Frontal activity in the Alpine region in convection permitting CCLM hindcasts and NWP products

Heimo Truhetz, András Csáki, Laurin Herbsthofer

Wegener Center for Climate and Global Globalen Wandel (wegcenter.uni-graz.at)
University of Graz, Austria

email: heimo.truhetz@uni-graz.at
Tel.: ++43 316 380 8442

COSMO User Seminar @ DWD, March 7, 2017

Acknowledgements

“Non-Hydrostatic Climate Modelling II” (NHCM-2)

funded by the Austrian Science Fund (FWF) (project ID: P24758-N29)
Jannuary 2013 to June 2017

Motivation

- **Develop a automated detection algorithm for a phenomenon-oriented model evaluation and model inter-comparison**

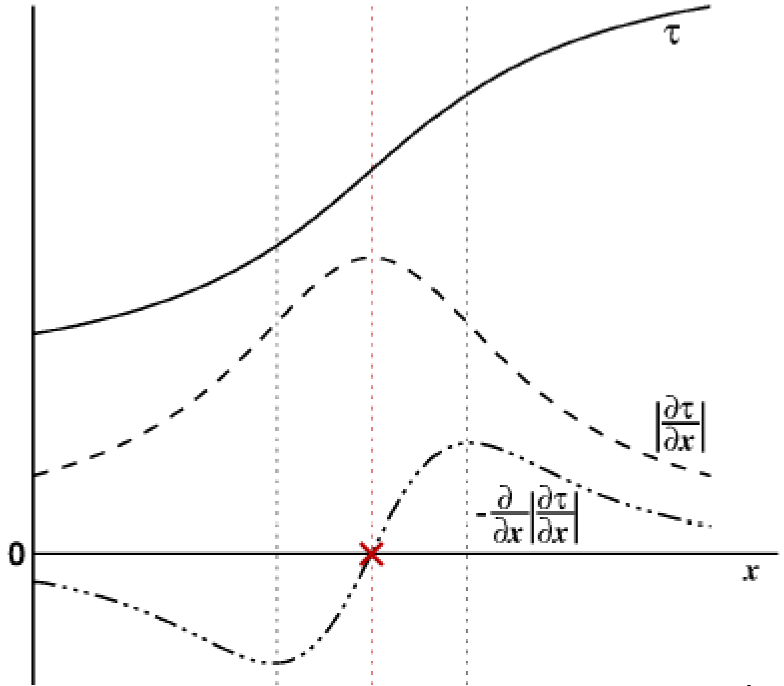
- **Requirements:**
- **(1) The algorithm should give similar results when compared to other front detection methods.**
  ➔ method inter-comparison; weather maps
- **(2) The algorithm should detect the same phenomenon in different models (also with different resolutions), when they simulate the same synoptic situation.**
  ➔ apply the method on different models (with different grids) that simulate the same synoptic situation

Outline

1. **Front detection algorithm**
2. Evaluation
3. Test case in the Alpine region
4. Conclusions

1. Front detection algorithm

- **Masterthesis by Robert Ritter (2014)**



Ritter (2014)

One-dimensional idealized thermal field τ. The red cross marks the position of the front, identified through TFP$= 0$.

TFP … thermal Front Parameter:

$$\mathrm{TFP} = -\nabla|\nabla\tau| \cdot \frac{\nabla\tau}{|\nabla\tau|} \stackrel{!}{=} 0\,.$$

τ … thermal field (**equiv. pot. temp**., **wet-bulb pot. temp**.)

masking condition for baroclinic zones:

$$|\nabla\tau| > K$$

***K* = 4.5 °C/100 km**
(Jenkner et al., 2010)

Condition to avoid local minima:

$$\nabla \cdot \nabla|\nabla\tau| < 0\,.$$

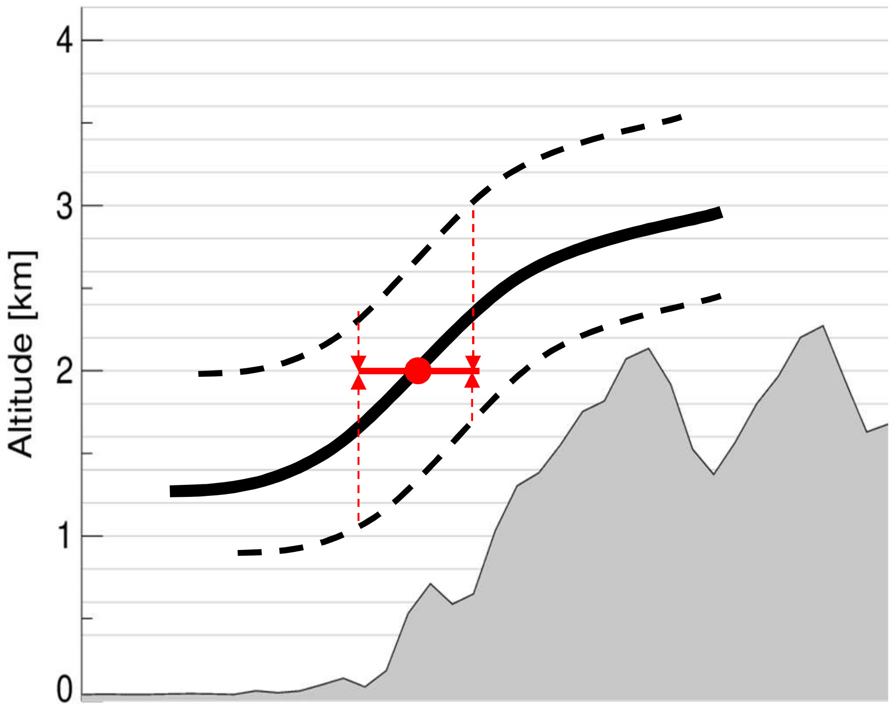
Front movement:

$$v_f = \mathbf{v} \cdot \frac{\nabla(\mathrm{TFP})}{|\nabla(\mathrm{TFP})|}$$

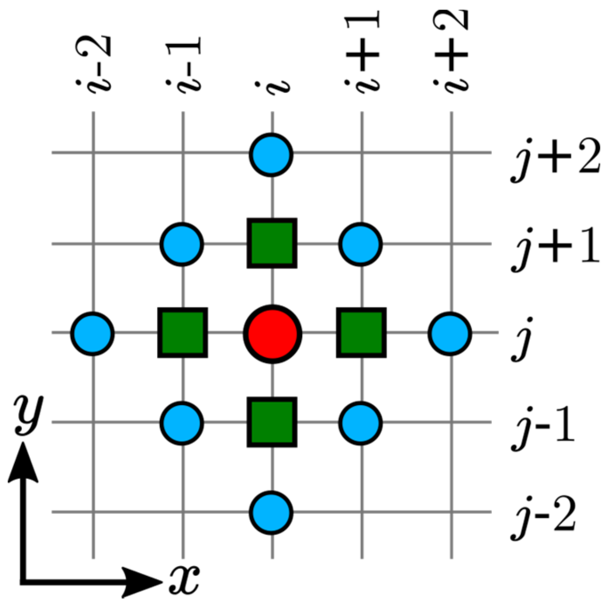
>0 … cold front
<0 … warm front

1. Front detection algorithm

- **Calculation of derivatives on true-z levels**



$$\mathrm{TFP} = -\nabla|\nabla\tau| \cdot \frac{\nabla\tau}{|\nabla\tau|} \stackrel{!}{=} 0 \,.$$

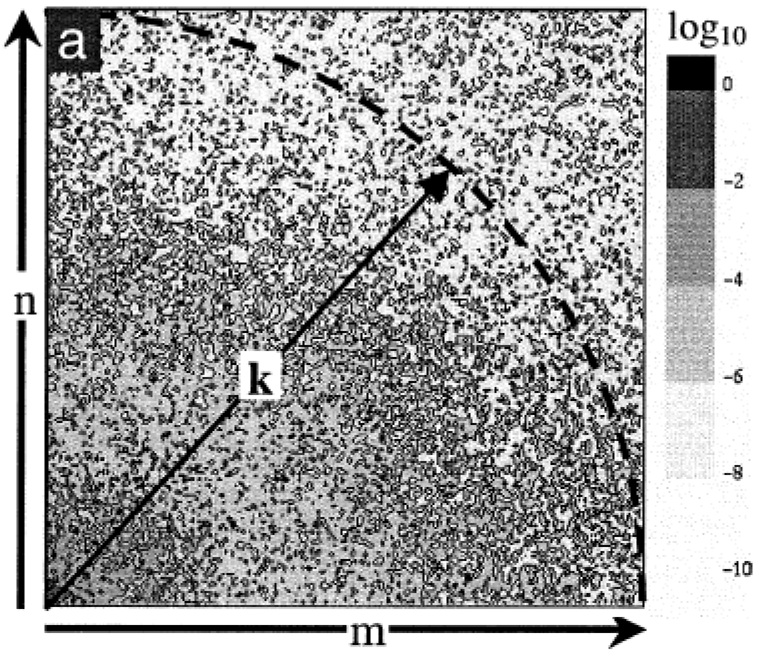


- **Input variables (P, T, Qv) of the surrounding points are onto the level of the detection height**
- **➔ Avoids artificial fronts on slanted detection surfaces**

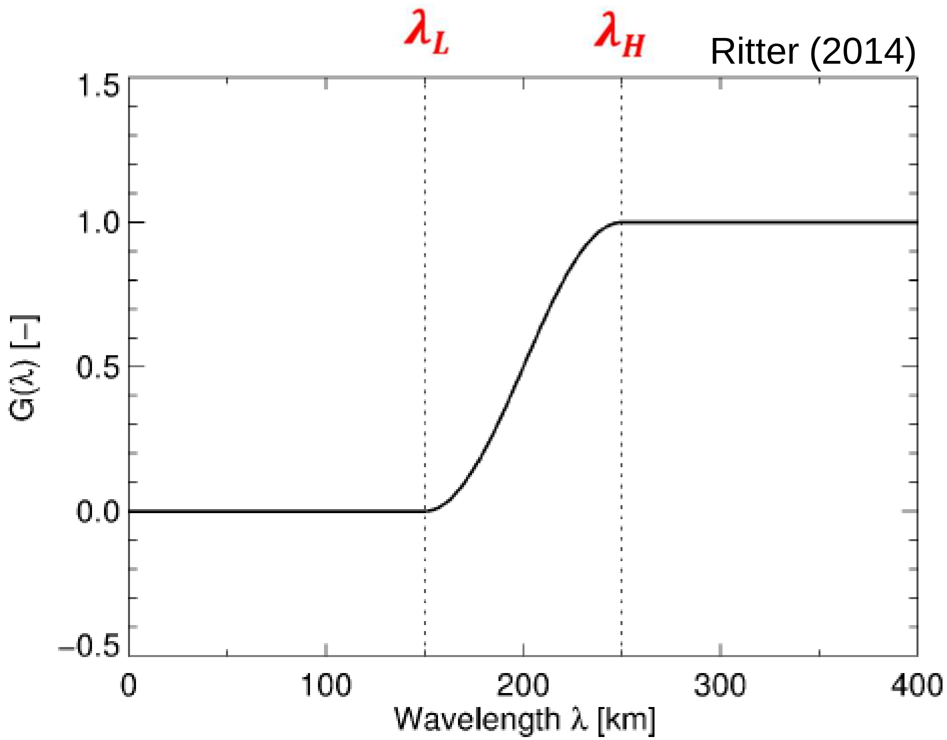
1. Front detection algorithm

- **Filtering via spectral separation**
  using the Discrete Cosinus Transform (DCS) (e.g. Denis et al., 2002)

$$F(m,n) = \beta(m)\beta(n) \sum_{i=0}^{i=N_i-1} \sum_{j=0}^{j=N_j-1} f(i,j) \cos\left[\pi m \frac{(i+\frac{1}{2})}{N_i}\right] \cos\left[\pi n \frac{(j+\frac{1}{2})}{N_j}\right]$$



Denis et al. (2002)

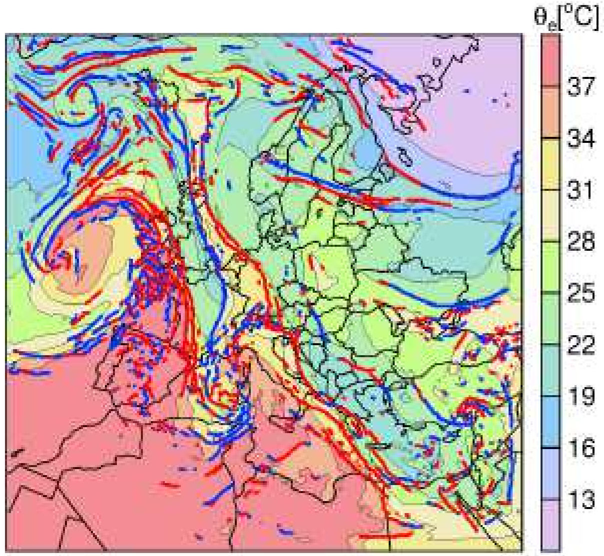


Filter function used in the DCT. Here, the filter function represents a low-pass filter. λ_L and λ_H mark the boundaries for the transition zone.

