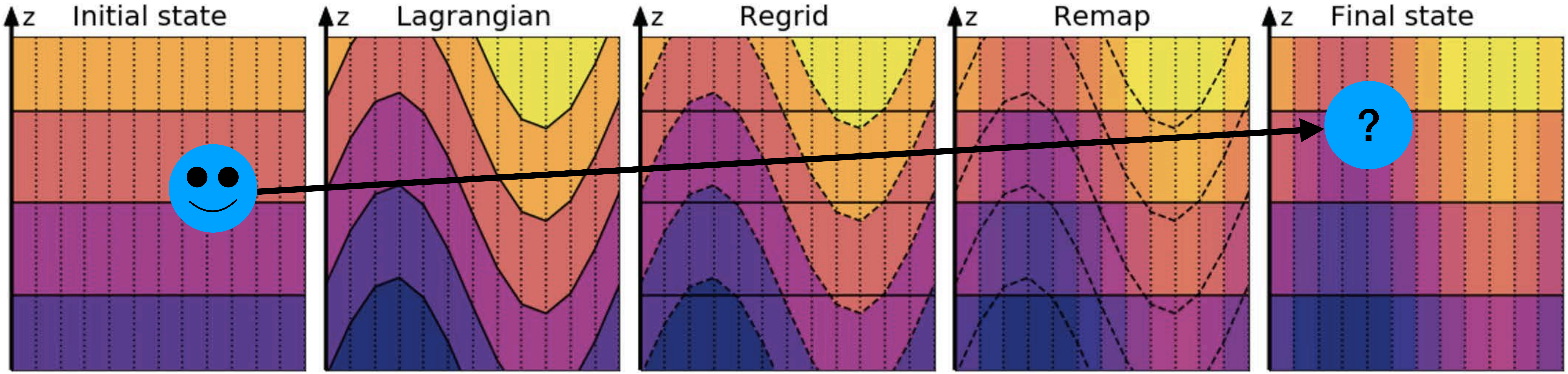
Online Lagrangian Particle Advection in MOM6



Spencer Jones (he/him)[1], Kaila Uyeda[1,2], Luyu Sun[3],
Alistair Adcroft[4], Matthew Harrison[4] and Stephen Griffies[4]

1. Texas A&M University, 2. UC, Irvine, 3. University of Maryland, 4. NOAA GFDL

Online particle advection has some advantages over offline advection

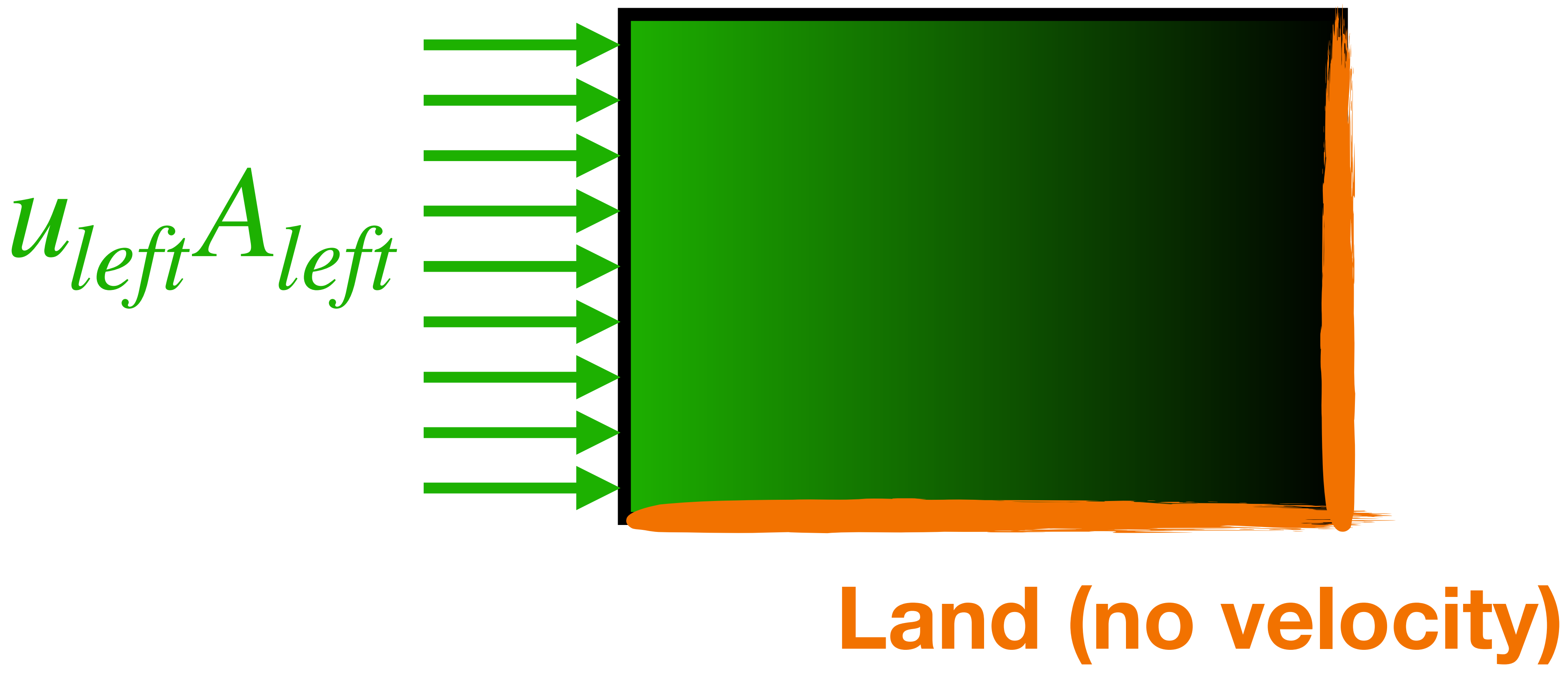
- Offline particle advection typically involves
    - saving time-averaged velocities
    - using them to advect virtual particles

- You can save more output to get more accurate trajectories, but this requires a lot of storage

- Online particle advection avoids this problem, because the particles are advected as part of the model run

In a model where the grid is stationary (not true in MOM6)

u, v, w are interpolated within each grid cell

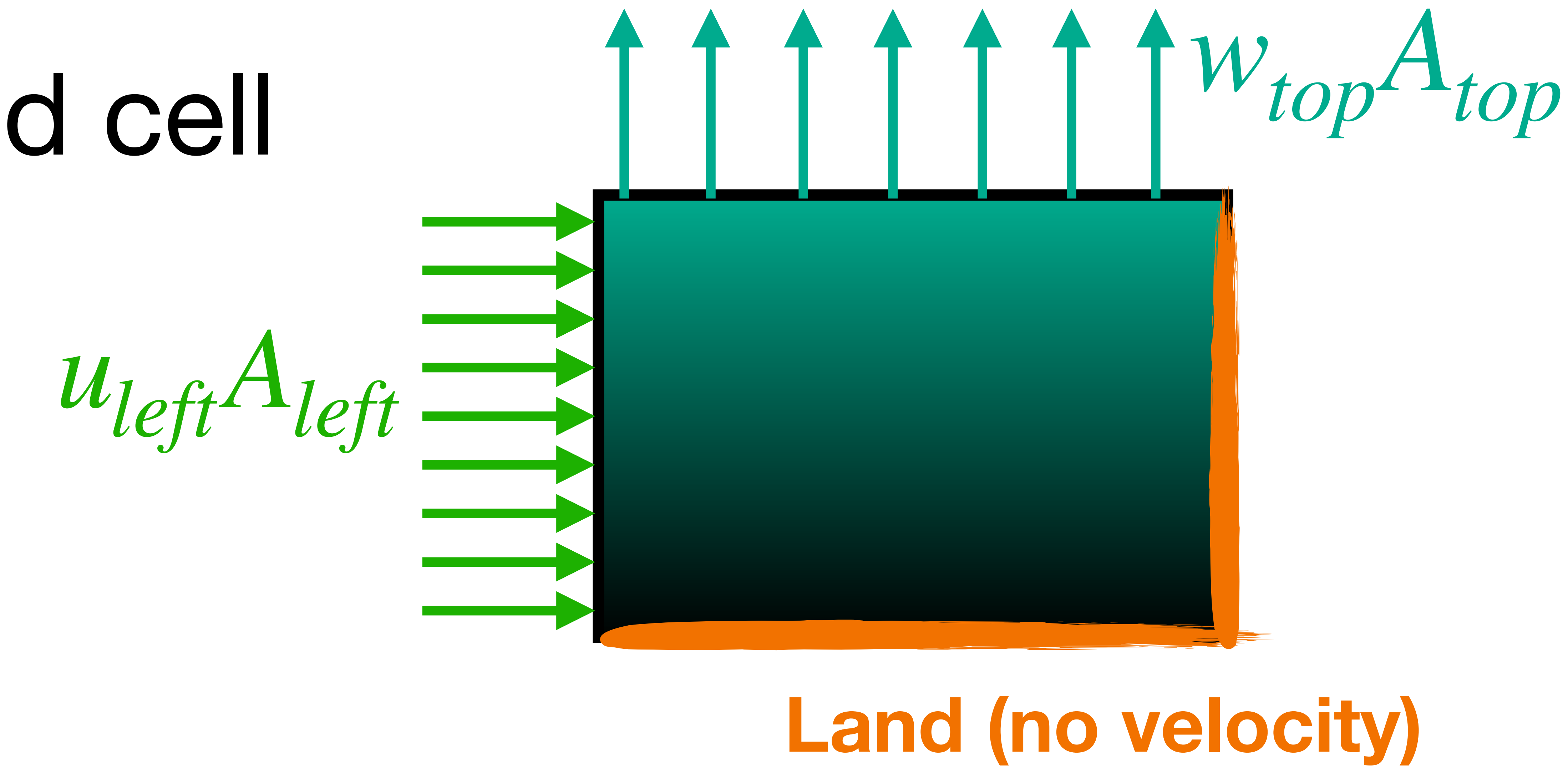


Most commonly,

- u is interpolated linearly in the x-direction, and is constant in y, z

In a model where the grid is stationary (not true in MOM6)

u, v, w are interpolated within each grid cell

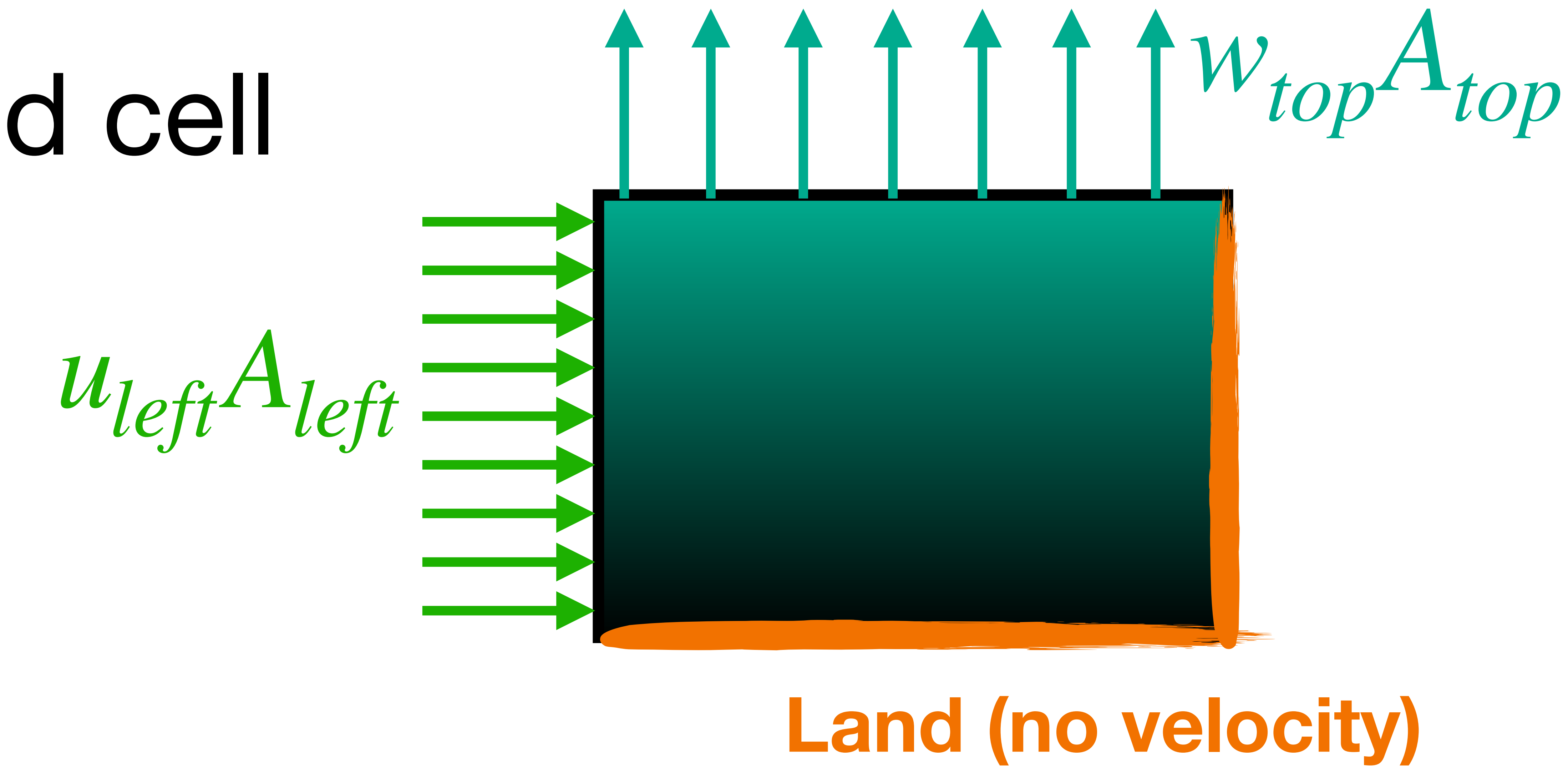


Most commonly,

- u is interpolated linearly in the x-direction, and is constant in y, z
- v is interpolated linearly in the y-direction, and is constant in x, z
- w is interpolated linearly in the z-direction, and is constant in x, y

In a model where the grid is stationary (not true in MOM6)

u, v, w are interpolated within each grid cell



Most commonly,

- u is interpolated linearly in the x-direction, and is constant in y, z
- v is interpolated linearly in the y-direction, and is constant in x, z
- w is interpolated linearly in the z-direction, and is constant in x, y

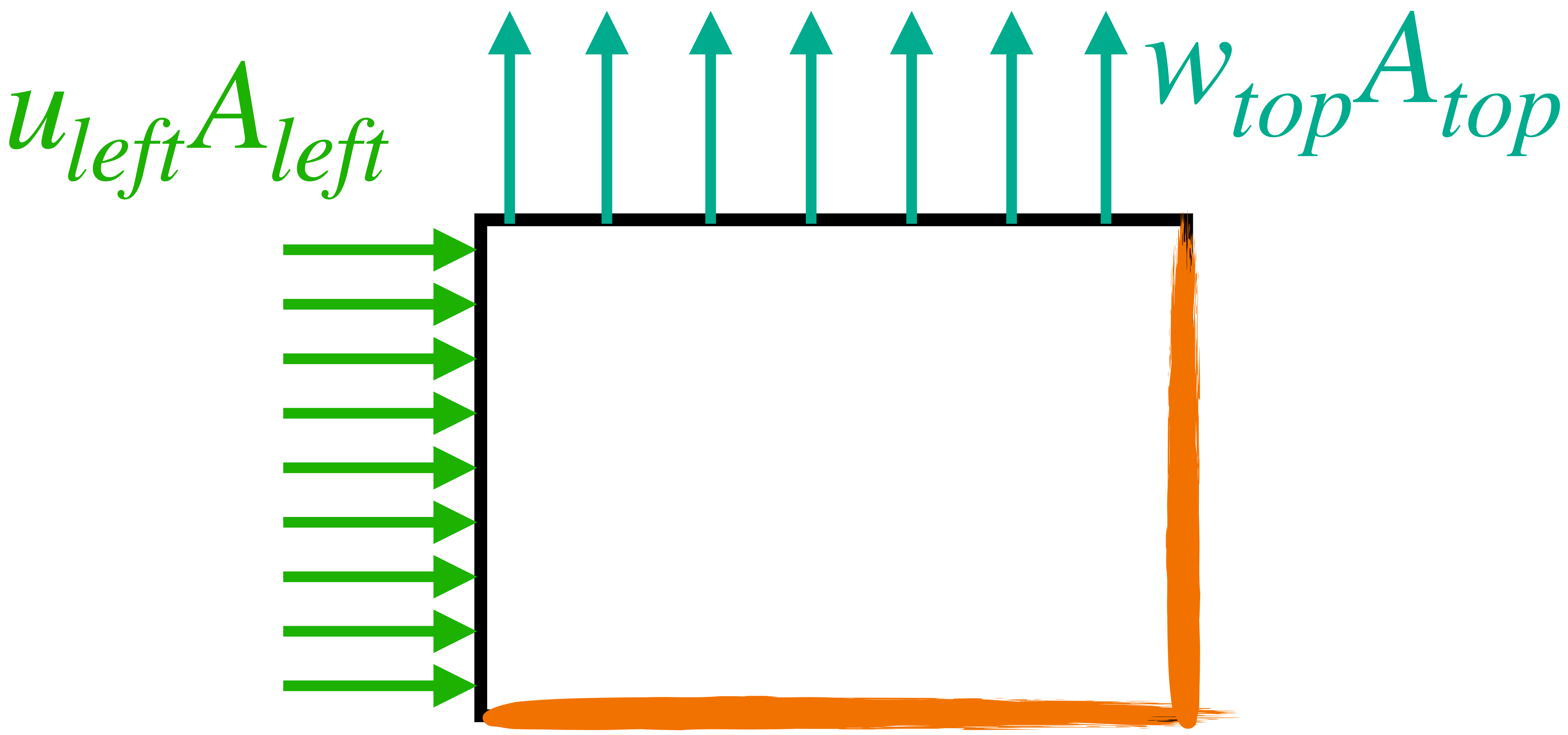
This is a “mass conserving” scheme.

Defining a “mass conserving” scheme.

If

- u is interpolated linearly in the x-direction, and is constant in y, z
- v is interpolated linearly in the y-direction, and is constant in x, z
- w is interpolated linearly in the z-direction, and is constant in x, y

In the ocean model, inputs and outputs match for each grid cell



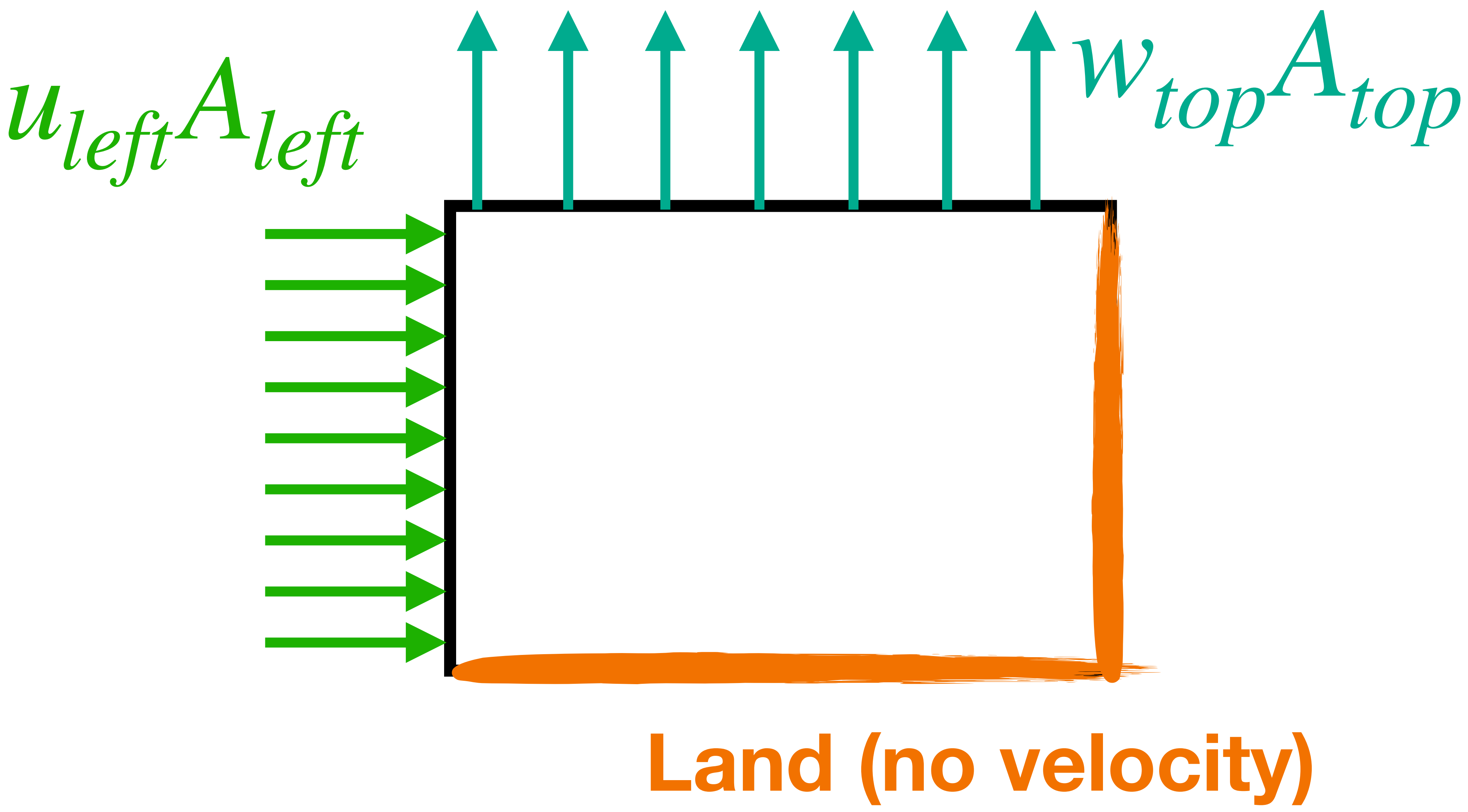
Land (no velocity)

Defining a “mass conserving” scheme.

If

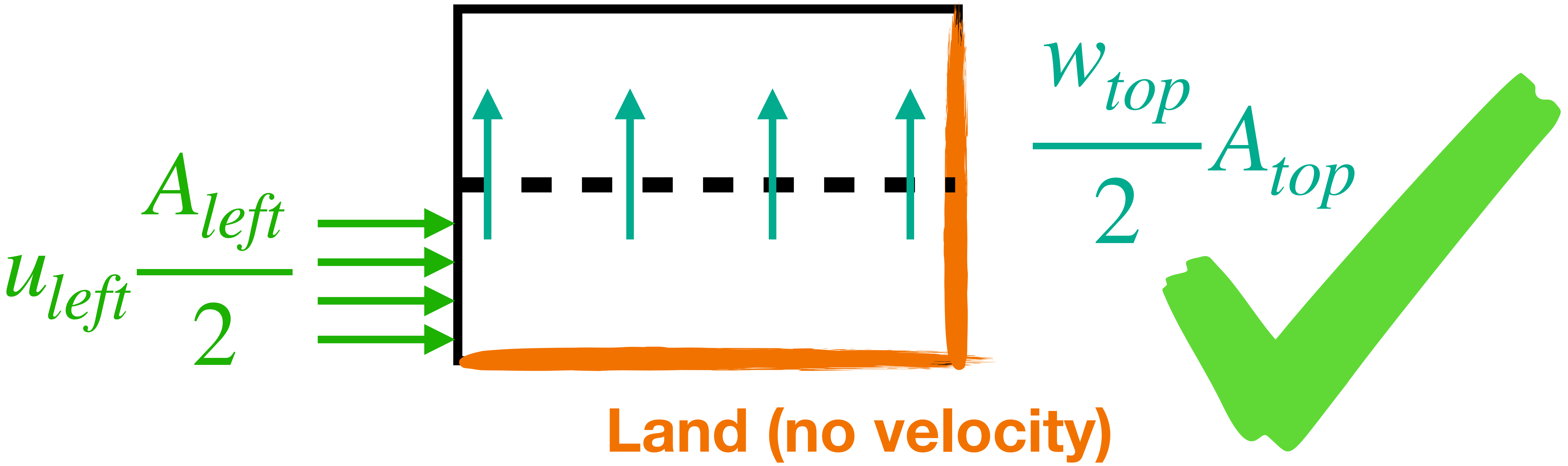
- u is interpolated linearly in the x-direction, and is constant in y, z
- v is interpolated linearly in the y-direction, and is constant in x, z
- w is interpolated linearly in the z-direction, and is constant in x, y

In the ocean model, inputs and outputs match for each grid cell



Subdividing the grid cell does not create convergence of mass

e.g.



