On the downward continuation of gravitational gradients (GOCE-GDC project)

Novák[1] P., Sebera[1,2] J., Vaľko[1] M., Baur[3] O.

1. NTIS, University of West Bohemia, Czech Republic
2. Astronomical Institute, Academy of Sciences of the Czech Republic
3. Space Research Institute, Austrian Academy of Sciences, Austria

GGHS, Venice, Italy
www.gravityfield.org/conference/

9 - 12 October, 2012

Insight into upward/downward continuation

Why?

"The main advantage of enhancing potential field by downward continuation with respect to other derivative-based techniques is that the physical dimensions of the transformed field are the same as in the original data" (Fedi and Florio 2002).

But!

"Continuing downward, the functions become rougher. The transition from the flight level to the level of the Earth causes an amplification of the high-frequencies" (Hofmann-Wellenhof and Moritz 1986).

⇒ Upward cont. - acts as a low pass filter

⇒ Downward cont. - vice versa: usually **strong error amplification**

Insight into upward/downward continuation

Why?

"The main advantage of enhancing potential field by downward continuation with respect to other derivative-based techniques is that the physical dimensions of the transformed field are the same as in the original data" (Fedi and Florio 2002).

But!

"Continuing downward, the functions become rougher. The transition from the flight level to the level of the Earth causes an amplification of the high-frequencies" (Hofmann-Wellenhof and Moritz 1986).

⇒ Upward cont. - acts as a low pass filter

⇒ Downward cont. - vice versa: usually **strong error amplification**

Insight into upward/downward continuation

Why?

"The main advantage of enhancing potential field by downward continuation with respect to other derivative-based techniques is that the physical dimensions of the transformed field are the same as in the original data" (Fedi and Florio 2002).

But!

"Continuing downward, the functions become rougher. The transition from the flight level to the level of the Earth causes an amplification of the high-frequencies" (Hofmann-Wellenhof and Moritz 1986).

$\Rightarrow$ Upward cont. - acts as a low pass filter

$\Rightarrow$ Downward cont. - vice versa: usually **strong error amplification**

Basic options for continuation of potential data

- Gradient method - the function is developed into a Taylor series and continued via $f = f(0) + \frac{f'}{1!}dx + \frac{f''}{2!}dx^2 + \ldots$
  - \+ Very simple, flexible and very fast
  - − The derivatives $f', f'', \ldots$ must be known
  - − Usually applicable for small differentials (dx)
- Poisson integral - standard tool for continuation of potential data
  - \+ Rigorous (solution of the Dirichlet problem) and quite flexible (curve, plane, sphere, ellipsoid)
  - \+ Plane - fast, solvable via FFT
  - − Curve, Sphere, Ellipsoid - integral equation must be solved
- Other methods - global basis functions (sph. harmonics, ...), local (wavelets, ...), collocation, combined strategies (google for more)

Basic options for continuation of potential data

- Gradient method - the function is developed into a Taylor series and continued via $f = f(0) + \frac{f'}{1!}dx + \frac{f''}{2!}dx^2 + ...$
  + Very simple, flexible and very fast
  – The derivatives $f', f'', ...$ must be known
  – Usually applicable for small differentials (dx)
- Poisson integral - standard tool for continuation of potential data
  + Rigorous (solution of the Dirichlet problem) and quite flexible (curve, plane, sphere, ellipsoid)
  + Plane - fast, solvable via FFT
  – Curve, Sphere, Ellipsoid - integral equation must be solved
- Other methods - global basis functions (sph. harmonics, ...), local (wavelets, ...), collocation, combined strategies (google for more)

Basic options for continuation of potential data

- Gradient method - the function is developed into a Taylor series and continued via $f = f(0) + \frac{f'}{1!}dx + \frac{f''}{2!}dx^2 + \ldots$
  - \+ Very simple, flexible and very fast
  - – The derivatives $f', f'', \ldots$ must be known
  - – Usually applicable for small differentials (dx)
- Poisson integral - standard tool for continuation of potential data
  - \+ Rigorous (solution of the Dirichlet problem) and quite flexible (curve, plane, sphere, ellipsoid)
  - \+ Plane - fast, solvable via FFT
  - – Curve, Sphere, Ellipsoid - integral equation must be solved
- Other methods - global basis functions (sph. harmonics, ...), local (wavelets, ...), collocation, combined strategies (google for more)