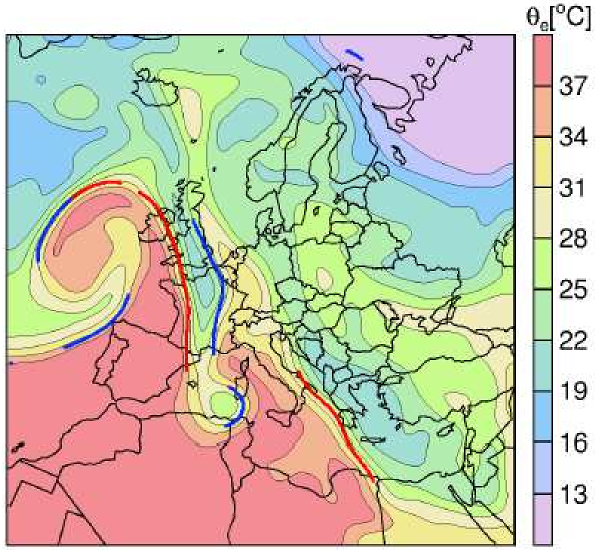
- ➔ **Comparability of different models with different resolutions**

1. Front detection algorithm

- **Sensitivity for spectral separation**
- Model data from EUR-11 CCLM hindcast (conducted by Klaus Keuler, BTU Cottbus)



(a) Unfiltered.



(d) Filtered for $\lambda_L = 300\,\text{km}$ and $\lambda_H = 500\,\text{km}$.

Ritter (2014)

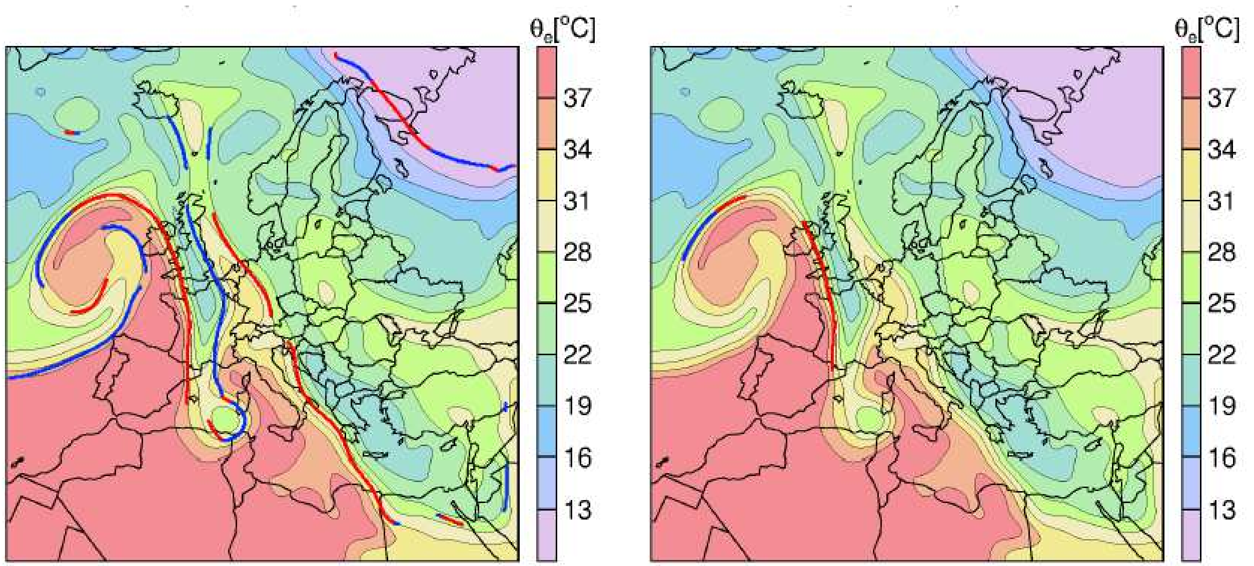
Effect of the filter setting on equivalent potential temperature θ_e and detected fronts for EU-011 on 6 January 1996 at 0600 UTC at 3000 m and $K = 4.5\,°\text{C}\,(100\,\text{km})^{-1}$. λ_L and λ_H refer to the lower and higher wavelength of the filter.

Filter settings matter

1. Front detection algorithm

- **Sensitivity for masking condition**

- Model data from EUR-11 CCLM hindcast (conducted by Klaus Keuler, BTU Cottbus)



(c) $\lambda_L = 300\,\text{km}$ and $\lambda_H = 500\,\text{km}$, $K = 3\,°\text{C}\,(100\,\text{km})^{-1}$.

(d) $\lambda_L = 300\,\text{km}$ and $\lambda_H = 500\,\text{km}$, $K = 6\,°\text{C}\,(100\,\text{km})^{-1}$.

Ritter (2014)

Effect of the thermal gradient threshold on detected fronts for EU-011 on 6 January 1996 at 0600 UTC at 3000 m for different filter settings. λ_L and λ_H refer to the lower and higher wavelength of the filter.

Masking condition matters

Outline

1. Front detection algorithm
2. **Evaluation**
3. Test case in the Alpine region
4. Conclusions

2. Evaluation (calibration)

- **Comparison with the Cyclone Database (CDB) of U.K. MetOffice (UKMO)**
  (Hewson, 1998; Hewson, 2001; Hewson and Titley, 2010)

  - **Automated derivation of fronts from operational MetOffice forecast model (0.44° grid spacing)**
  - **Extraction of well defined front points → stored in database (period 2000 to 2005)**

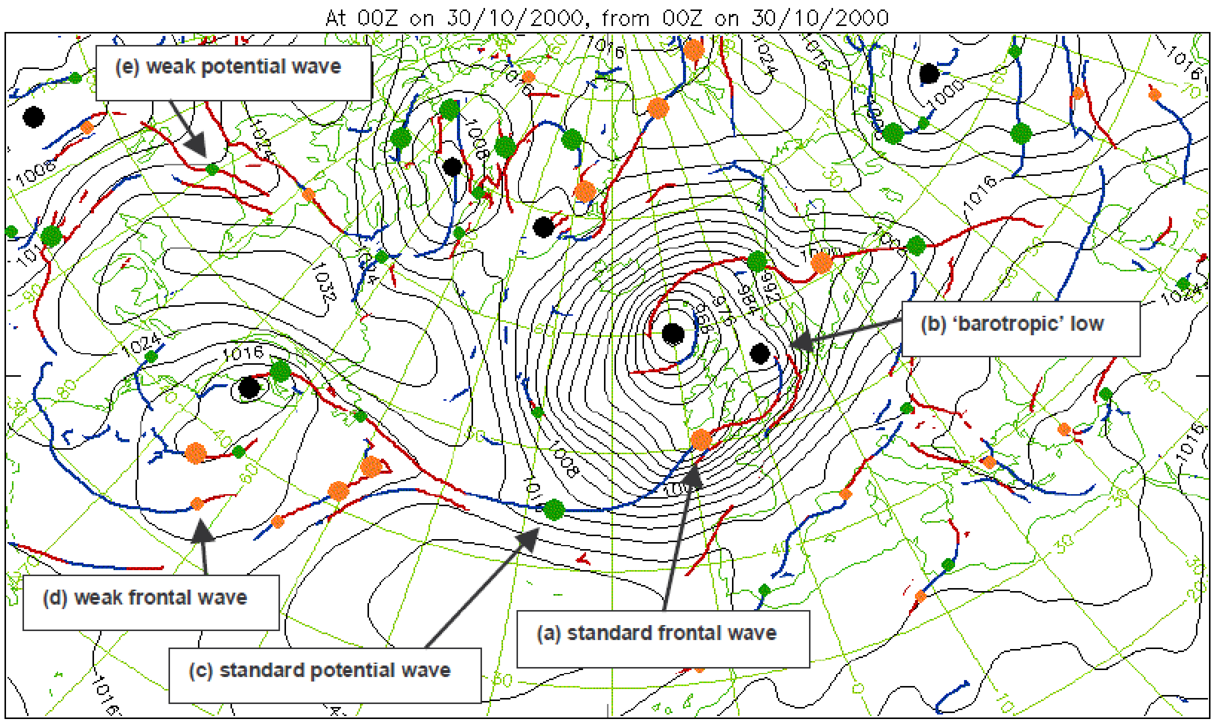


Figure 1: Standard and weak warm and cold objective fronts are shown as red and blue lines, derived using the surface 'gptl ht. above orography = 1km'. Black contours show mslp at 4mb intervals. Each spot type represents a different type of cyclonic feature, as labelled. The five types were plotted according to the *hierarchy* (a),(b),(c),(d),(e), which means that a 'co-location mask', applied after plotting each type, precludes identification of any other cyclonic feature later in the list within a 300km radius. Similarly, any two features of the same type closer than 300km initially will generally have been combined into one, located halfway between. Less reliance should be placed on any features over high topography

Hewson (2001)

- **1-2-1 smoother** is applied in advance
- Meeting point between cold and warm front
- Local maximum of the vorticity along a front in the across-front geostrophic wind
- Separation between weak and strong fronts

→ Mimic MetOffice model (0.44°) by remapping IFS (0.36°) onto EUR-44 grid

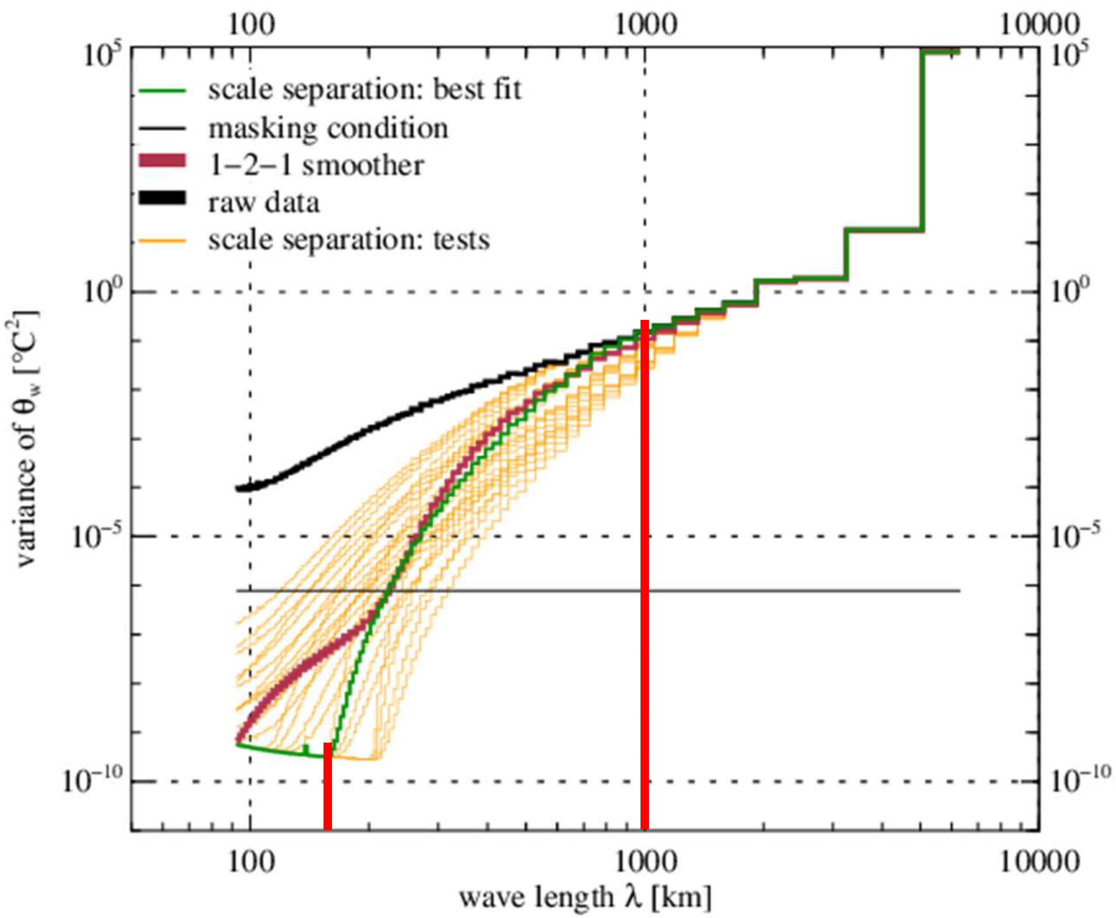
→ Derive filter settings and masking condition empirically by means of parameter testing

→ compare every front point in CDB with its closest counterpart from our algorithm

2. Evaluation (calibration)

- **Calibration of filter settings from spectral analysis (period 2001 to 2005)**

IFS remapped onto EUR-44 grid



$\lambda_L \downarrow \quad \lambda_H \longrightarrow$

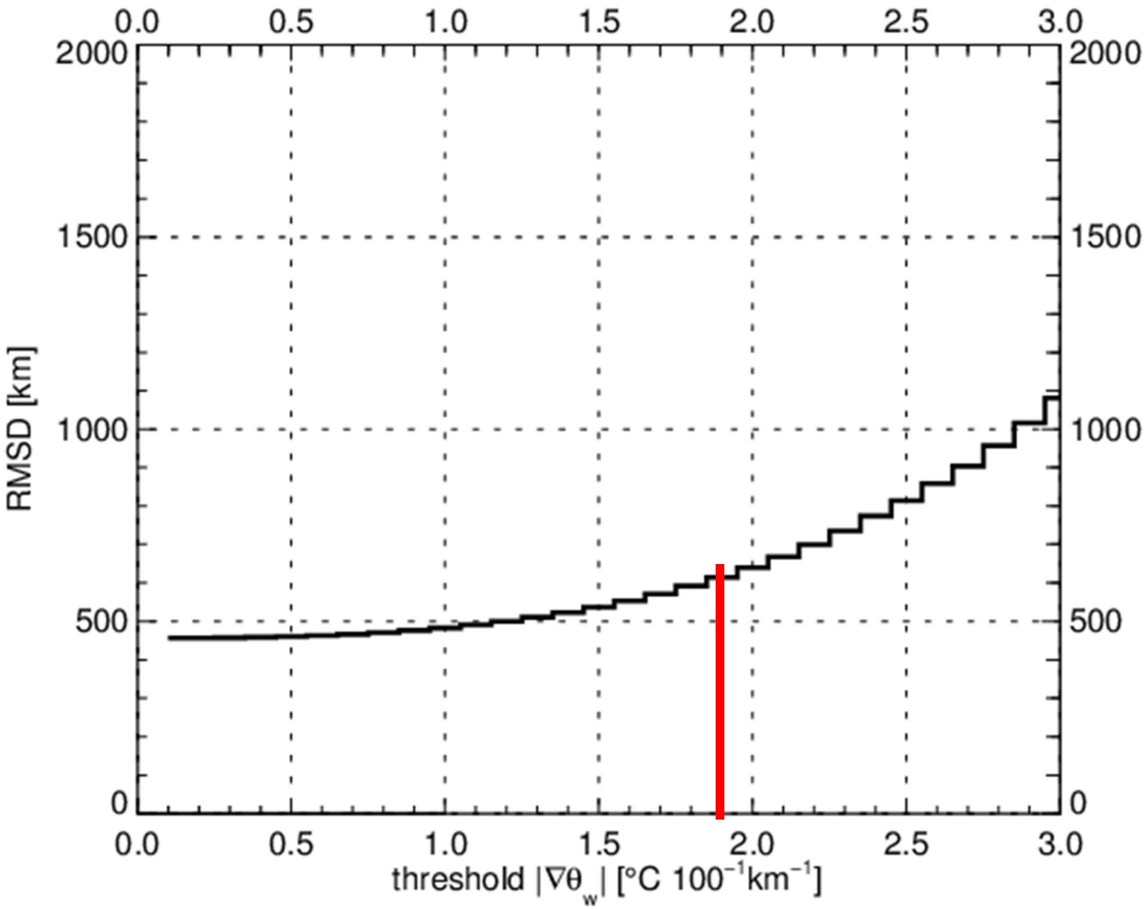
[km]	700	1000	1500	2000
0	x	x	x	x
25	x	x	x	x
50	x	x	x	x
100	x	x	x	x
150	x	x	x	x
200	x	x	x	x

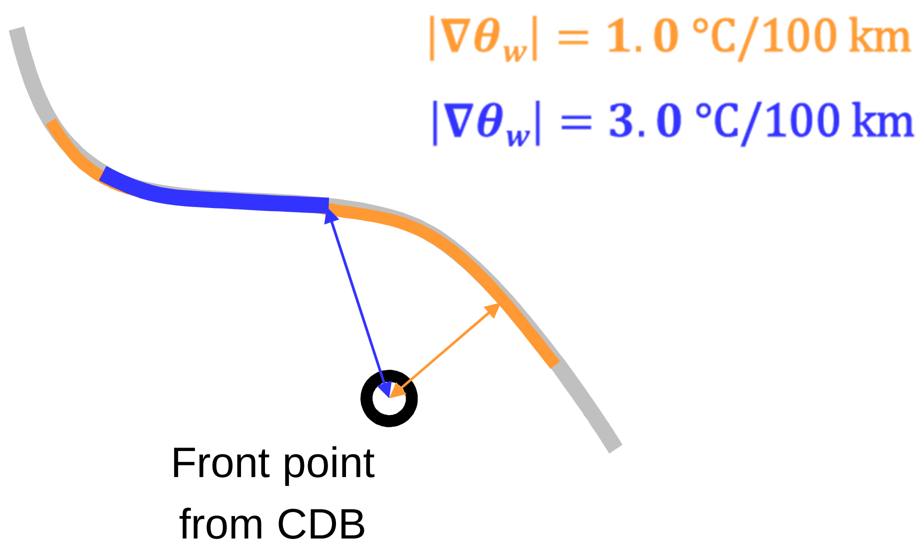
$\lambda_L = 150\ \mathrm{km}$

$\lambda_H = 1000\ \mathrm{km}$

2. Evaluation (calibration)

- **Calibration of masking condition (period 2001 to 2005)**





$|\nabla\theta_w| = 1.9$ °C/100 km

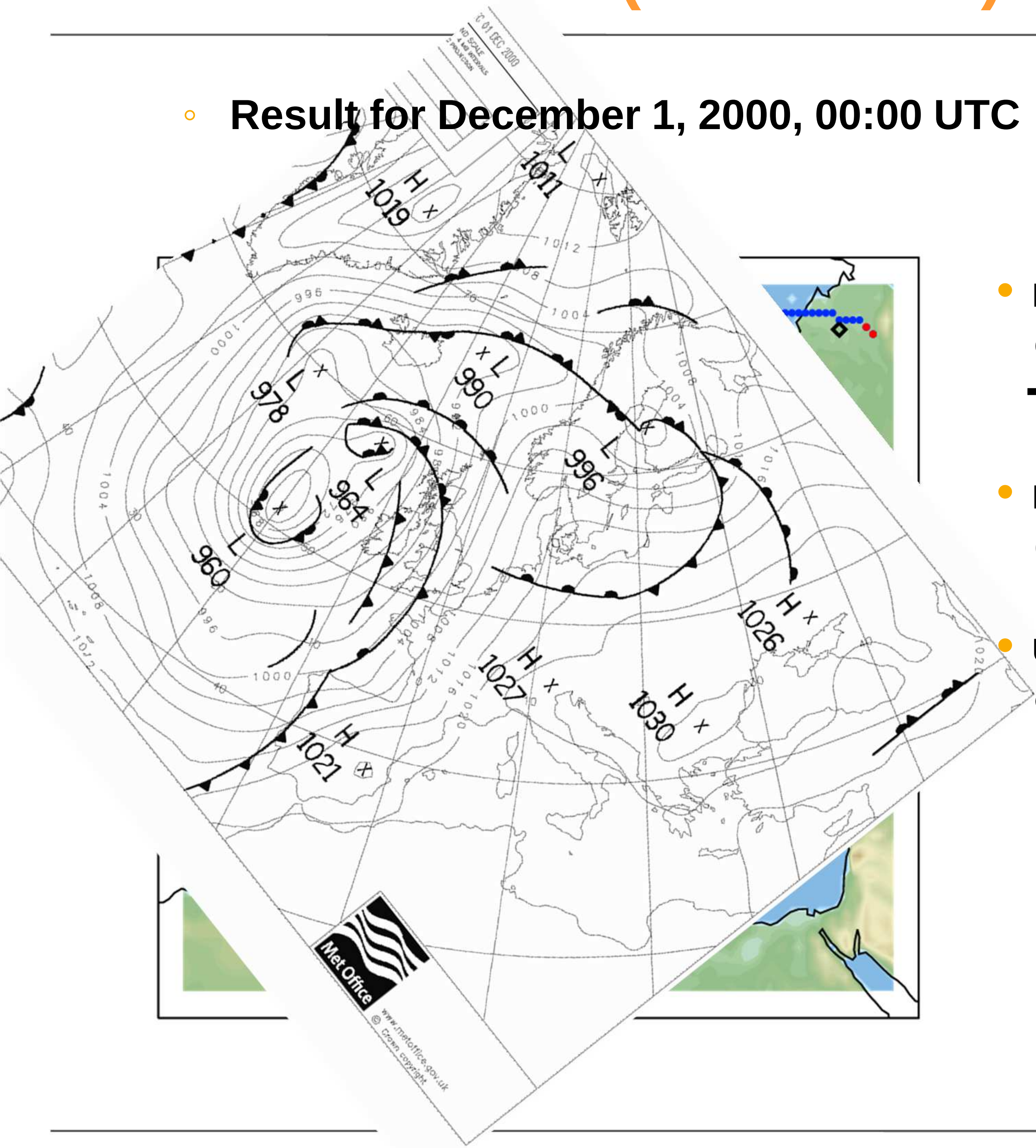
(weak definition)

threshold is too small
→ fronts are too long,
no change in RMSD

threshold is too large
→ fronts are too short
→ number of fronts is too low

2. Evaluation (calibration)

◦ **Result for December 1, 2000, 00:00 UTC**



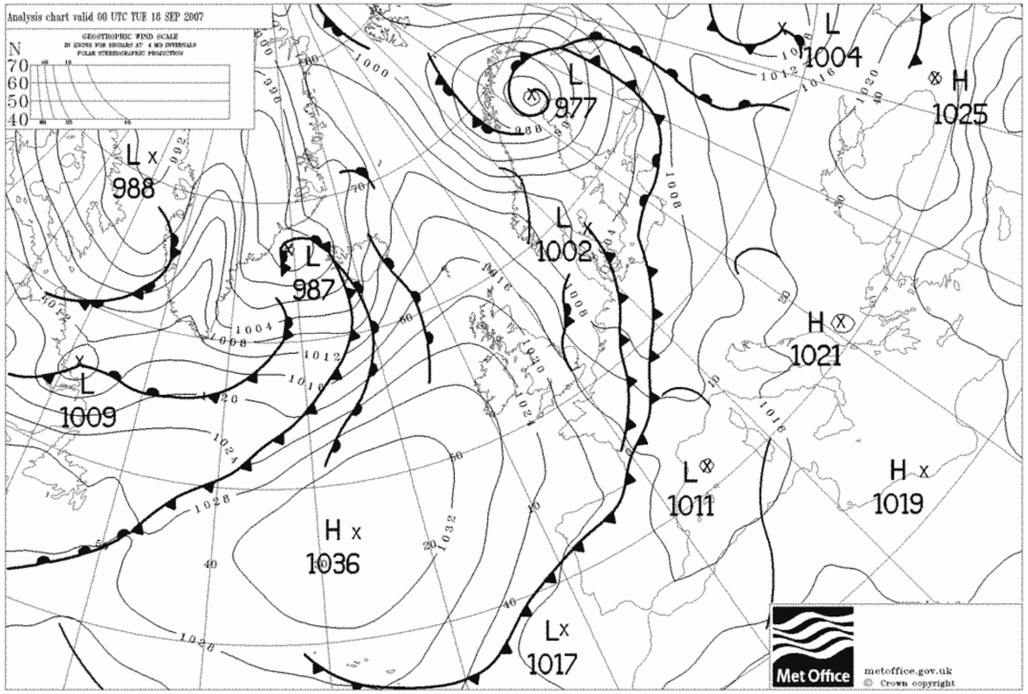
- **Fronts from smoothed and filtered data do largely agree**
  ➔ demonstrates success of scale separation
- **Deviations from CDB front points remain (different grids, algorithm)**
- **UKMO meteorologist does also not agree on CDB…**

Outline

1. Front detection algorithm
2. Evaluation
3. **Test case in the Alpine region**
4. Conclusions

3. Test case in the Alpine region

- **September 18, 2007**
  - **Vienna Enhanced Resolution Analysis (VERA; www.univie.ac.at/amk/vera) used in the MesoVICT project (Mesoscale Verification Inter-Comparison over Complex Terrain) in US (NCAR/UCAR) (Dorninger et al., 2013)**



00:00 UTC

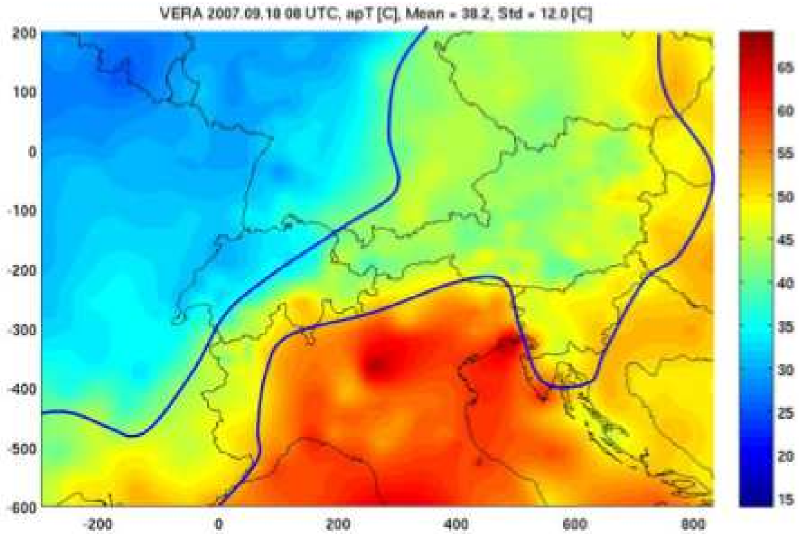
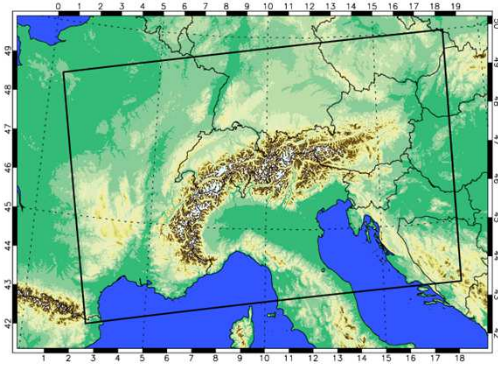


Figure 16: Equivalent potential temperature analysis for 18 September 2007, 08 UTC. Blue lines indicate frontal positions.

Dorninger et al. (2013)

3. Test case in the Alpine region

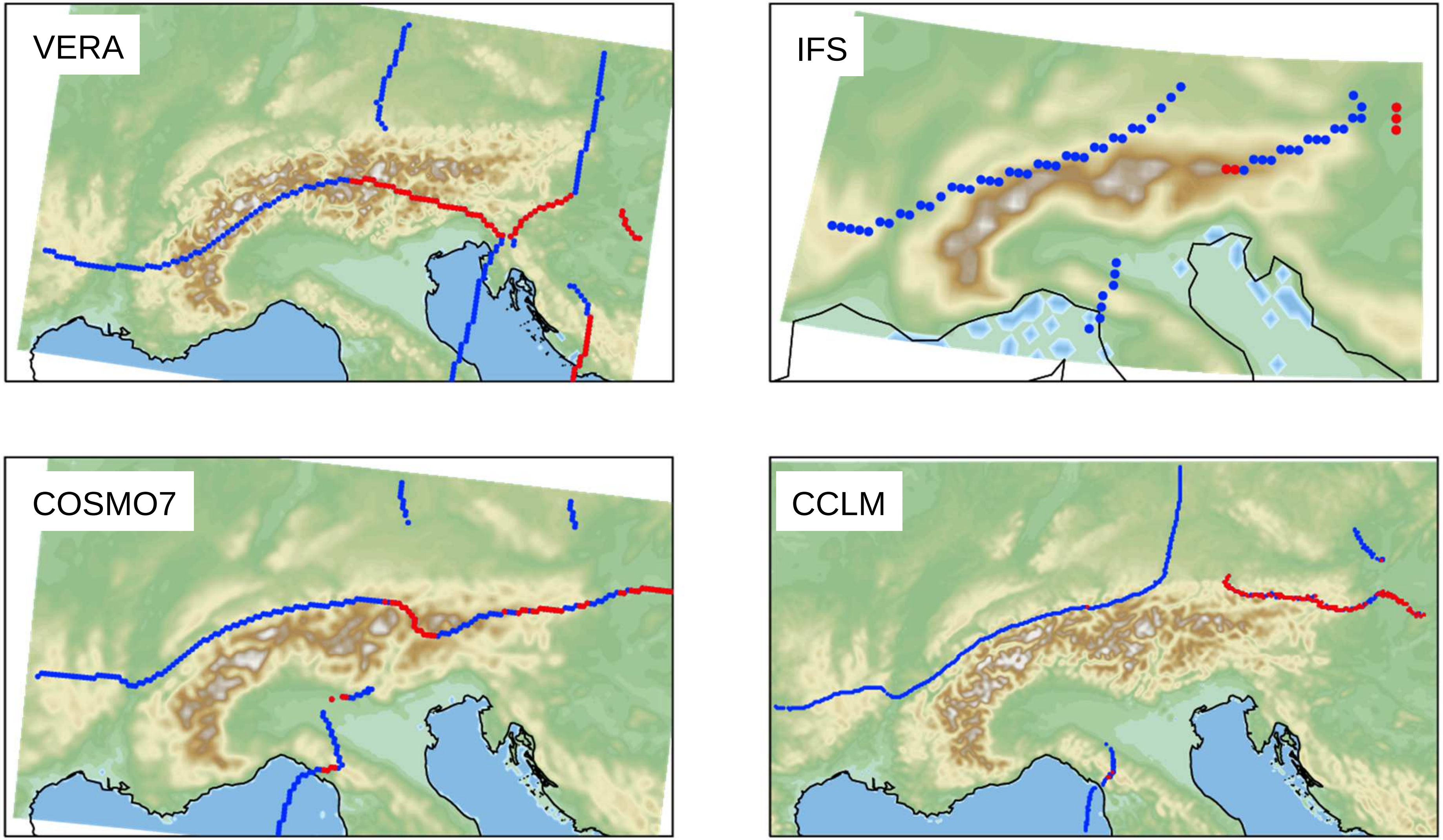
- **Models**
  - **cosmo_131108_5.00_clm1 (0.0275° grid spacing)**
    no deep convection parameterisation
  - **Driving data: IFS (0.225° grid spacing)**
  - Period: Nov 2005 – Dec 2010
  - Initialised by a 17 year spin-up simulation with
    cosmo_090213_4.8_clm17
    (driven by EUR-11 hindcast from Klaus Keuler, BTU Cottbus)
  - **Analysis fields of COSMO7 (0.06° grid spacing)**
  - **Analysis and forecast fields of IFS (0.225° grid spacing)**



➔ **consistency in synoptics is ensured**

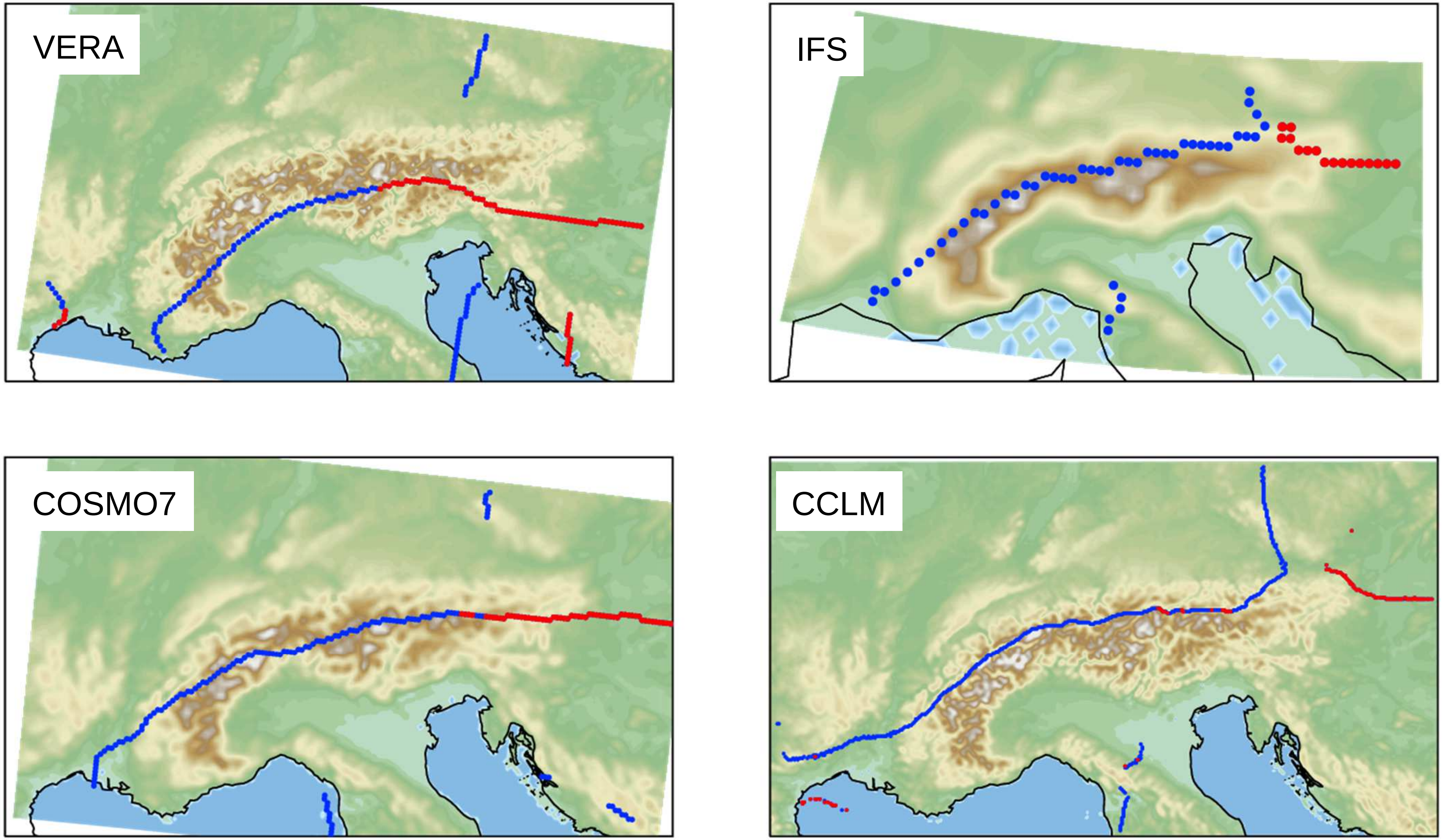
3. Test case in the Alpine region

- **09 UTC**



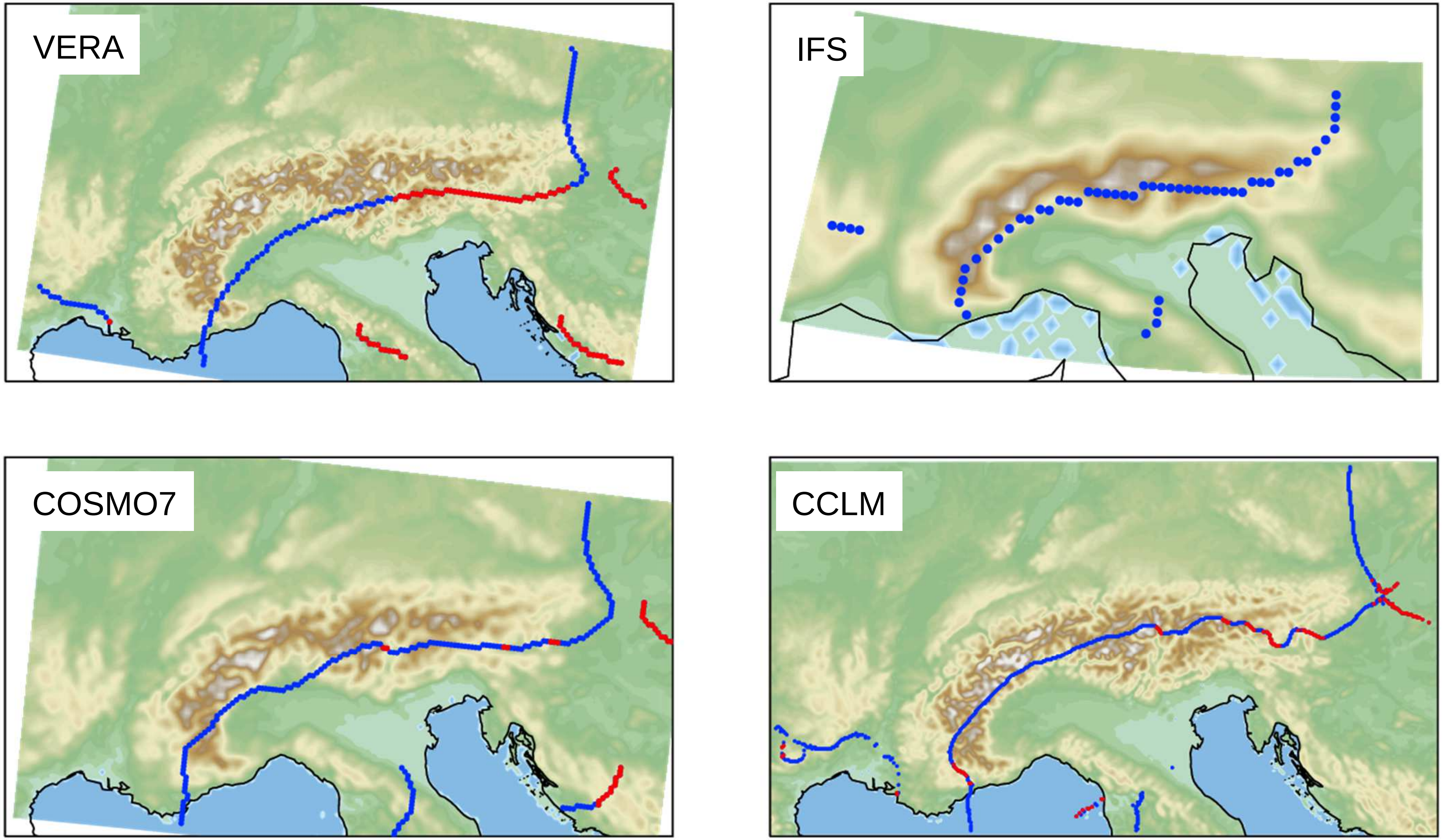
3. Test case in the Alpine region

- **12 UTC**



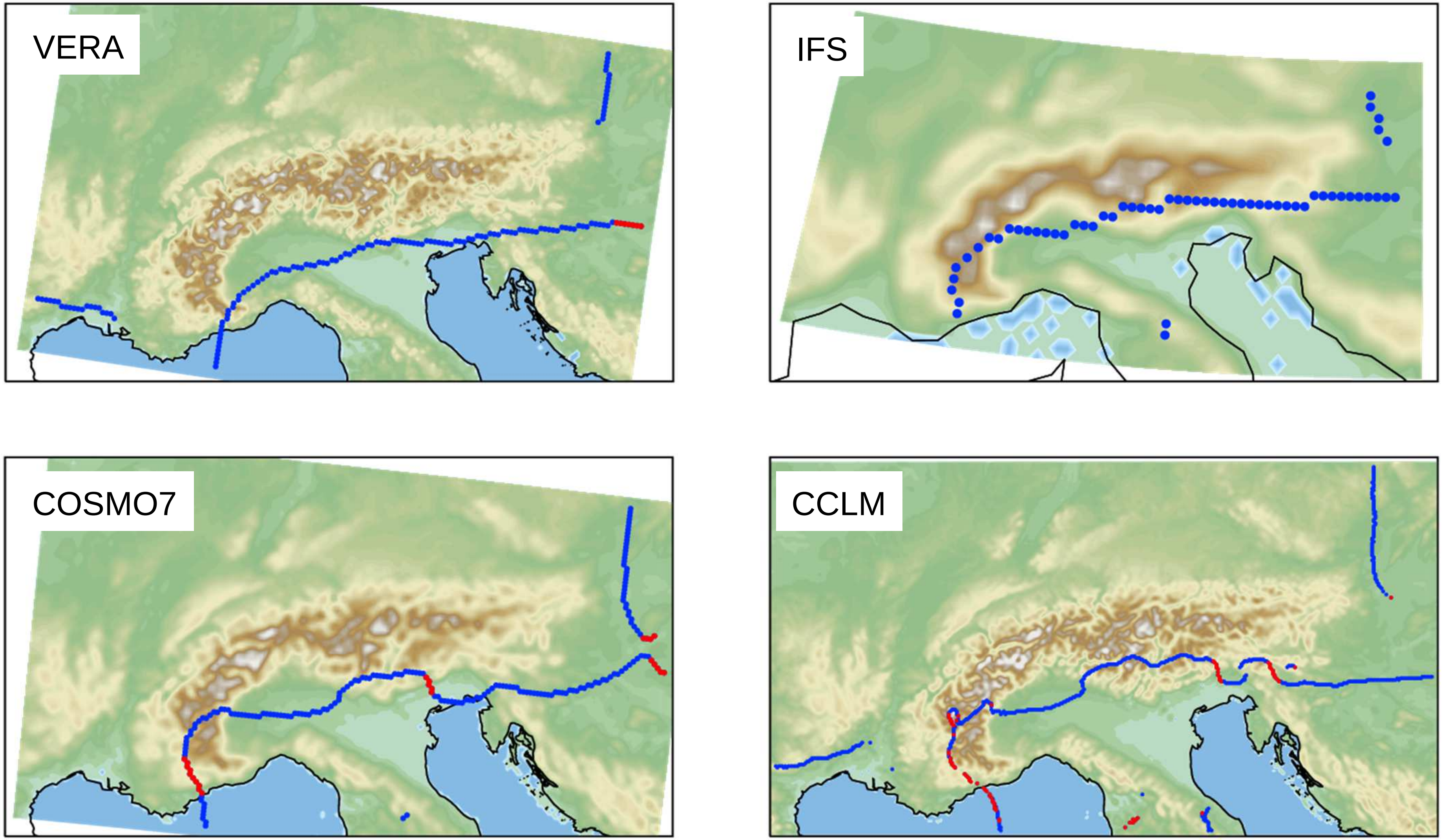
3. Test case in the Alpine region

- **15 UTC**



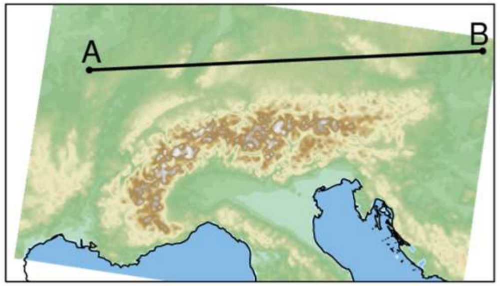
3. Test case in the Alpine region

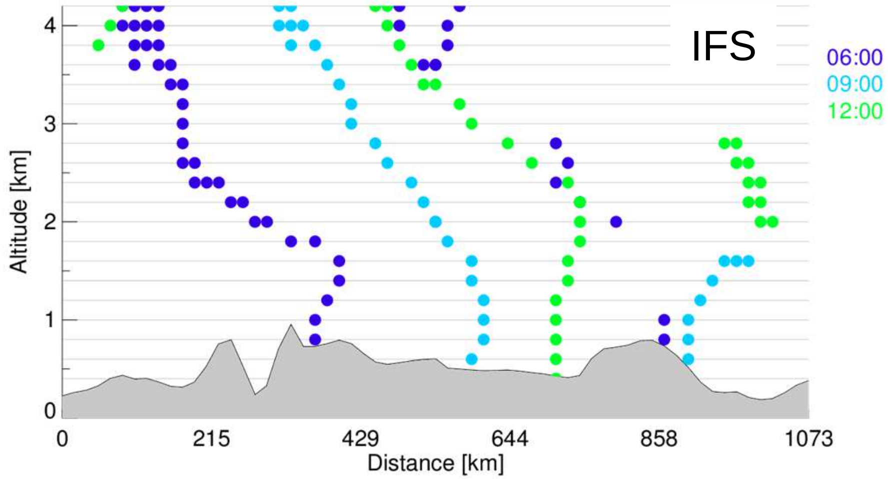
- **18 UTC**

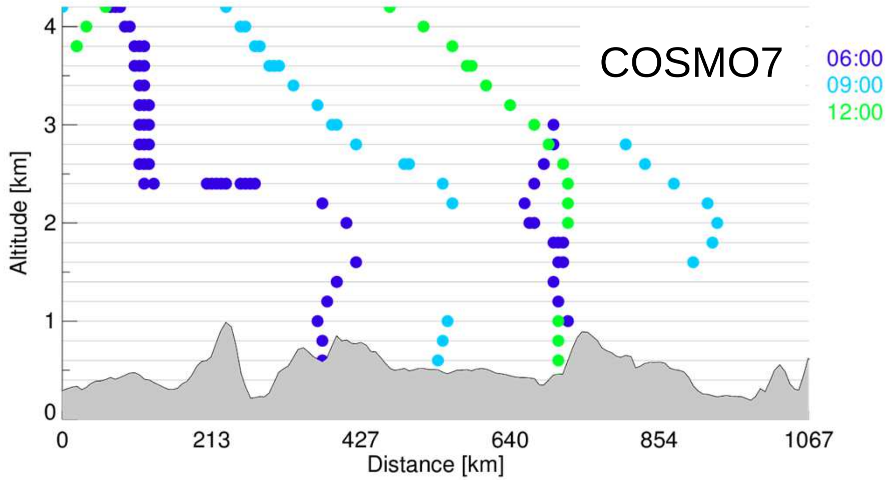


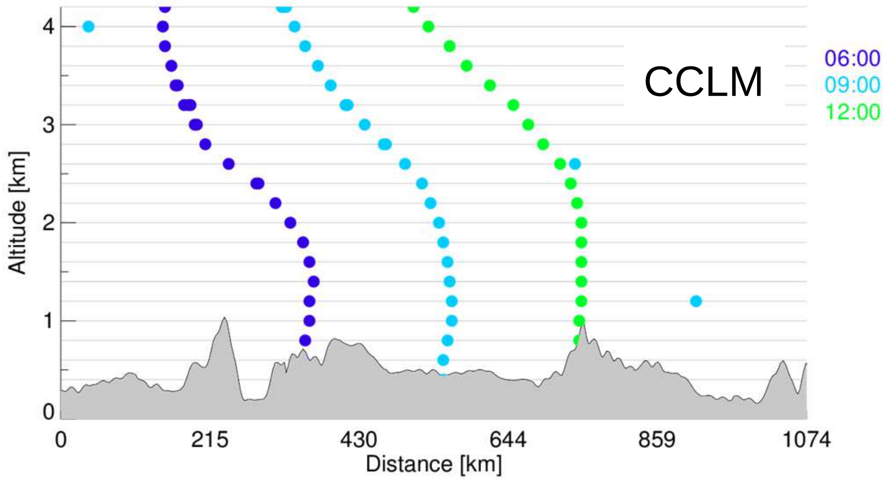
➔ **Orography has significant impact**

3. Test case in the Alpine region



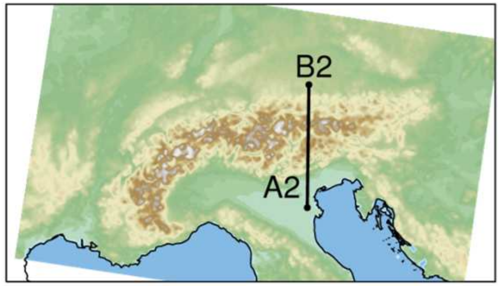


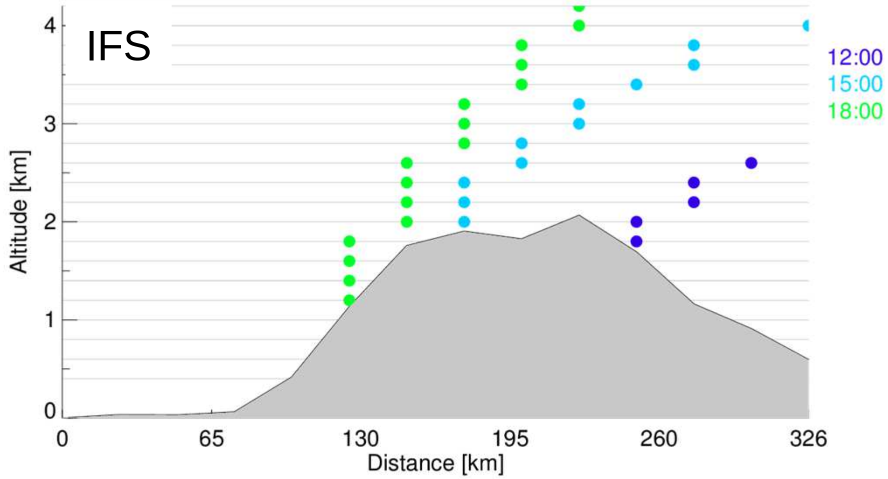




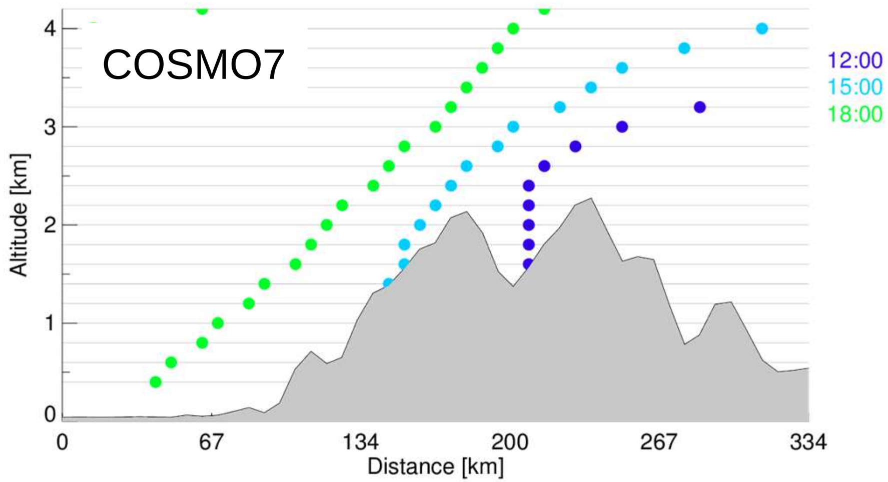
➔ **Deep convection parameterisation has significant impact**