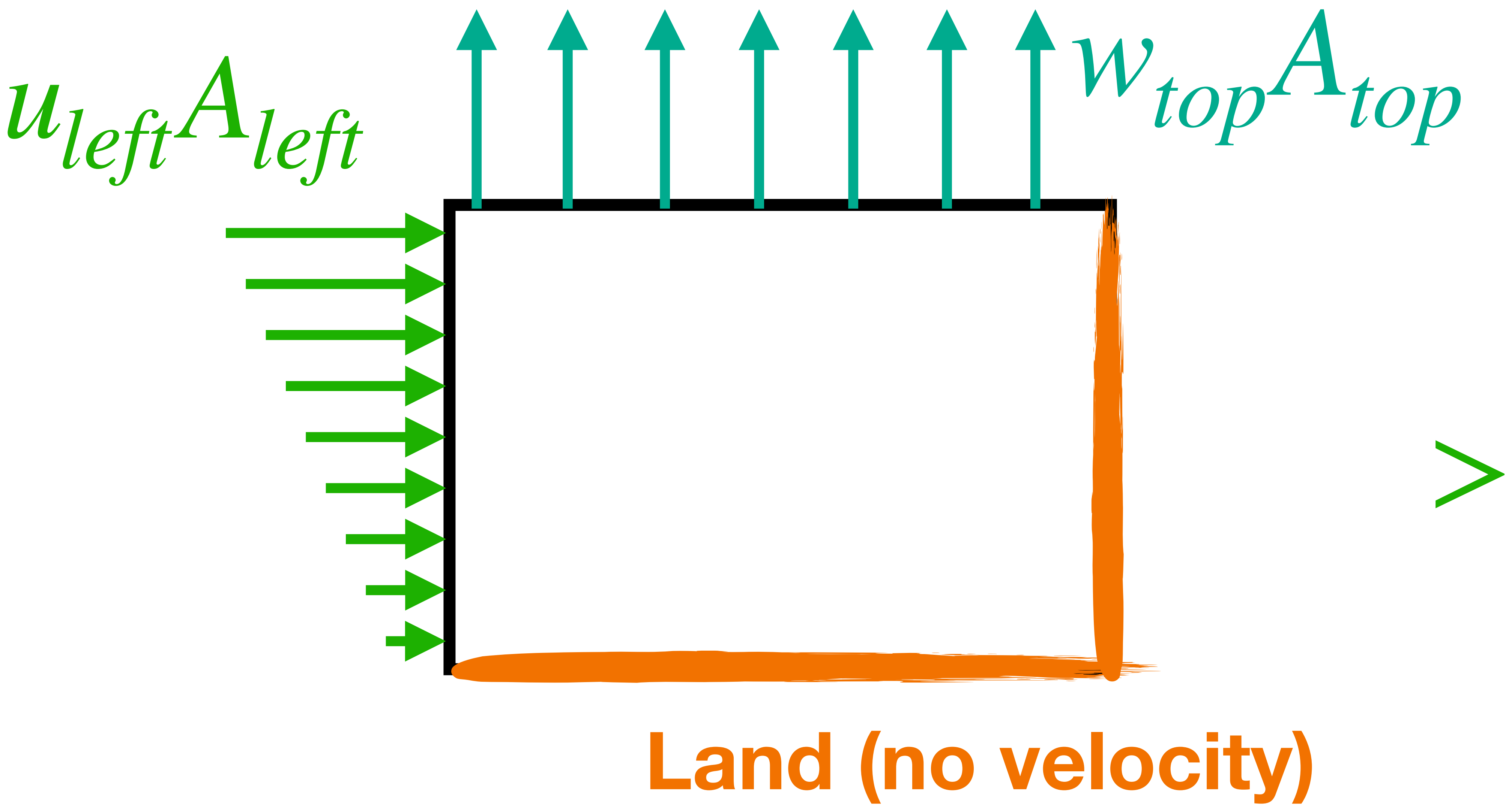
Example of a scheme that doesn't conserve mass

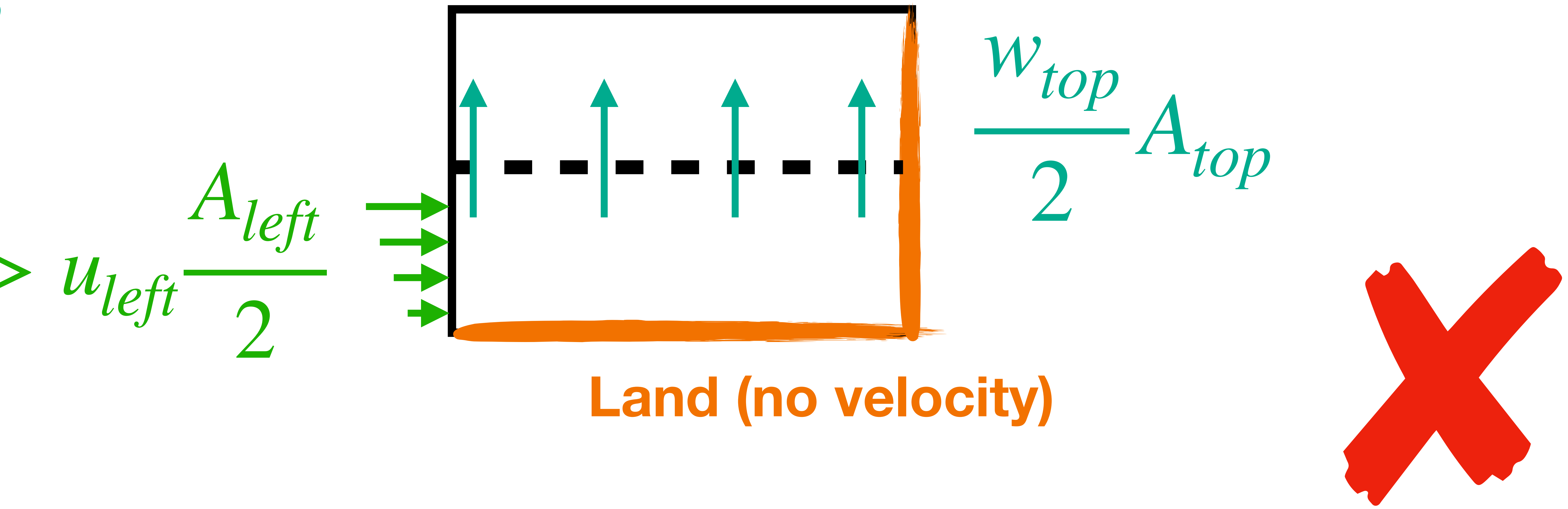
- u is interpolated linearly in the x-direction, is constant in y and varies linearly in z
- w is interpolated linearly in the z-direction, and is constant in x, y

(Perhaps with the motivation of having u and v decrease towards the ocean floor)

In the ocean model, inputs and outputs match for each grid cell

Convergences appear when we subdivide our grid cells



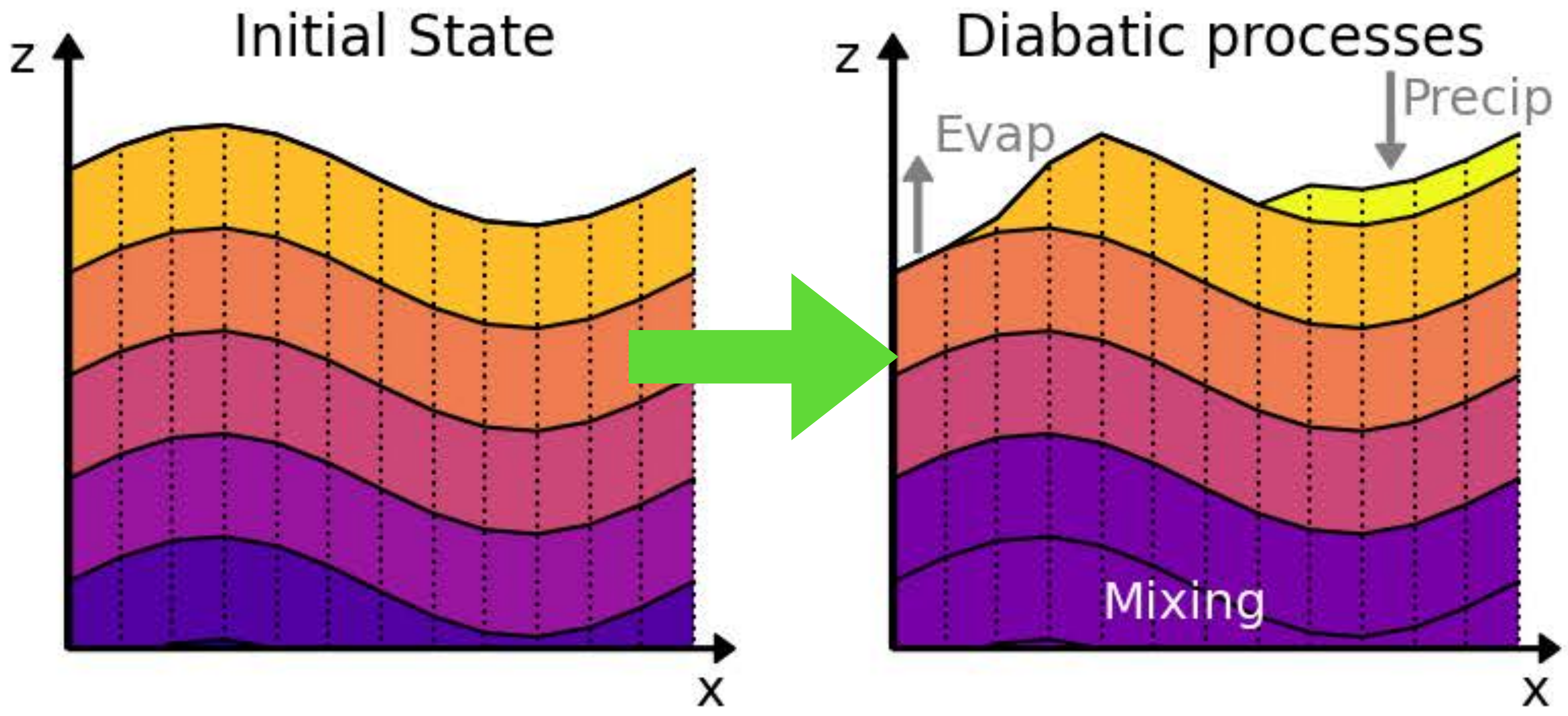


Online in MOM6, we cannot use a traditional scheme

w does not exist

During the dynamics step,
the layers move together with
the fluid

Order of operations usually
proceeds like this:



Online in MOM6, we cannot use a traditional scheme

w does not exist

During the dynamics step, the layers move together with the fluid

Order of operations usually proceeds like this:

Online in MOM6, we cannot use a traditional scheme

w does not exist

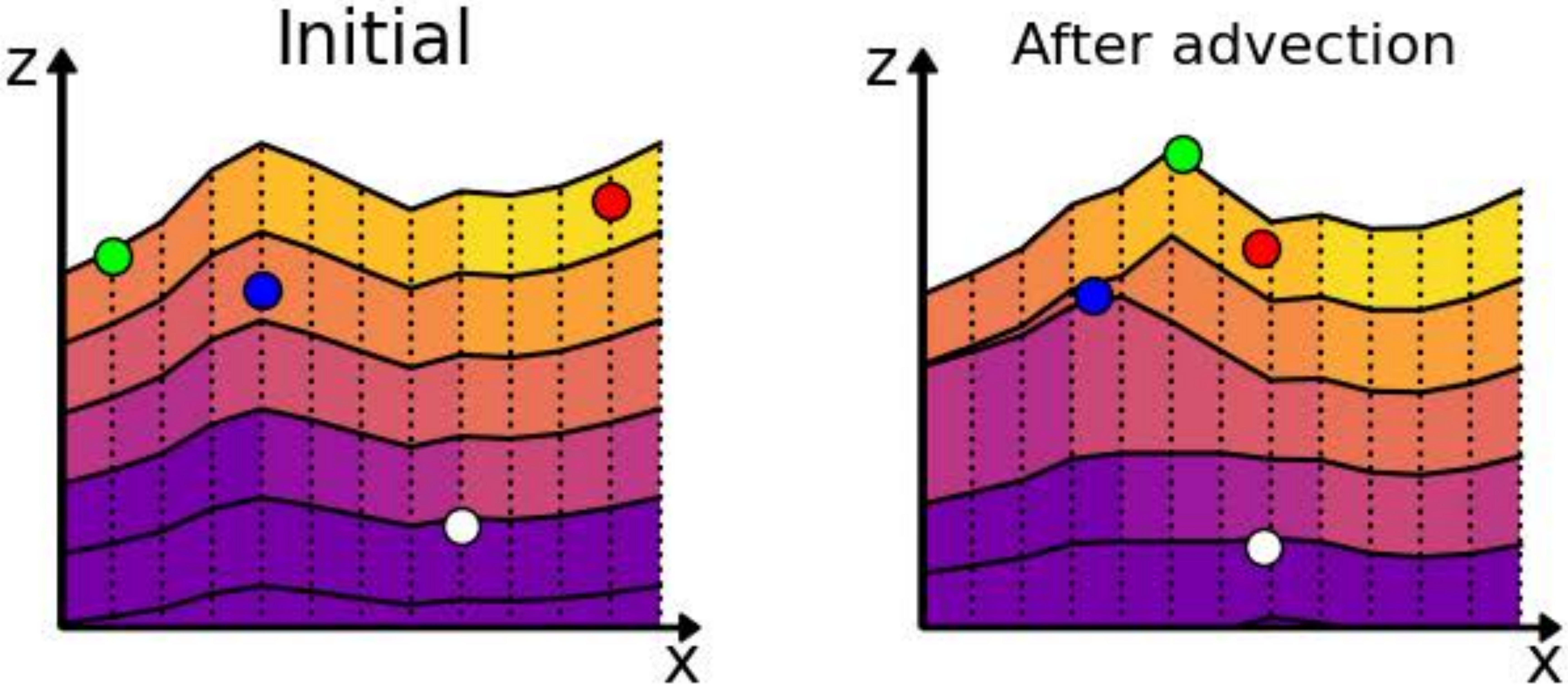
During the dynamics step, the layers move together with the fluid

Order of operations usually proceeds like this:

Online in MOM6, we cannot use a traditional scheme

That's a lot of steps! But let's first just focus on the dynamics step, in which the water actually moves

