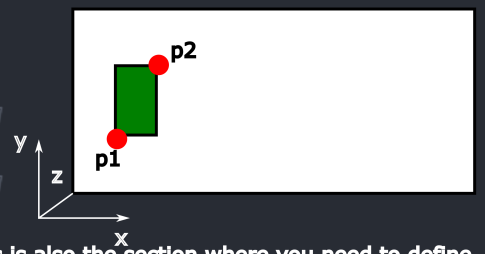
```yaml
# Grid parameters
grid:
    # Number of cells [Nx, Ny, Nz]
    dimension: [170, 150, 1]
    # Size of each cell [dx, dy, dz]
    cell size: [1.0, 1.0, 1.0]

# Physics parameters
physics:
    porosity: 0.2
    molecular diffusion: 0.000000060 #m^2/min
    longitudinal dispersivity: 0.01
    transverse dispersivity: 0.001
    velocity:
        # Type of the file imported (e.g., modflow)
        type: modflow
        # Path to the file (e.g, .ftl file generated by the
        # LMT Modflow package)
        file: tmp/model-{}.ftl

# Simulation parameters
simulation:
    particles:
        # Number of particles
        N: 100000
        # Define the volume where
        # p1x <= x <= p2x
        # p1y <= y <= p2y
        # p1z <= z <= p2z
        start:
            # Point 1 [p1x, p1y, p1z]
            p1: [25, 65, 0]
            # Point 2 [p2x, p2y, p2z]
            p2: [37, 85, 1]

    # Time step #days
    dt: 0.05
    # Number of steps
    steps: 100000
```

here you need to define the grid parameters of your model

here you need to define the physics parameters and indicate which is the velocity fied that has to be used in the transport simulation. Following the structure of this Jupyter Notebook, the flow field is generated by MODFLOW and the file containg the data is in the tmp folder. The "{}" is going to be replaced the Monte Carlo iteration number.

here you need o define the simulation parametes, Since we are working with a Lagrangian model, the contamiant plume is discretized into particles. The number of particles has to be choosen such that if increased, the result of the simulation doesn't change. This is a variable that needs to be tested with a single scenario, before implementing the full Monte Carlo analysis.
In this section you need also to define the intial position of the contaminat plume:



this is also the section where you need to define the time discritization and the number of steps of the simulation.

```yaml
# Output parameters
output:
    # csv file containing post-proce
    csv:
        # Path to the file, The '{}
        # Monte Carlo iteration numb
        file: output/result-{}.csv
        # Data is computed every 'skip' steps
        skip: 1
        # List of items to be computed. You can add any number of items
        items:
            - label: cbtx x=117.0
              # Compute fraction of particles that have crossed a plane
              # located at x=117.0
              type: after-x
              x: 117.0
            #
            - label: cbtx x=168.0
              # Compute fraction of particles that have crossed a plane
              # located at x=168.0
              type: after-x
              x: 168.0

    # Export snapshots (optional)
    snapshot:
        # Path to the file. The '*'
        # step number. A file is ge
        # Note that exporting a large number of snapshots may
        # slow down the simulation.
        file: output/snap-{}-*.csv
        # List of time steps
        # steps: [0, 50000, 100000]
        # Instead of 'steps', it is possible to use 'skip' as for
        # the csv file, for example
        skip: 400
```

here you need to define where you want your simulation results to be saved, and to what they are related. As writted in the Jupiter Nottebook, the output is a .csv file that contains the data related to the cbt curves at selected control planes (items). Here the simulations results are saved in the result-{}.csv file, where the {} is replaced with the Monte Carlo iteration number.
NB, you can add as many items you need, just coping and pasting from line 52 to line 57 and changing the position of the evaluated control plane.

here you can ask to generate the snap-{}-*.cvs file which contains the particles postions (in which the plume has been discretized into) at the selected time step. To do that, you can define the time steps that you are interested in, or (if they are many) you can define the interval (skip) between them. In the name of the file: