High Performance Computing
CLUSTERTECH

EGU2020-6653
Session AS1.1
Development of Customized Variable-Resolution CPAS for Meteorological Simulation

Chi Chiu Cheung

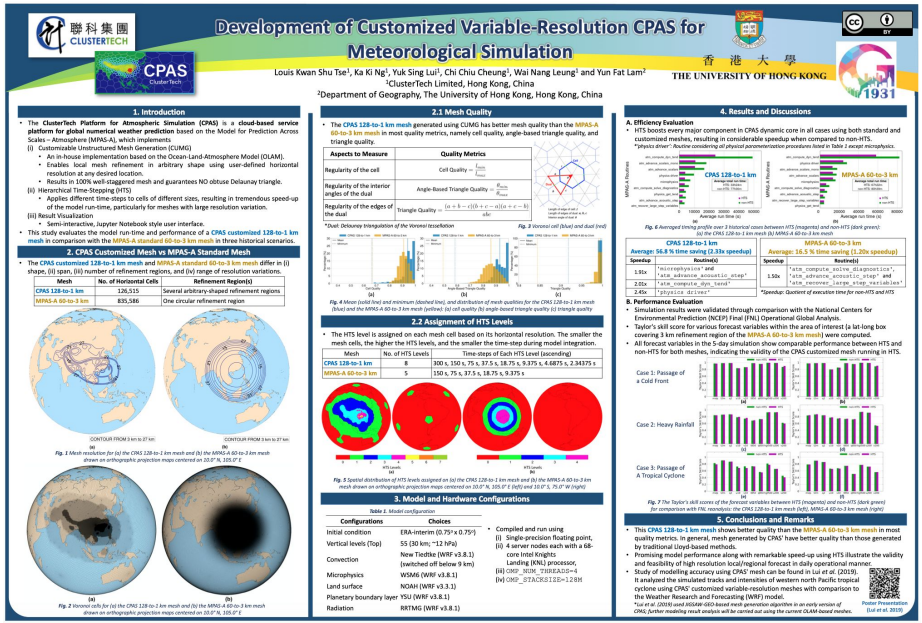
Leader of Computational Software Team, Partner and Senior Director of Technology, ClusterTech Ltd.

8 May 2020

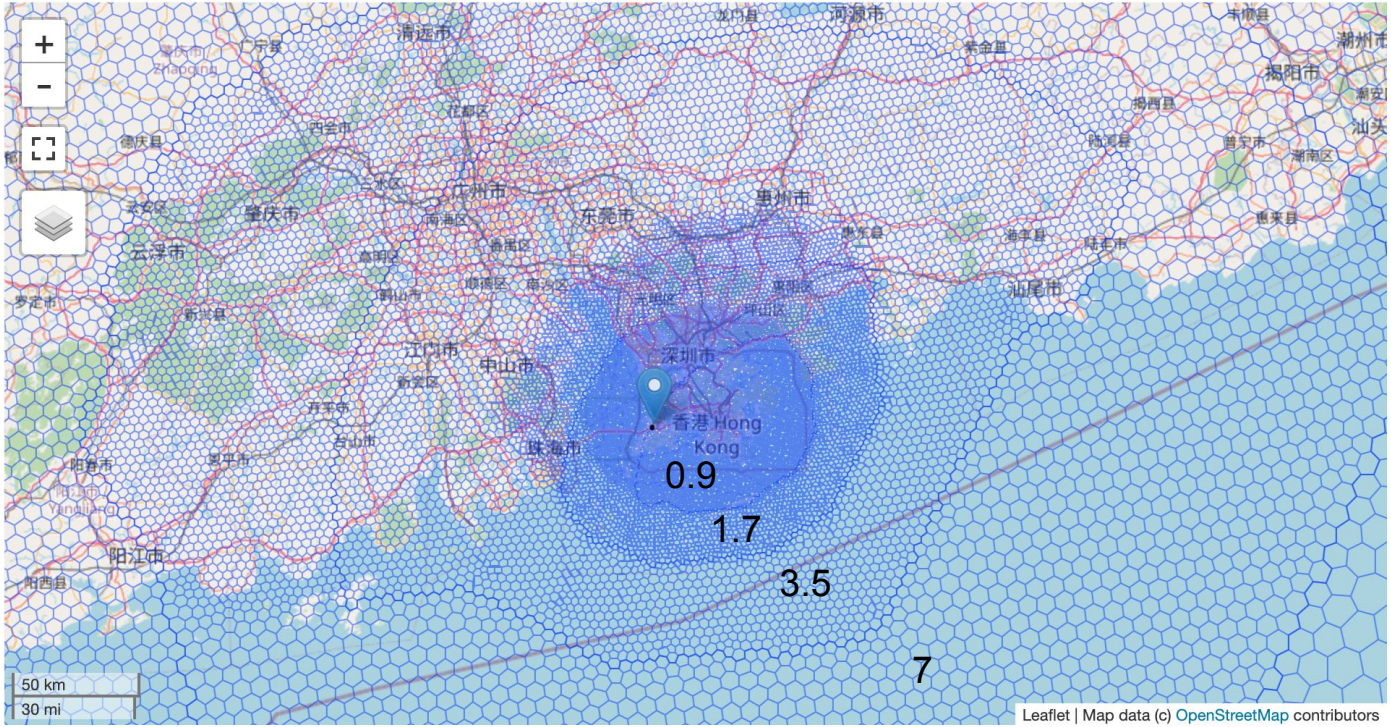
聯科集團
CLUSTERTECH

Poster

https://presentations.copernicus.org/EGU2020/EGU2020-6653_presentation.pdf



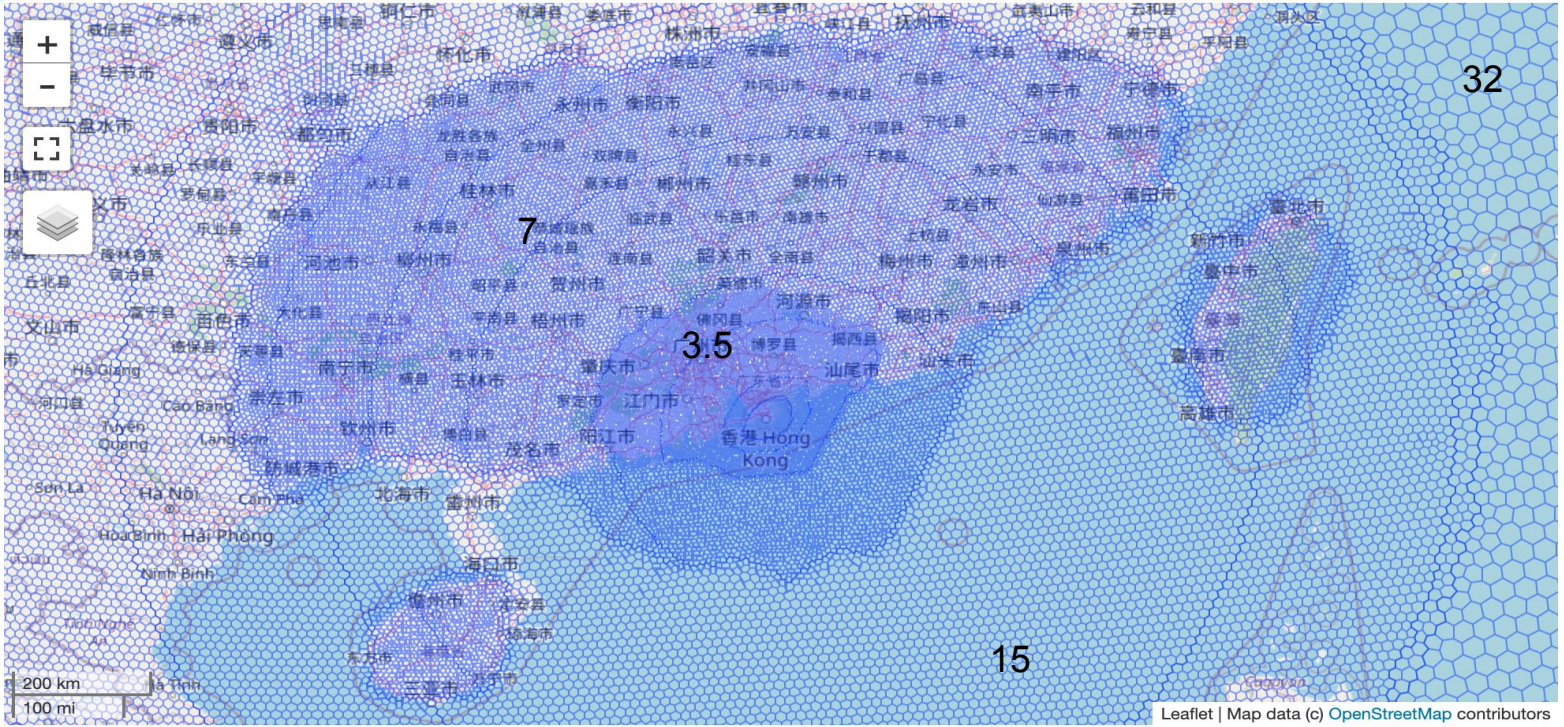
Customized mesh zoomed in near Hong Kong



Resolution finer than 60-3km mesh locally near Hong Kong

Numbers are resolution in km

Customized mesh near South China



Resolution coarser than 60-3km mesh except very near Hong Kong

Numbers are resolution in km

Demo of Result Visualization

Case 1: Passage of a Cold Front

Evaluate against observation data

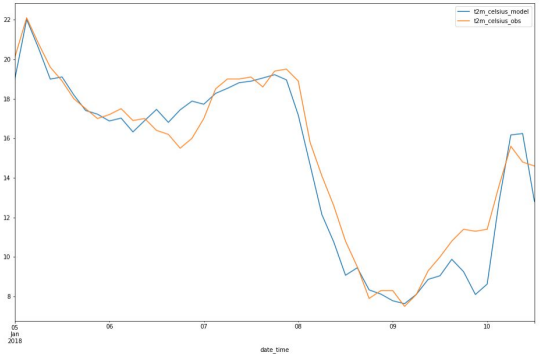
High Performance Computing
CLUSTERTECH

Hong Kong Airport weather station

5.1 Model evaluation against your observation data (sample code)

Free free to change the code for your observation data upload. See this blog article.

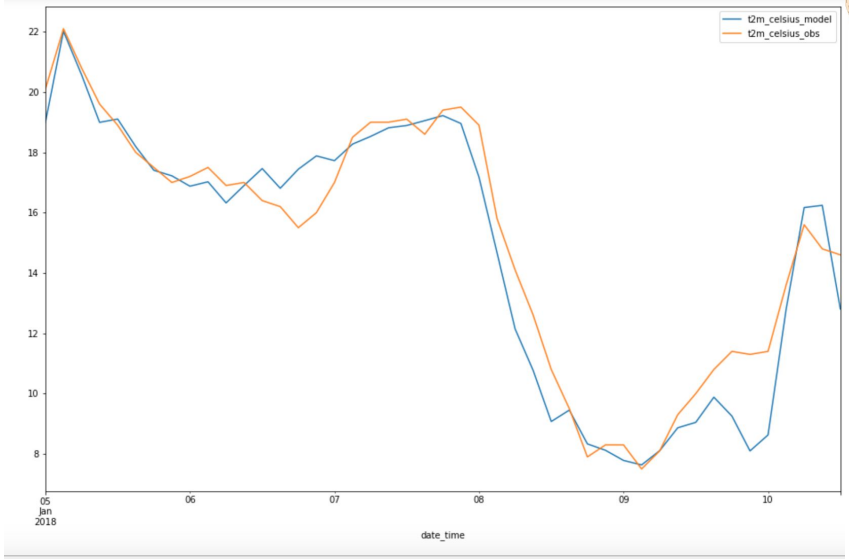
```
In [6]: import numpy as np
        import pandas as pd
        from datetime import timedelta
        from cpas.evaluation import ModelEvaluation, csv2df

        (my_lat, my_lon) = (22+18.534/60, 113+54.84/60)  # Naha airport
        local_time_header = 'Local time in Hong Kong (airport)'  # Column header in CSV file
        local_time_offset = 8
        my_csv = 'my_obs_data/45007.04.01.2018.13.01.2018.1.0.0.en.utf8.00000000.csv'

        raw_obs_df = csv2df(my_csv, sep=';', comment='#', index_col=False)
        obs_df = pd.DataFrame()
        obs_df['date_time'] = pd.to_datetime(raw_obs_df[local_time_header], format='%d.%m.%Y %H:%M
        (obs_df['lat'], obs_df['lon']) = (my_lat, my_lon)
        obs_df['t2m_celsius'] = raw_obs_df['T']

        me = ModelEvaluation(ui.mesh_ncfile, ui.diag_ncfile, obs_df)
        me.met_combine()
        comp = me.get_comparison()

        mask = ~np.isnan(comp['t2m_celsius_model'])
        filtered = comp[mask]
        filtered.xs(my_lat, level='lat').xs(my_lon, level='lon').plot(y=['t2m_celsius_model', 't2m

        Interpolating t2m......
        Interpolating q2......
        Interpolating ui0......
        Interpolating vi0......
        Interpolating msip.....
Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3cb0a7ed90>
```

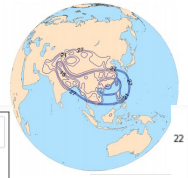


Customized 128km-1km mesh, 126,515 cells

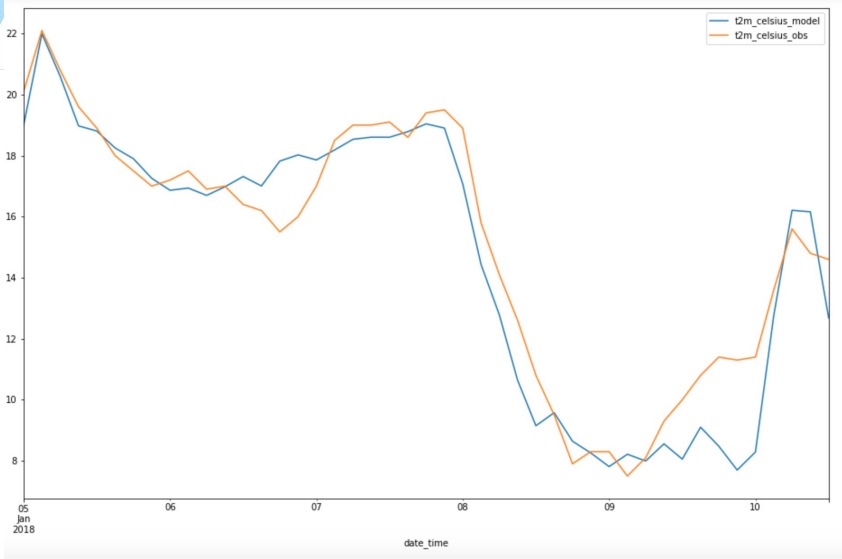
High Performance Computing CLUSTERTECH

CPAS HTS

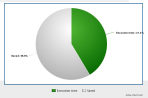




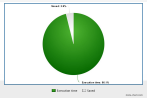
MPAS-A



Execution time:



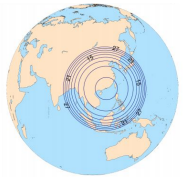
Execution time:

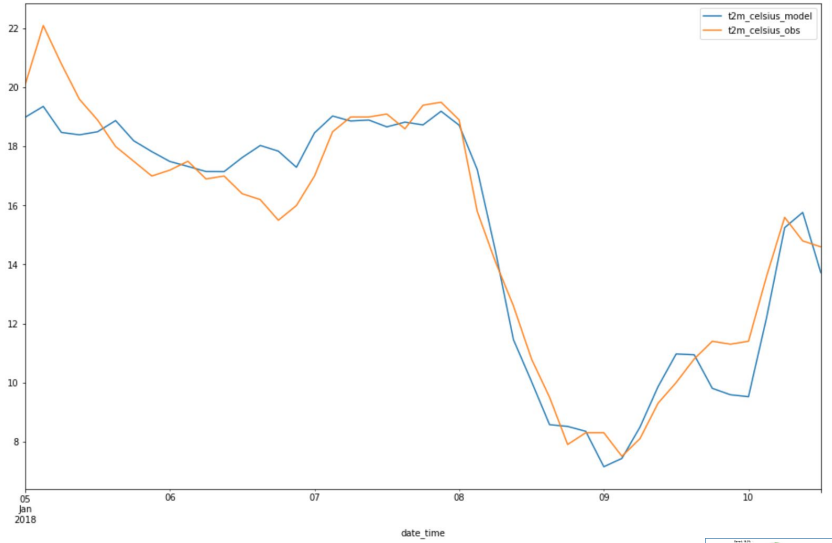


Standard 60km-3km mesh, 835,586 cells

High Performance Computing
CLUSTERTECH

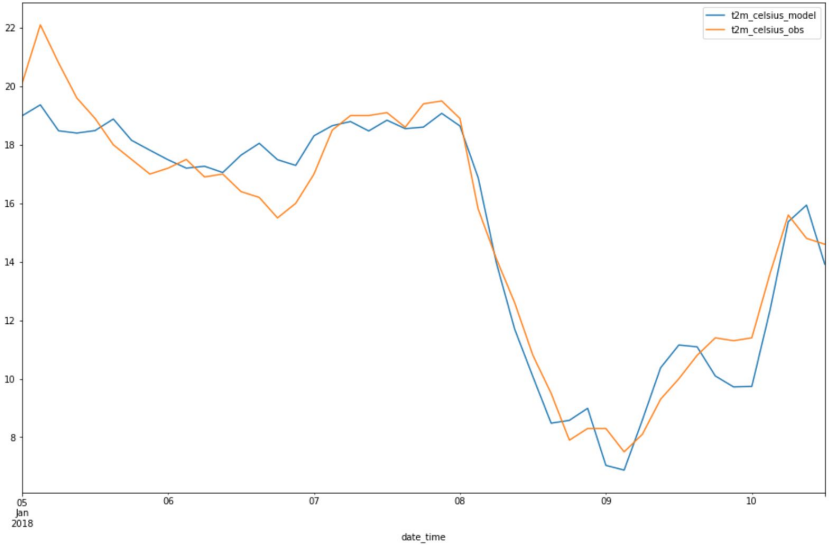
CPAS HTS



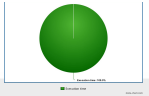


Execution time:

MPAS-A



Execution time:

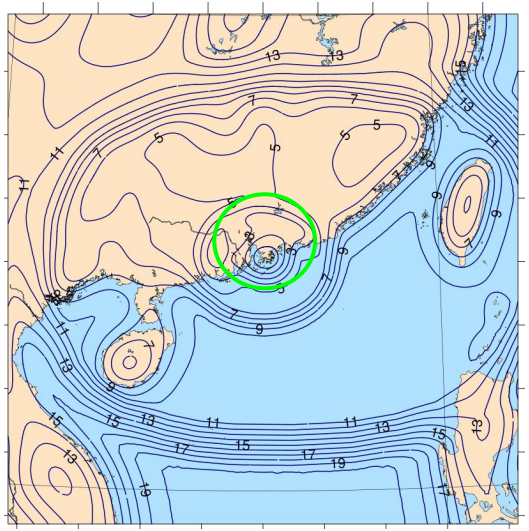


Comparison on resolutions

128km-1km mesh:

Small finer than 3km-resolution area

Most area much coarser than 3km



60km-3km mesh:

Large 3km-resolution area

More evaluation results

Whitepaper 2

High Performance Computing
CLUSTERTECH

https://cpas.earth/technology/publications

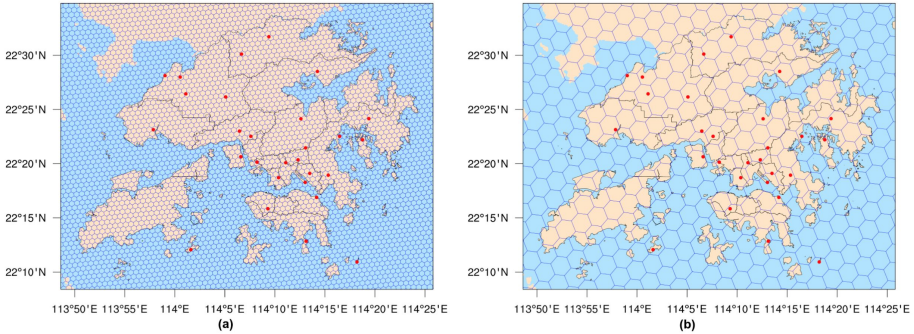


Figure 10: Geographic locations (red dots) of the 28 HKO stations for model evaluation, and the voronoi mesh for (a) the customized 128-to-1 km mesh and (b) the standard 60-to-3 km mesh

Better predictions in district (within city) length scale!

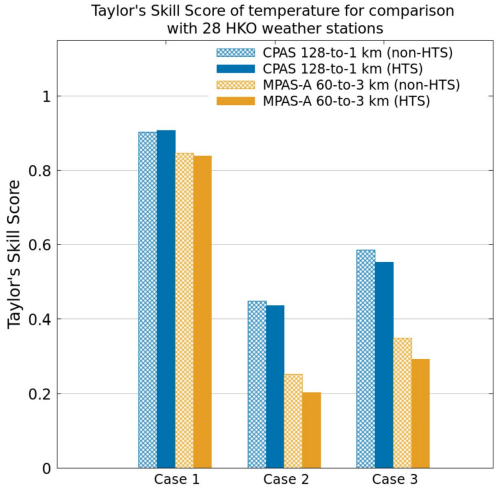
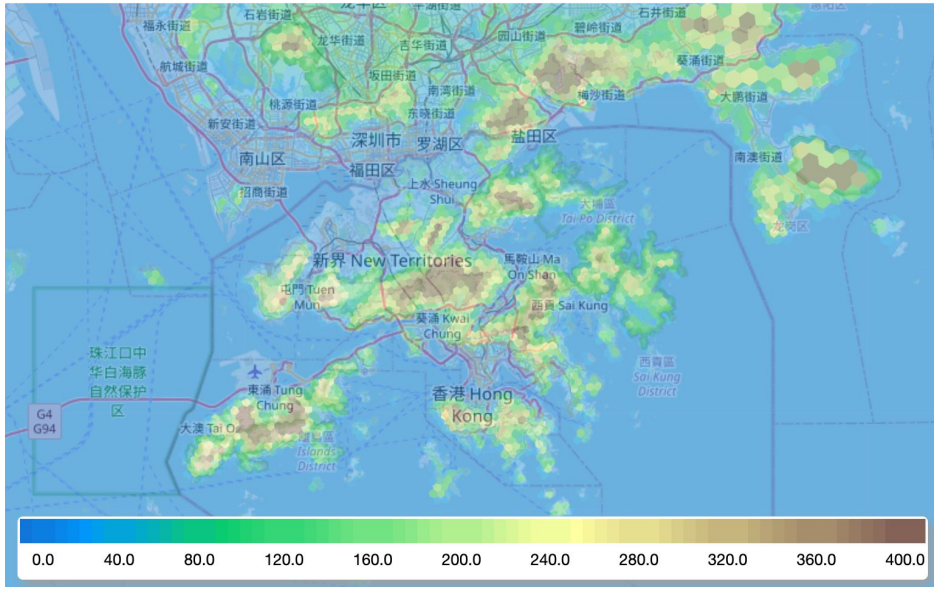
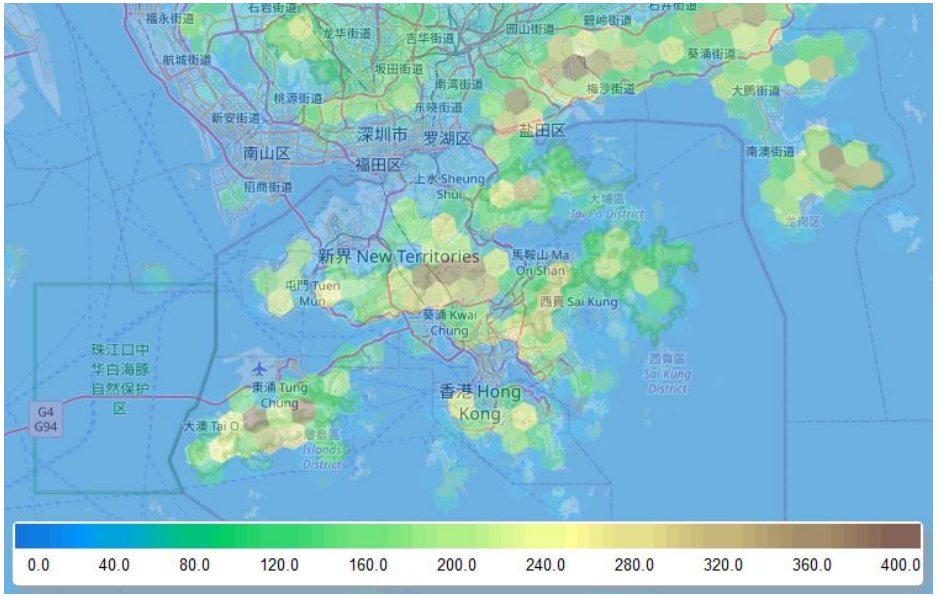


Figure 12: The Taylor's skill scores of temperature for comparison with 28 HKO weather stations

Better resolving of terrain



Terrain data in 1km resolution



Terrain data in 3km resolution

Sign-up cpas.earth to see demo data

https://cpas.earth/

jupyter

```
In [1]: from cpas.ui import UI
        ui = UI().select_job()
```

Select Data: My data | Demo data

Select Project:
2020-05-04 08:24:56Z EGU2020 HK_128-to-1km (generated by experimental OLAM-based algorithm; customized CPAS v0.4.0 experi
2020-04-07 01:25:10Z Customized 160km-1km mesh for Hong Kong (generated by Lloyd algorithm; MPAS-A v5.2 experiments)
2020-04-07 01:24:13Z Standard 92km-25km mesh centered at Hong Kong (downloaded from MPAS-A; CPAS v0.3.2 HTS experiments
2020-04-07 01:22:41Z Standard 60km-3km mesh centered at Hong Kong (downloaded from MPAS-A; CPAS v0.3.2 HTS experiments)

Select Mesh: 2020-05-04 03:10:19Z HK_128-to-1km #500 (experimental OLAM-based mesh generation algorithm)

Simulation | Shallow Water Test

Select Simulation:
2020-05-04 04:10:21Z 20180105 00Z Cold front (HTS)
2020-05-04 04:10:16Z 20180105 00Z Cold front (nonHTS)
2020-05-04 04:10:08Z 20180611 00Z Heavy rain (HTS)
2020-05-04 04:10:02Z 20180611 00Z Heavy rain (nonHTS)
2020-05-04 04:09:46Z 20180912 00Z Mangkhut (HTS)
2020-05-04 04:09:22Z 20180912 00Z Mangkhut (nonHTS)

```
Found data files:
 grid.nc | static.nc | mesh.nc | diag.nc
Ready to do plot_mesh() and visualize_mesh().
Ready to do plot_diag_contour() and visualize_sim().
```

Q&A / Discussion

Learn more &
start free trial now!
https://cpas.earth