Basic options for continuation of potential data

- Gradient method - the function is developed into a Taylor series and continued via $f = f(0) + \frac{f'}{1!}dx + \frac{f''}{2!}dx^2 + \ldots$
  - \+ Very simple, flexible and very fast
  - − The derivatives $f', f'', \ldots$ must be known
  - − Usually applicable for small differentials (dx)
- Poisson integral - standard tool for continuation of potential data
  - \+ Rigorous (solution of the Dirichlet problem) and quite flexible (curve, plane, sphere, ellipsoid)
  - \+ Plane - fast, solvable via FFT
  - − Curve, Sphere, Ellipsoid - integral equation must be solved
- Other methods - global basis functions (sph. harmonics, ...), local (wavelets, ...), collocation, combined strategies (google for more)

Basic options for continuation of potential data

- Gradient method - the function is developed into a Taylor series and continued via $f = f(0) + \frac{f'}{1!}dx + \frac{f''}{2!}dx^2 + ...$
  - \+ Very simple, flexible and very fast
  - – The derivatives $f', f'', ...$ must be known
  - – Usually applicable for small differentials (dx)
- Poisson integral - standard tool for continuation of potential data
  - \+ Rigorous (solution of the Dirichlet problem) and quite flexible (curve, plane, sphere, ellipsoid)
  - \+ Plane - fast, solvable via FFT
  - – Curve, Sphere, Ellipsoid - integral equation must be solved
- Other methods - global basis functions (sph. harmonics, ...), local (wavelets, ...), collocation, combined strategies (google for more)

Step 1: From real orbit to a mean sphere (gradient method)

- Why? It can simplify (speed up) any further GOCE data processing (gridding, SHA, ...)
- Why not? The GOCE orbit $\vec{r}$ is nearly circular: $d\vec{x} = \Delta\vec{r}$ is small ($|\Delta\vec{r}| \leq 16$ km)
- Preserves the time series
- $f' = \frac{dVij}{dr}$ can be computed with help of GOCE models
- Check: the data statistics/parameters should not change significantly

Step 1: From real orbit to a mean sphere (gradient method)

- Why? It can simplify (speed up) any further GOCE data processing (gridding, SHA, ...)
- Why not? The GOCE orbit $\vec{r}$ is nearly circular: $d\vec{x} = \Delta\vec{r}$ is small ($|\Delta\vec{r}| \leq 16$ km)
- Preserves the time series
- $f' = \frac{dVij}{dr}$ can be computed with help of GOCE models
- Check: the data statistics/parameters should not change significantly

Step 1: From real orbit to a mean sphere (gradient method)

- Why? It can simplify (speed up) any further GOCE data processing (gridding, SHA, ...)
- Why not? The GOCE orbit $\vec{r}$ is nearly circular: $d\vec{x} = \Delta\vec{r}$ is small ($|\Delta\vec{r}| \leq 16$ km)
- Preserves the time series
- $f' = \frac{dVij}{dr}$ can be computed with help of GOCE models
- Check: the data statistics/parameters should not change significantly

Step 1: From real orbit to a mean sphere (gradient method)

- Why? It can simplify (speed up) any further GOCE data processing (gridding, SHA, ...)
- Why not? The GOCE orbit $\vec{r}$ is nearly circular: $d\vec{x} = \Delta\vec{r}$ is small ($|\Delta\vec{r}| \leq 16$ km)
- Preserves the time series
- $f' = \frac{dVij}{dr}$ can be computed with help of GOCE models
- Check: the data statistics/parameters should not change significantly

Step 1: From real orbit to a mean sphere (gradient method)