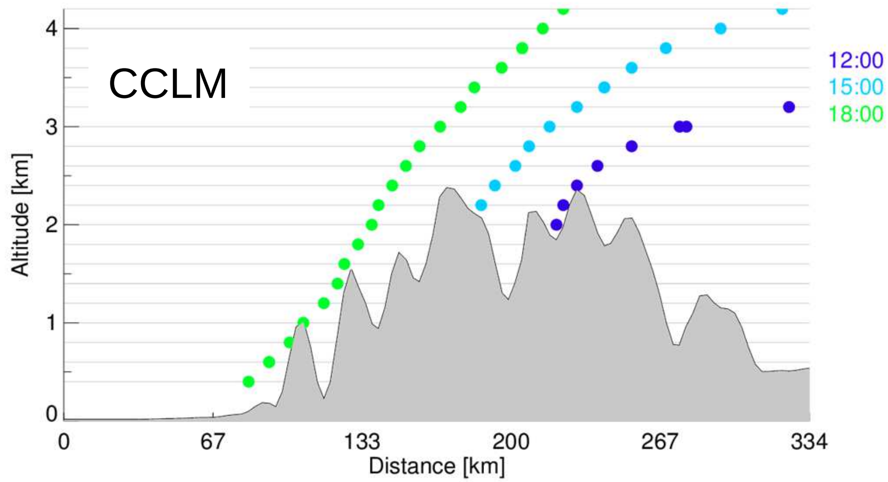
3. Test case in the Alpine region





➔ **Small deviations over mountains**





Outline

1. Front detection algorithm
2. Evaluation
3. Test case in the Alpine region
4. **Conclusions**

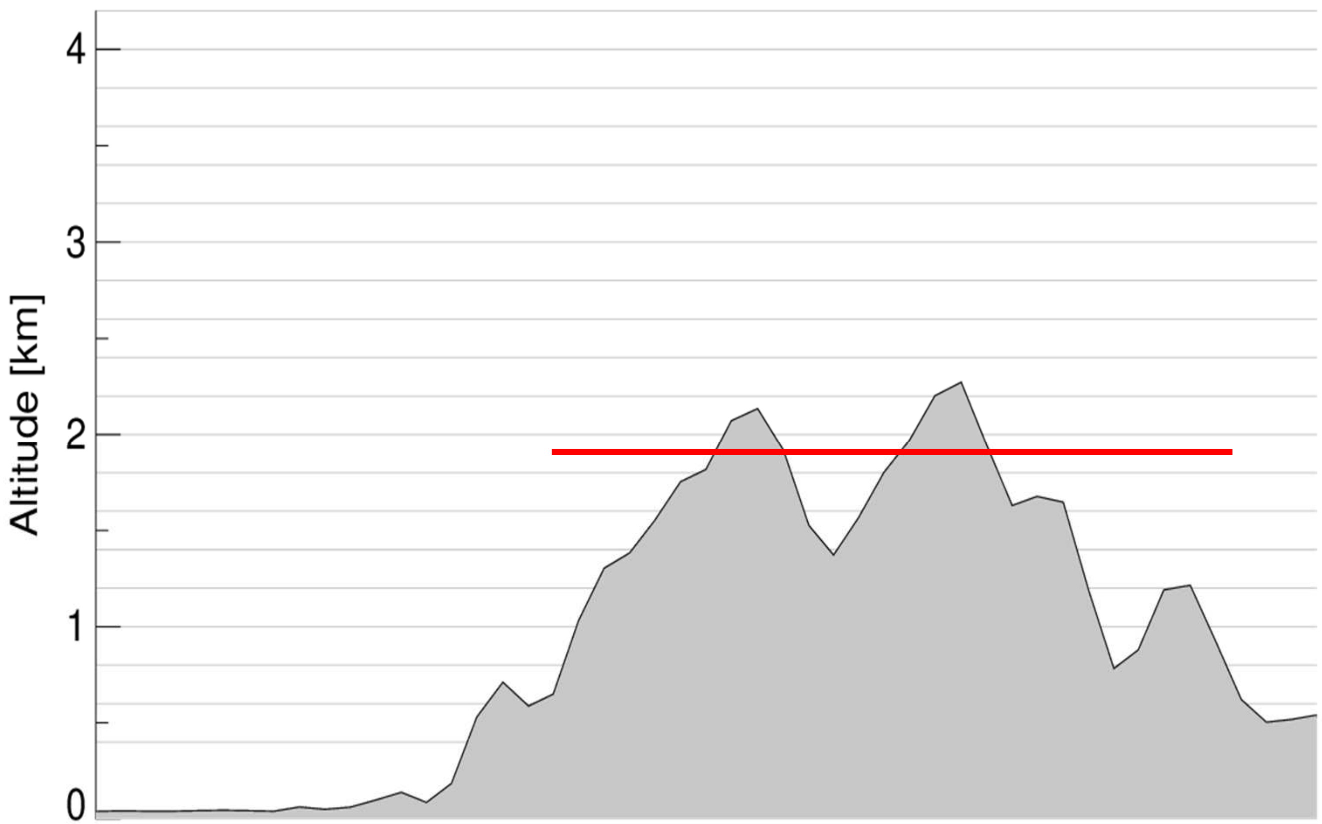
4. Conclusions

- **Develop a automated detection algorithm for the purpose of model evaluation and inter-comparison**

- **Requirements:**
- **(1) The algorithm should give similar results when compared to other front detection methods.**
  ➔ method inter-comparison; weather maps ➔ fulfilled
- **(2) The algorithm should detect the same phenomenon in different models (also with different resolutions), when they simulate the same synoptic situation.**
  ➔ fulfilled
  ➔ the phenomenon is differently captured by different models
  ➔ parameterisation of deep convection has significant influence
  ➔ over mountains the deviations are smallest

1. Front detection algorithm

- Calculation of derivatives on true-z levels



- **Scale separation does not allow missing data**

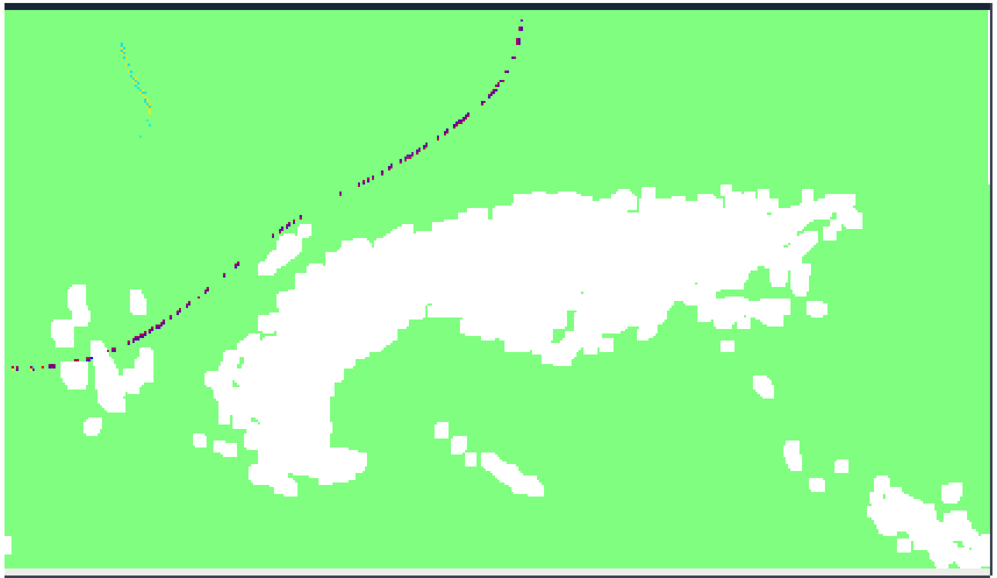
  ➔ **subsurface areas need to be filled**

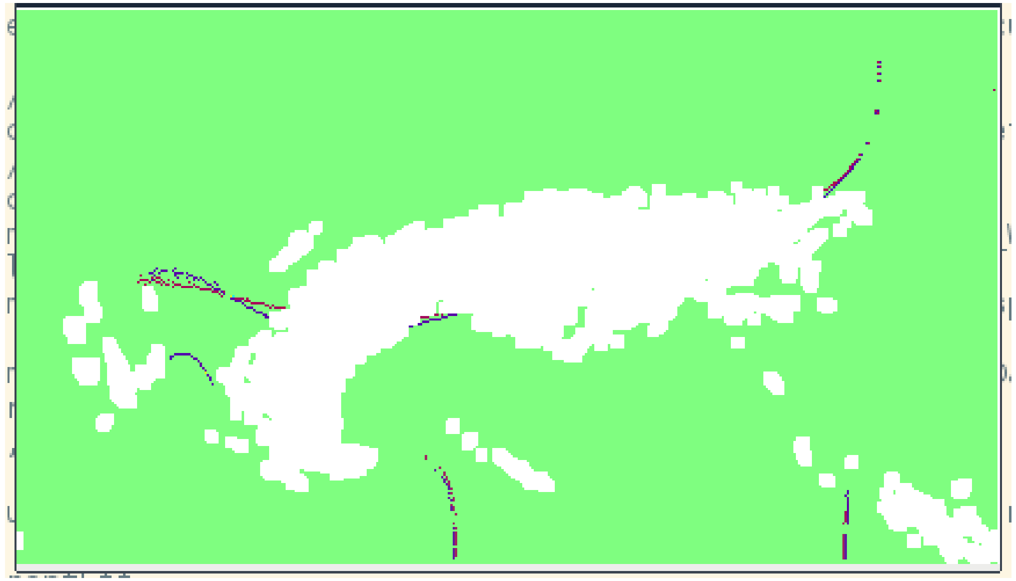
  ➔ **filling based on spatial anomalies from the level above (no local extrapolation)**

  ➔ **filling is robust, but it matters**

- **Detection "surface" itself matters**

How to evaluate the front detection algorithm?





How to evaluate the front detection algorithm?

