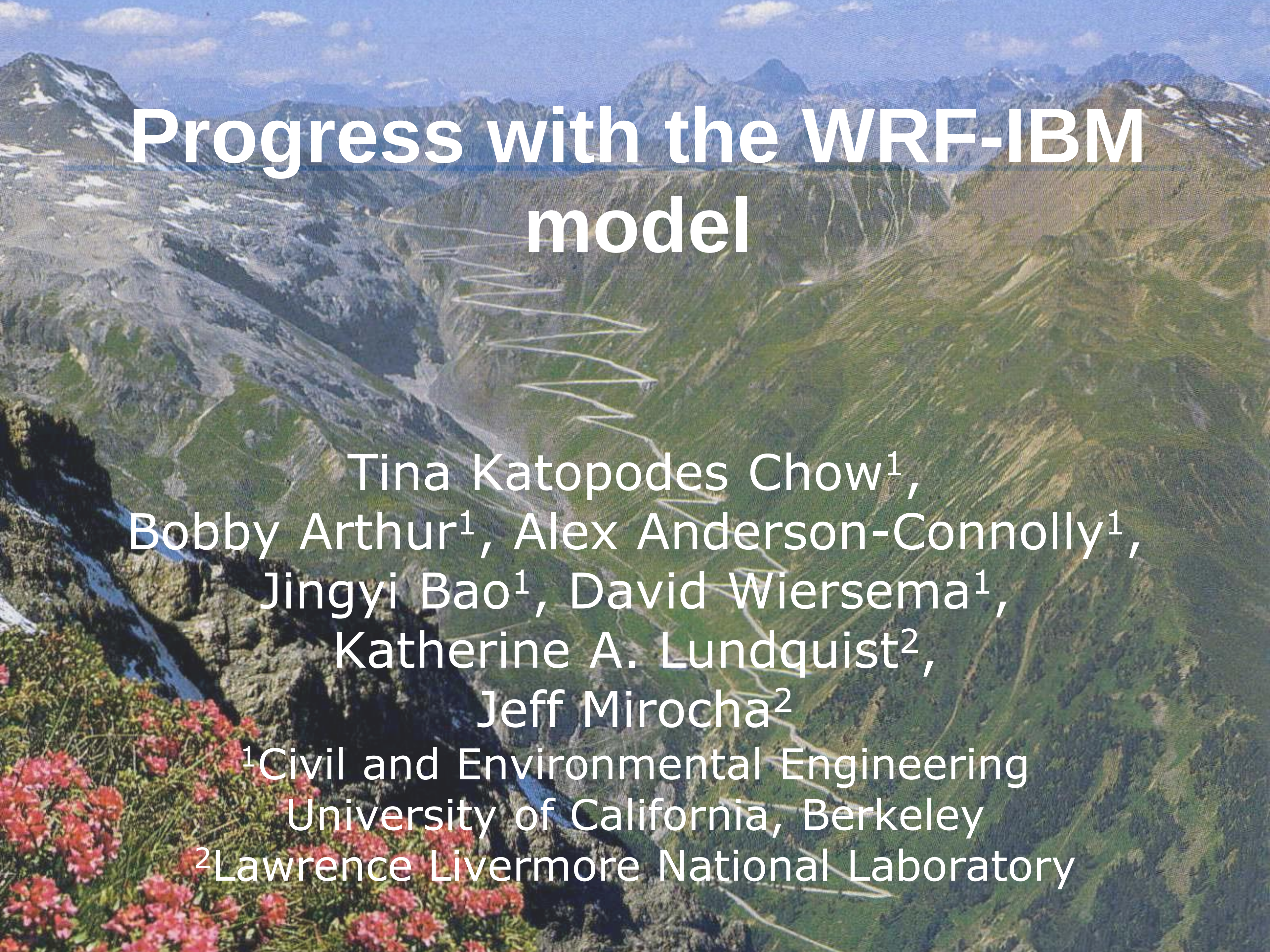
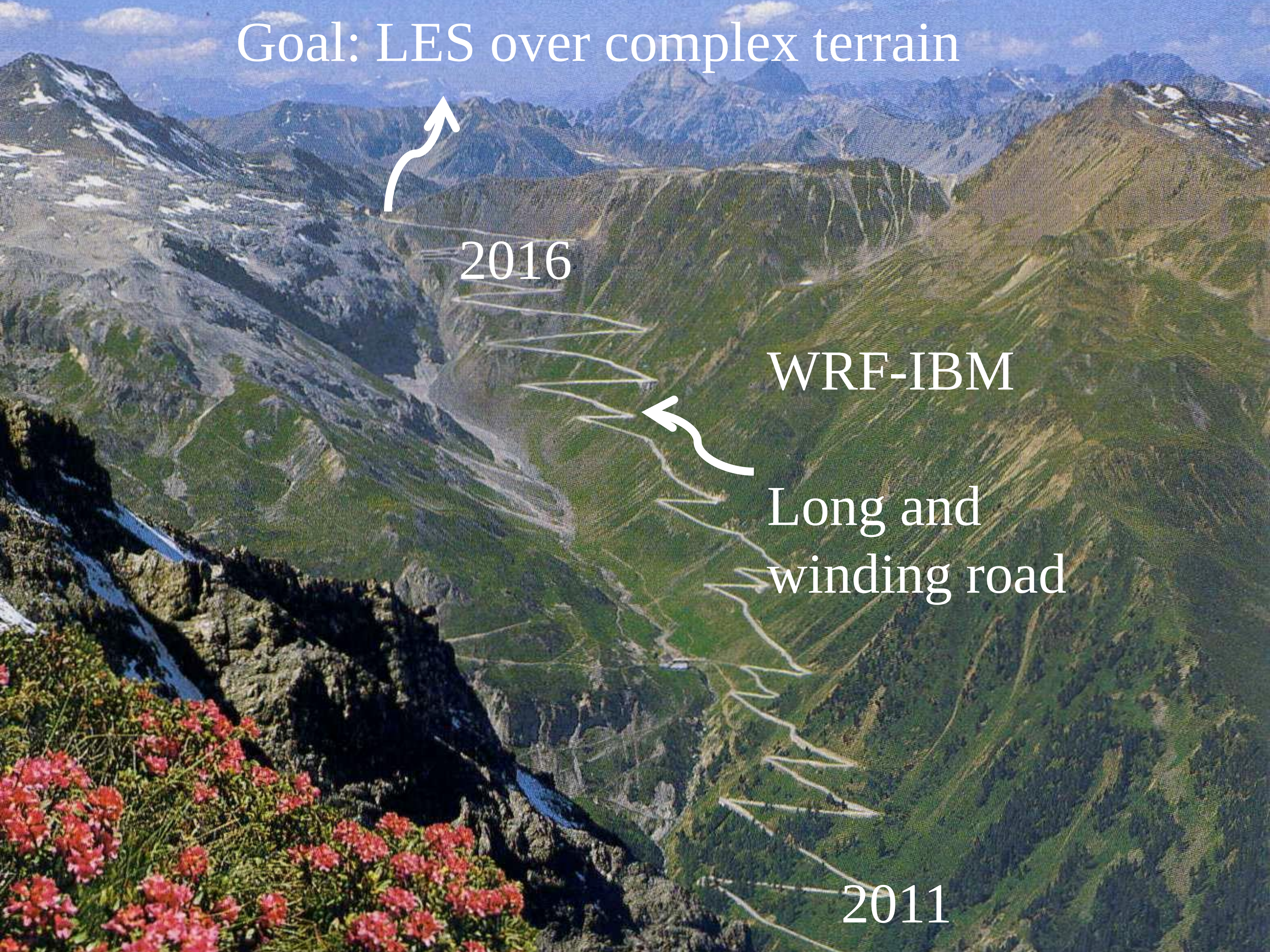


Progress with the WRF-IBM model

Tina Katopodes Chow[1],
Bobby Arthur[1], Alex Anderson-Connolly[1],
Jingyi Bao[1], David Wiersema[1],
Katherine A. Lundquist[2],
Jeff Mirocha[2]
[1]Civil and Environmental Engineering
University of California, Berkeley
[2]Lawrence Livermore National Laboratory

Goal: LES over complex terrain



2016

WRF-IBM

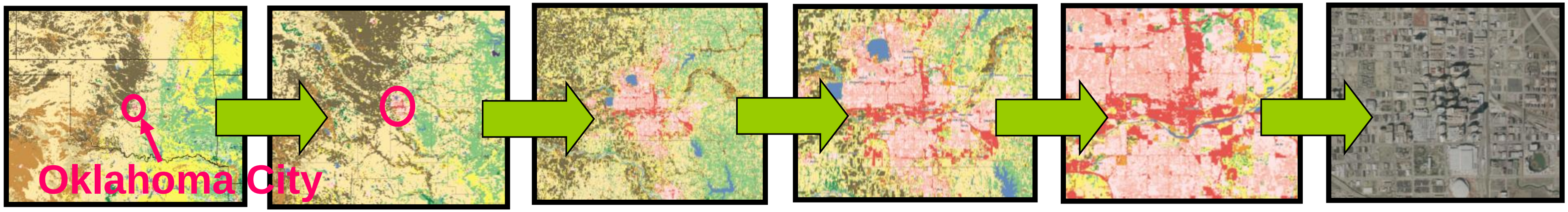
Long and winding road

2011

Road map

- WRF-IBM for complex terrain
- Implementing the log law
- Current/future work

WRF to WRF-IBM – seamless grid nesting



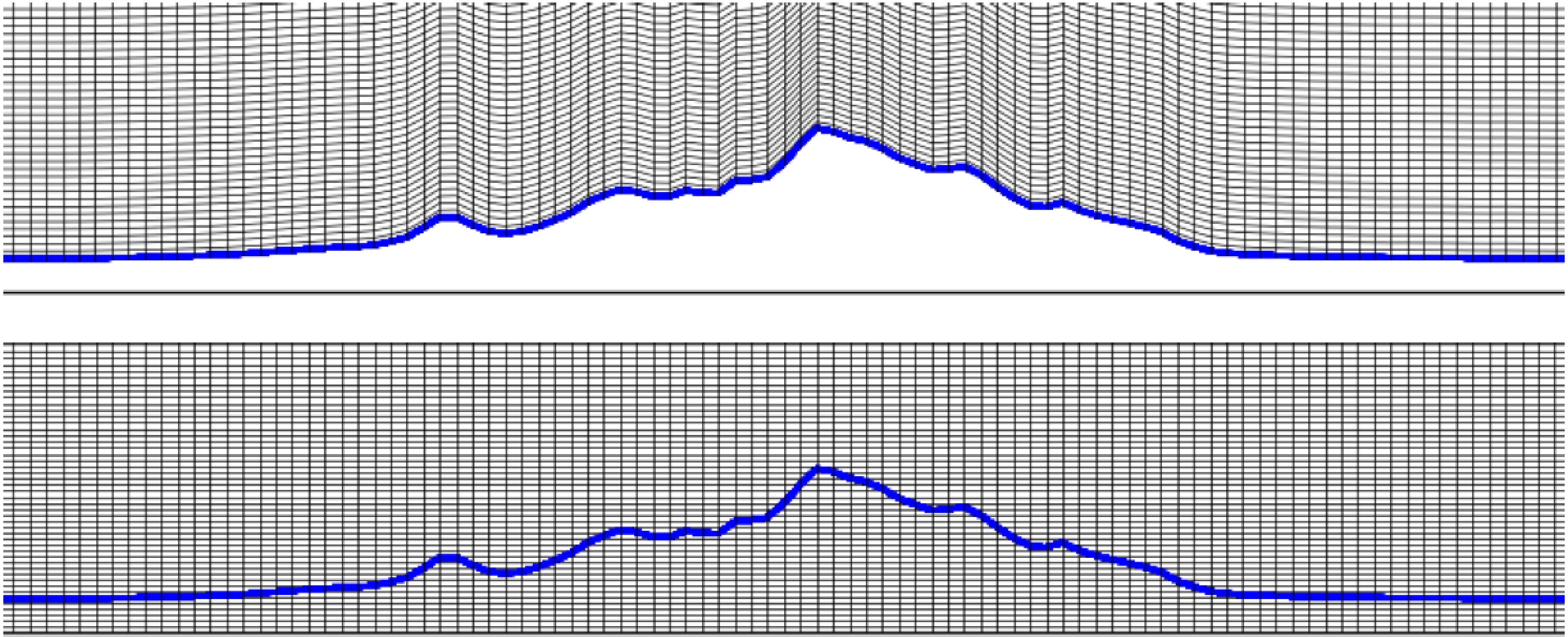
3 km | 1 km | 350 m | 100 m | 30 m | 5-10 m

WRF -> WRF-IBM

transition

- Weather and Research Forecasting (WRF) model
  - Mesoscale to microscale
- One tool for all scales
  - Improved turbulence models for LES
  - Immersed boundary method (IBM) for steep terrain

