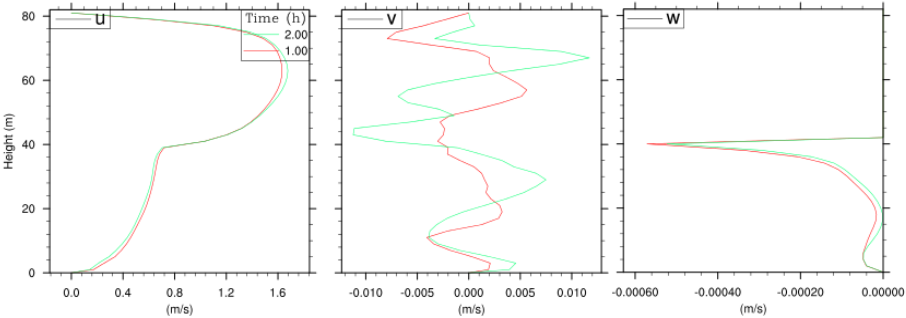
Streamlines (1-h average) for the same cross sections as seen in Frame 10 for the w-velocity





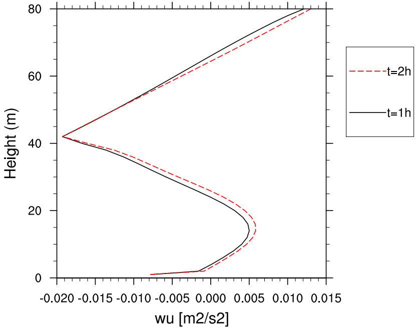
Question 2: Velocity and momentum flux profiles

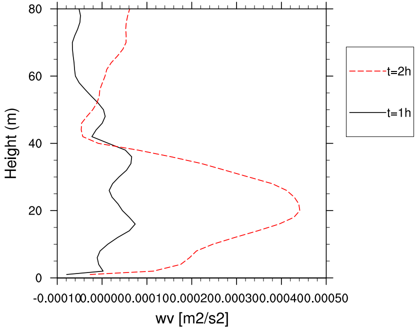
Vertical profiles of 1-h and horizontally averaged u-, v- and w-velocity



Question 2: Velocity and momentum flux profiles

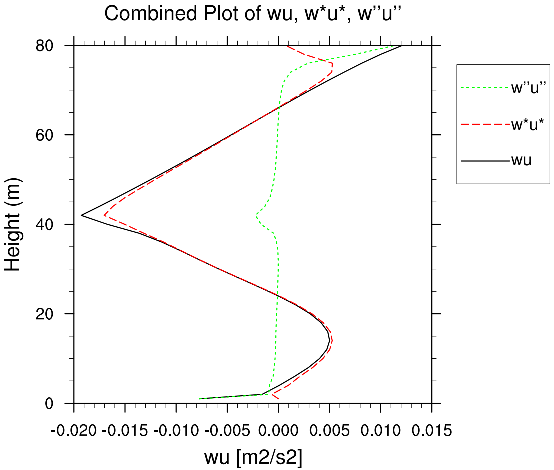
Vertical profiles of 1-h and horizontally averaged total turbulent momentum fluxes *wu* and *wv*





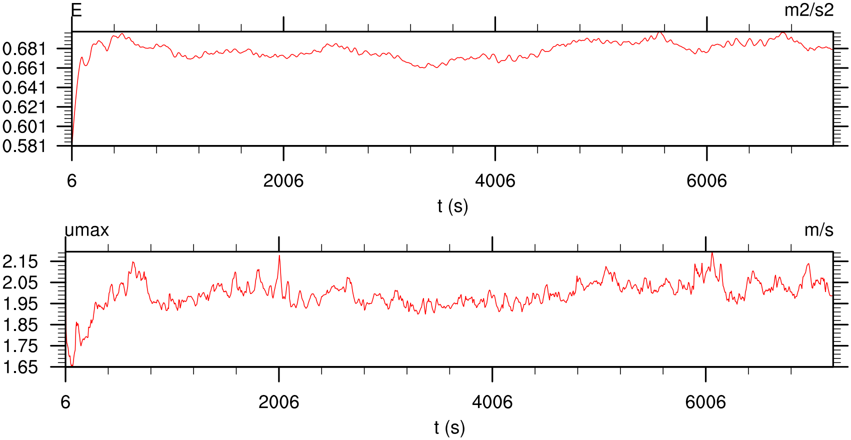
Question 3: LES? - Fluxes

Vertical profiles of 1-h and horizontally averaged momentum fluxes: total (wu), resolved-scale ($w^* u^*$) and subgrid-scale ($w'' u''$) fluxes



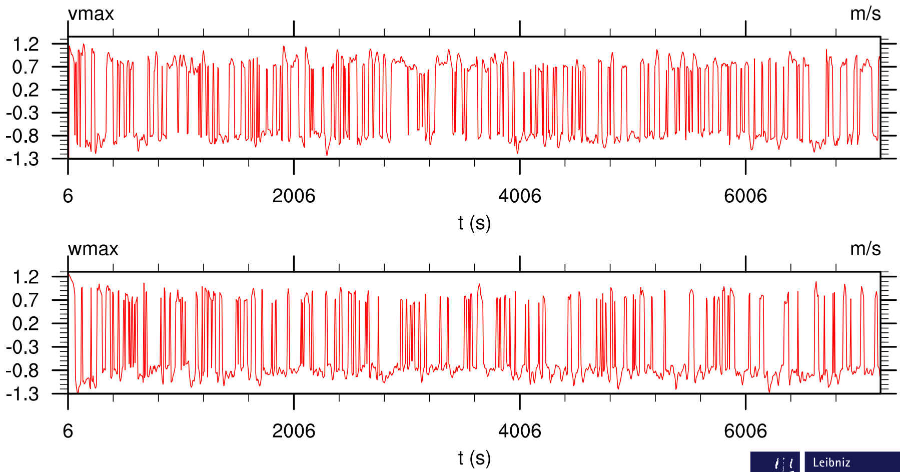
Question 3: LES? - Time Series (I)