Particle advection during dynamics step

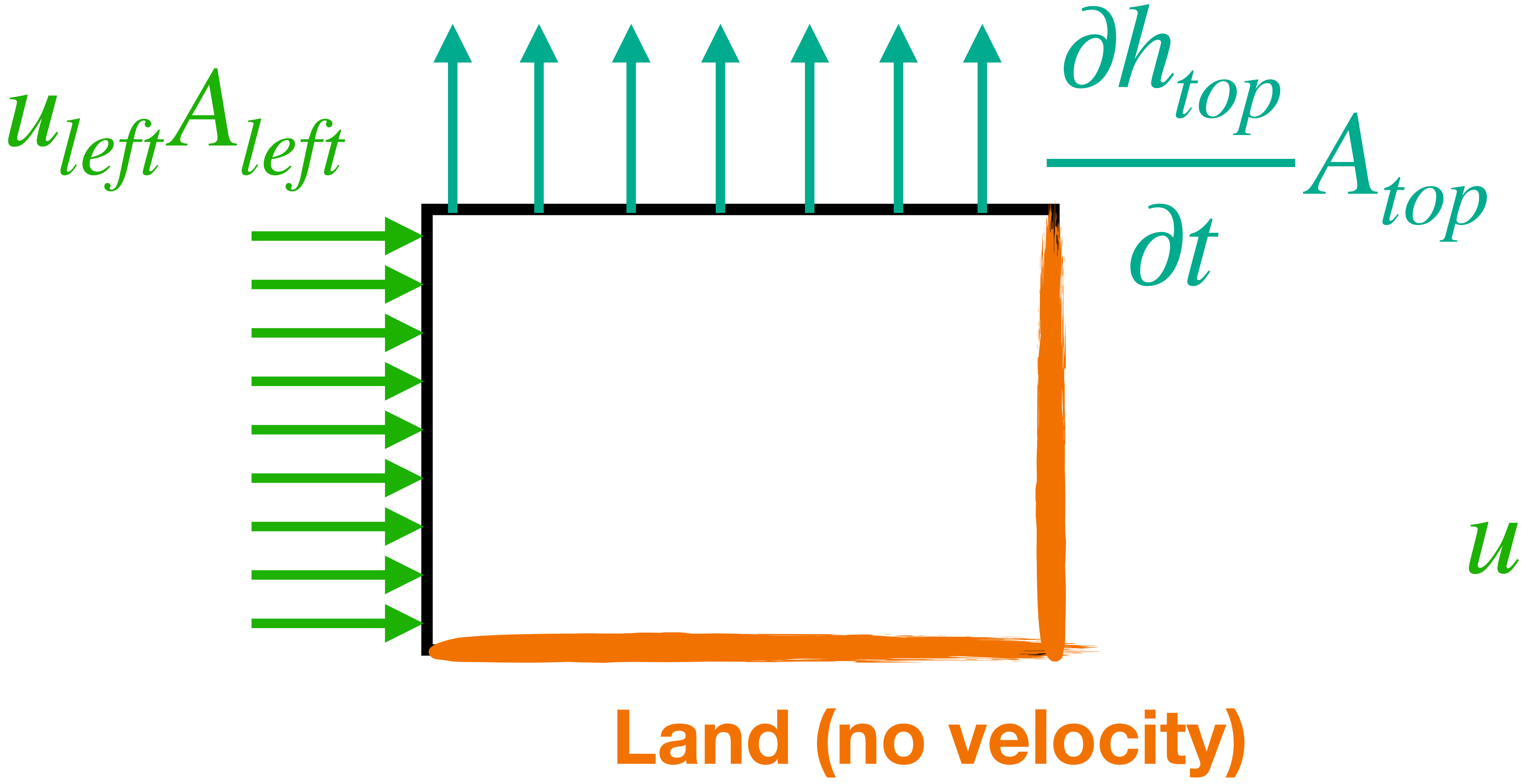


Particles are advected horizontally and maintain their fractional position within each layer

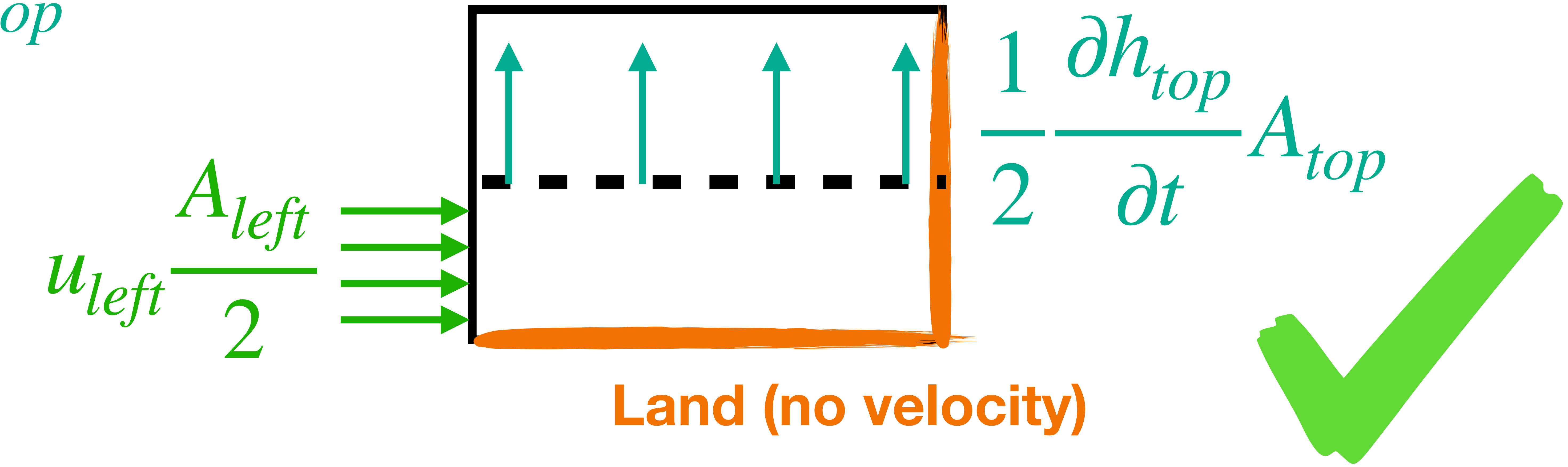
Mass conservation in this new setup

- u is interpolated linearly in the x-direction, and is constant in y, z
- v is interpolated linearly in the y-direction, and is constant in x, z
- Particle maintains its fractional position in the cell

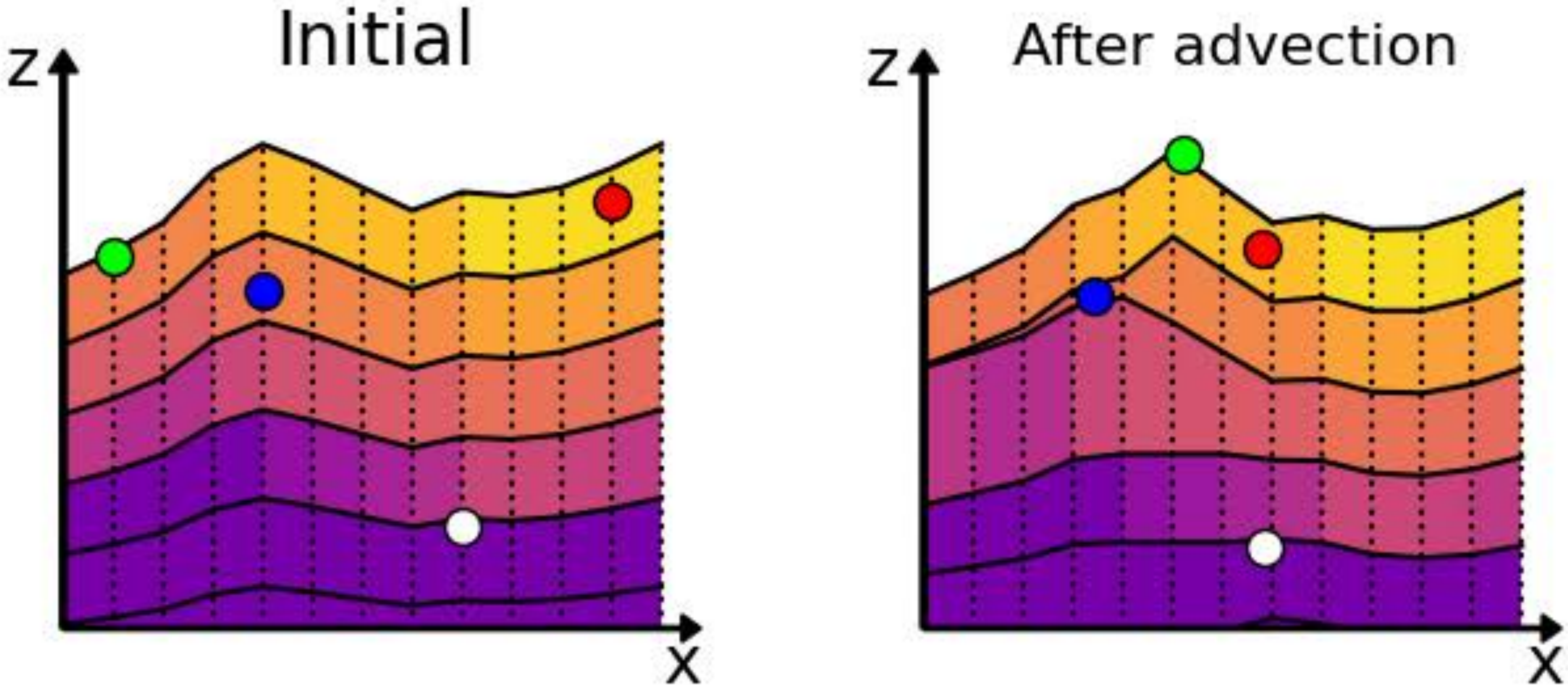
Volume is conserved because top interface moves



Subdividing the grid cell does not create convergence of mass, because “center” interface moves half the distance of top interface



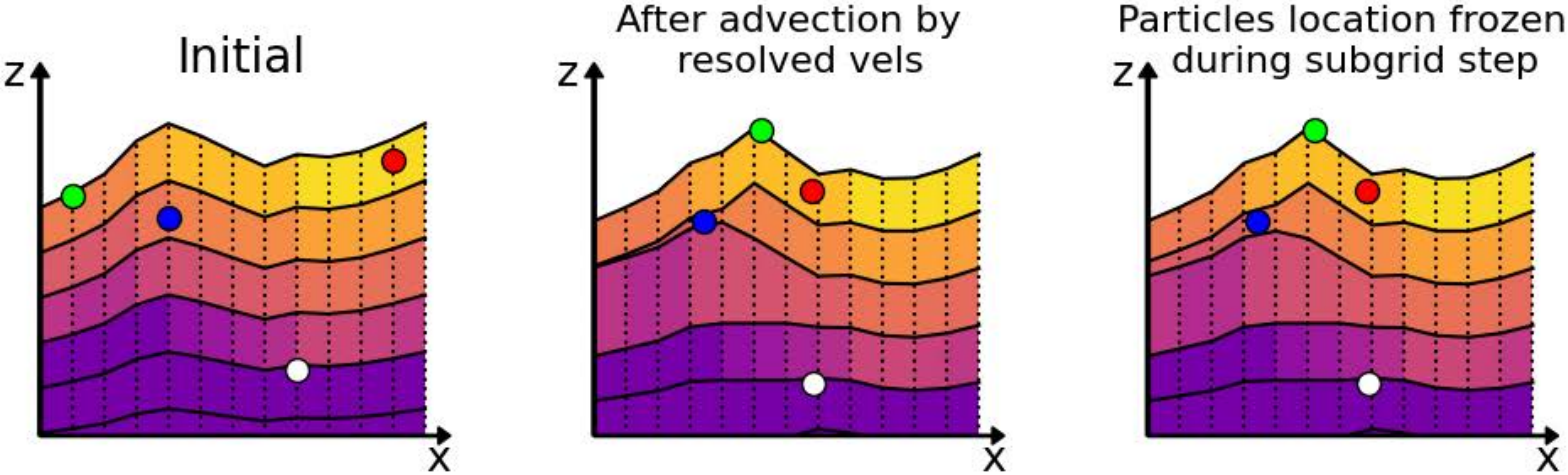
Our code has 2 modes



Advect particles with resolved velocities

Advect particles with residual velocities

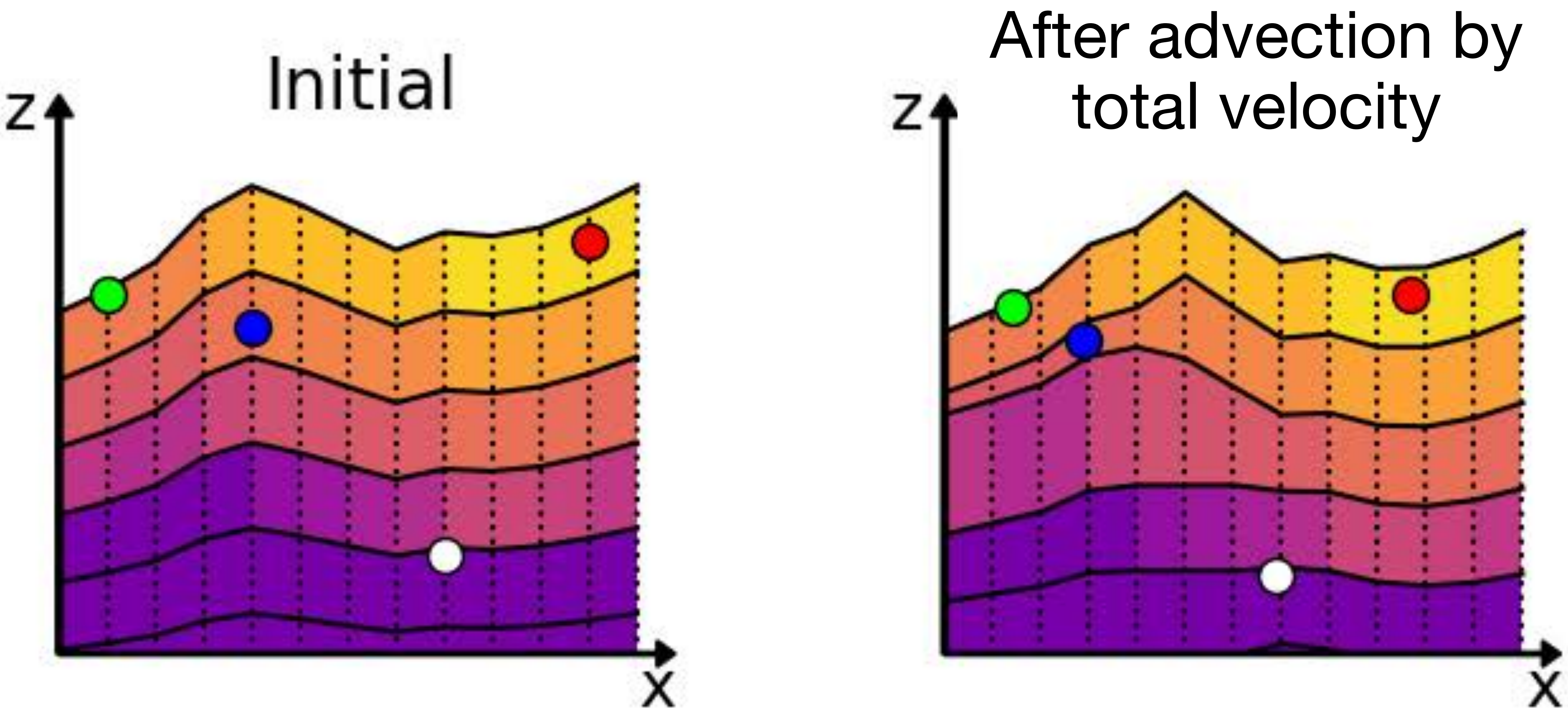
Advection with resolved velocities only



Particles are advected horizontally using the resolved velocities

Particles are frozen in depth and time during the part of the timestep where sub grid scale velocities are used