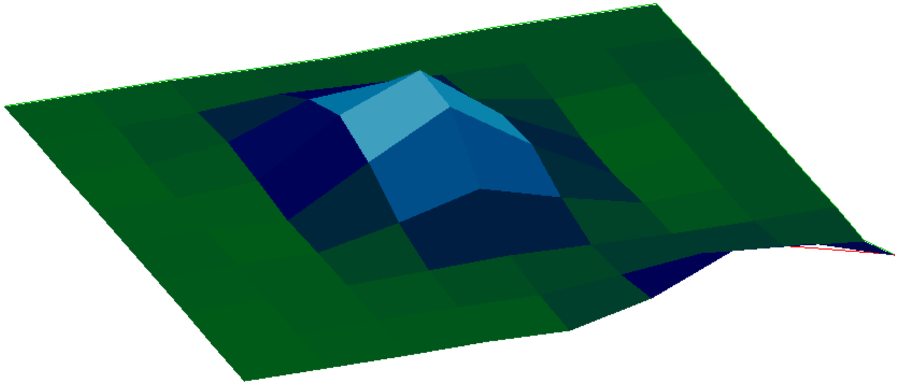
Immersed boundary method



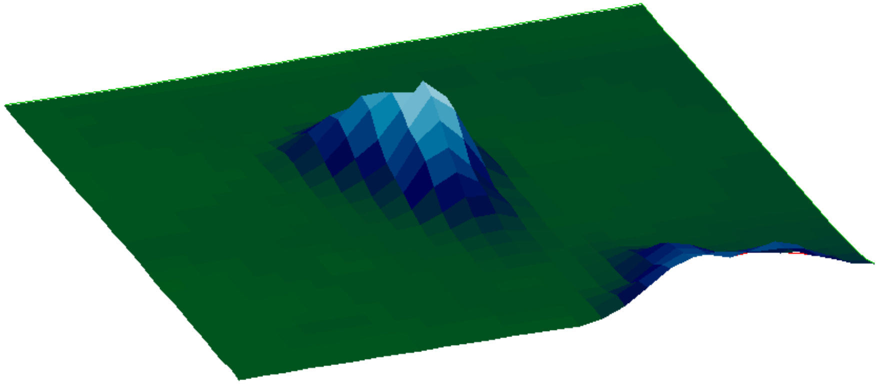
Granite Mountain, Utah

Increasing resolution → steeper slopes

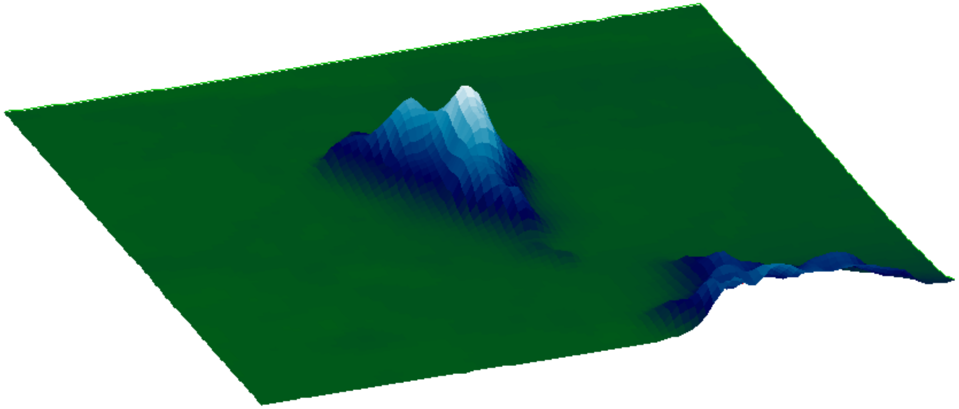
3 km, max slope ~4°



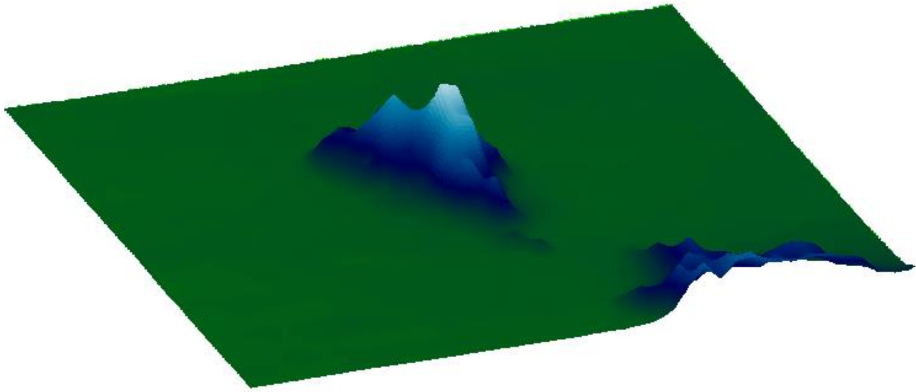
1 km, max slope ~14°



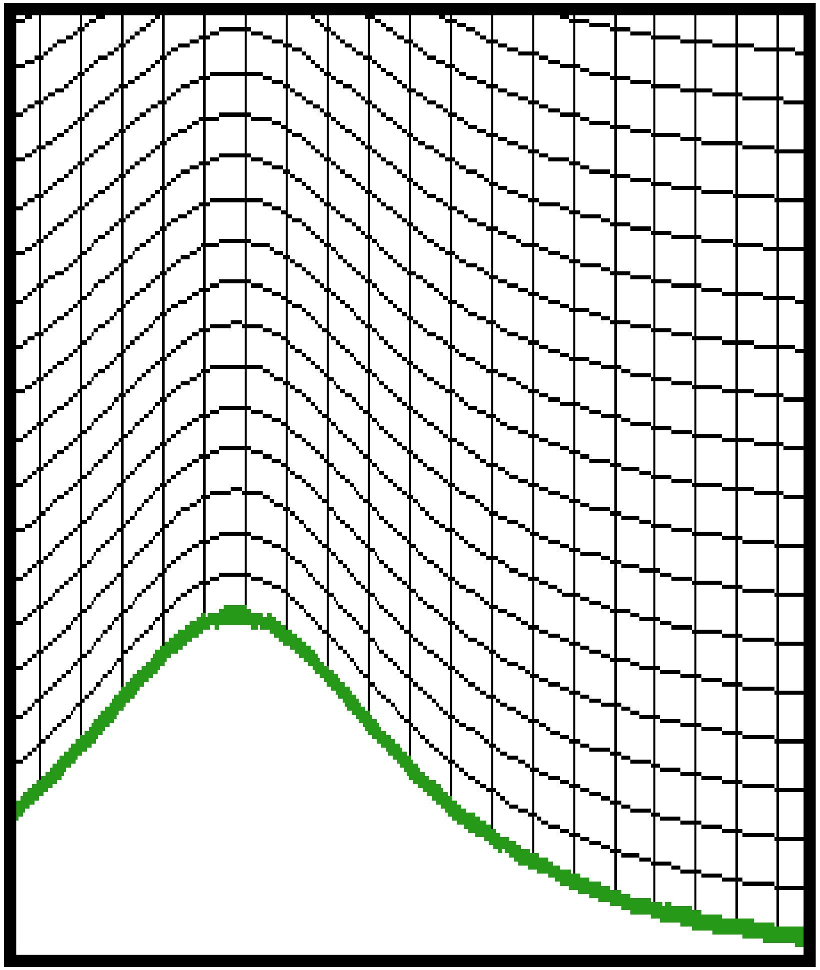
300 m, max slope ~28°



100 m, max slope ~40°



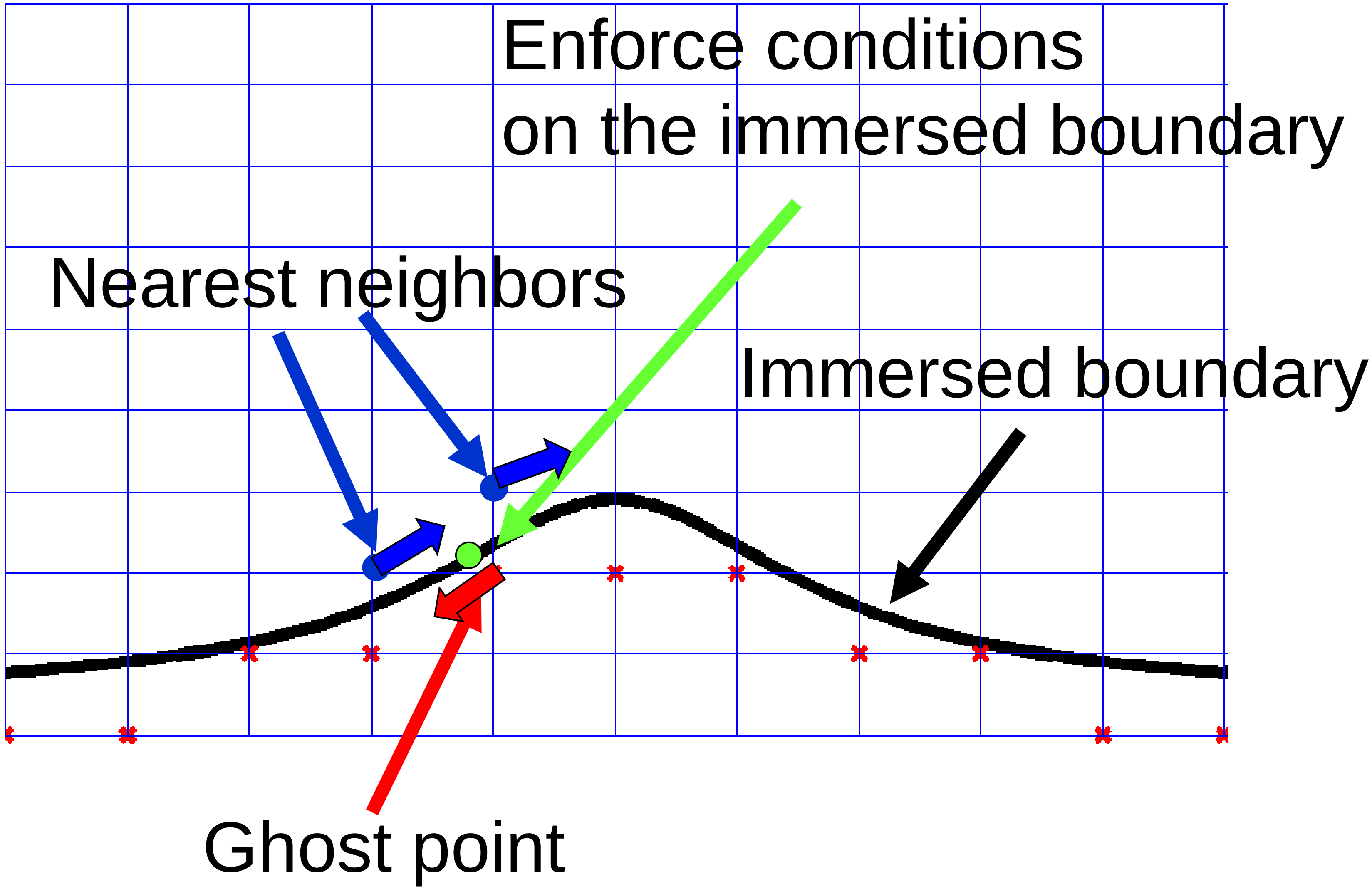
Terrain slope limit



Terrain-following coordinates

- Horizontal pressure gradient errors
  - 45° limit, usually ~30° starts causing problems (e.g. Mahrer 1984)
- Grid aspect ratio limitations
- Numerical stability

Ghost-cell immersed boundary method



Enforce conditions
on the immersed boundary

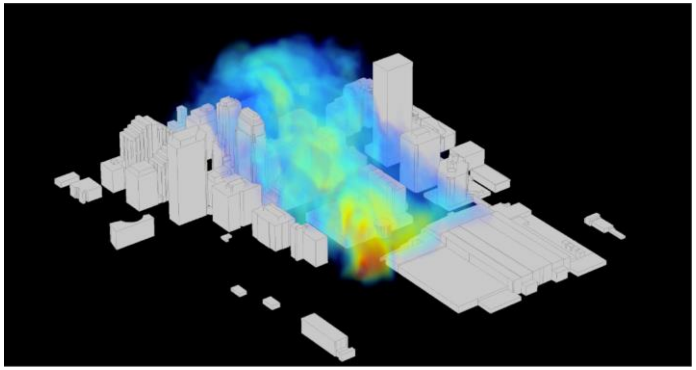
Nearest neighbors

Immersed boundary

Ghost point

Complex terrain applications

- Current implementation for no-slip
  - Good for urban environments at ~1 m resolution
- Need log law wall stress for complex terrain



IBM-WRF for Oklahoma City

$$U = \frac{u_*}{\kappa} \ln\left(\frac{z + z_0}{z_0}\right) \qquad C_D = \left[\frac{1}{\kappa} \ln\left(\frac{z_1 + z_0}{z_0}\right)\right]^{-2}$$

$$\tau_{wall} = -u_*^2 = -C_D \left|U_1\right| U_1$$

WRF implementation of log law

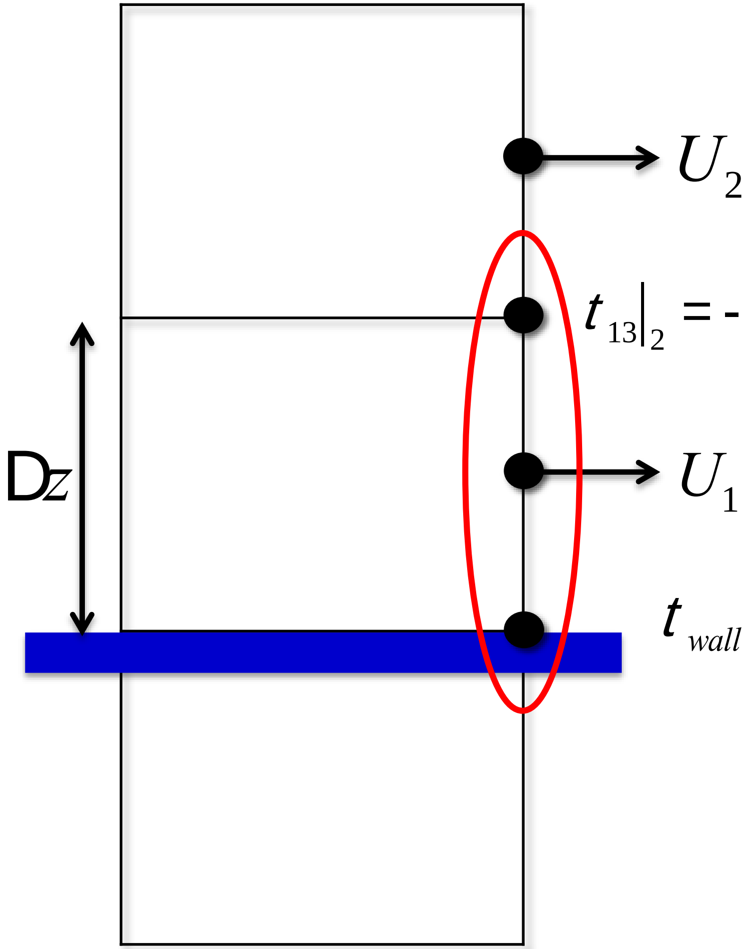
- Momentum equation in U direction

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + W\frac{\partial U}{\partial z} = -\frac{1}{\rho}\frac{\partial P}{\partial x} - \left(\frac{\partial \tau_{11}}{\partial x} + \frac{\partial \tau_{12}}{\partial y} + \frac{\partial \tau_{13}}{\partial z}\right)$$

- Requires gradient in τ_{13}

WRF implementation of log law

$$\frac{\partial U}{\partial t} = -\frac{\partial \tau_{13}}{\partial z} = -\frac{\tau_{13}\big|_2 - \tau_{wall}}{\Delta z}\ldots$$

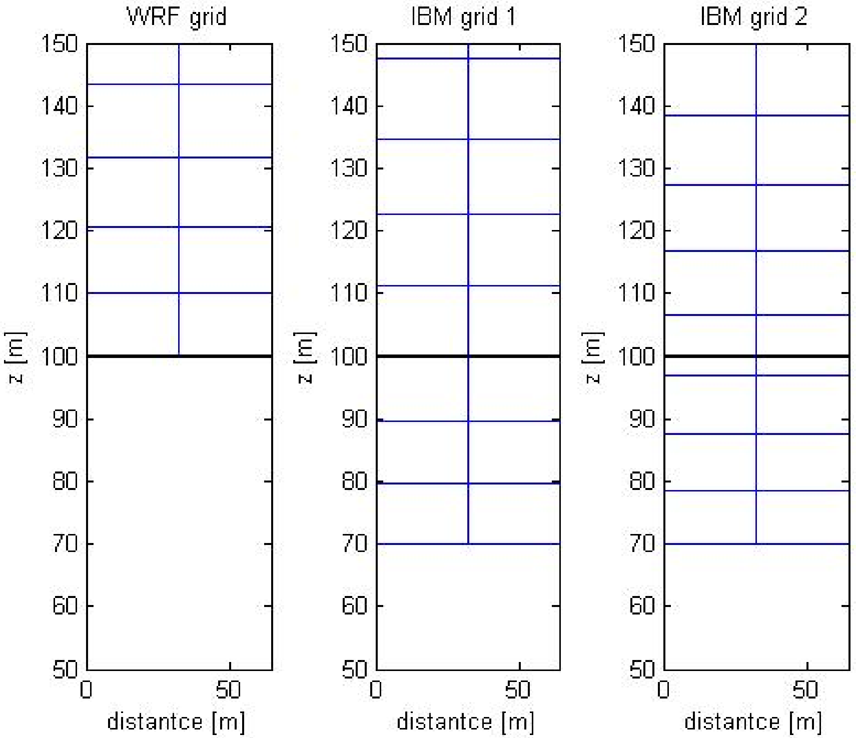


$$\tau_{13}\big|_2 = -\nu_T \frac{\partial U}{\partial z} = -\nu_T \frac{U_2 - U_1}{\Delta z}$$

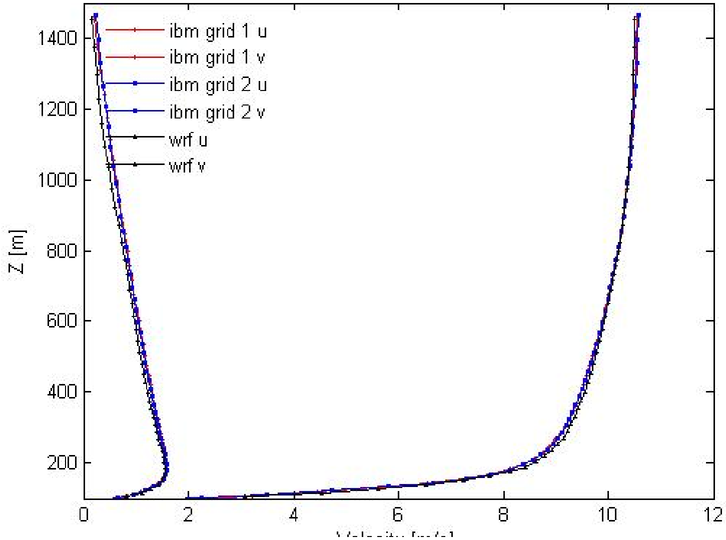
$$\tau_{wall} = -u_*^2 = C_d |U_1| U_1$$

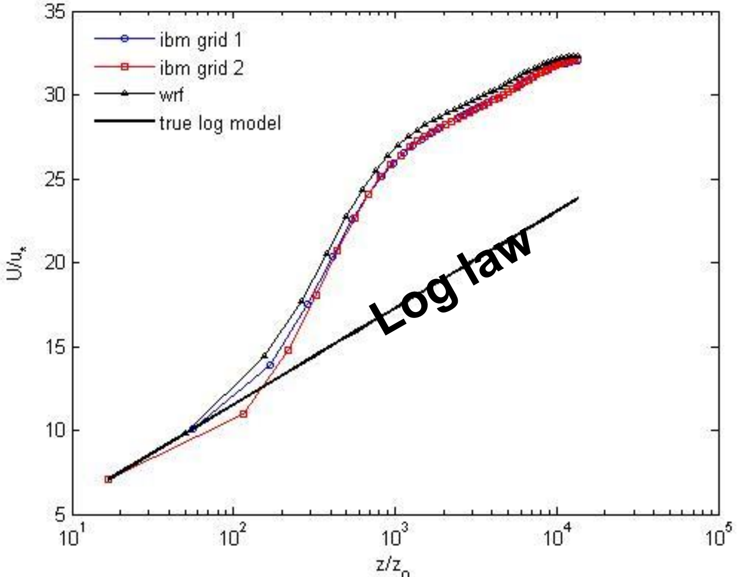
WRF-IBM log law - testing

Neutral boundary layer



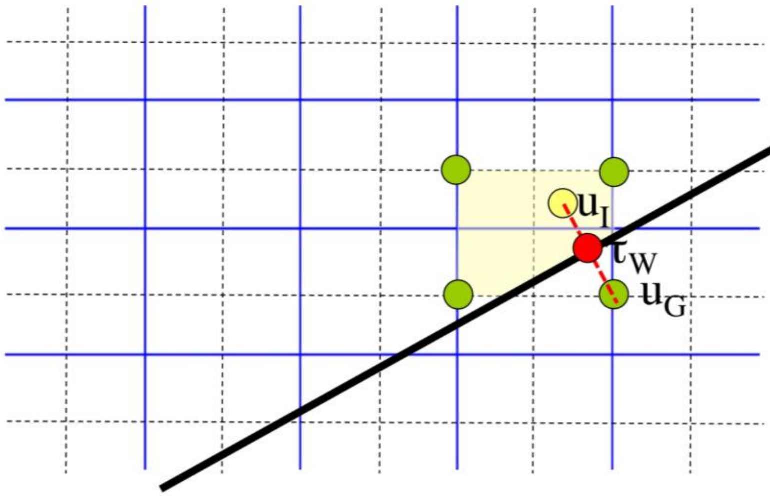
Grid Setup



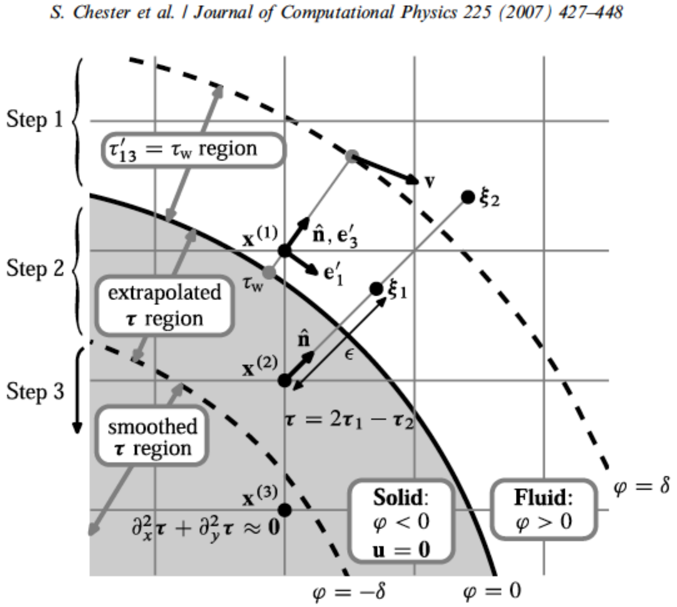


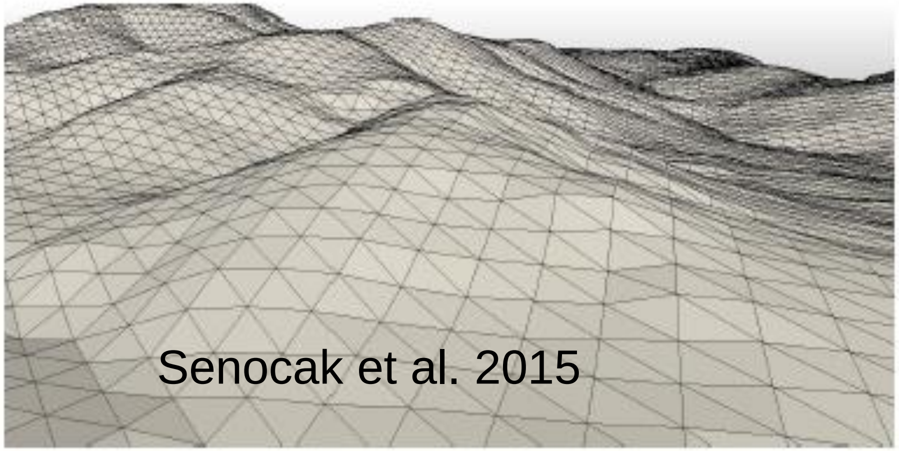
Log law + WRF-IBM

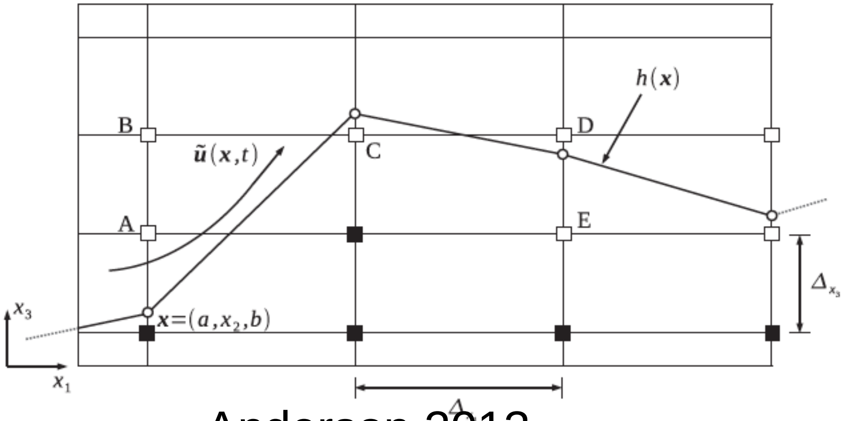
- It’s all in the details



Chester et al. 2007


S. Chester et al. / Journal of Computational Physics 225 (2007) 427–448


Senocak et al. 2015



Anderson 2013

Velocity gradient based IBM

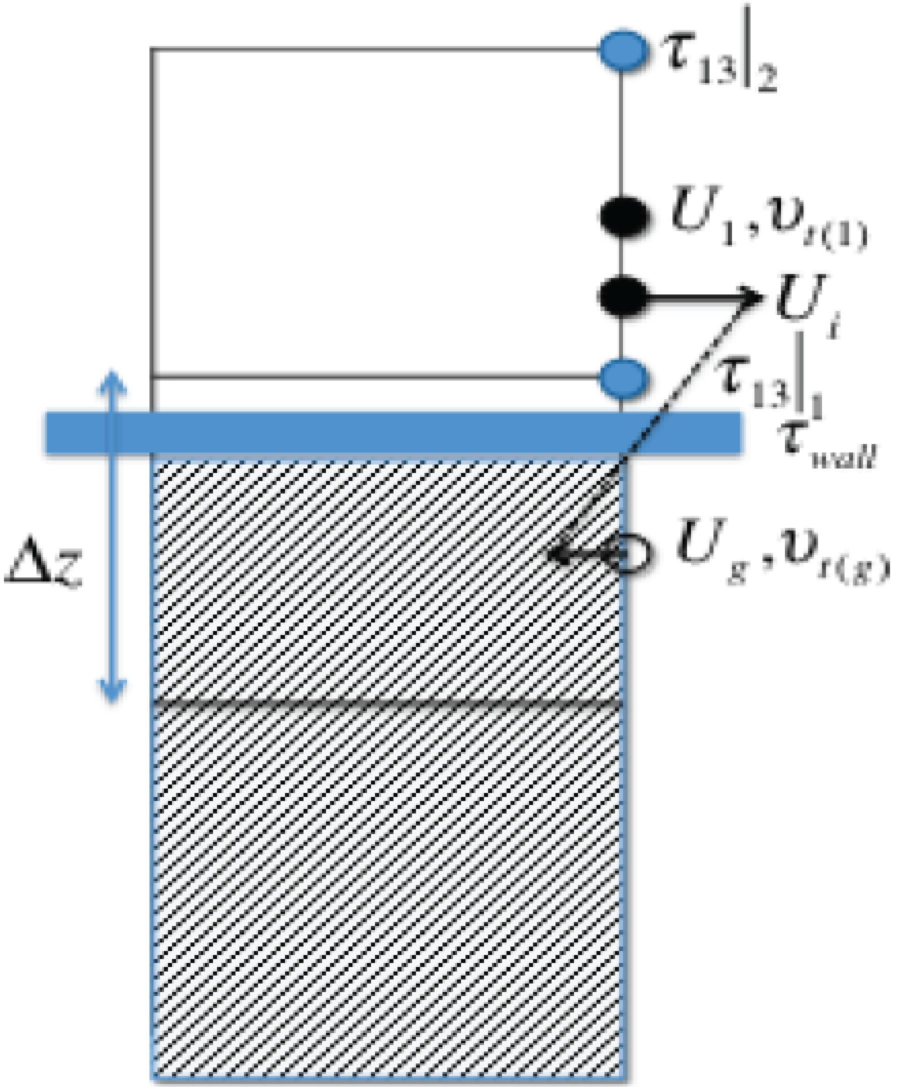
- Reconstruct velocity at ghost cell based on wall stress

$$\tau_{wall} = -\nu_t \frac{\delta u}{\delta z}$$

$$\frac{\delta u}{\delta z} = -\frac{\tau_{wall}}{\nu_{t(wall)}}$$

$$\tau_{wall_{xz}} = -\frac{k}{\ln \frac{z_1}{z_0}}^2 |U|\,U$$

$$\tau_{wall_{yz}} = -\frac{k}{\ln \frac{z_1}{z_0}}^2 |U|\,V$$



What's the eddy viscosity at the wall?

- Prandtl's mixing length $u_* k z$
  - Agrees with log law at wall
  - Very small values near the wall
- But WRF uses other eddy viscosity models even at the wall
  - E.g. Smagorinsky
  - Much larger than $\nu_{t(wall)}$ from Prandtl's model
- Large near-wall gradients and does not agree with standard WRF results
- Consider 3 new ways to estimate $\nu_{t(wall)}$

Estimating eddy viscosity at the wall

- Smagorinsky closure $\nu_{t(g)} = \nu_{t(1)} = \nu_{t(wall)}$
    - Set equal to value above
- Mason and Thompson 1992
    - Blending of length scales

$$\frac{1}{(l^*)^2} = -\frac{1}{(c_s\triangle)^2} + \frac{1}{(\kappa z)^2} \qquad \nu_t = -(l^*)^2\,|S_{ij}|$$

- Use this to set $\nu_{t(1)}$
- Then $\nu_{t(g)} = \nu_{t(1)} = \nu_{t(wall)}$

Estimating eddy viscosity at the wall

- Moeng 1984
  - Set $\frac{\delta u}{\delta z}$ according to log law at first point
  - This affects S_{ij} and hence $\nu_{t(1)}$

$$\nu_{t(g)} = \nu_{t(1)} = \nu_{t(wall)}$$

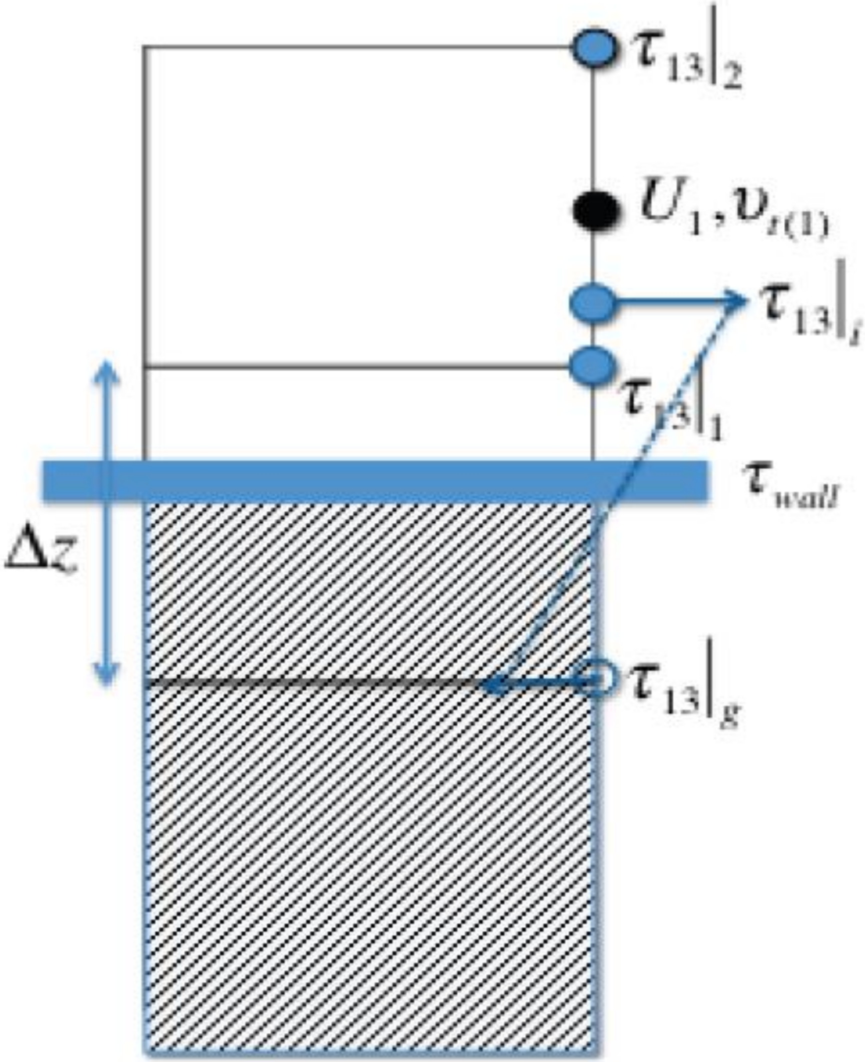
Shear stress reconstruction IBM

- Most similar to WRF’s BC

$$\tau_{wall_{xz}} = -\frac{k}{\ln\frac{z_1}{z_0}}^2 |U|U$$

$$\tau_{wall_{yz}} = -\frac{k}{\ln\frac{z_1}{z_0}}^2 |U|V$$

- My favorite
- But requires rotation of stress tensor

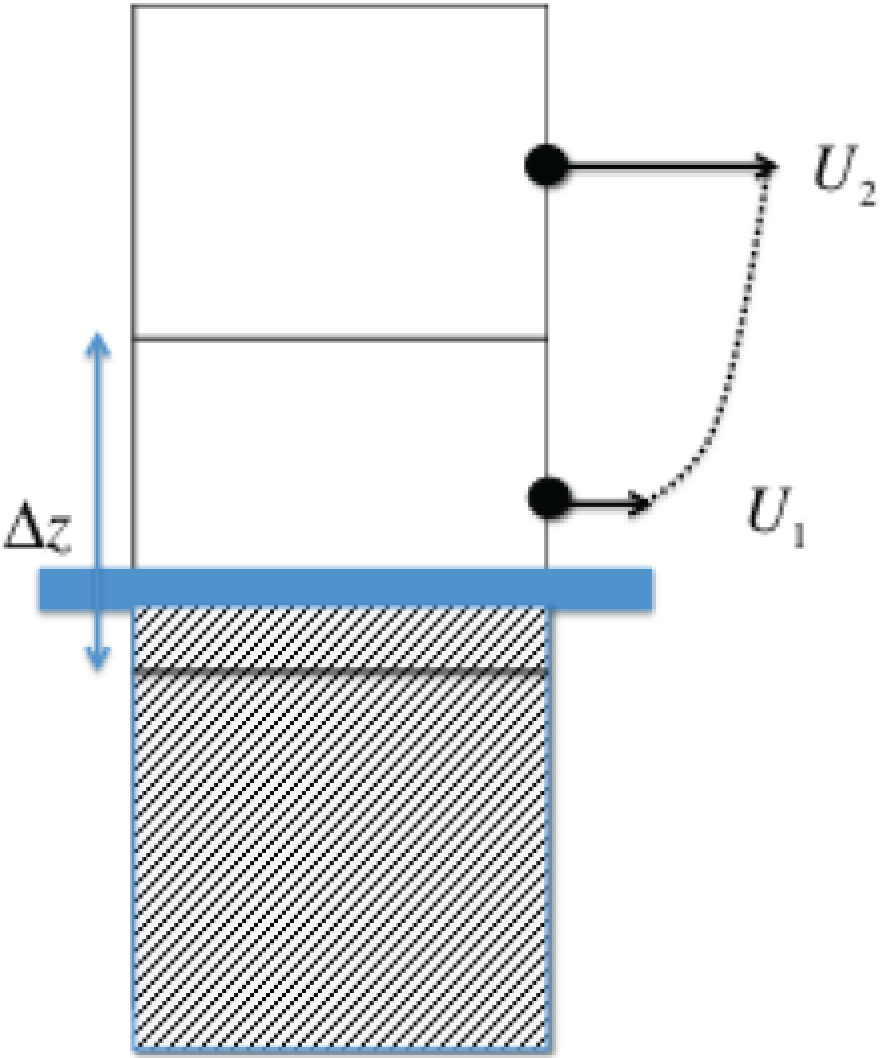


Velocity reconstruction IBM

- Reconstruct velocity according to log law
  - Fadlun et al. 2000
  - Senocak et al. 2004

$$\frac{U}{u^*} = \frac{1}{k} \ln \frac{z}{z_0}$$

$$U_1 = U_2 \frac{\ln \frac{z_1}{z_0}}{\ln \frac{z_1}{z_0}}$$

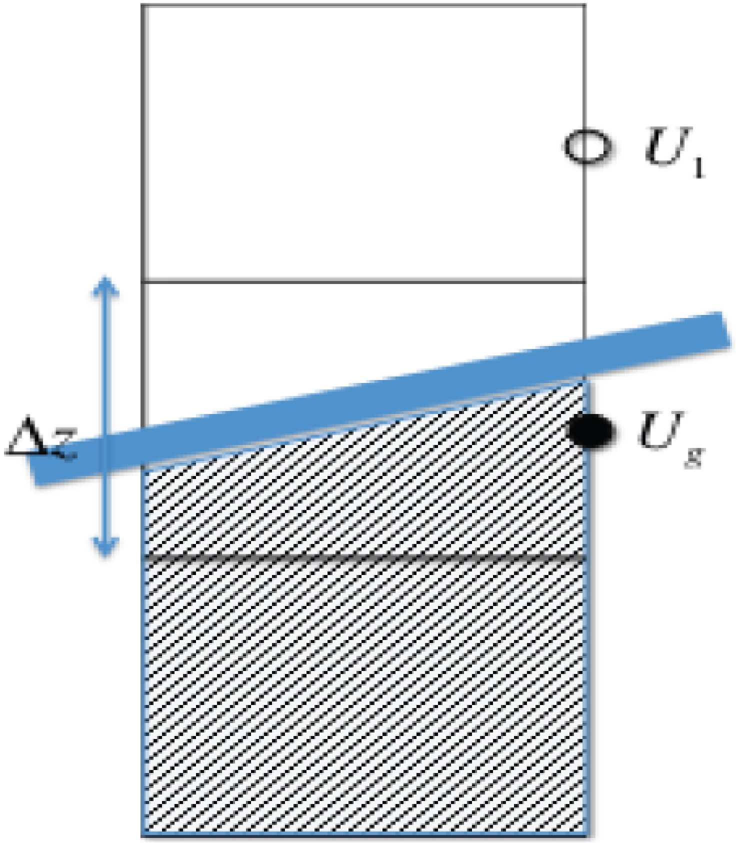


Canopy method IBM

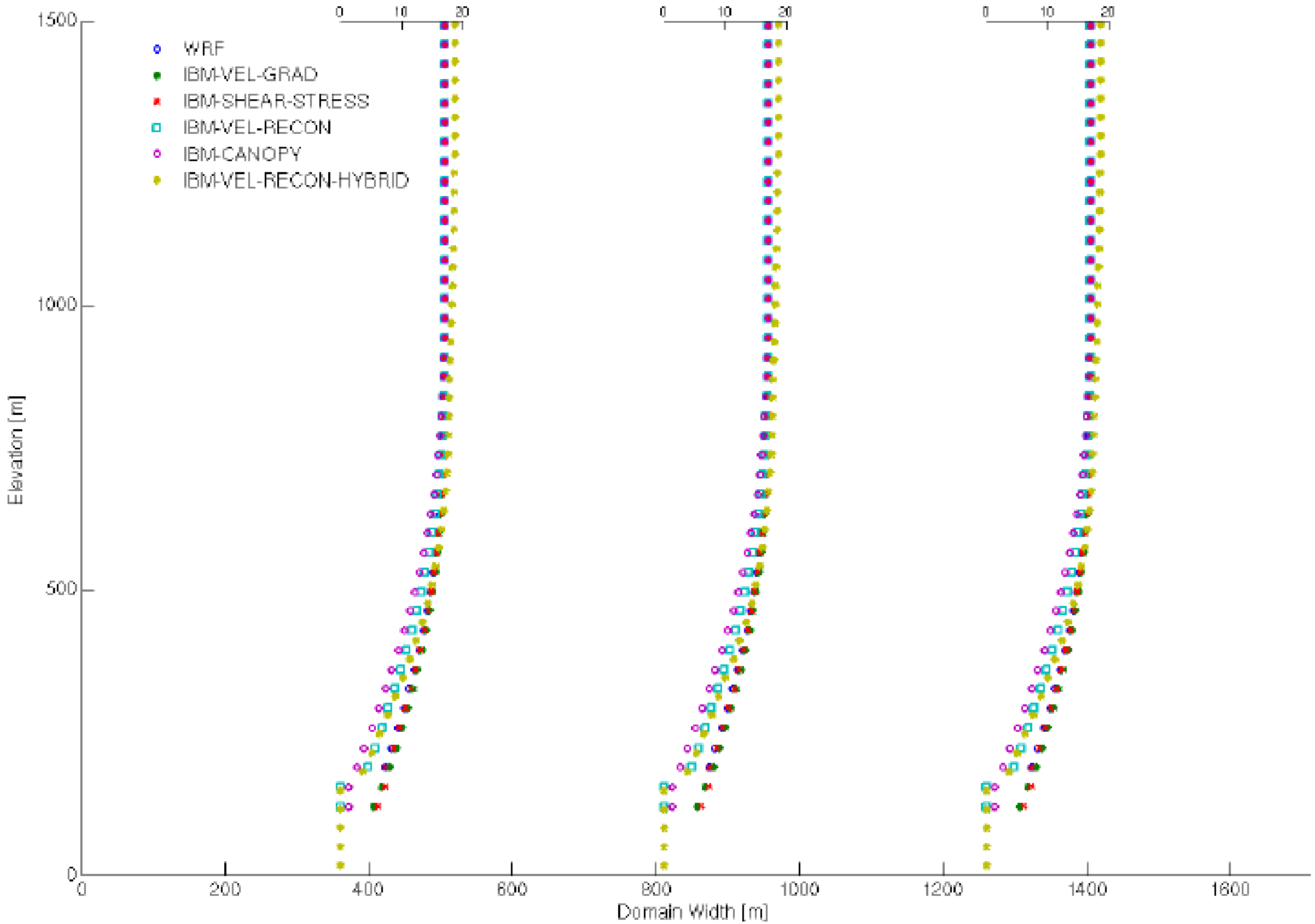
- Drag applied at cut cells
- All internal nodes set to zero
  - Anderson 2013
  - Frontal area *A(x)*

$$f(x,t) = -\frac{1}{2} C_D A(x) \, |U| \, U$$

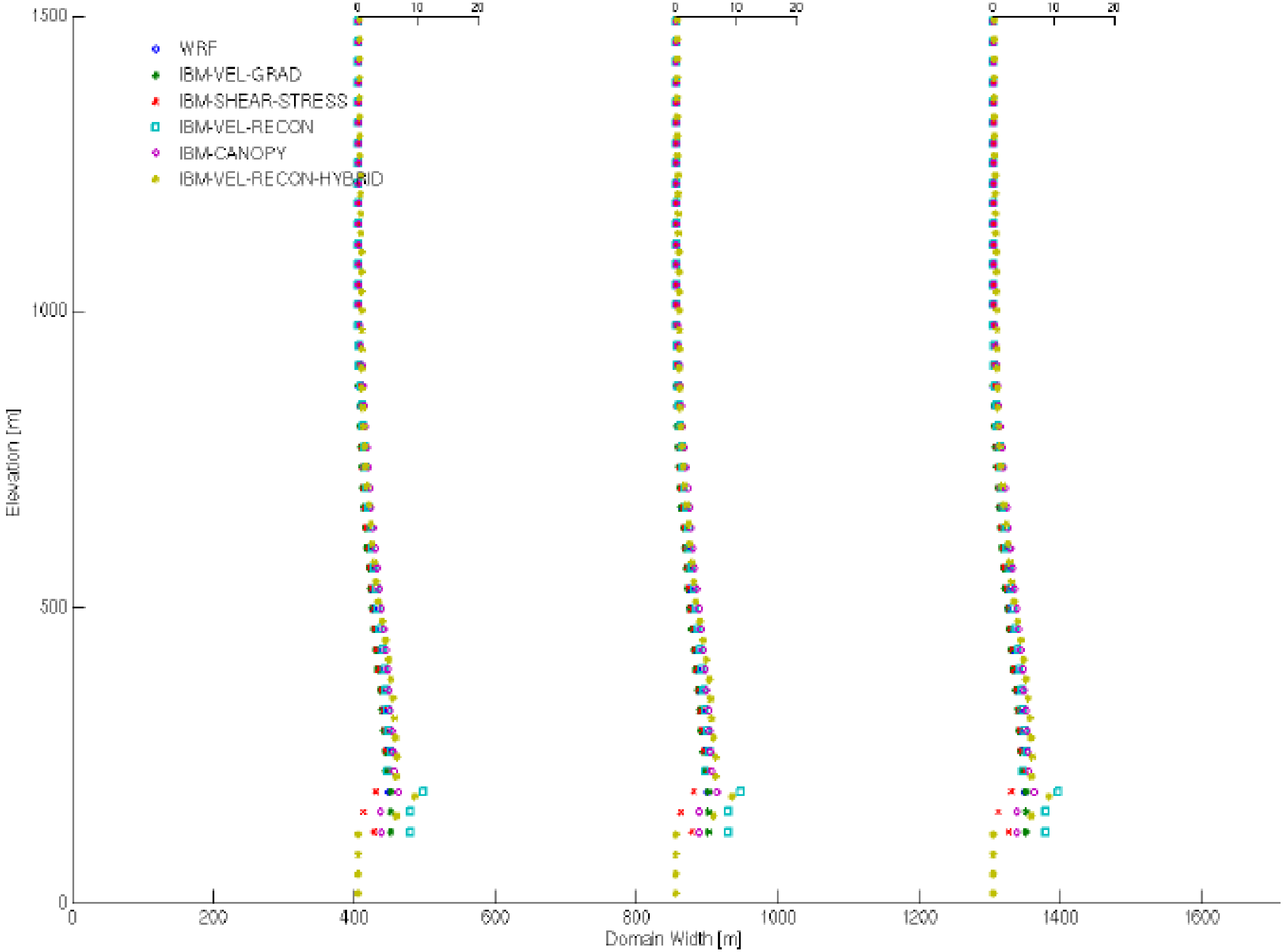
  - Form drag for immersed obstacle



IBM flat terrain – U velocity



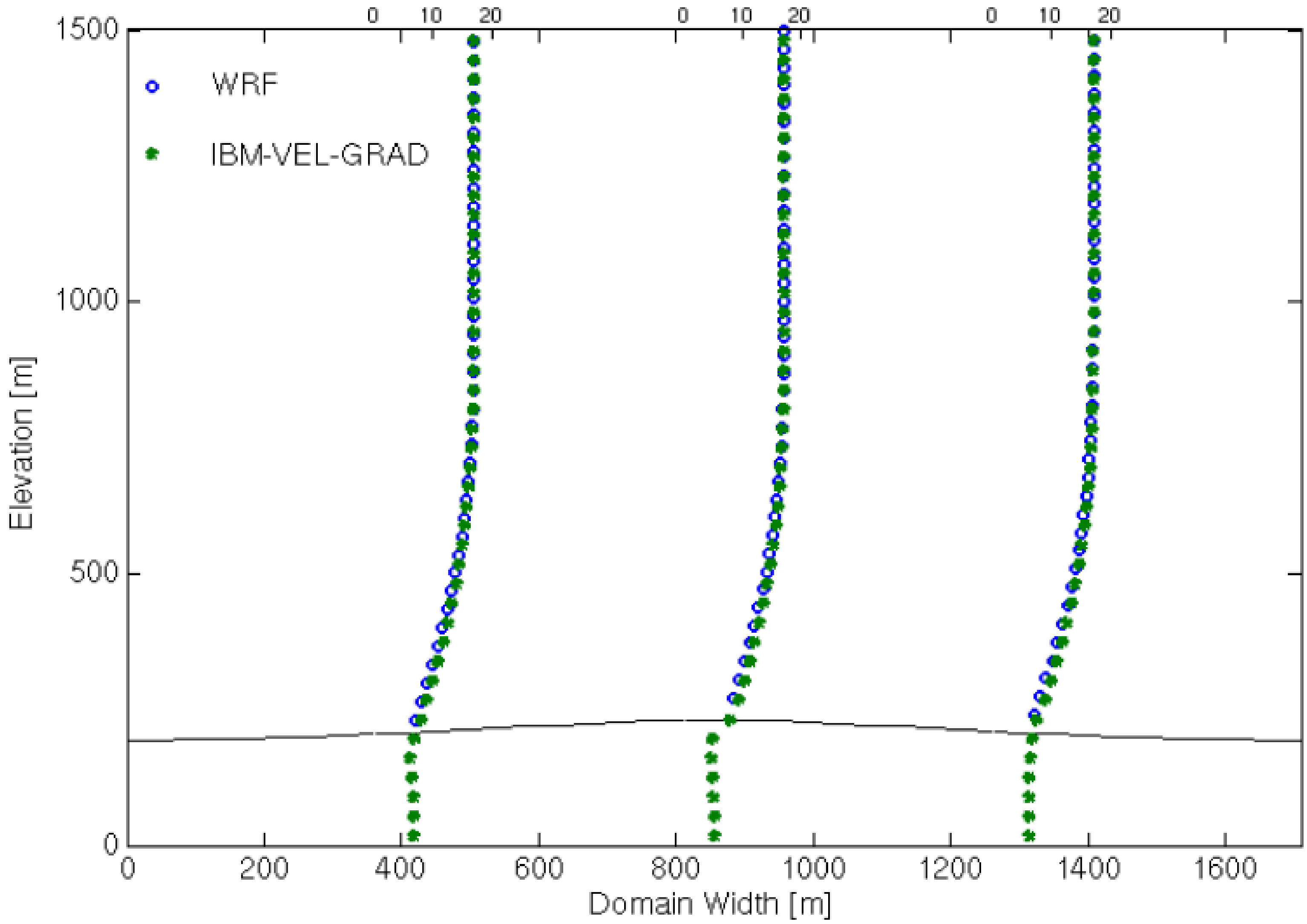
IBM flat terrain – eddy viscosity



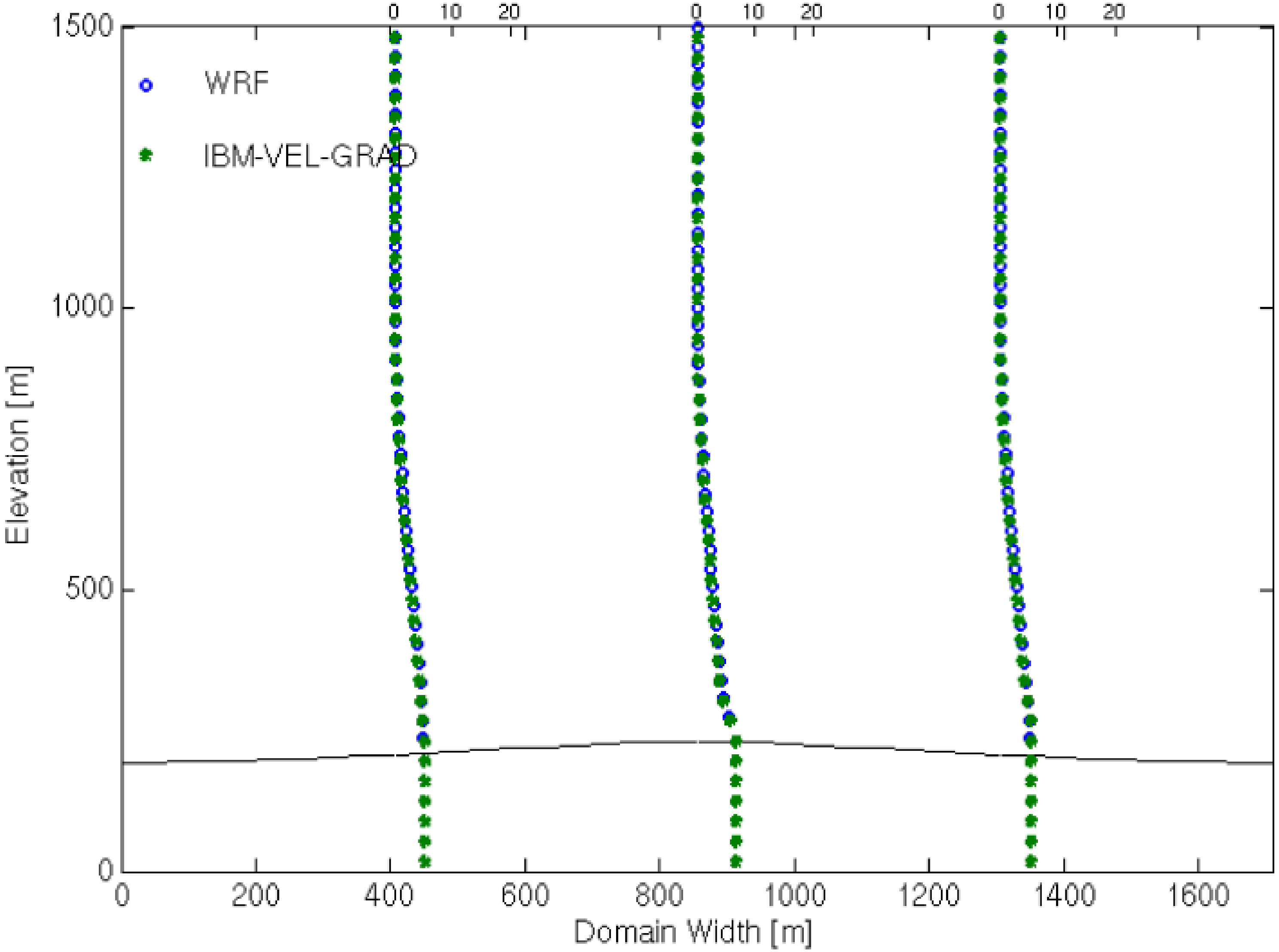
Moving to terrain

- Velocity reconstruction and canopy methods not good over flat terrain
- Focus on velocity gradient method and shear stress reconstruction

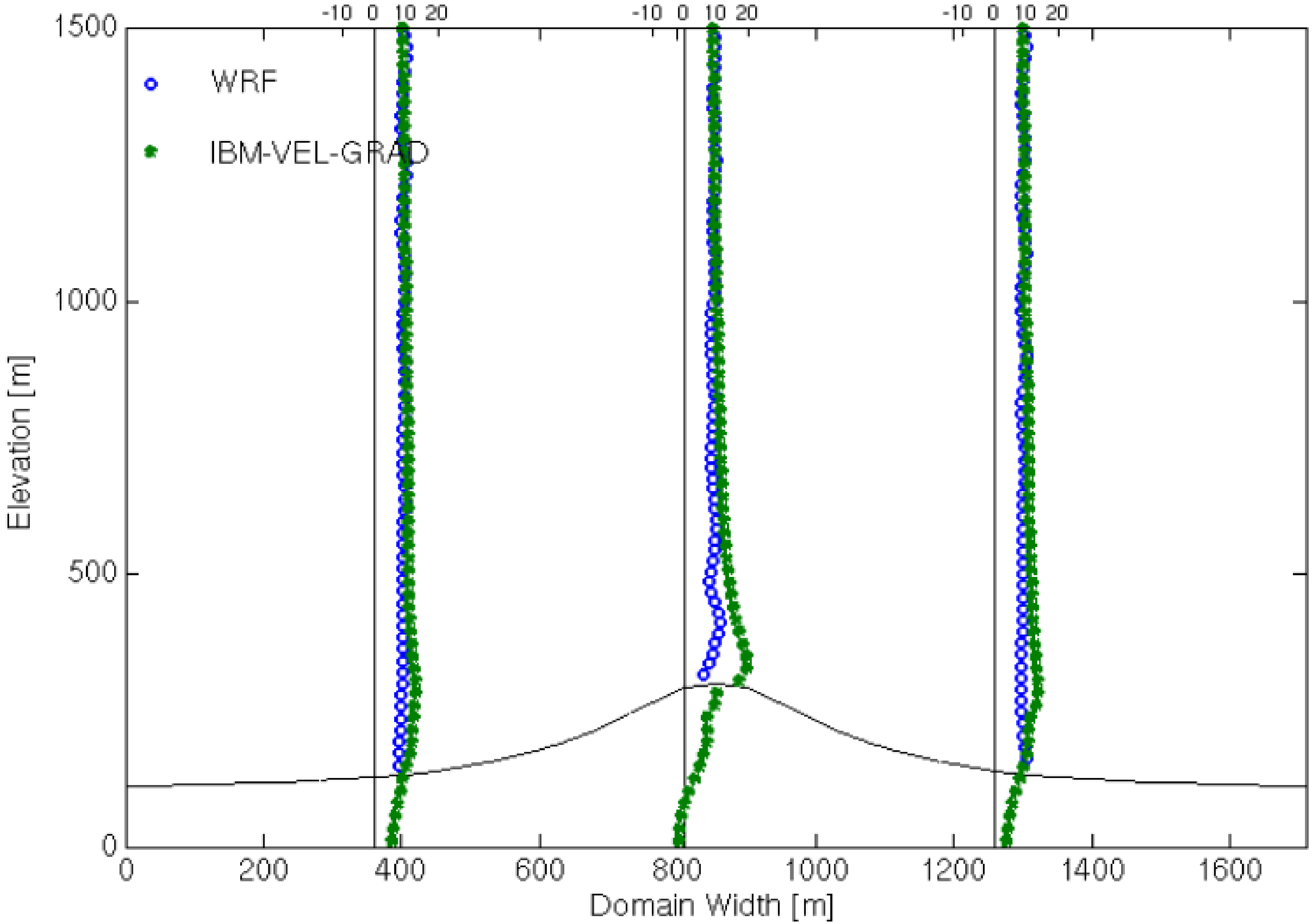
IBM 5° hill – U velocity



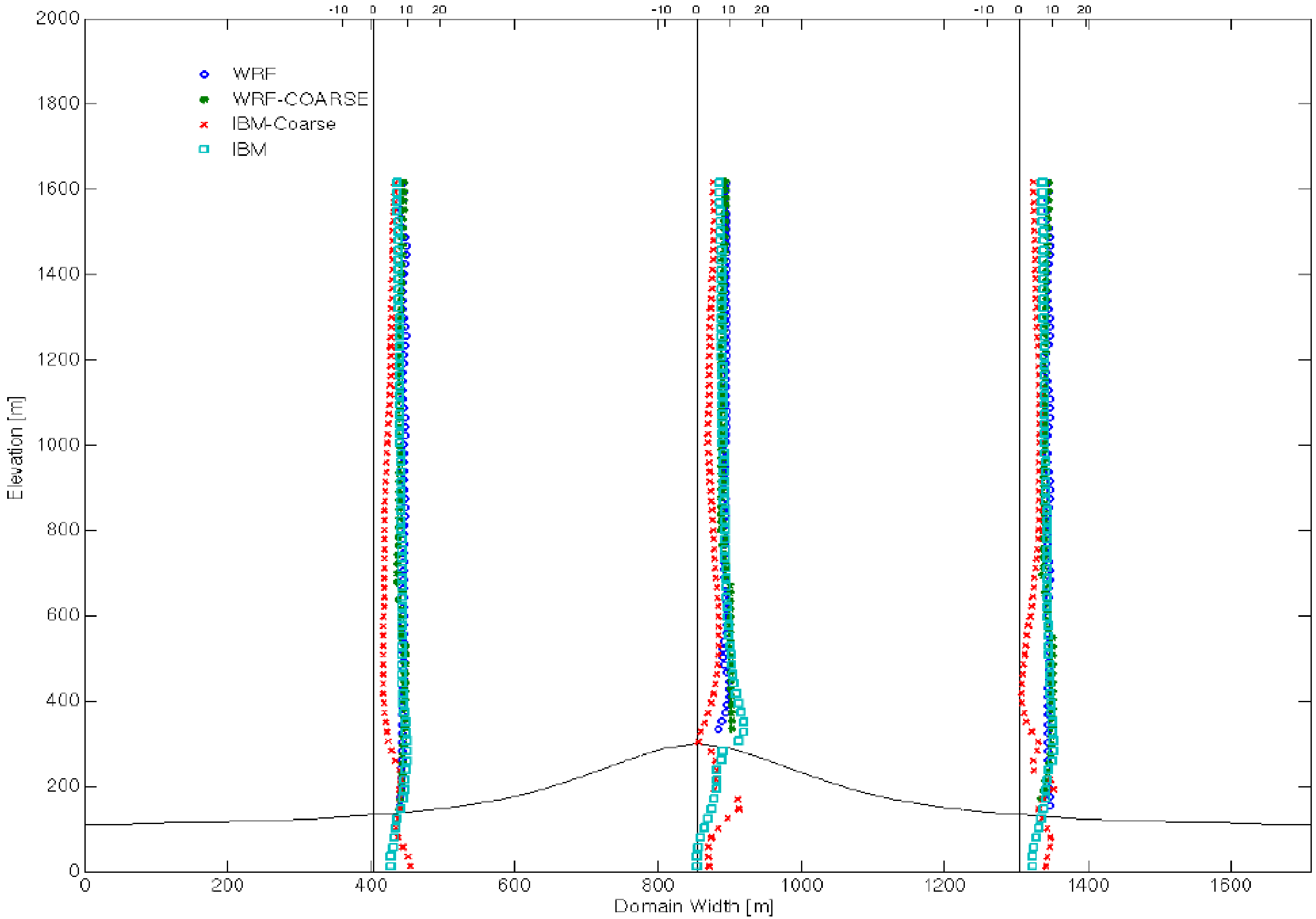
IBM 5° hill – eddy viscosity



IBM 20° hill – U velocity



IBM 20° hill – grid dependence

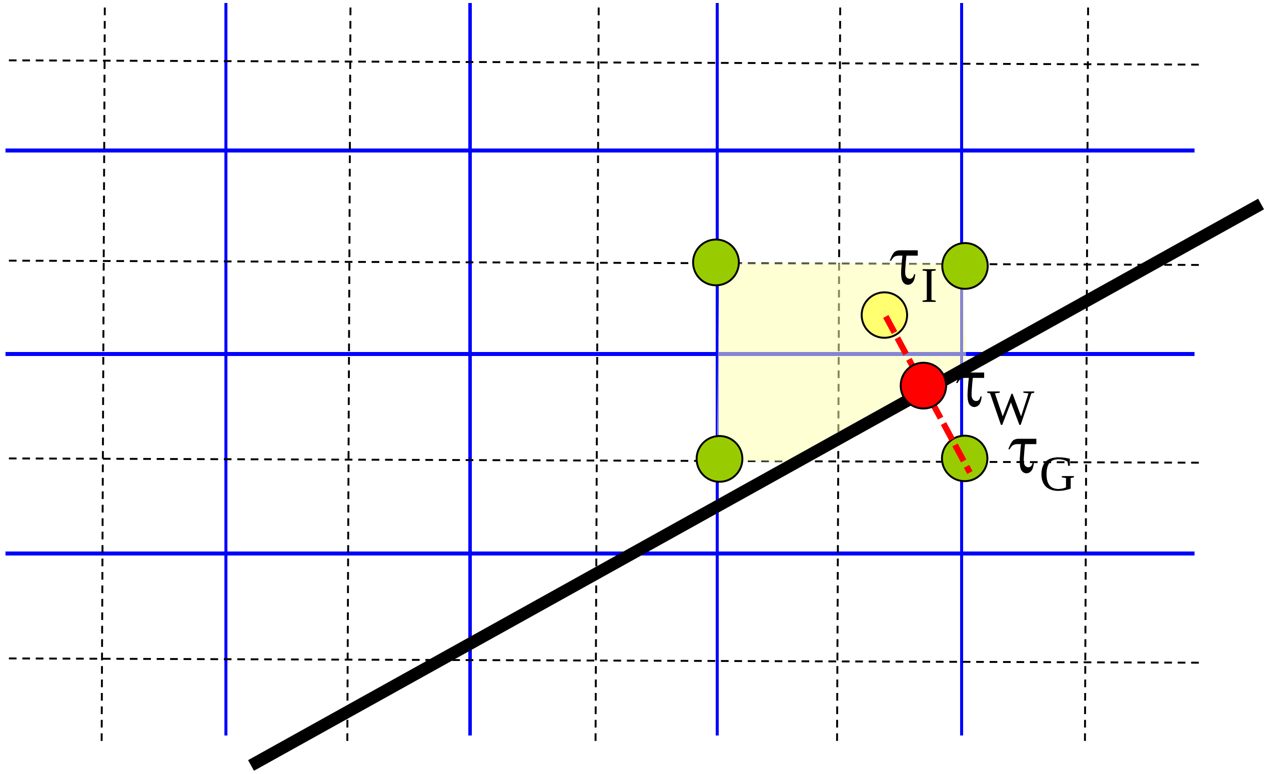


Challenges: velocity gradient method

- WRF terrain-following grid skewed
  - Introduces errors
  - Runs at finer resolution (45 m) blow up
  - Coarse case agrees better with IBM fine…
- IBM interpolation errors
  - Agreement is worst
  - Nearest neighbors are too far away

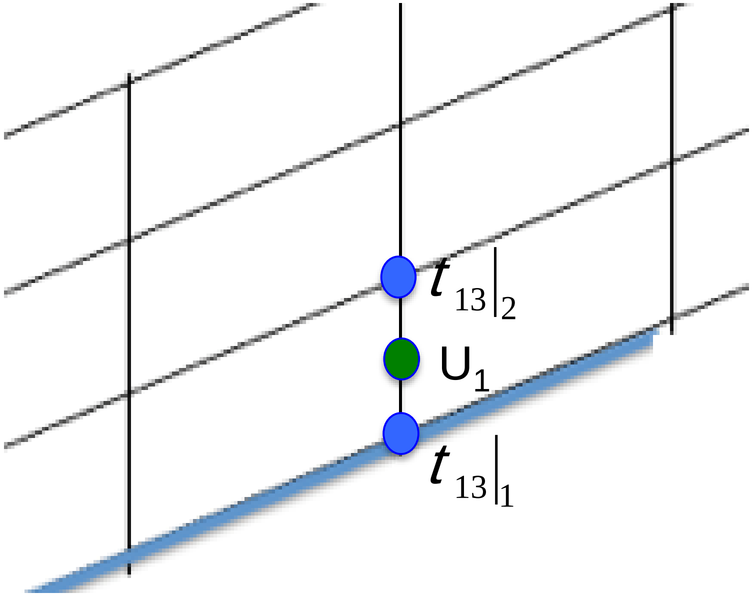
- Move to shear stress reconstruction

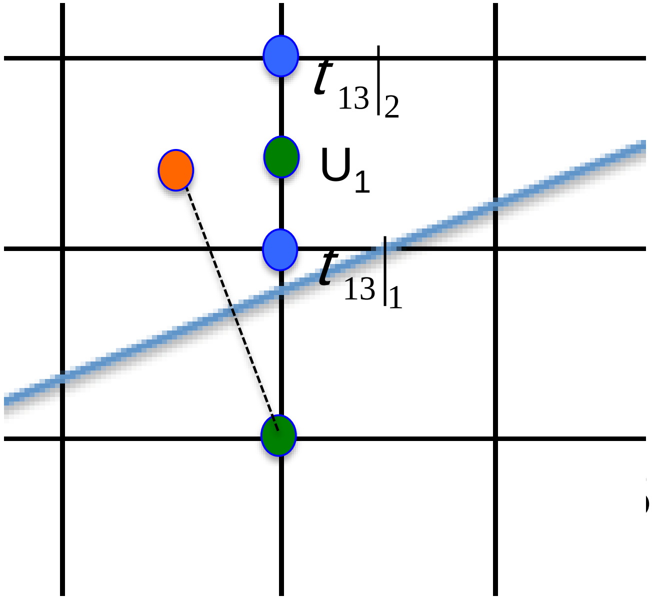
3D shear stress reconstruction



Implementation challenges

- WRF uses vertical gradients, not normal derivatives
- Error in WRF eddy viscosity

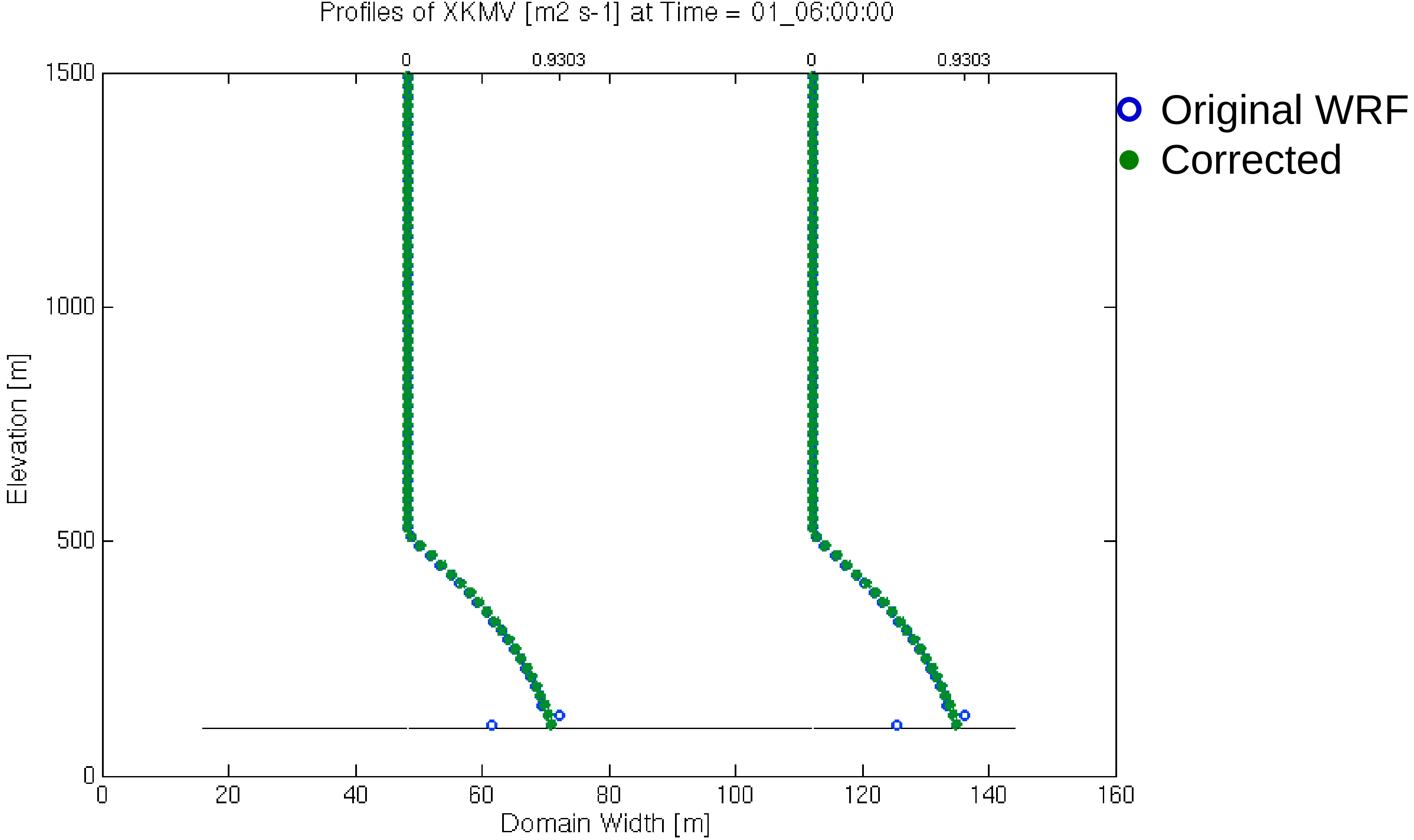




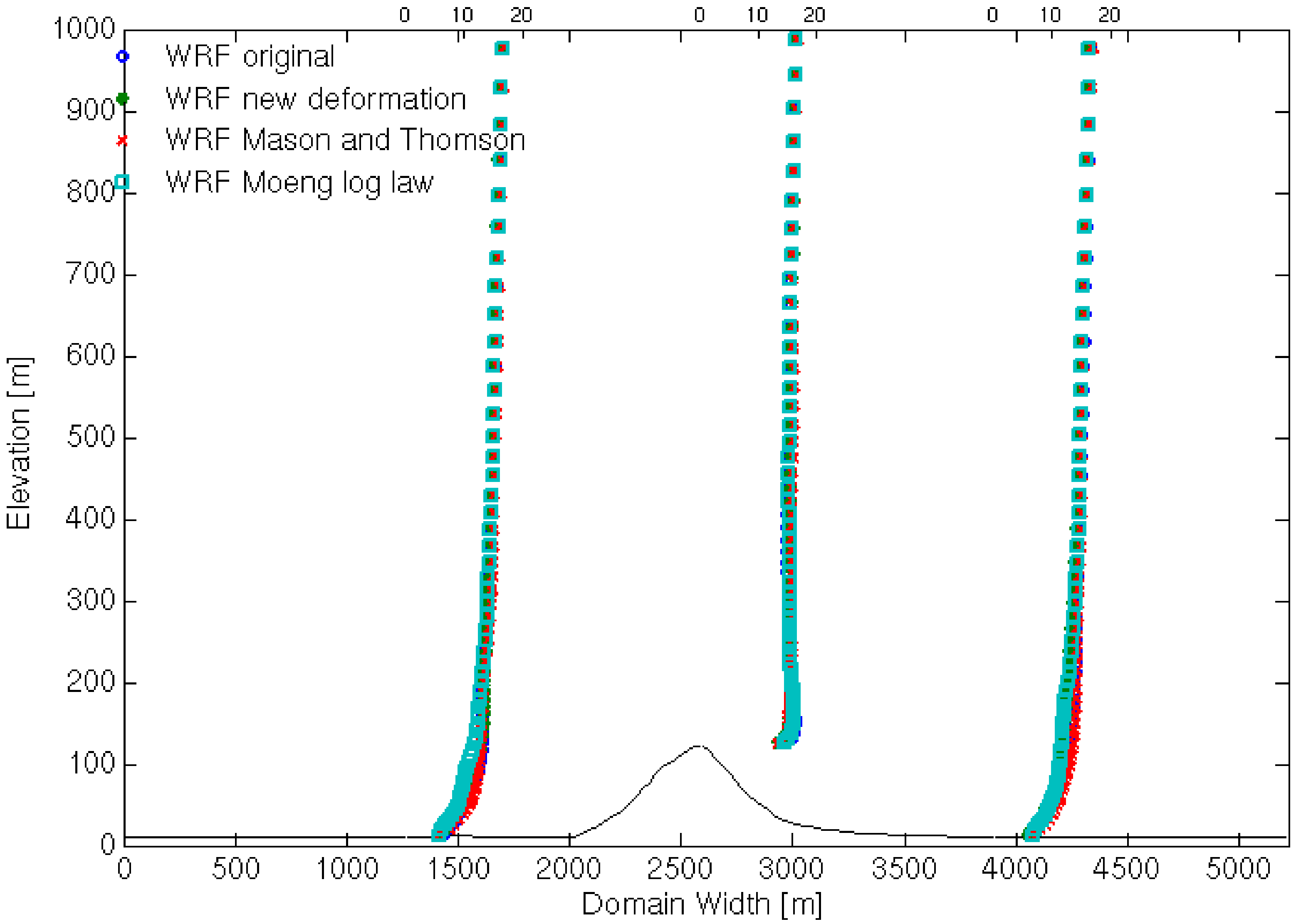
M

WRF eddy viscosity bug fix

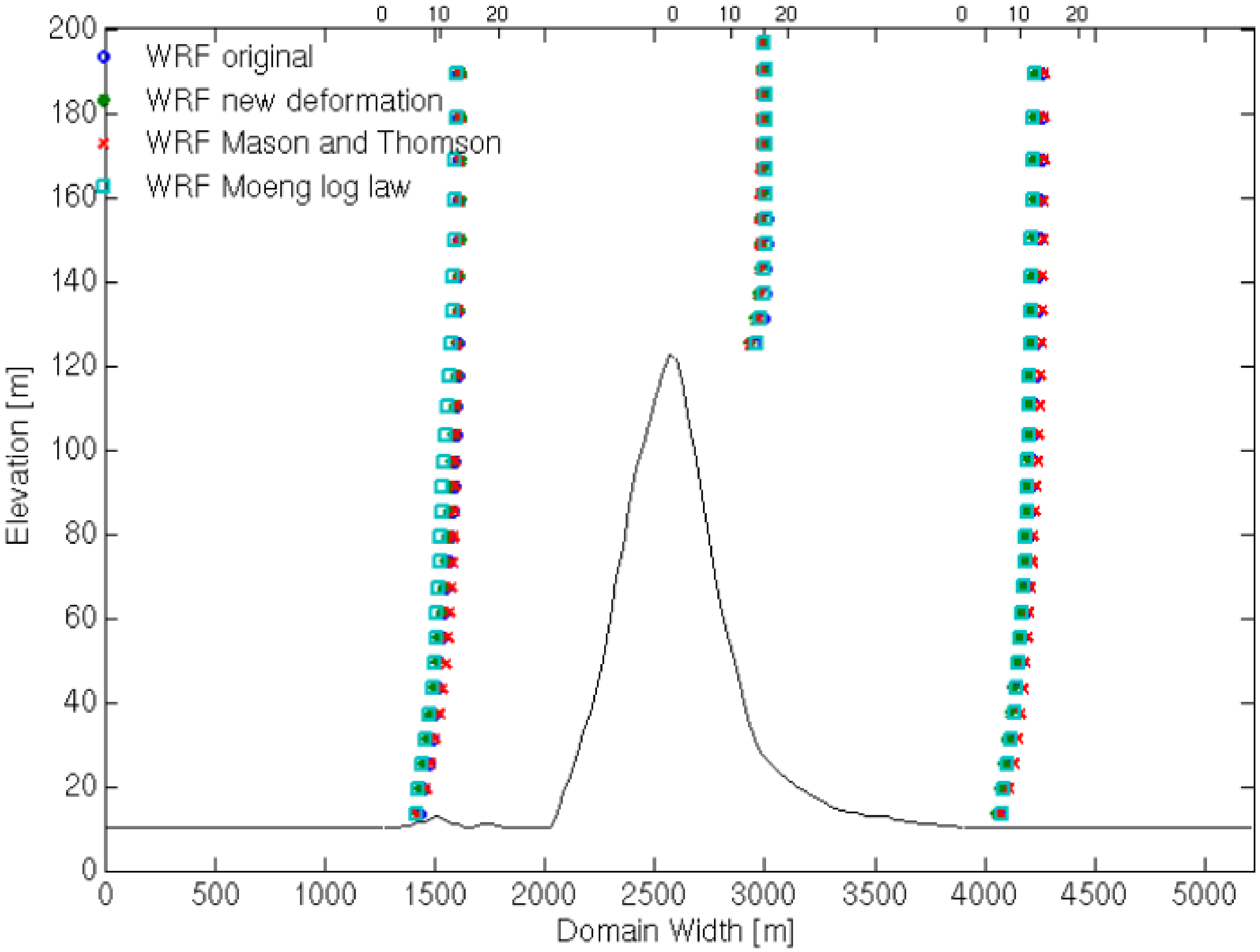
- WRF using zero deformation at surface
- Katie Lundquist



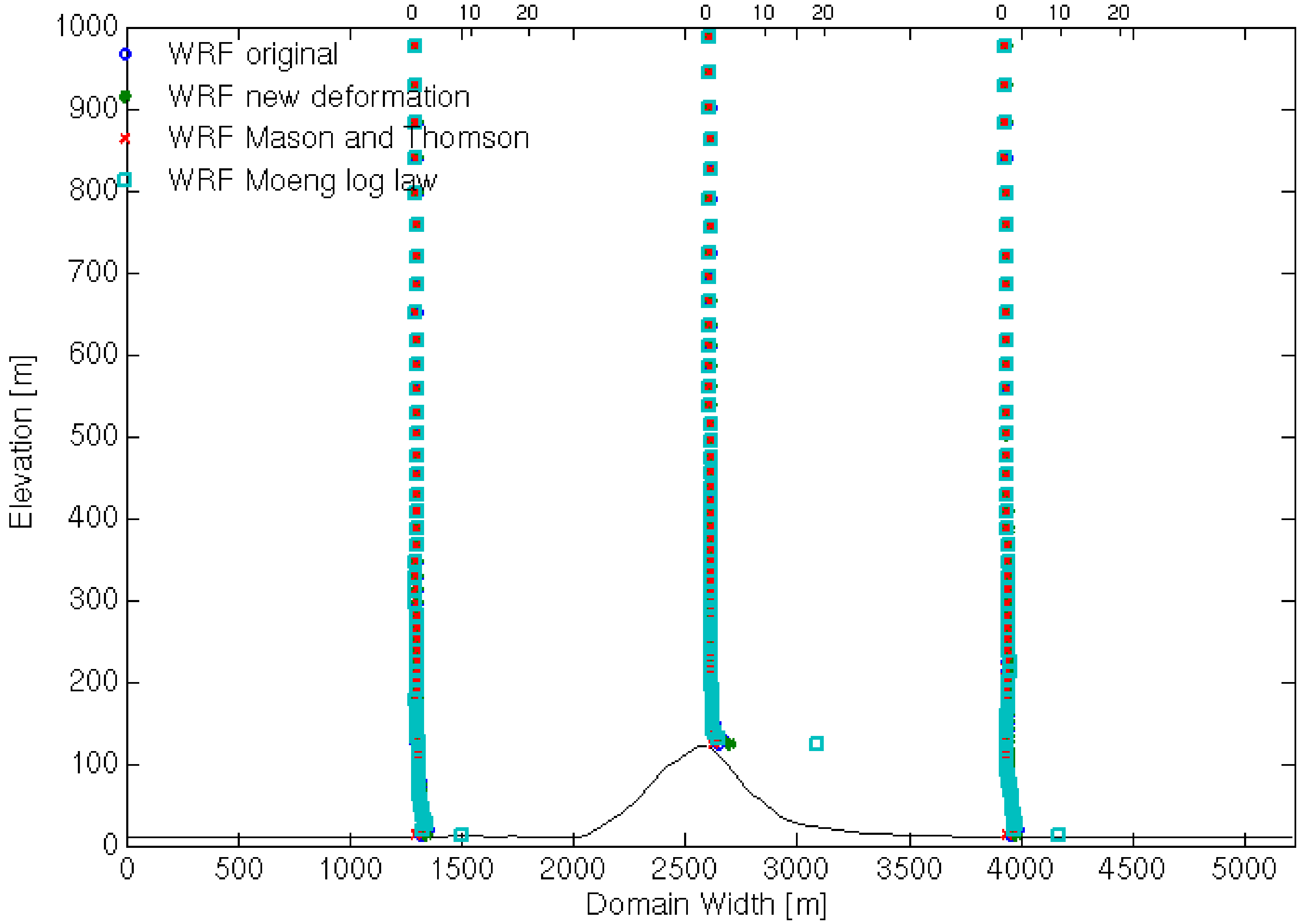
WRF Askervein simulations – U vel.



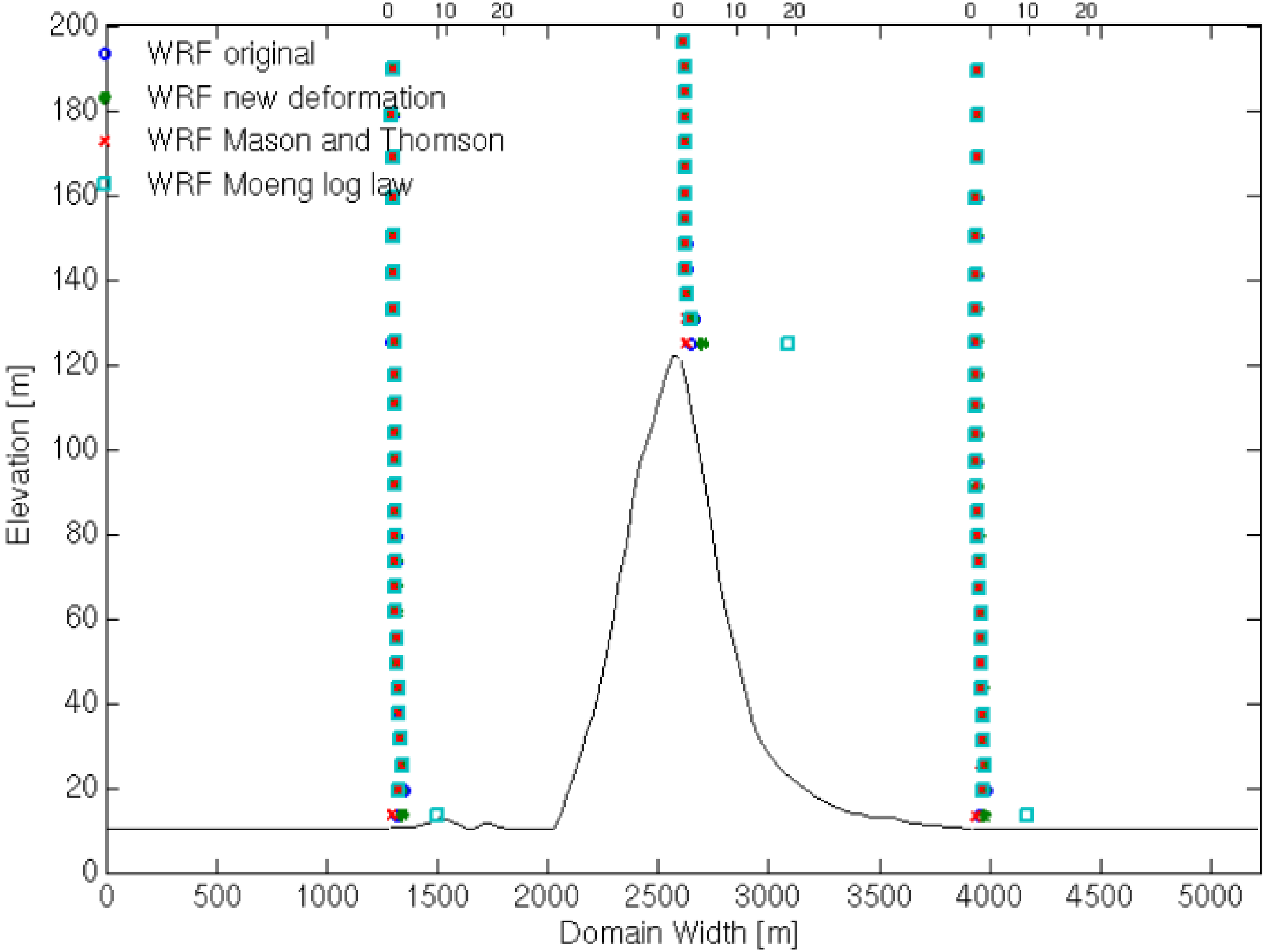
WRF Askervein simulations – U vel.



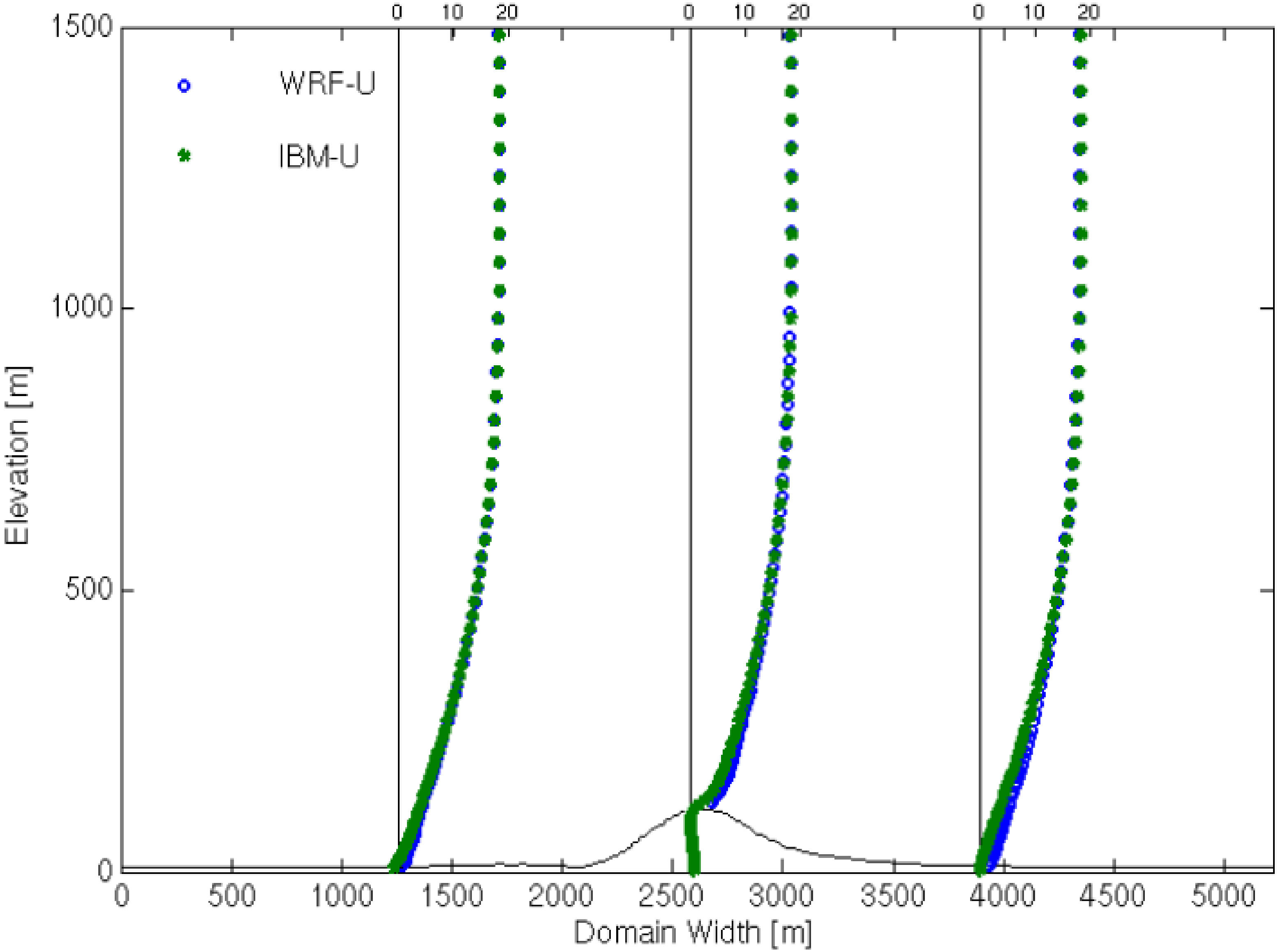
WRF Askervein – eddy viscosity



WRF Askervein – eddy viscosity

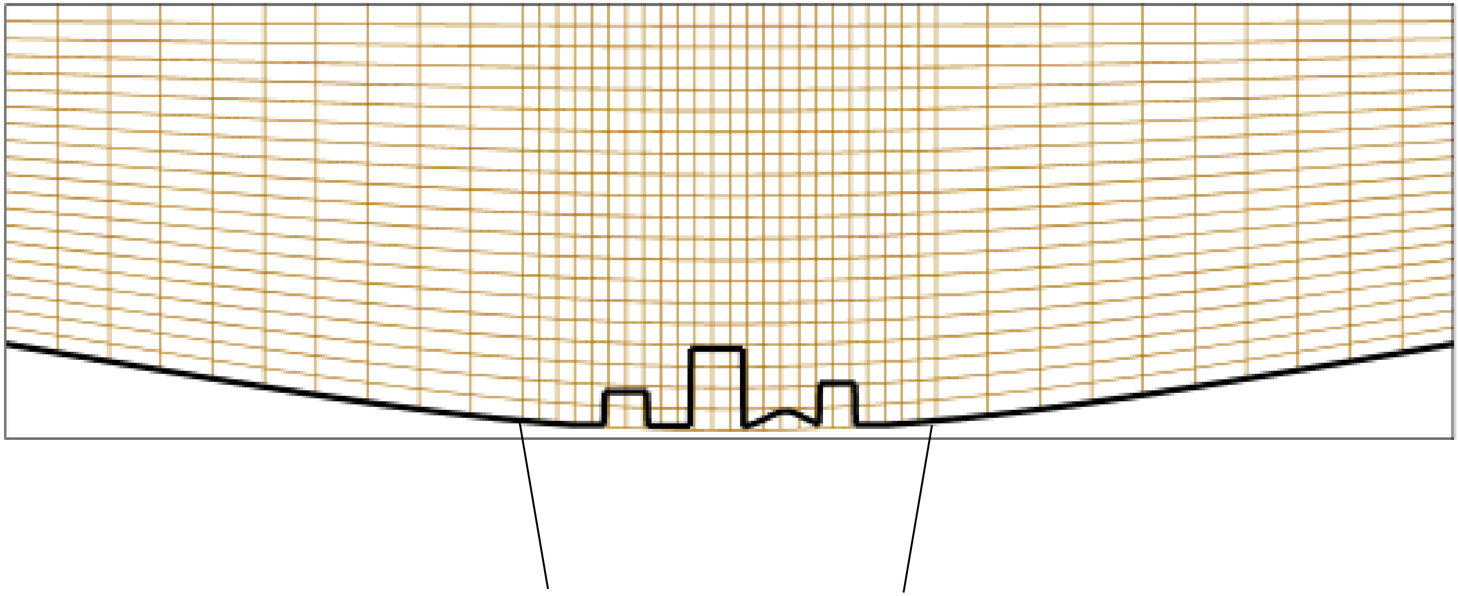


No-slip Askervein simulation



WRF to WRF-IBM interface

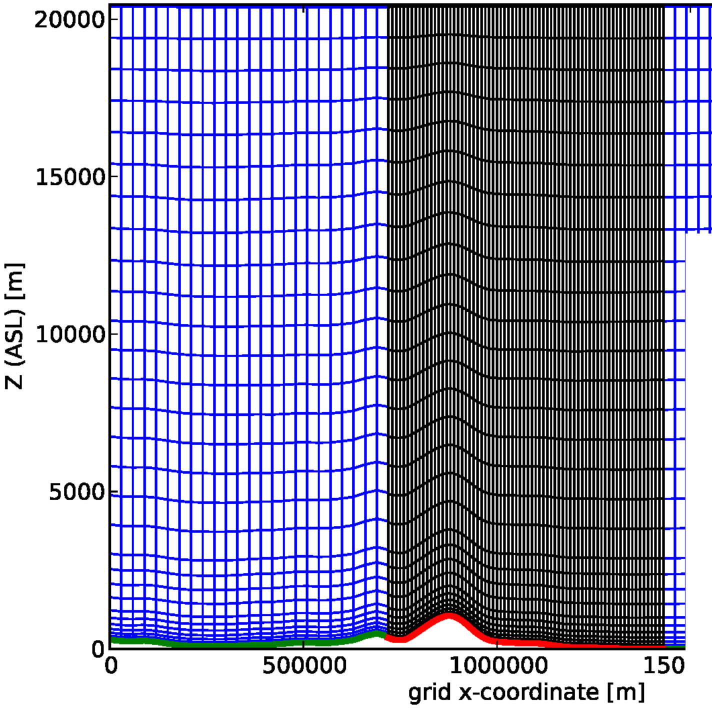
- Develop interpolation framework
- Grid nesting from WRF to IBM grids
  - And from IBM to IBM grids



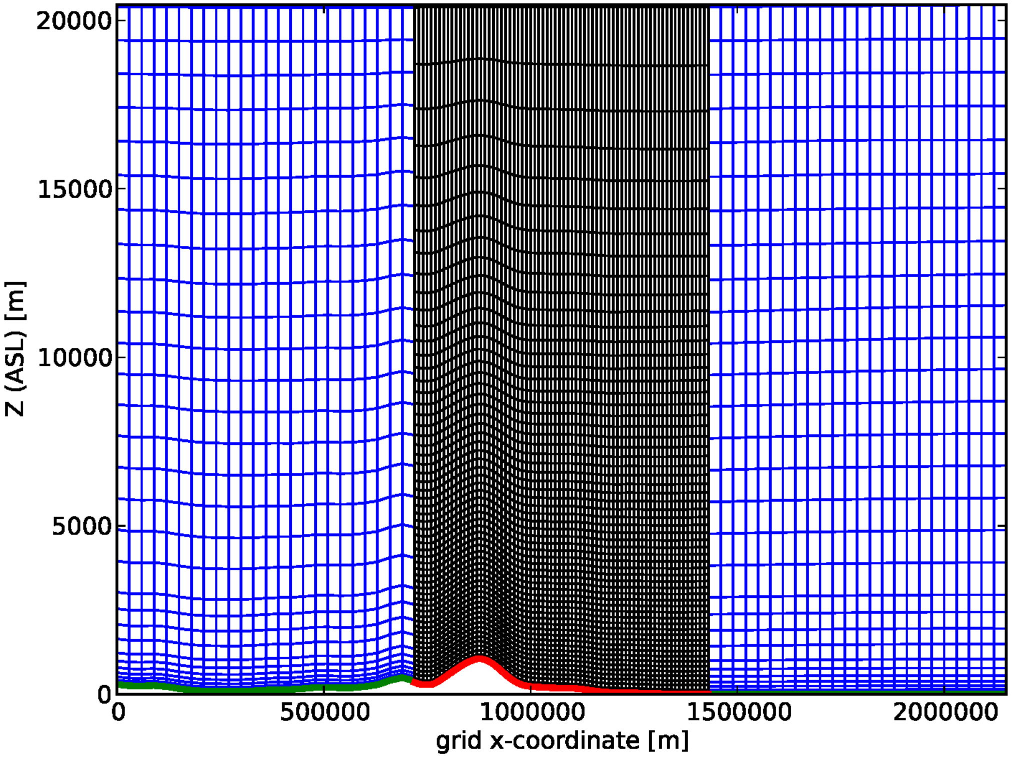
Nested IBM grid

WRF to WRF-IBM nesting

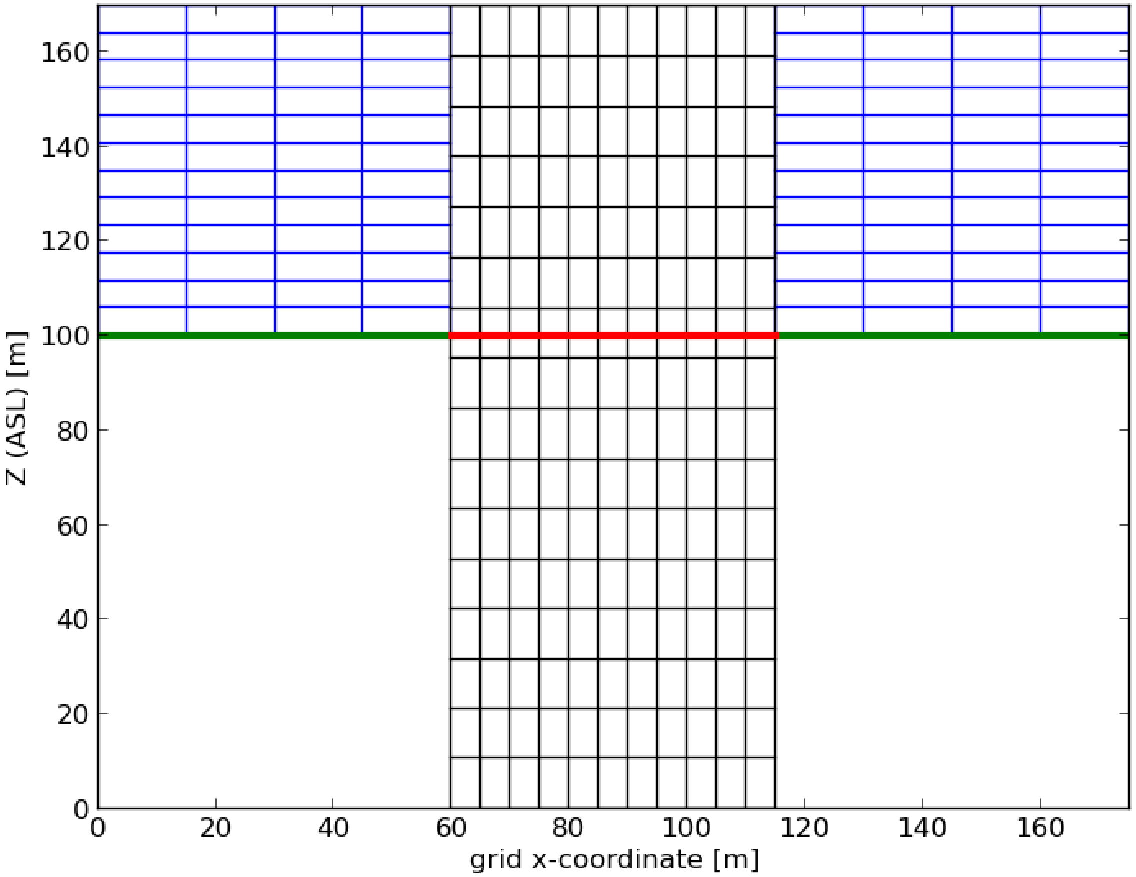
- WRF: same vertical levels
- IBM: interpolation needed
- Vertical nesting



Nested grid



Nesting WRF to WRF-IBM



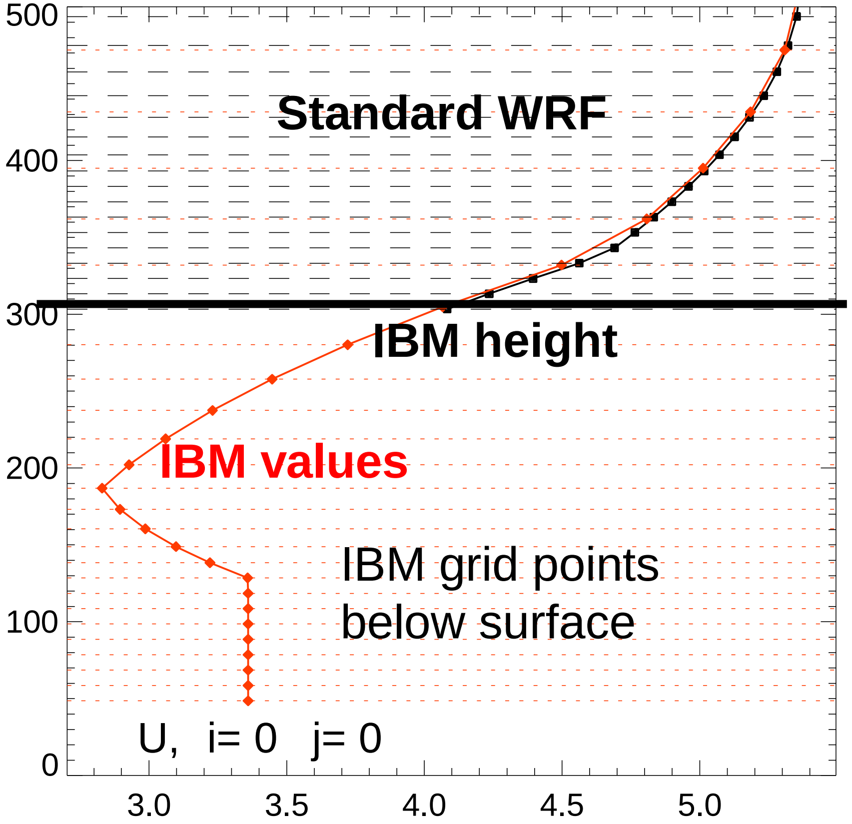
Current and future work

- Finish shear stress reconstruction implementation
  - Jingyi Bao
- IBM simulations for Granite Mountain
  - Bobby Arthur
  - ~10 m resolution, HPC
- Nested WRF simulations for Fall IOP 6
  - Alex Anderson-Connolly
  - This afternoon!
  - 100 m resolution
  - Work in progress

Extra slides

Initialization with meteorological data

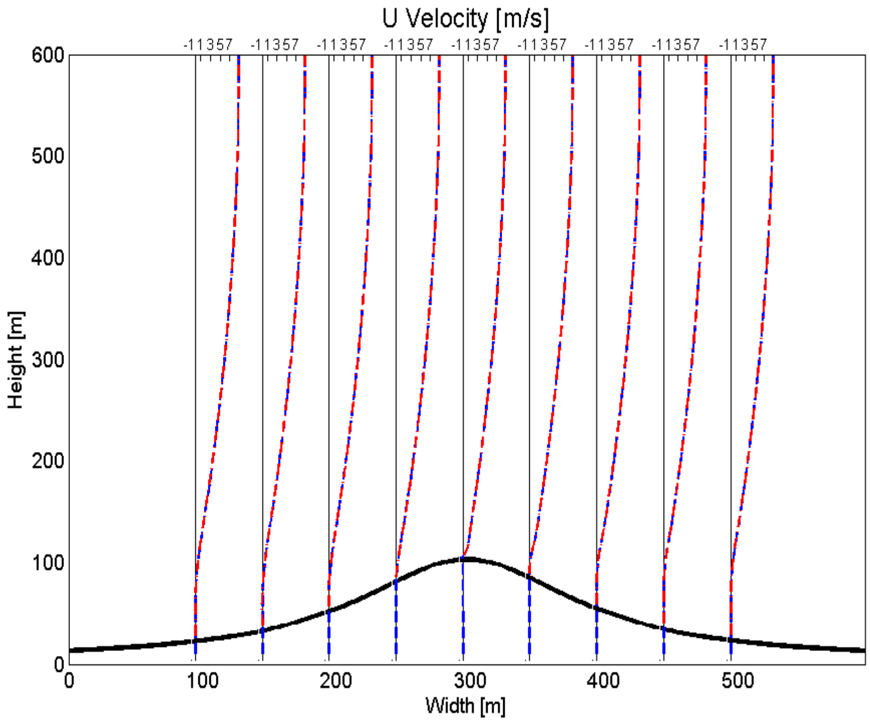
- IBM domain extends below the lowest terrain height
- Interpolate met. data onto IBM grid for initialization and boundary forcing



Idealized hill

- Goal: match WRF and WRF-IBM results
- Notes about log law:
  - WRF implements d/dz instead of d/dn
  - WRF results depend on choice of dz

No slip conditions



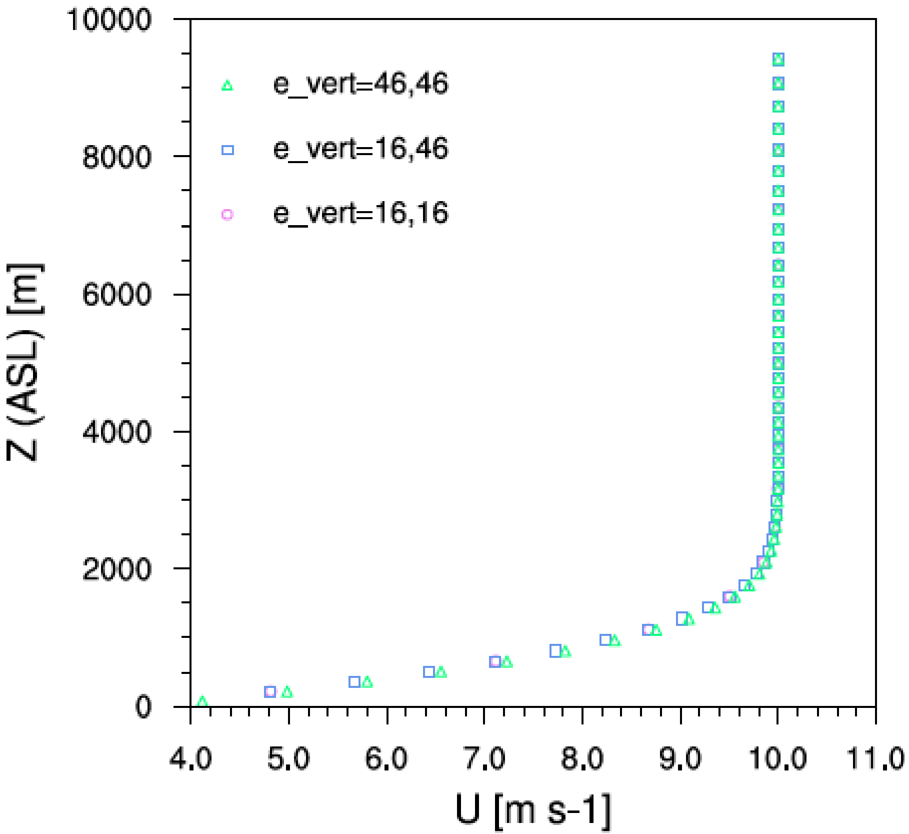
Red – terrain following coordinates (WRF)

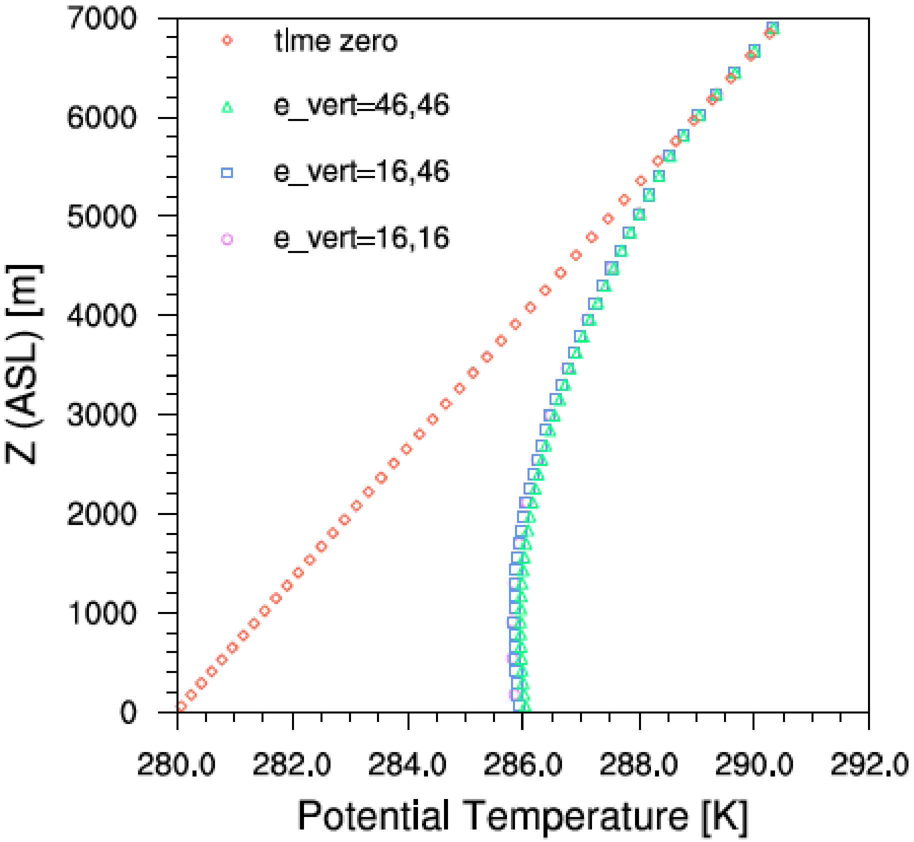
Blue – Immersed Boundary Method (IBM-WRF)

Lundquist et al. 2010, 2012

Vertical nesting in WRF

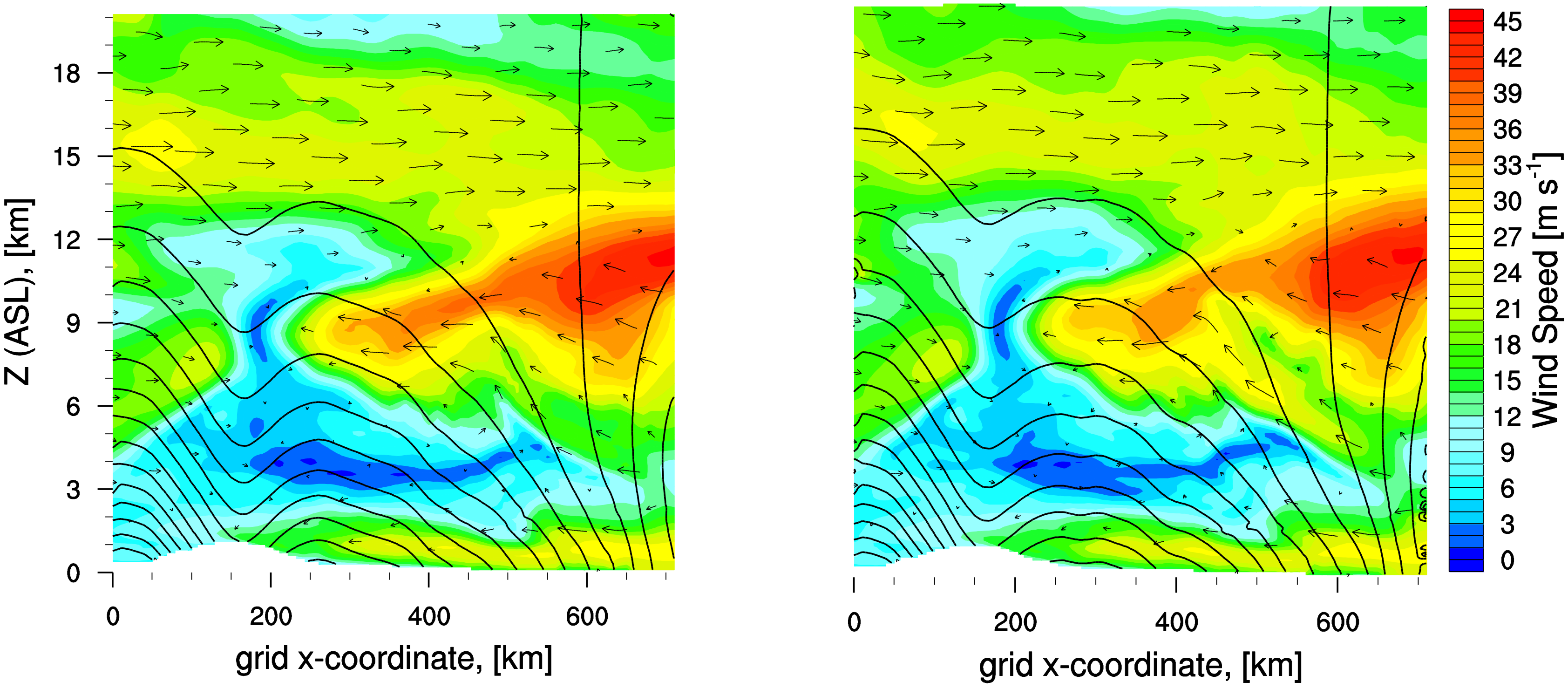
Idealized tests – flow over flat plate, heated flat plate





Vertical nesting in WRF

Real test cases – Jan 2000 snowstorm

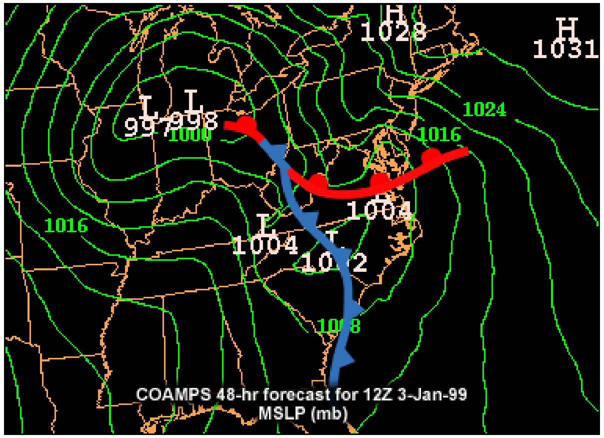


Same vertical levels, 30:30

With vertical nesting, 30:60

Initialization with meteorological data

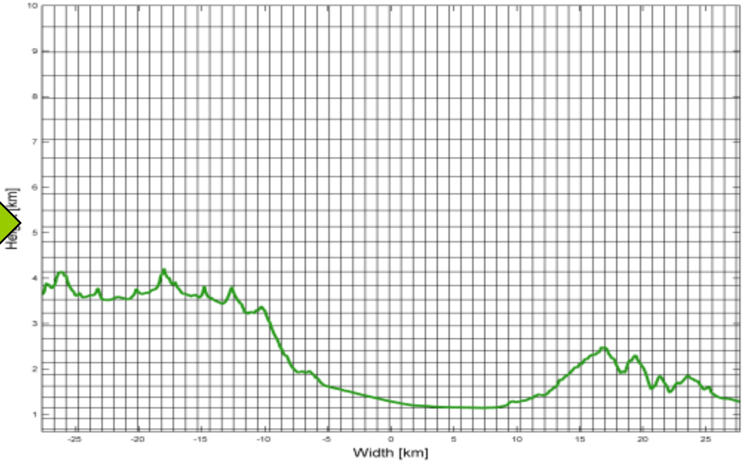
- Run WRF-IBM with direct forcing from met. data



COAMPS 48-hr forecast for 12Z 3-Jan-99
MSLP (mb)

Marine Meteorology Division, Naval Research Laboratory

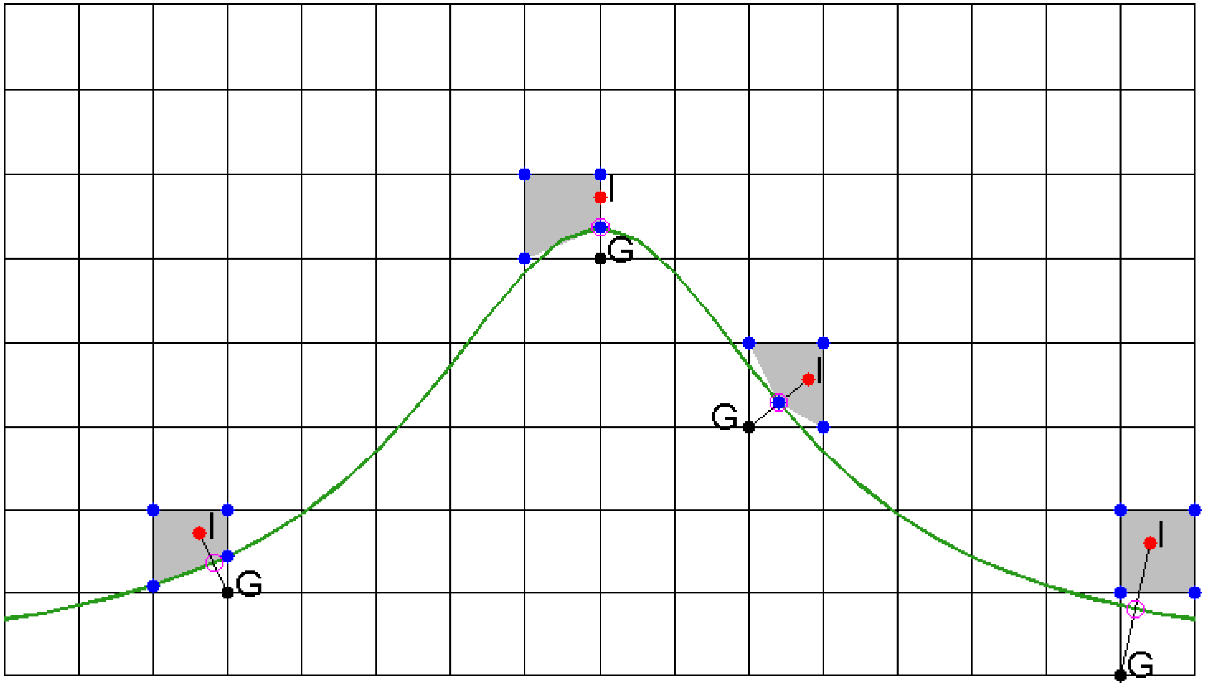
NAM



WRF-IBM

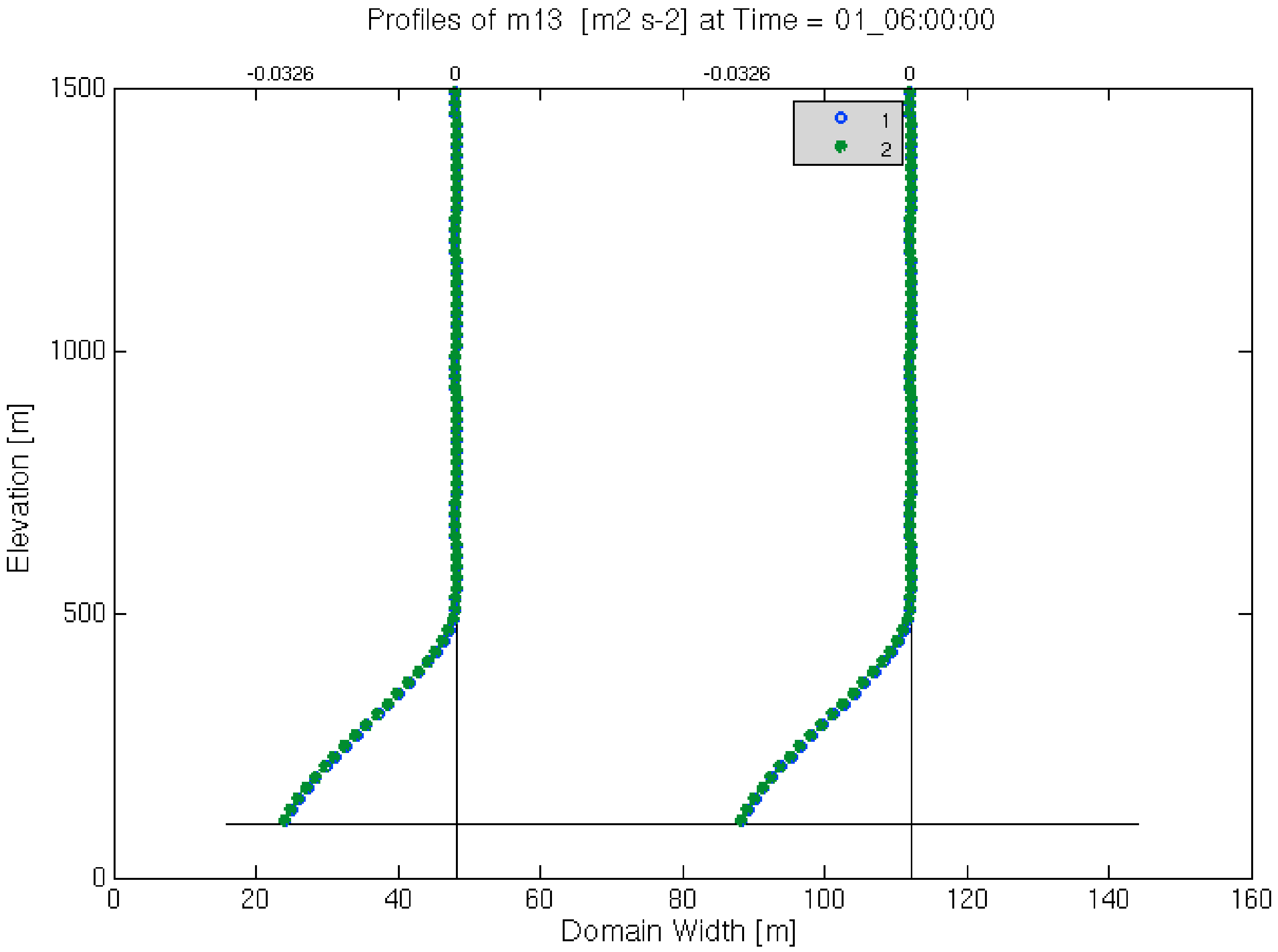
IBM - Boundary reconstruction

- IBM implemented in WRF
- 2 different interpolation algorithms
- Handles highly complex topography

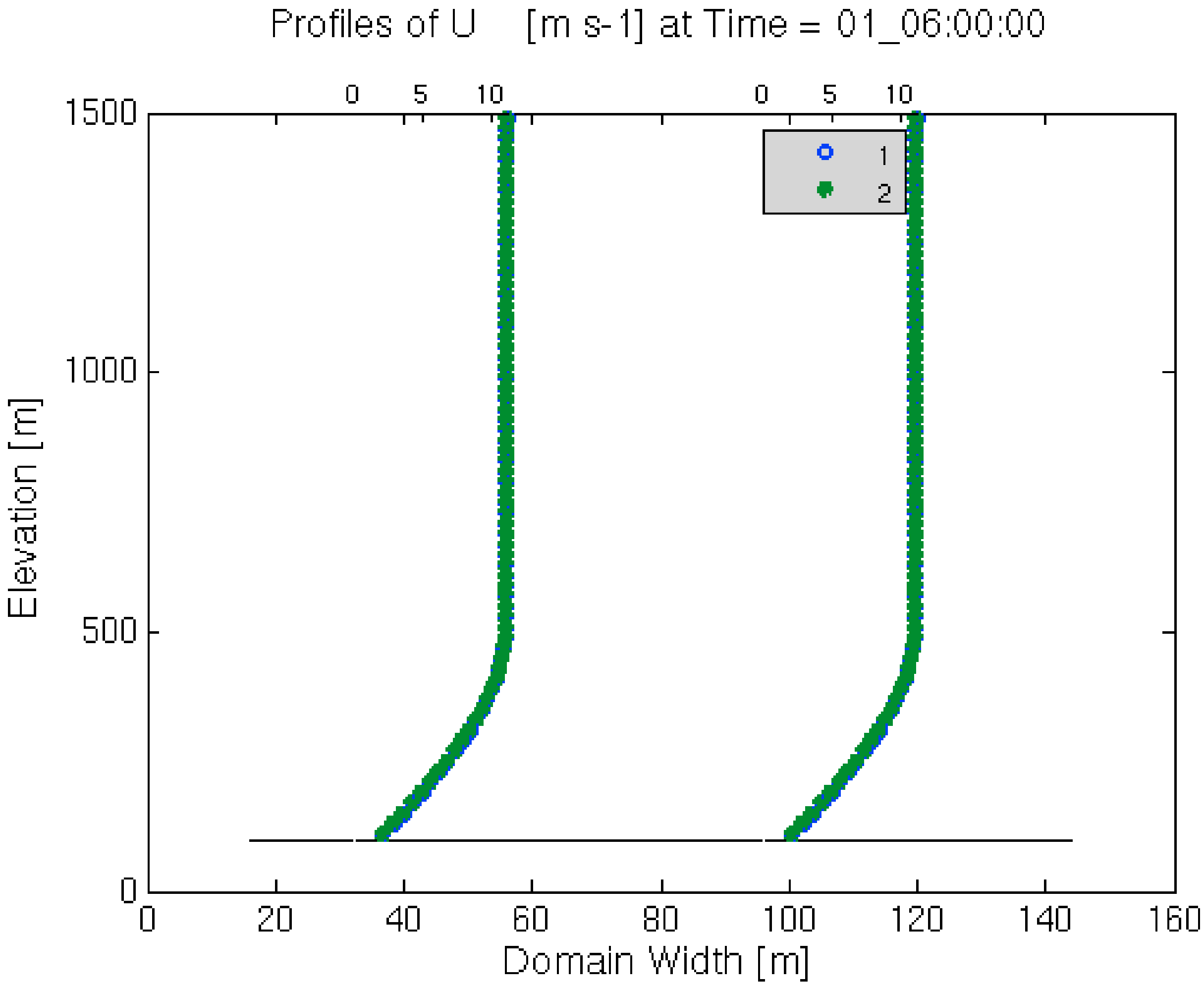


Lundquist et al. MWR 2010, 2012

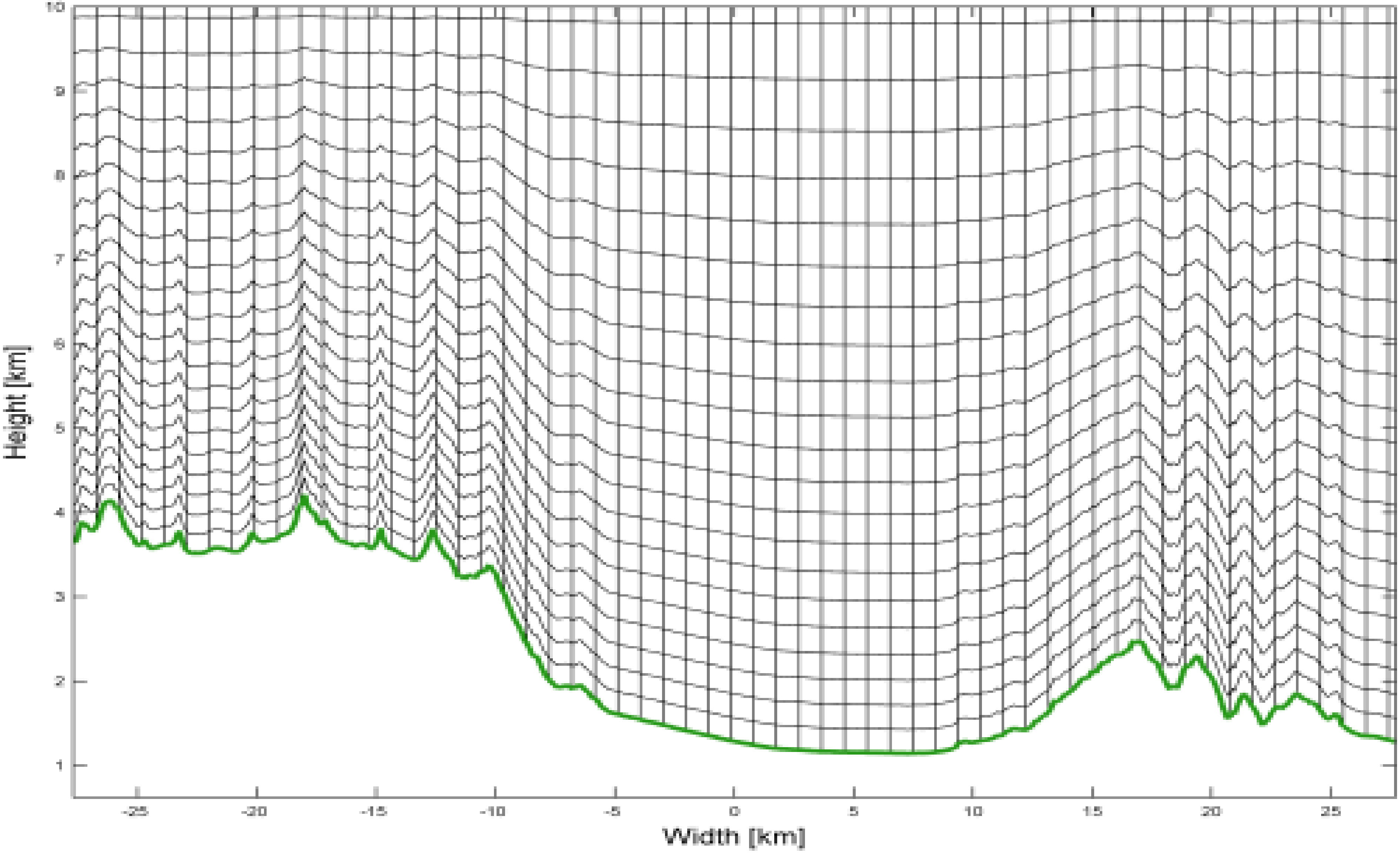
Katie’s WRF eddy viscosity fix



Katie’s WRF eddy viscosity fix



Terrain-following coordinates



Immersed boundary method