Total kinetic energy E of the flow and maximum u-velocity in the model domain



Question 3: LES? - Time Series (II)

Maximum v- and w-velocity in the model domain



Answer to question 1 (I)

Can you identify any interesting flow patterns around the cube and what do they tell us?

The 1-h-averaged near-surface horizontal velocity components *u* and *v* show (see Frame 9):

- reversed streamwise flow in the gap between leeward and windward cube wall,
- diverging spanwise flow in the gap with nearly same magnitude as reversed spanwise flow.

The *w*-velocity fields complete the picture (see Frame 10), we see:

- descending mean flow near the windward cube wall,
- ascending mean flow near the leeward cube wall.

Answer to question 1 (II)

Can you identify any interesting flow patterns around the cube and what do they tell us?

Streamlines in Frame 11 show an overall view of the mean horizontal (left; near surface) and the mean streamwise-vertical (right; center of cube wall) flow:

- left: in the gap between leeward and windward cube wall, streamlines are directed in opposite direction to the prescribed flow direction, and they diverge in the spanwise direction,
- left: starting at the corners of the leeward cube wall, these diverging streamlines converge with the streamlines of the flow forced around the side walls of the cube,
- right: above the cube roof, the mean flow is horizontal and directed as prescribed,
- right: in the streamwise gap, we find a rotor-like vortex, explaining the mean downward motion in the largest part of the gap, the upward motion at the leeward cube wall, and the reversed streamwise flow, covering almost fully the gap dimensions.

Note: Flow patterns can change significantly when the size of the gaps between buildings changes (see e.g. Oke, T. R. *Street Design and Urban Canopy Layer Climate*. Energy and Buildings, 11 (1988)).

Answer to question 2 (I)

How do the horizontally and temporally averaged velocity and momentum flux profiles look like?

Frame 12 shows 1-h and horizontally averaged vertical profiles of velocity components u, v and w:

- u: Channel flow causes zero velocity at bottom and top domain wall. Upper domain half: Velocities increase with distance from upper channel wall, peaks at around 60m, and decreases quickly closer towards cube top. Lower domain half: u further decreases towards bottom channel wall, due to roughness of the wall, and u is much smaller here than in upper domain half, due to presence of cube.
- v: In the horizontal average, v-component is much smaller than u, and it fluctuates around zero. Time average should be increased to further eliminate these fluctuations. Flow is forced by u-component, and cube does not induce significant v in horizontal mean.
- w: Zero above, small negative values below cube top. In fully developed LES with sufficient domain size and averaging, horizontally averaged w profile should be zero.

Answer to question 2 (II)

How do the horizontally and temporally averaged velocity and momentum flux profiles look like?

Frame 13 shows 1-h and horizontally averaged vertical profiles of *u* and *v* components of total turbulent vertical momentum flux, for two ouput times:

- *wv* is one order of magnitude smaller than *wu* (flow is forced with the *u*-component), hence, the *wv* profile is not smooth, it strongly fluctuates with heigt and time.
- In contrast, the *wu* profile is smooth and barely changes from one 1-h average to the next, indicating sufficient averaging time.

Answer to question 2 (III)

How does the horizontally and temporally averaged momentum flux profile look like?

- This *wu* profile of channel flow around a cube strongly deviates from the typical *wu profile* in a neutral obstacle-free atmospheric boundary layer (ABL). In the latter, *wu* takes largest negative values at the surface and increases towards zero at the top the boundary layer. This means, the flow is decelerated everywhere within the ABL due to surface friction. In the cube-flow, the *wu* profile can be split into three regions:
  - z=40 to 80m: linear increase with height, i.e. the flow is decelerated in this part. Up to 65m, *wu* is negative, i.e. the roughness of the cube top causes the deceleration. Above, *wu* is positive, i.e. the flow is decelerated due to the no-slip boundary condition at the domain top.
  - z=15 to 40m: decreasing with height, i.e. the flow is accelerated here, which can be attributed to the above-cube flow.
  - z=0 to 15m: increasing with height, meaning flow deceleration, due to surface friction.

Note: Such momentum flux profiles (wu) are typical for urban and vegetation canopy flows.

Answer to question 3

Is it really a fully developed large-eddy simulation?

- Frame 14: Except near the surface and at the domain top, subgrid-scale momentum flux $w''u''$ is one order of magnitude smaller than the resolved-scale counterpart w^*u^*, hence we can conclude, that the grid spacing is sufficiently small in order to resolve the energy-containing eddies within this neutral flow around a solid cube.
- Frame 15: Timeseries of the kinetic energy E and the maximum u value in the flow indicate that two hours of simulation time are sufficient for the spin up of the model. Both quantities level out towards the end of the simulation.
- Frame 16: The temporal evolution of maximum v and w values indicates that the flow shows turbulent features, since both components frequently change signs.