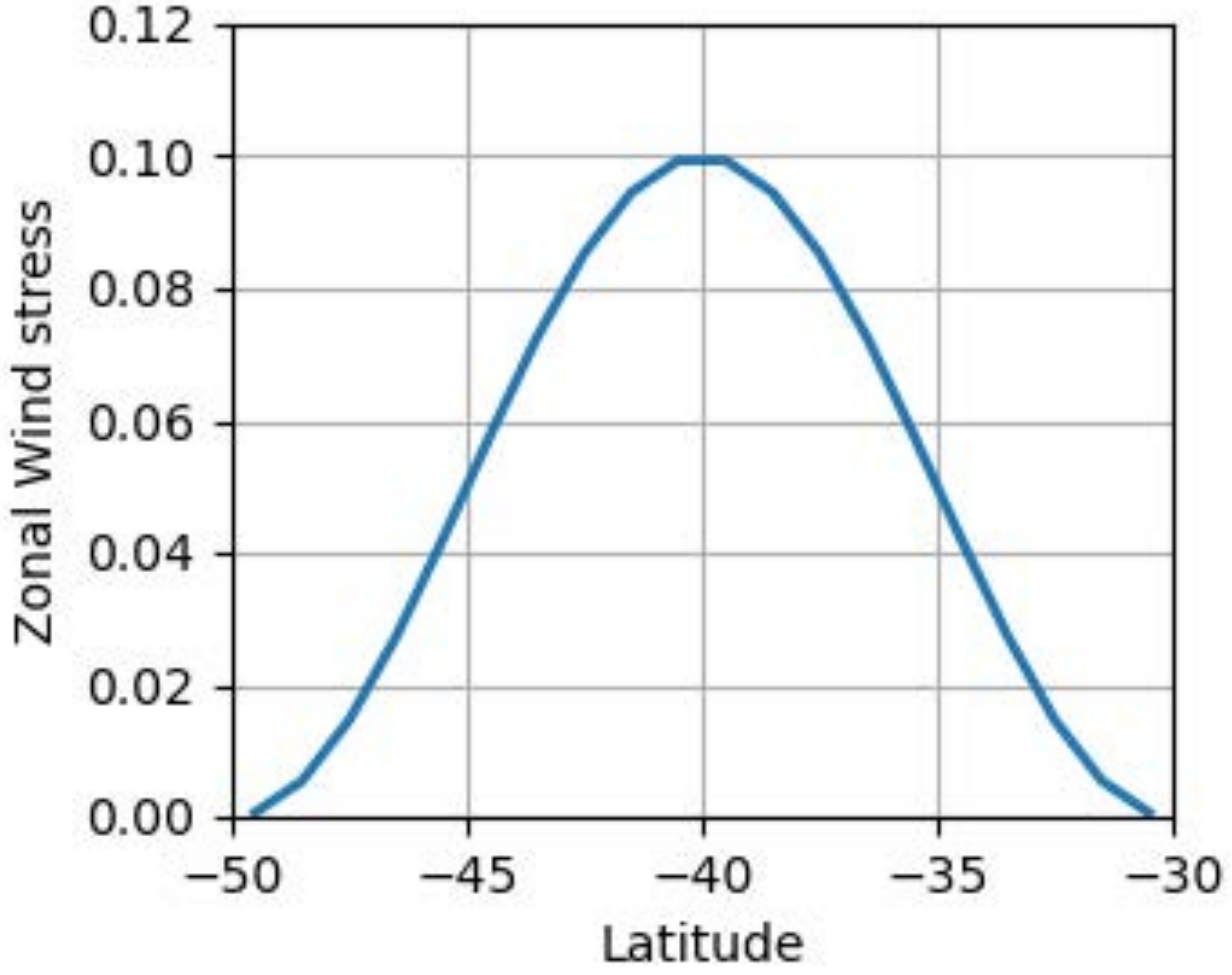
Advection with full velocity field

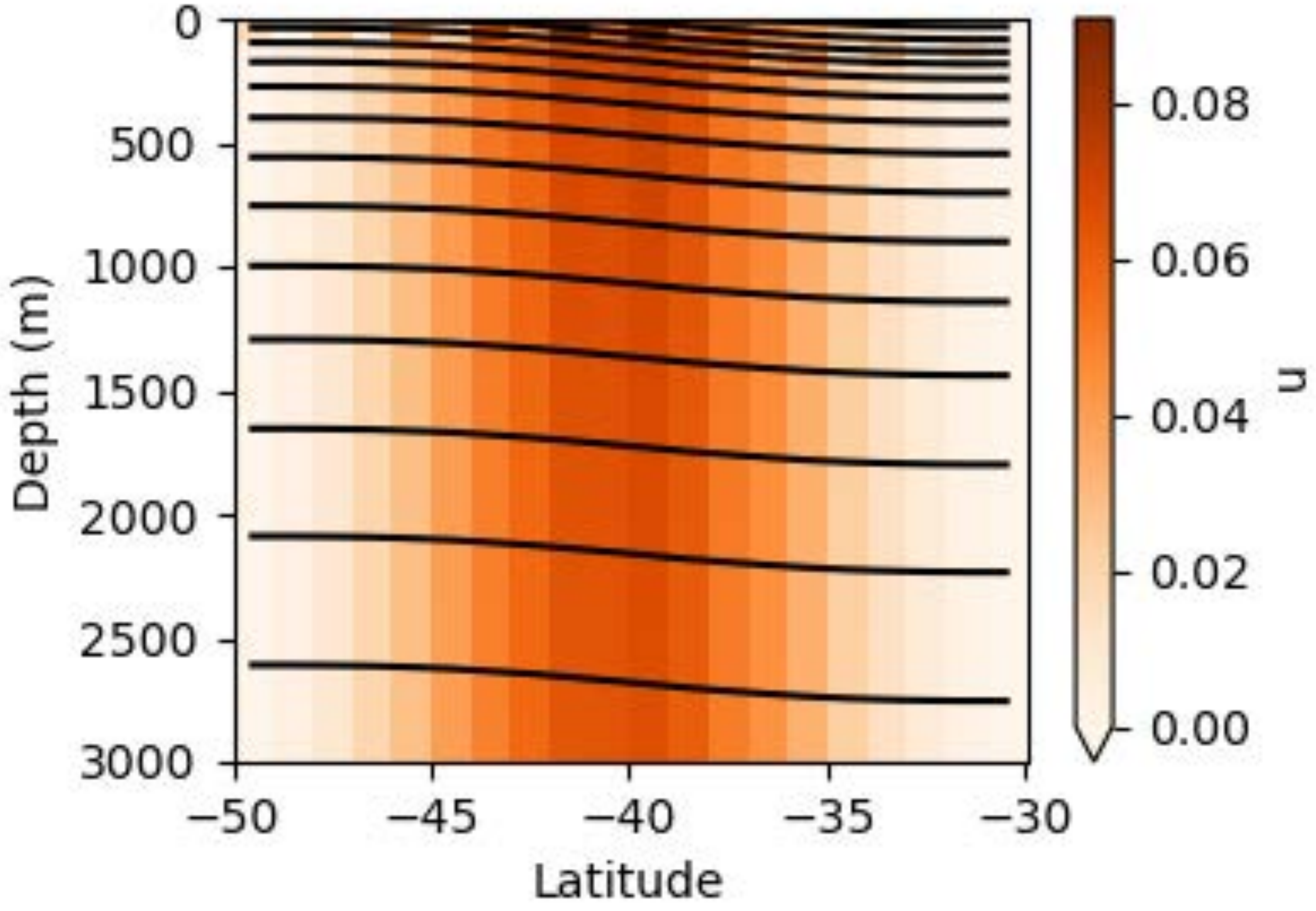


In this case, we actually use $\dfrac{\mathbf{u}h}{h}$ to advect the particles

Examples in adiabatic “stacked shallow water” channel

First examples use 1 degree resolution and $\kappa_{gm} = 8000 m^2/s$


Zonal wind

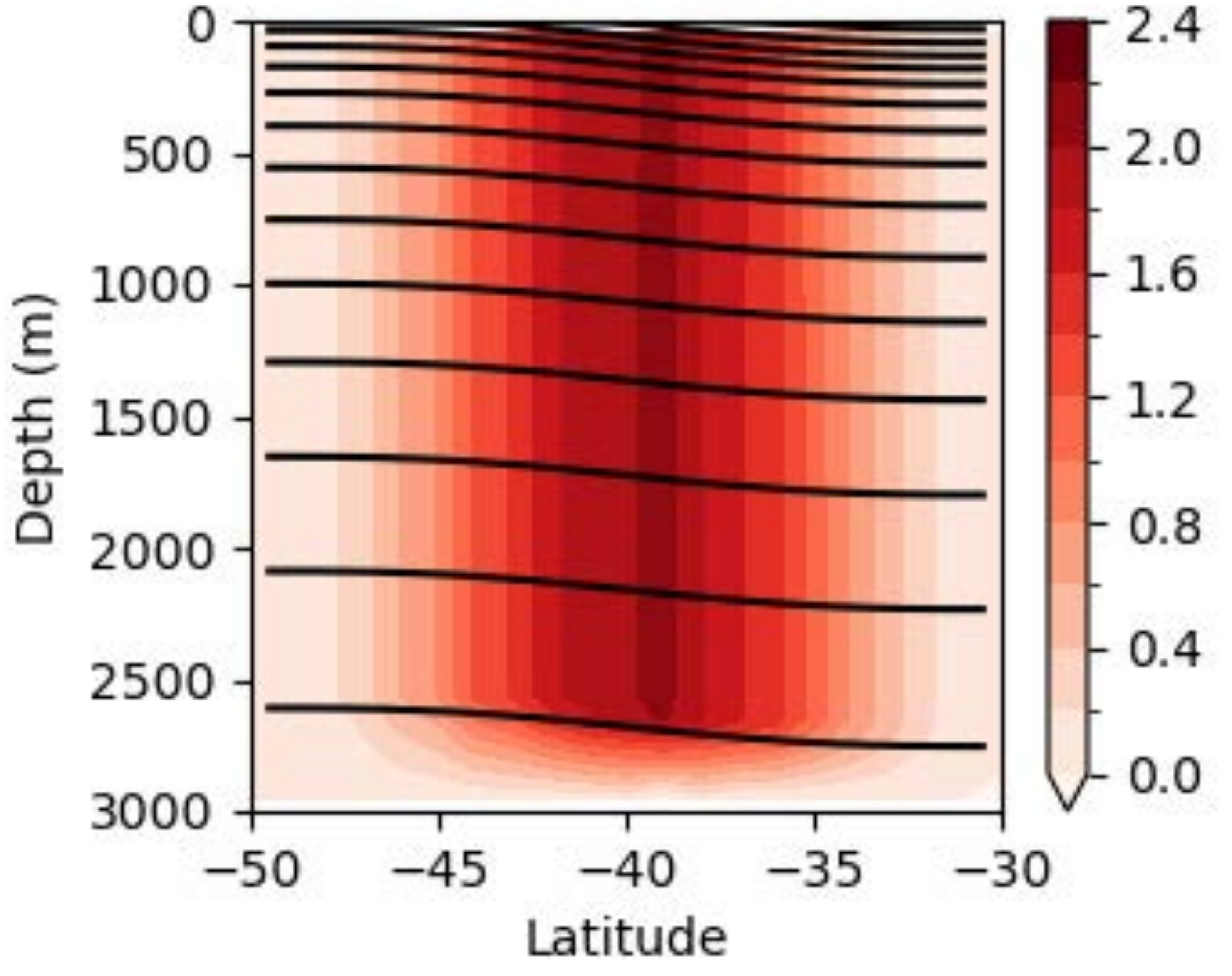

Zonal velocity

Examples in adiabatic “stacked shallow water” channel

First examples use 1 degree resolution and $\kappa_{gm} = 8000 m^2/s$

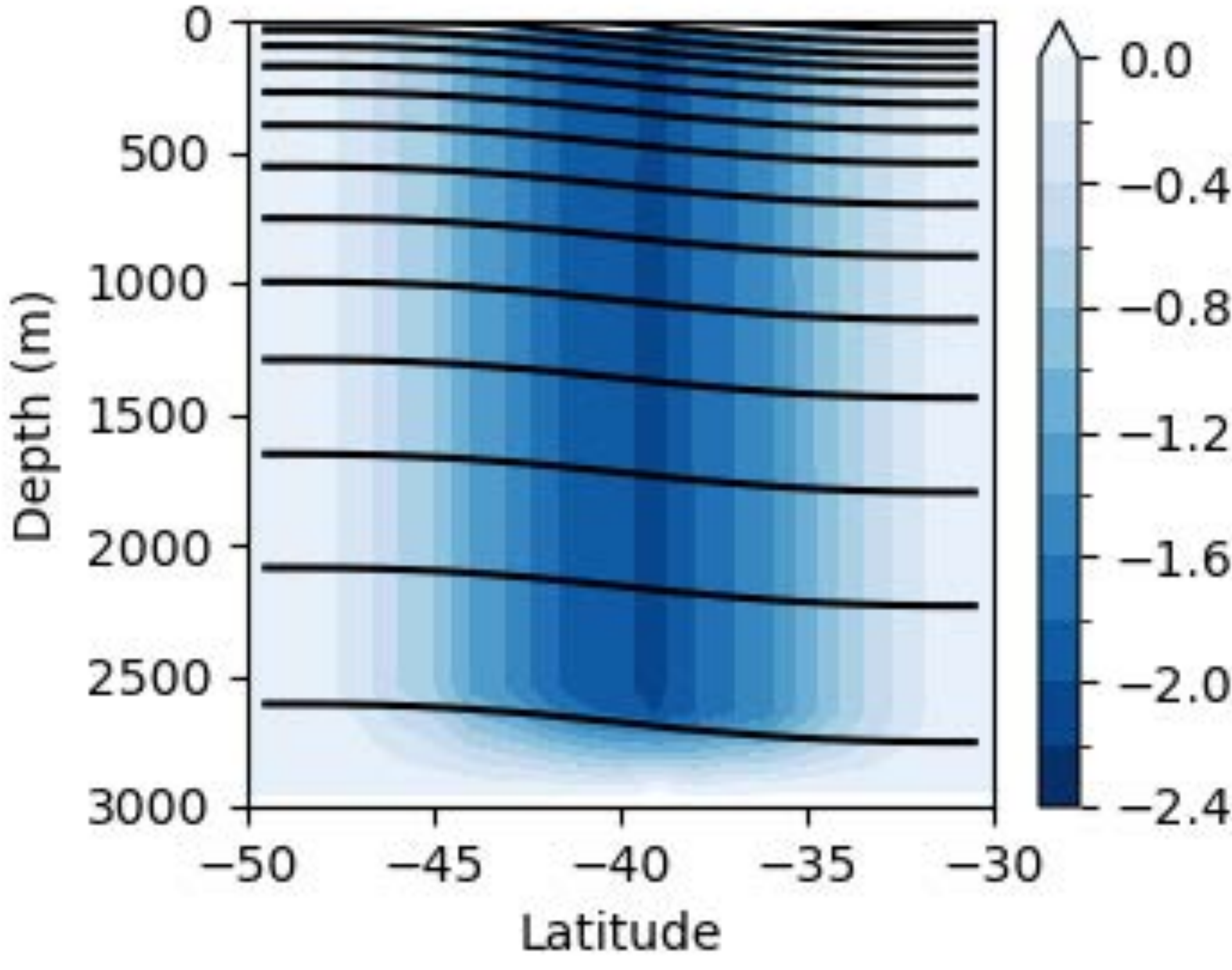
In adiabatic mode, there are no buoyancy fluxes, so the residual circulation must be zero everywhere

Overturning streamfunction for time-mean circulation, Ψ



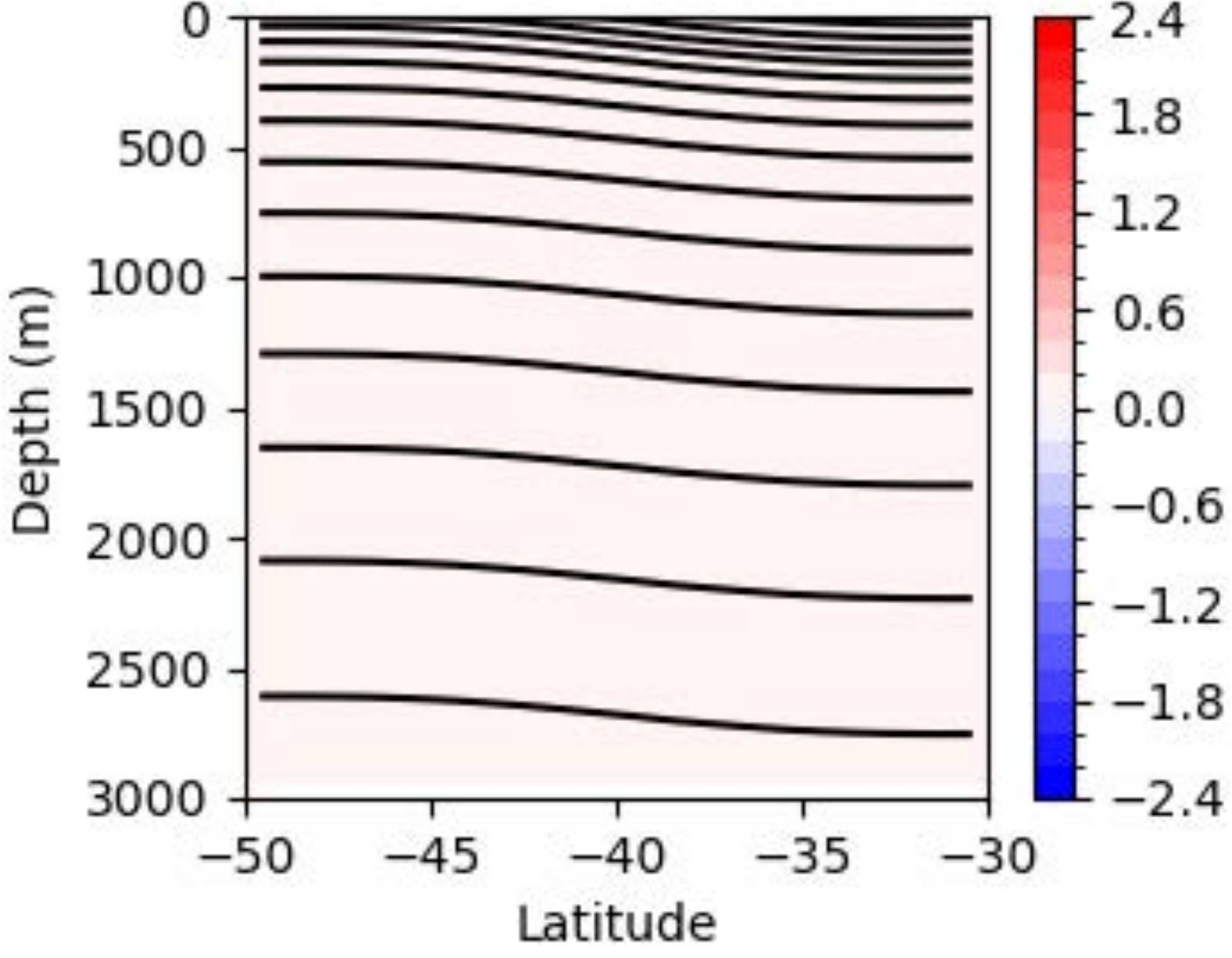
+

Overturning streamfunction for eddy-driven circulation, $\Psi*$

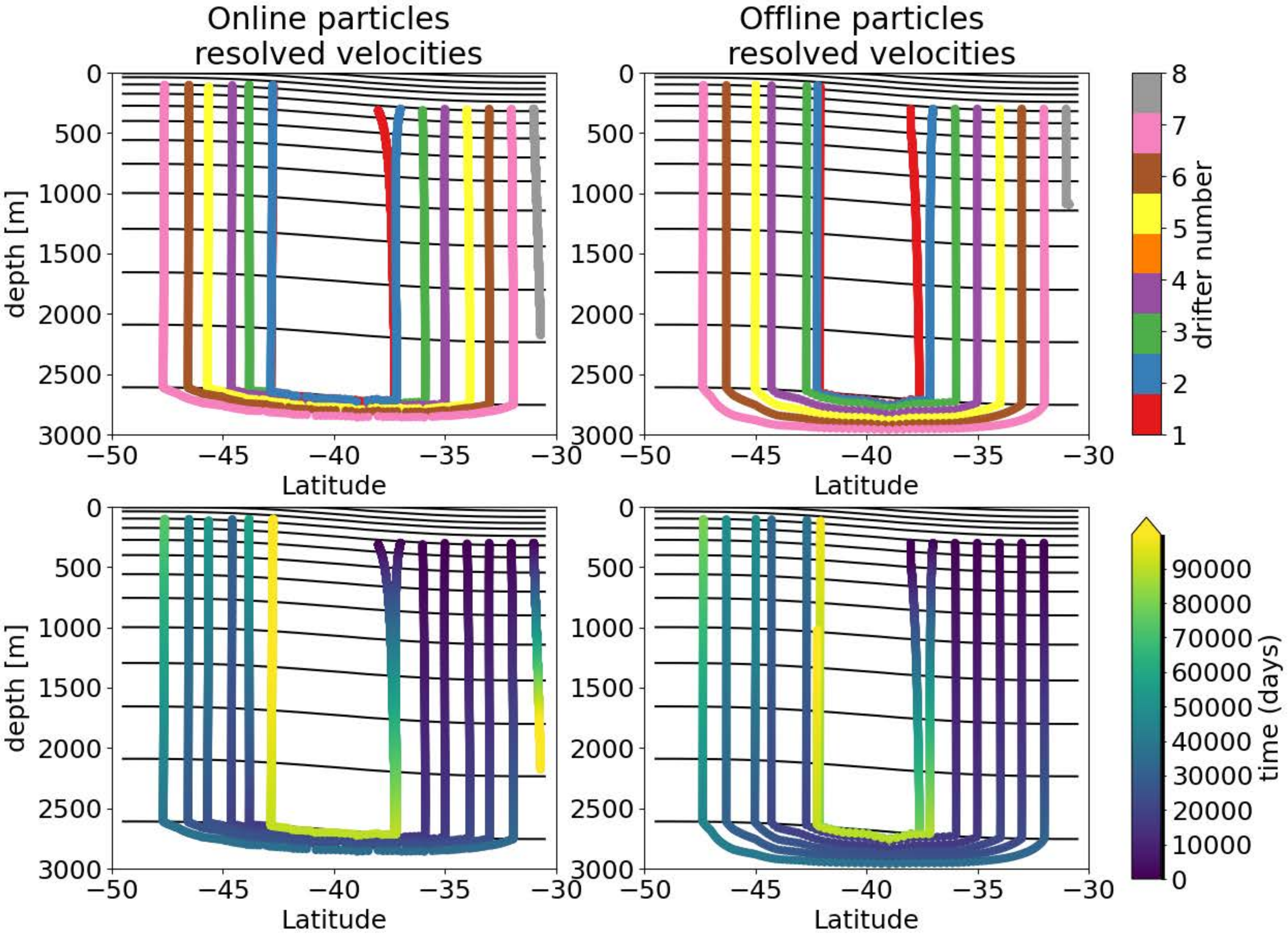


=

Overturning streamfunction for residual circulation, Ψ_{res}



Online vs offline trajecories in adiabatic channel (resolved)