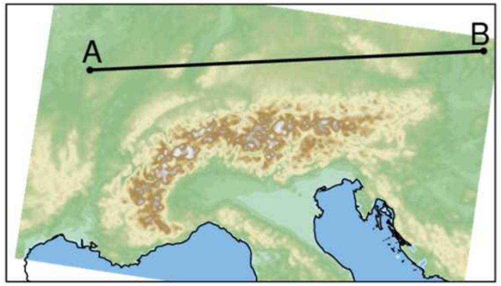
- **Detection surface**

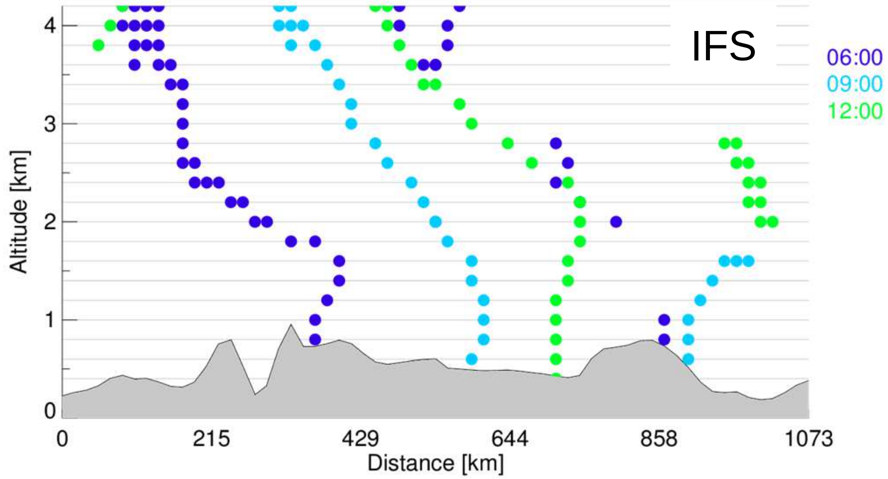
  - **Cyclone Database:**
    1000 m above surface geopotential smoothed with 1-2-1 smoother

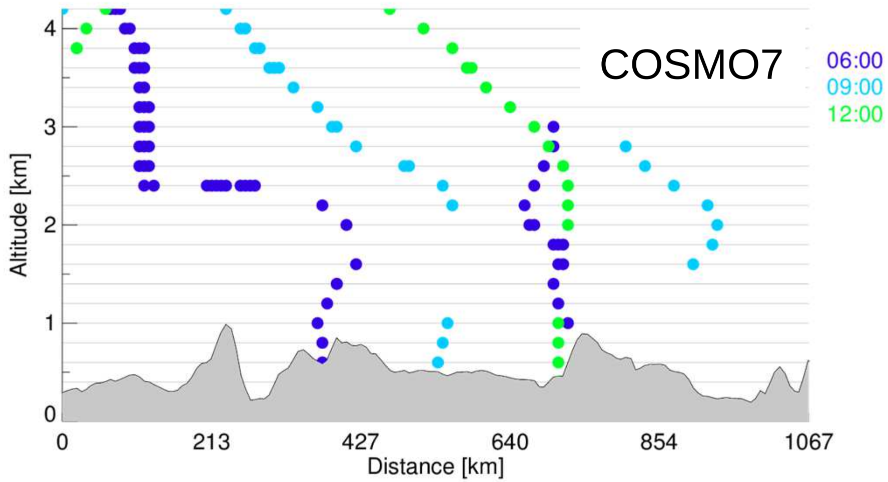
  - **Apply same procedure on all model grids**

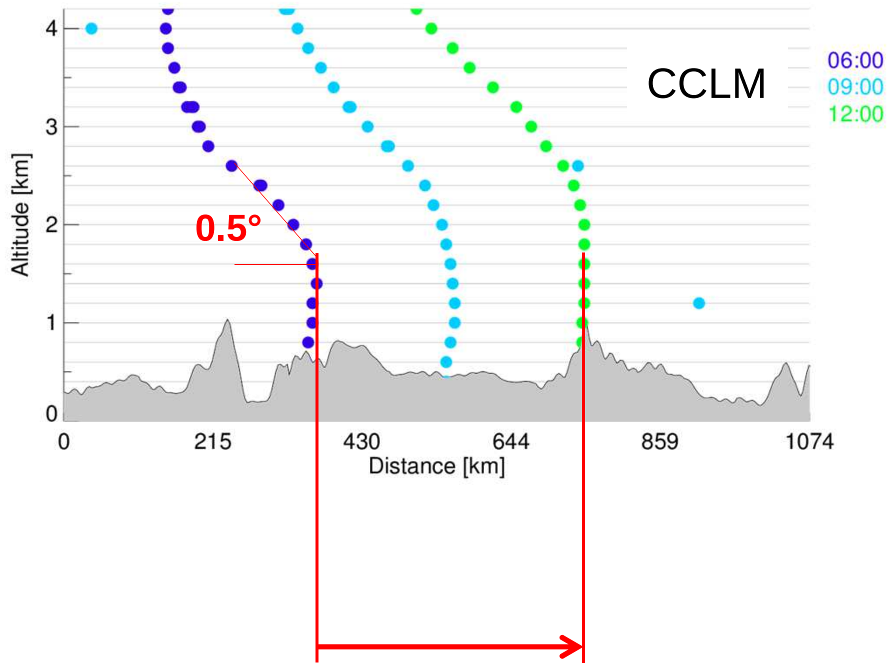
  ➔ **consistent definition, but maybe unsatisfying**

3. Test case in the Alpine region





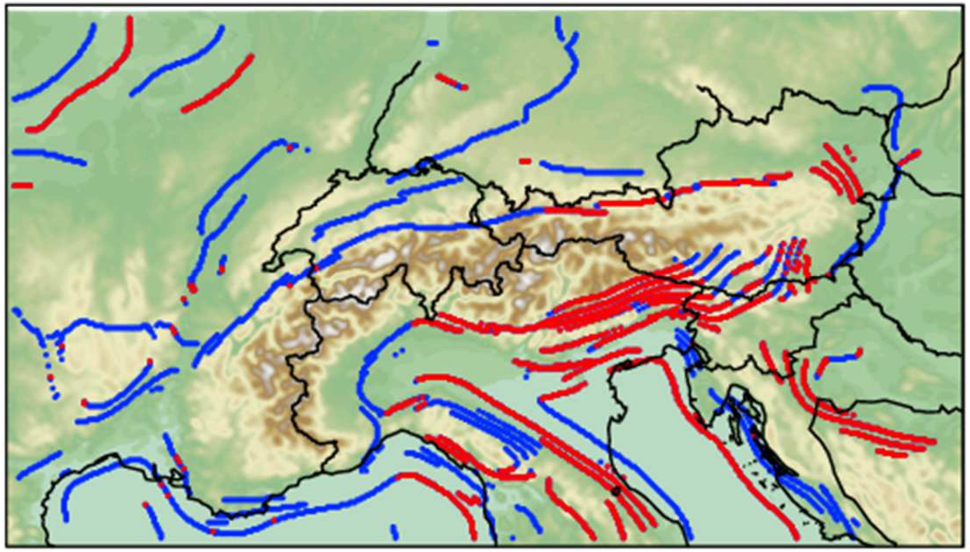




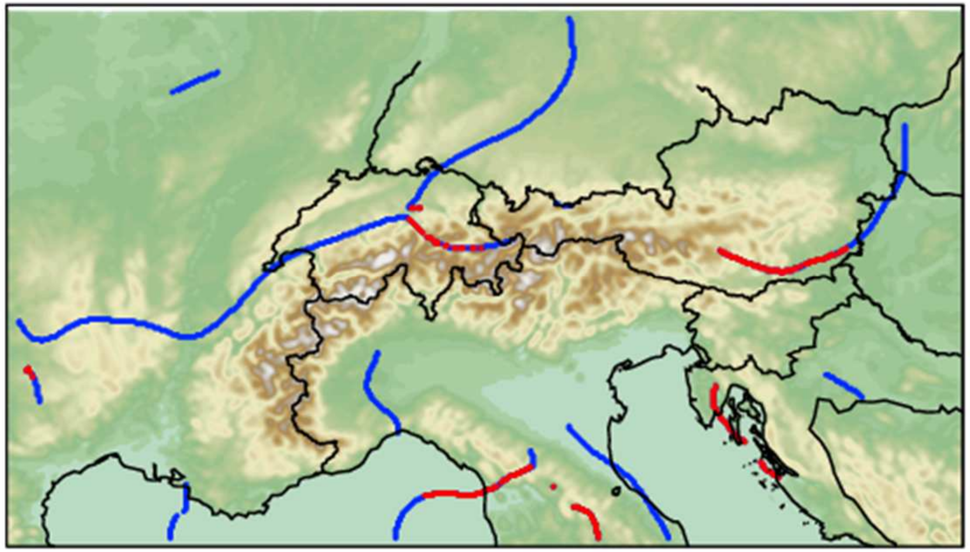
~390 km in 6 hours → ~**65 km/h**

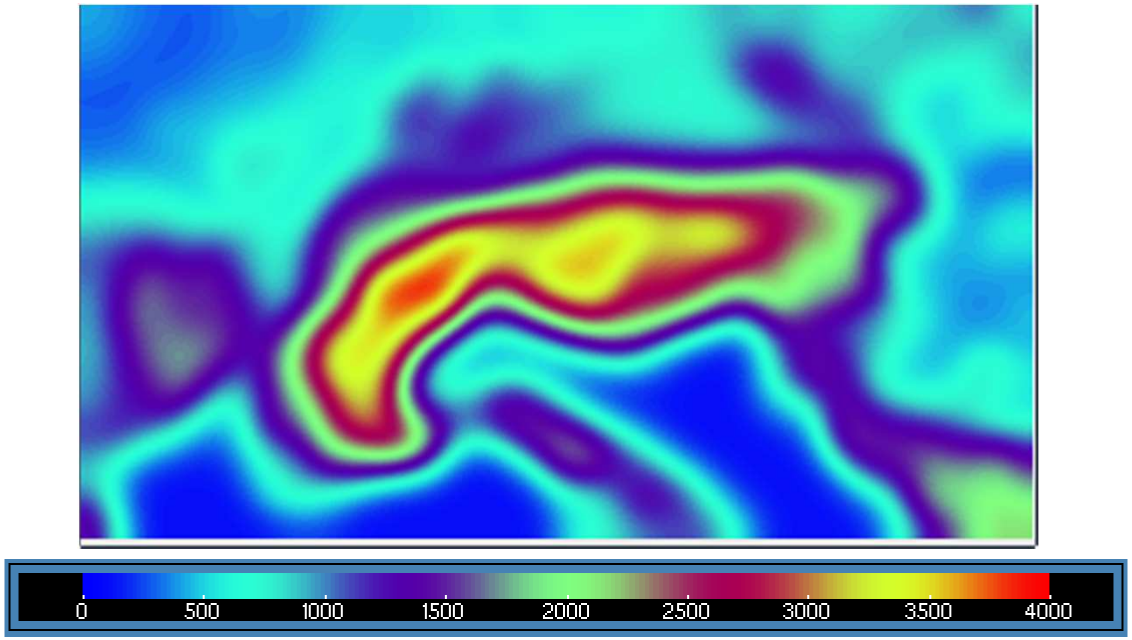
Lookback

Derivatives calculated on slanted surface



Derivatives calculated on true z-levels





Event at September 18, 2007, 08:00 UTC

CCLM (3 km grid spacing) driven by IFS;
TKE-SV scheme with shallow conv.

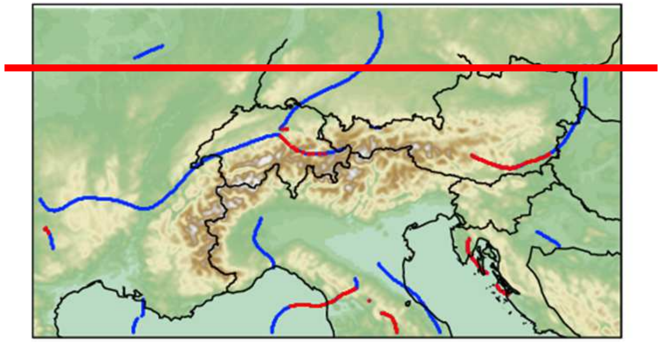
Looks better now, but steepness of the detection surface still matters

Lookback

- **Cross-sections**

- CCLM (3 km grid spacing) driven by IFS;

  TKE-SV scheme with shallow conv.



September 18, 2007

4400 m

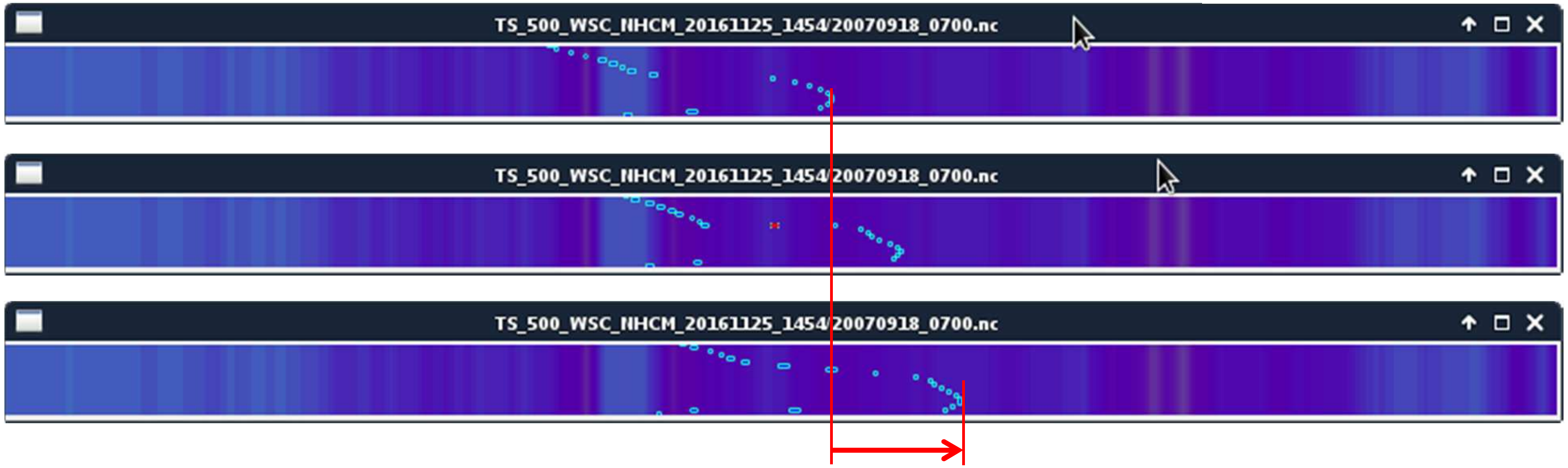
400 m

4400 m

400 m

4400 m

400 m



07:00 UTC

08:00 UTC

09:00 UTC

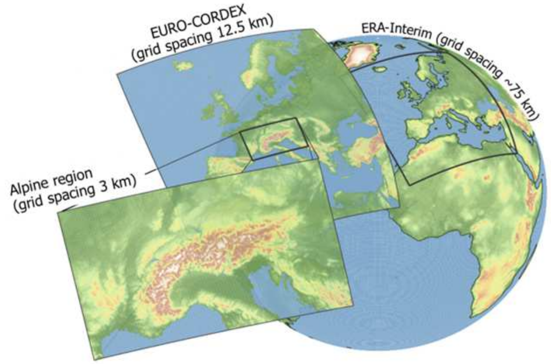
~100 km in 2 hours

Difficulty: fronts are very flat (slope <15°) ➔ definition of the detection surface?