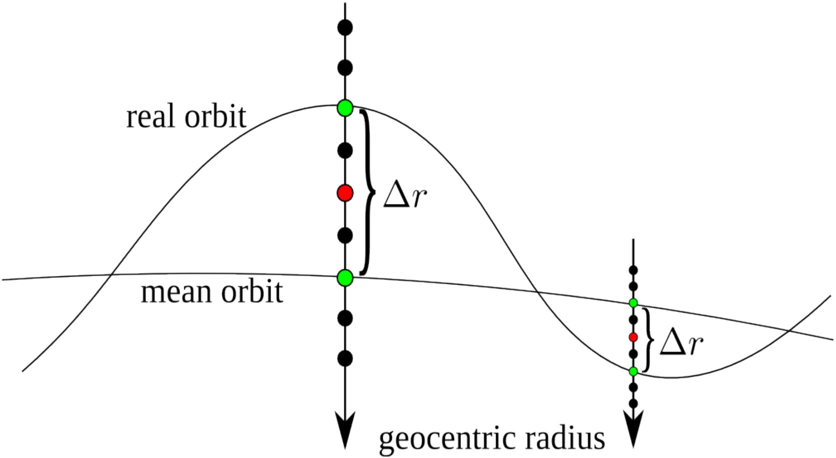
- Why? It can simplify (speed up) any further GOCE data processing (gridding, SHA, ...)
- Why not? The GOCE orbit $\vec{r}$ is nearly circular: $d\vec{x} = \Delta\vec{r}$ is small ($|\Delta\vec{r}| \leq 16$ km)
- Preserves the time series
- $f' = \frac{dVij}{dr}$ can be computed with help of GOCE models
- Check: the data statistics/parameters should not change significantly

We have used 9-point stencil for UD continuation of GOCE data

$$V_{ij}^{cont} = V_{ij}^{\mathrm{GOCE}} \pm \Delta r \frac{\mathrm{d}V_{ij}^{0}}{\mathrm{d}r}$$

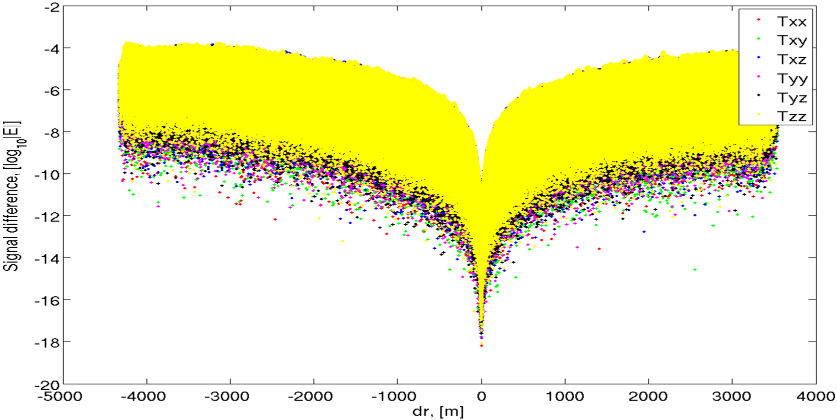
$$\frac{\mathrm{d}V_{ij}^{0}}{\mathrm{d}r} = \{aV_{ij}^{-4} - bV_{ij}^{-3} + cV_{ij}^{-2} - dV_{ij}^{-1} + eV_{ij}^{0} + dV_{ij}^{+1} - cV_{ij}^{+2} + bV_{ij}^{+3} - aV_{ij}^{+4}\}/\mathrm{d}r$$

with $a = \frac{1}{280}, b = \frac{4}{105}, c = \frac{1}{5}, d = \frac{4}{5}, e = 0$

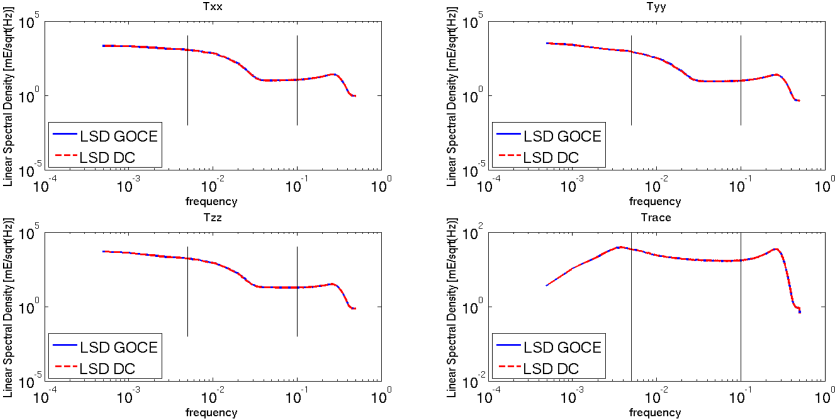


Simulation - height vs. accuracy ($\Delta r = 4dr$)