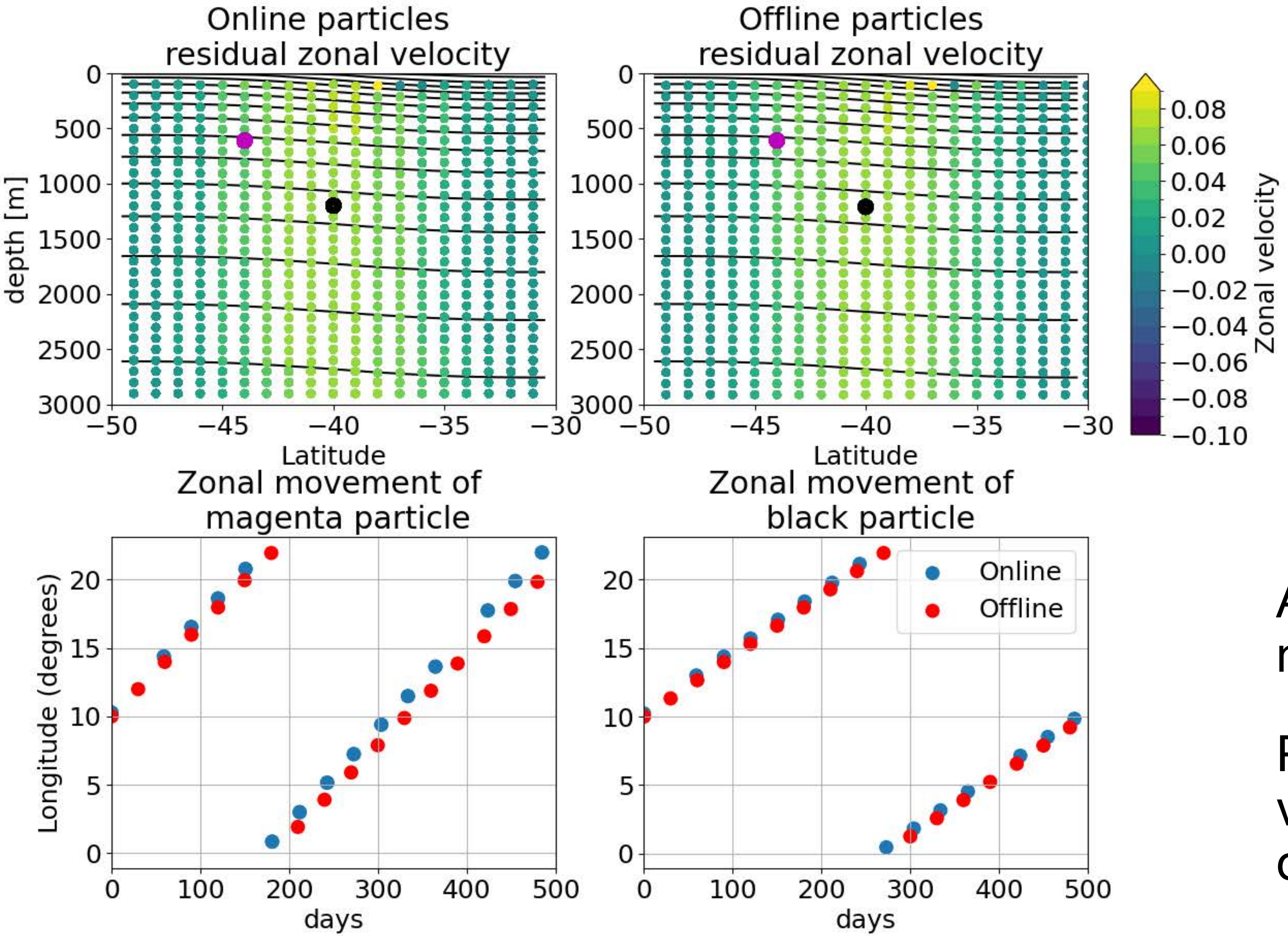


Offline particle advection performed using Parcels

Some small differences that I am still trying to understand

Overall, online particles are spread evenly in upward branch, suggesting mass conservation

Online vs offline trajecories in adiabatic channel (residual)

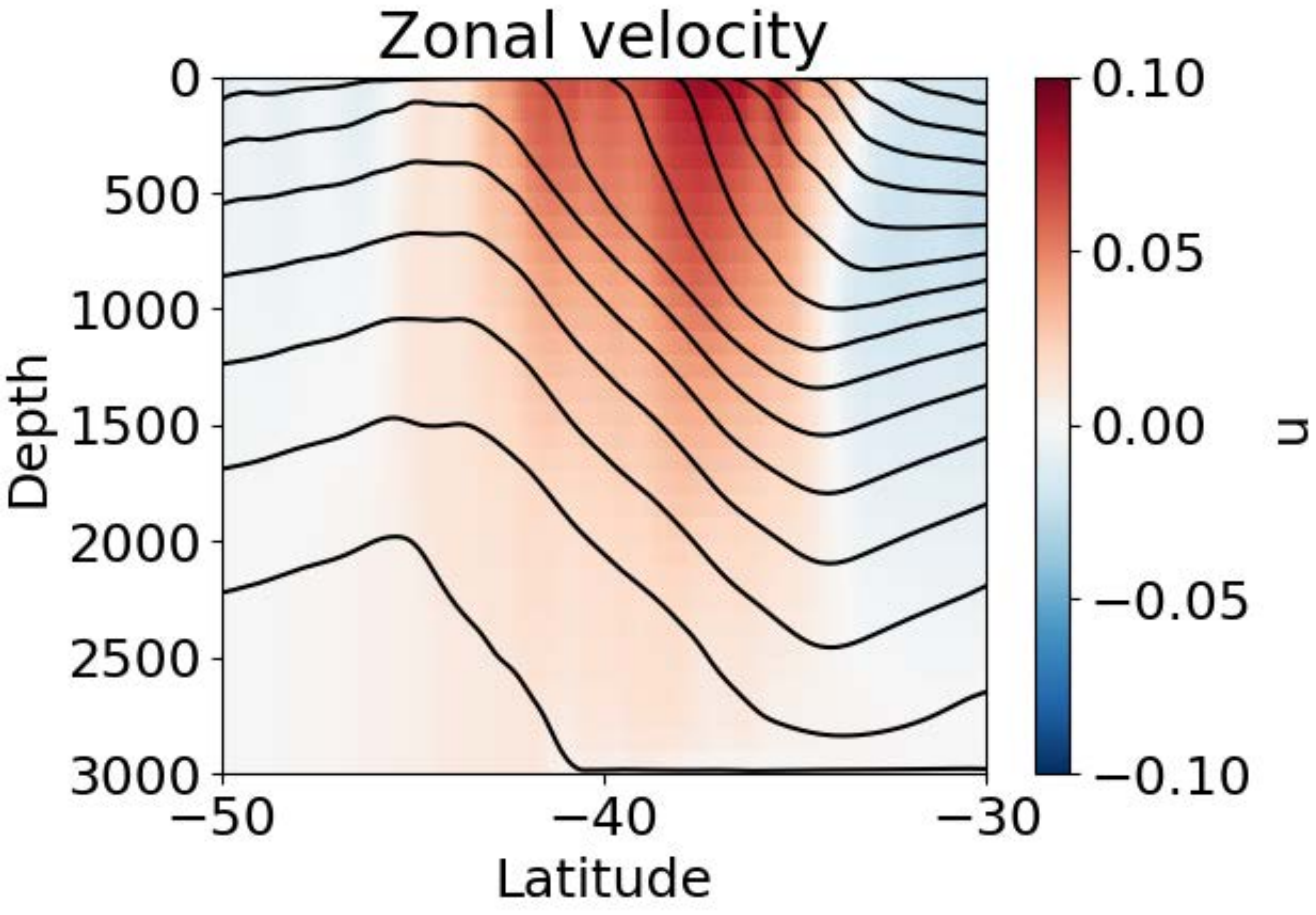


As expected, no flow in meridional or z directions

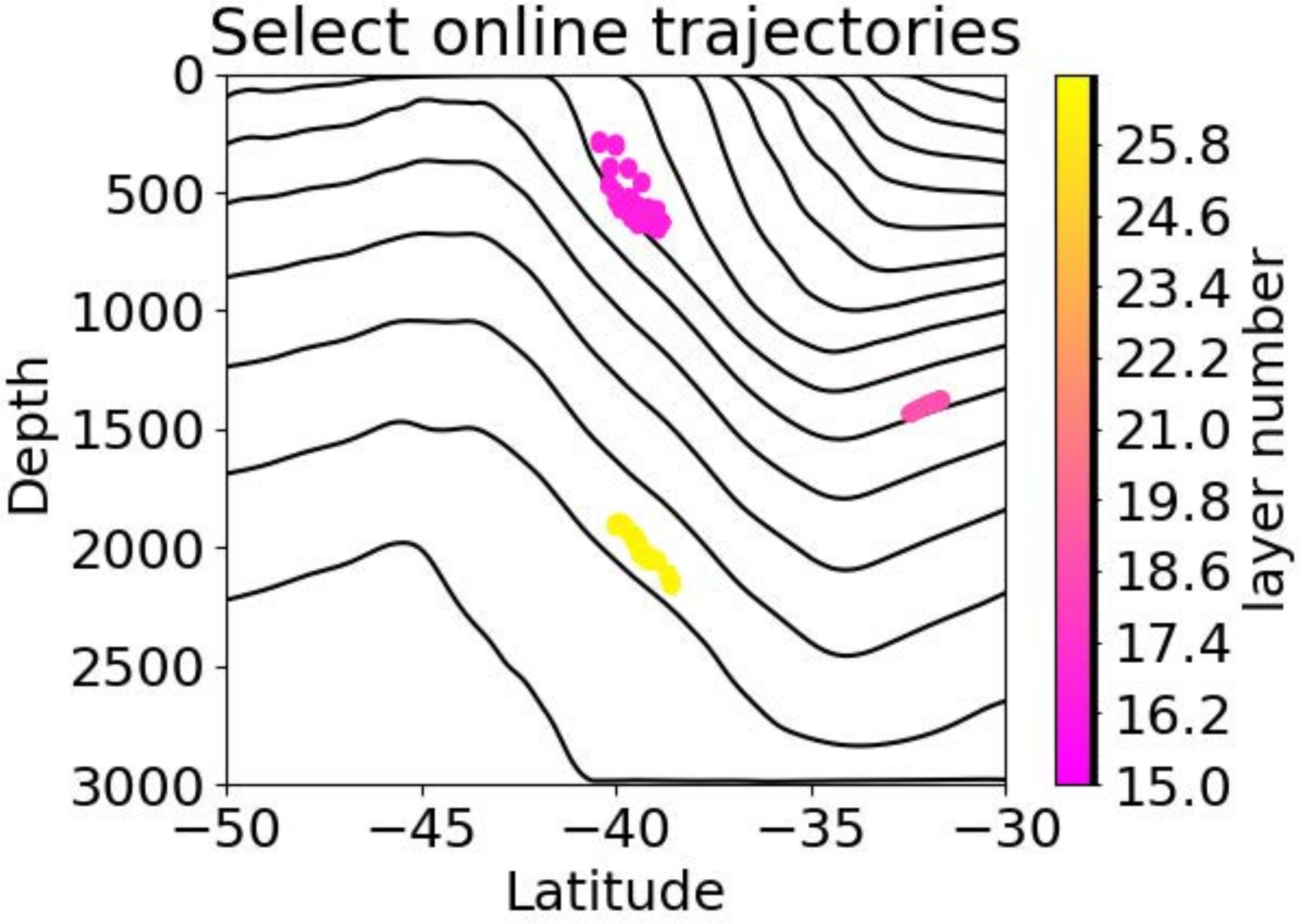
Particle trajectories match very closely in the zonal direction

Examples in adiabatic “stacked shallow water” channel

Now we go to 0.1 degree resolution and switch off the eddy parameterization

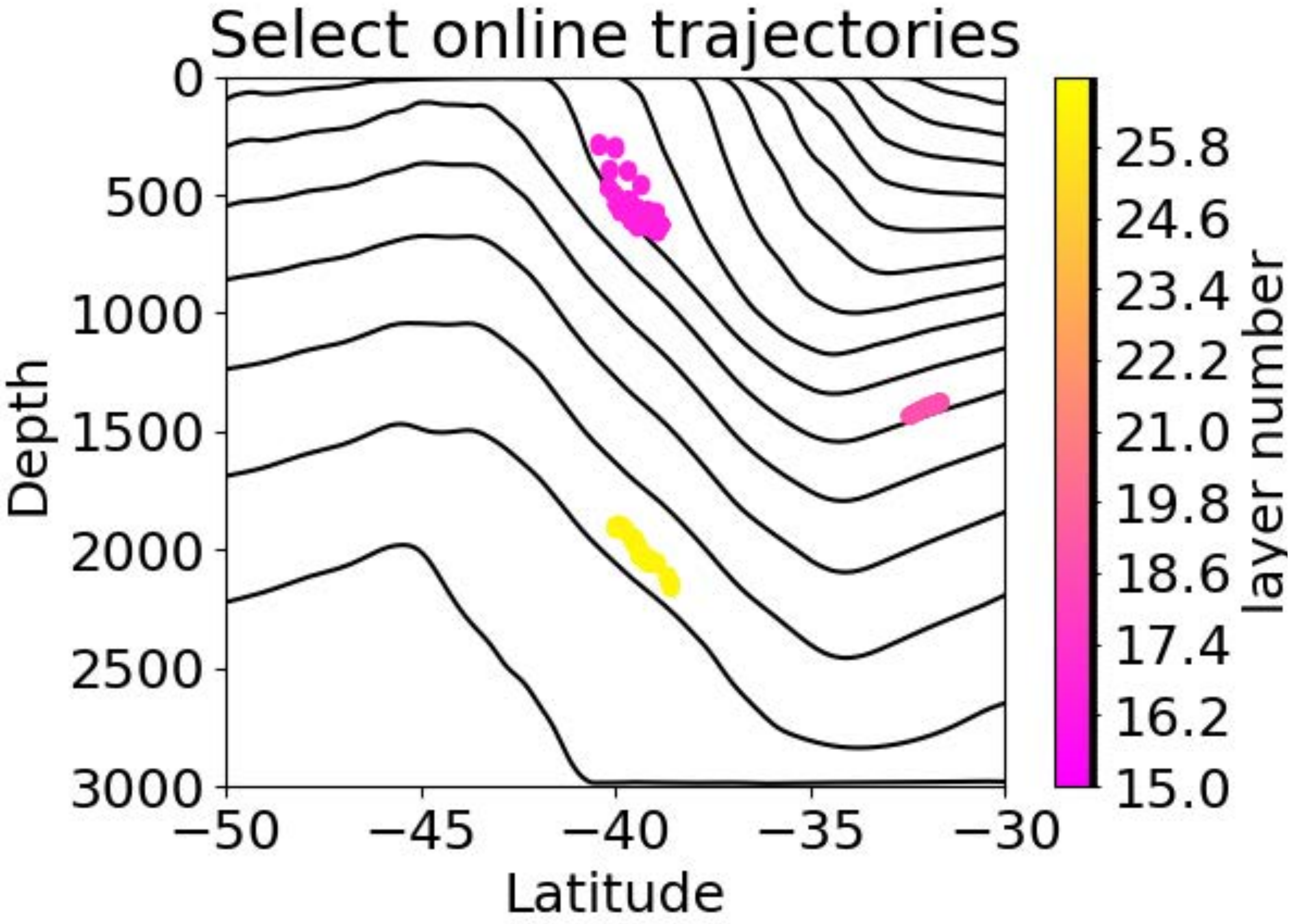


(This run is not in equilibrium yet)