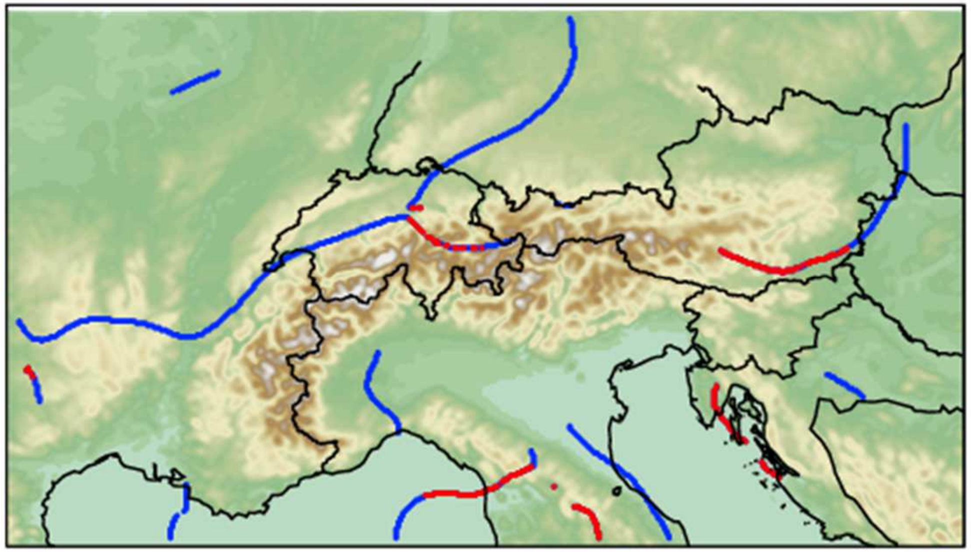
and move very fast (~50 km/h) ➔ observational data?

Lookback

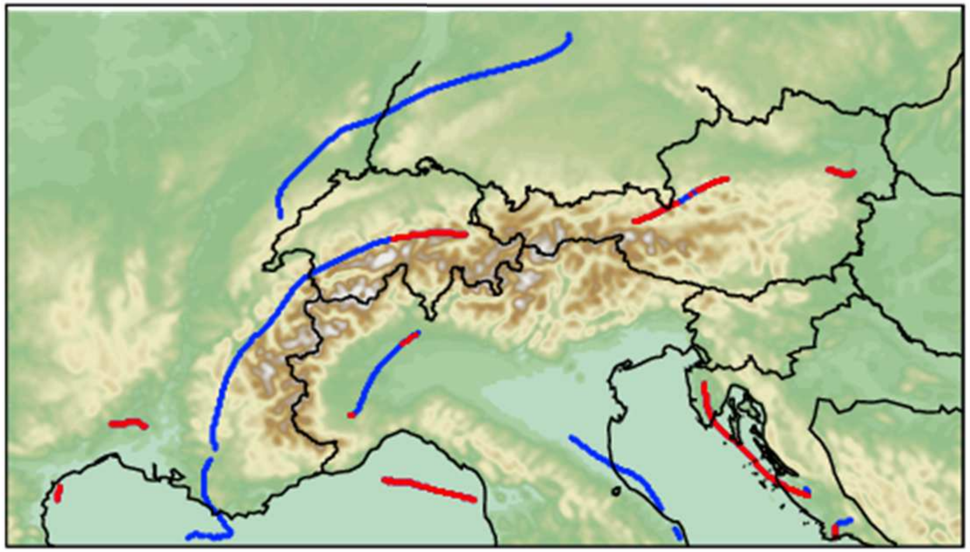
- **“Incomparability” with observational date due to model decoupling in large domains**



Event at September 18, 2007, 08:00 UTC



CCLM (3 km grid spacing) driven by IFS;
TKE-SV scheme with shallow conv.



CCLM (3 km grid spacing) driven by CCLM EUR-11;
default TKE scheme with shallow conv.

How to evaluate the front detection algorithm?

- **Ultimate goal:**
  The front detection algorithm should be used for model evaluation

- **Requirements:**
- **(1) The algorithm should give similar results when compared to other front detection methods.**
  ➔ method inter-comparison; weather maps?
- **(2) The algorithm should detect the same front line in different models (also with different resolutions), when they simulate the same weather situation.**
  ➔ apply the method on different models (with different grids) that simulate the same weather situation
- **European domain and in the Alpine region**

How to evaluate the front detection algorithm?

- **But:**
- **How to define filter settings (blocked wave lengths for low-pass filter)?**
  ➔ Cyclone Database?
- **How to derive detection surface to avoid artificial fronts?**
  ➔ change smoothness of surface in long term simulations until "stationary" fronts disappear?
- **Where to find front lines (or other proxies) to evaluate the algorithm?**
  ➔ radio-soundings?

How to evaluate the front detection algorithm?

- **Comparison with weather maps**

  - Masterthesis of Robert Ritter, Jenkner et al. (2010) (single events),

    Hope et al. (2014): correlation between number of rain days and fronts in the period 1979-2006 in Australia (weather maps are manually analysed)

    Lupikasza (2016): extreme precipitation events and their correlation with fronts (50 y period) (weather maps are manually analysed)

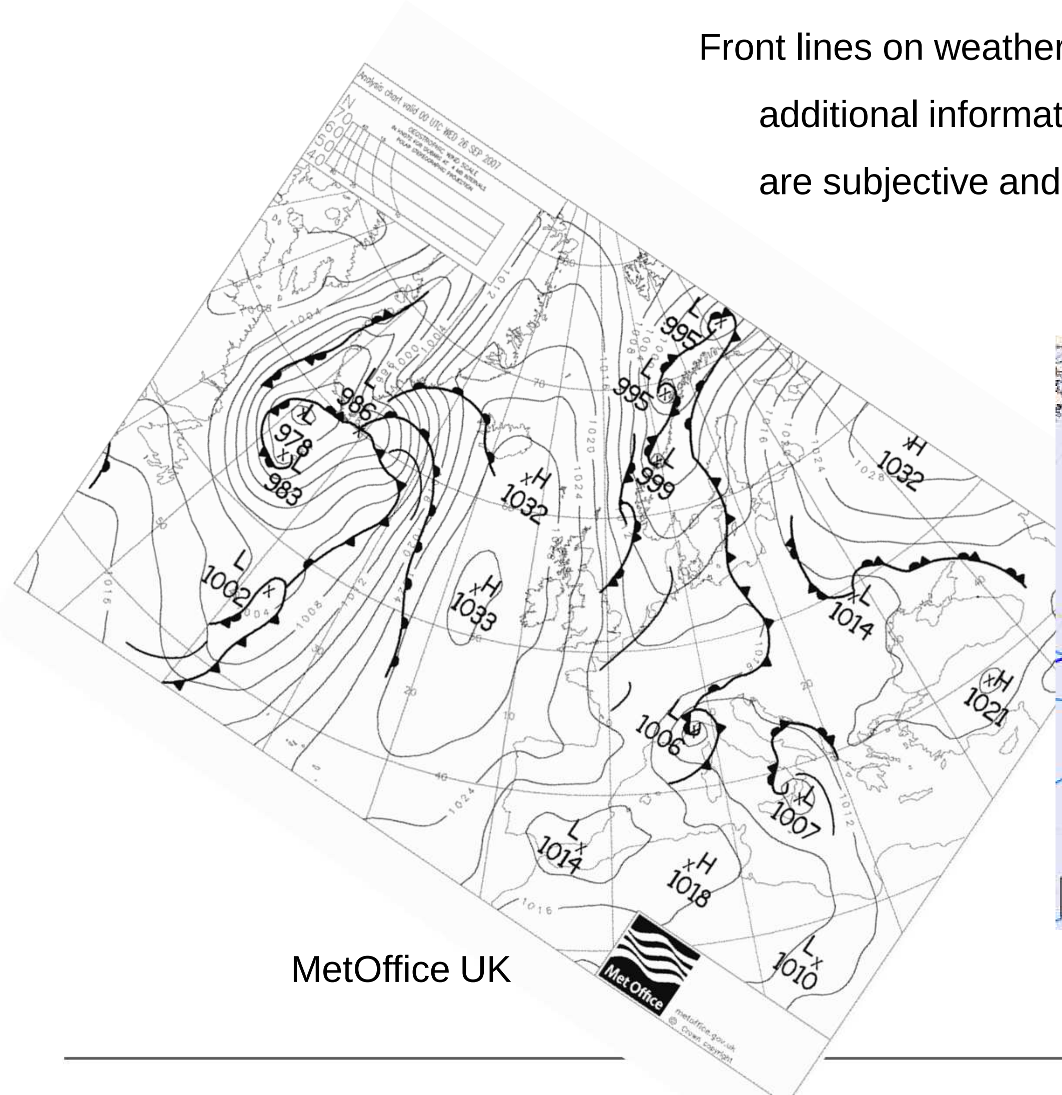
    Simmonds et al. (2012): produced climatology of fronts in the Southern Hemisphere based on ERA-Interim; no evaluation at all

    Catto et al. (2014): climate change effects on fronts; no evaluation at all

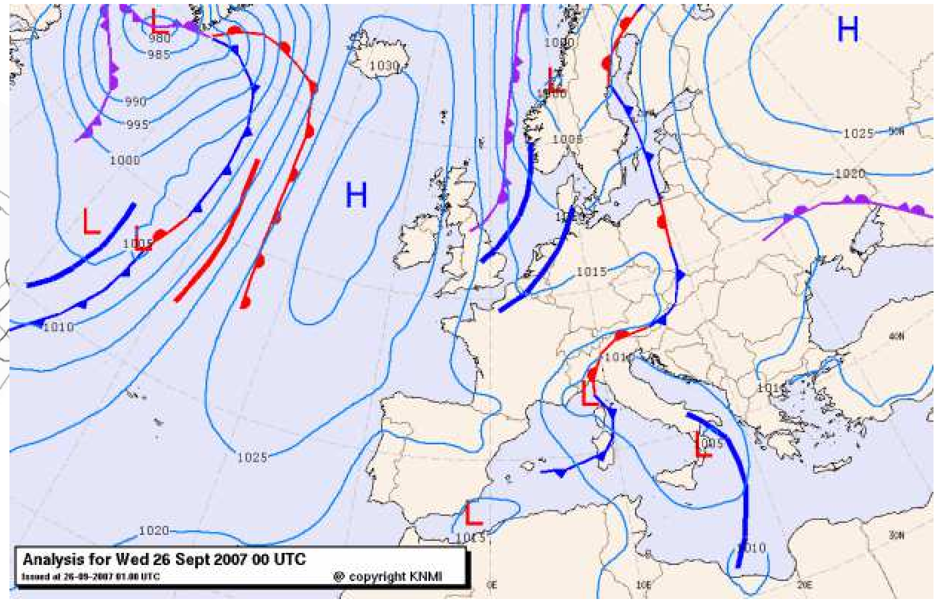
How to evaluate the front detection algorithm?

- **Comparison with weather maps**

Front lines on weather maps are manually drawn by meteorologists supported by additional information (e.g. TFP from model output and satellite images). The front lines are subjective and look different from different met services.



MetOffice UK



KNMI