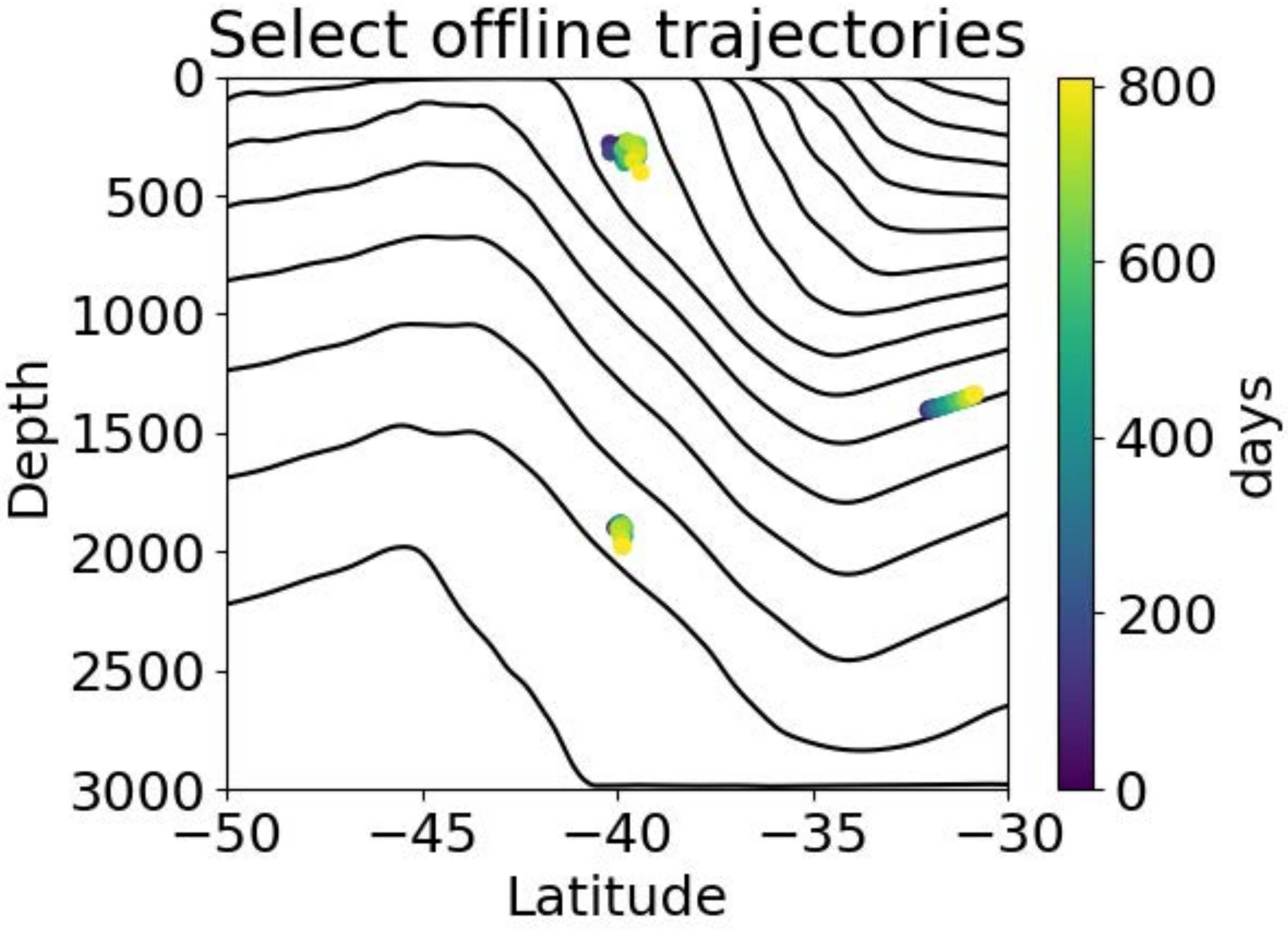


Online trajectories stay in the same layer

Offline trajectories are sensitive to output frequency



Select online trajectories

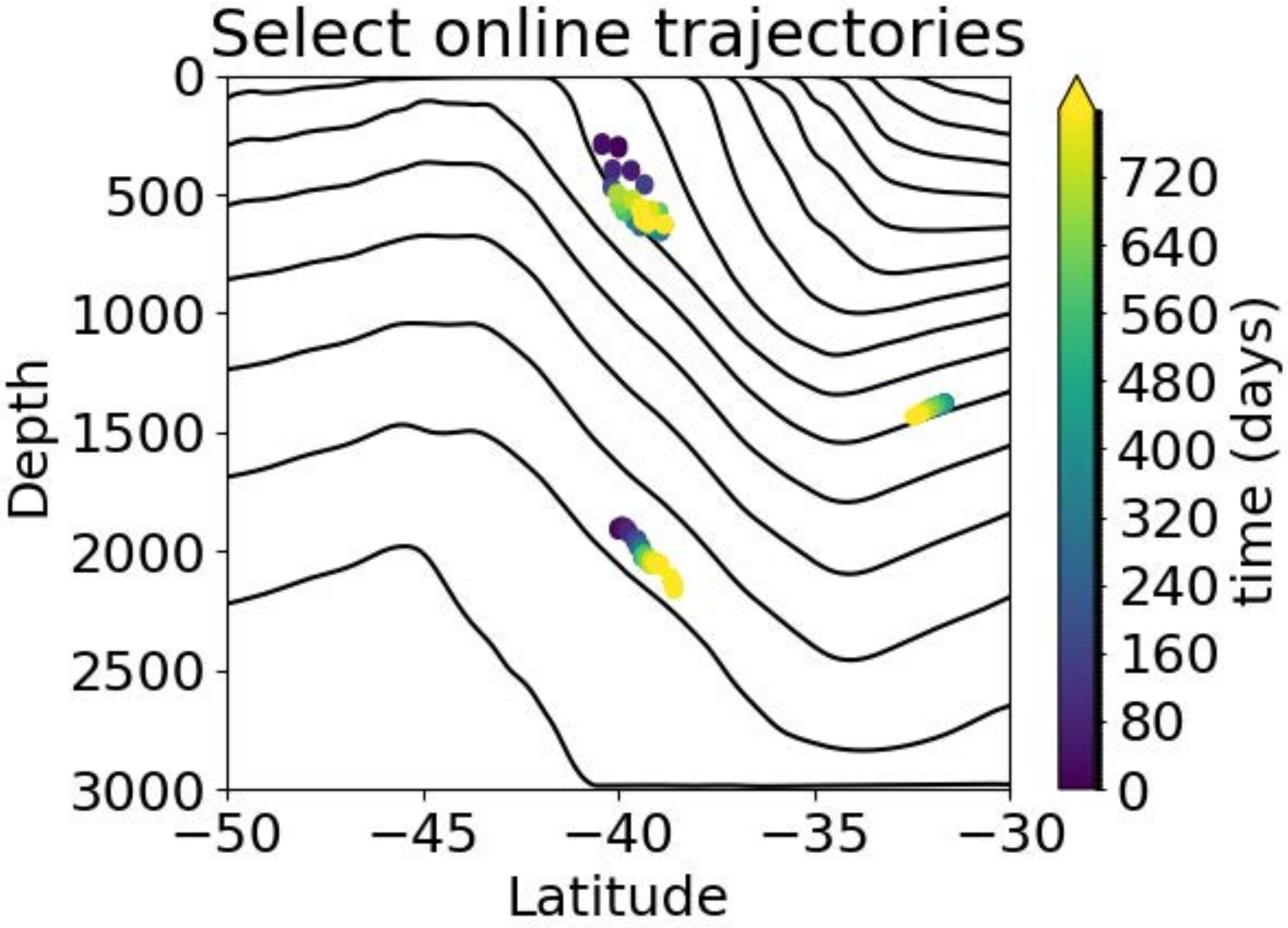


Select offline trajectories

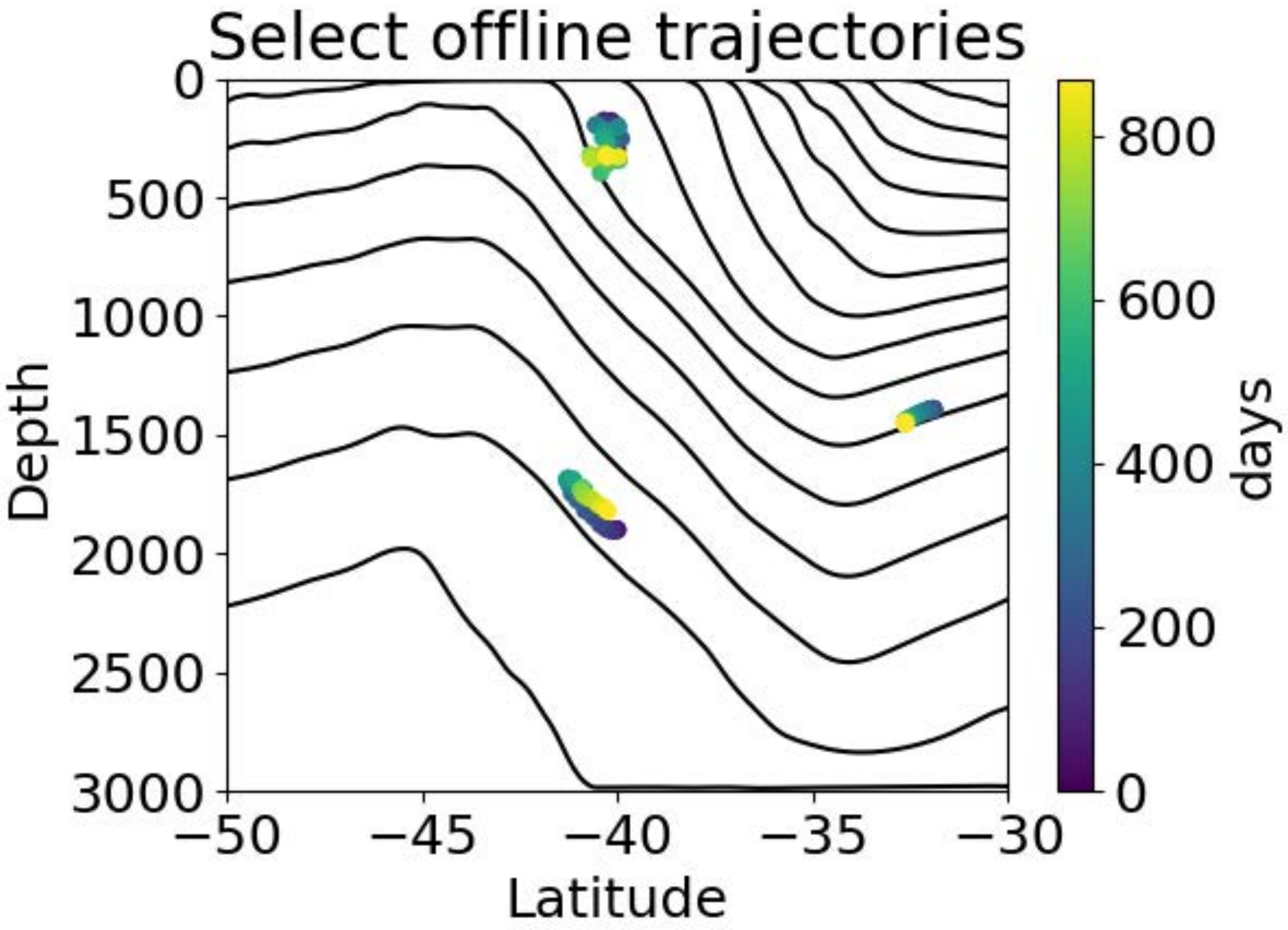
Online trajectories stay in the same layer

e.g. if velocities are averaged over 120 days, particles don't move much because this averages over the eddies

Offline trajectories are sensitive to output frequency



Online trajectories stay in the same layer



If velocities are averaged over 1day, offline trajectories are more similar to online trajectories

Conclusions

- The traditional method for online particle advection will not work in models with a Lagrangian vertical coordinate

- The new method presented here conserves mass

- We can advect particles using both the resolved and the residual velocity field (though not at the same time)

- Online methods generally match offline methods at coarse resolution

- Online methods should have less spurious vertical movement for high resolution

spencerjones@tamu.edu