- Estimated when applied to synthetic data from GOCE-only model
- Minimal accuracy = 0.1 mE with RMS $\simeq 10^{-6}$ E
- $\Rightarrow$ GOCE data are lucky, they can be UD continued to a mean sphere with gradient method

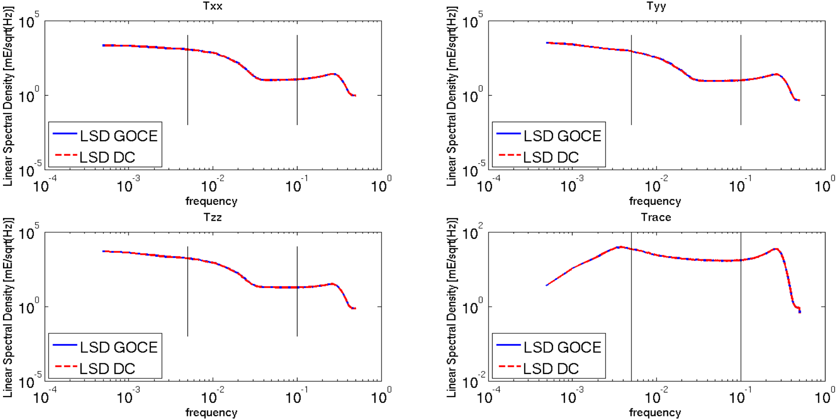


Real-data example - 2 months of data



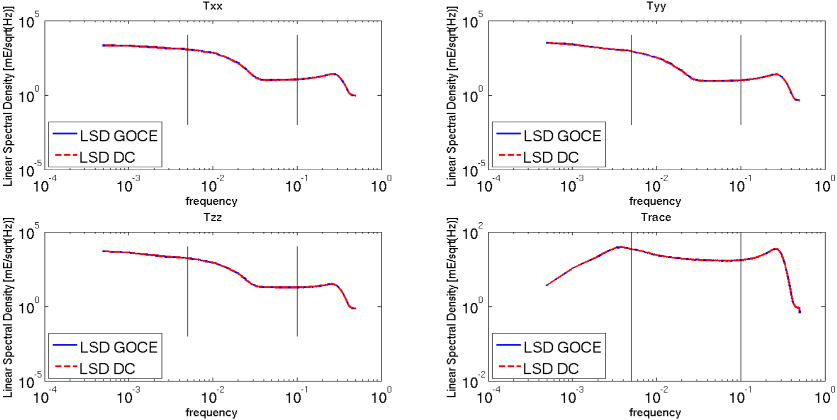
- SDs of the original GOCE (LNOF) and continued data almost the same
- ⇒ UDC from real orbit to mean sphere preserves spectral properties
- ⇒ Easily applicable to the GRF data

Real-data example - 2 months of data



- SDs of the original GOCE (LNOF) and continued data almost the same
- ⇒ UDC from real orbit to mean sphere preserves spectral properties
- ⇒ Easily applicable to the GRF data

Real-data example - 2 months of data



- SDs of the original GOCE (LNOF) and continued data almost the same
- ⇒ UDC from real orbit to mean sphere preserves spectral properties
- ⇒ Easily applicable to the GRF data

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$T(P) = \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma$$

$$T_z(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma$$

$$T_{zz}(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma$$

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$T(P) = \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma$$

$$T_z(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma$$

$$T_{zz}(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma$$

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$T(P) = \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma$$

$$T_z(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma$$

$$T_{zz}(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma$$

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$T(P) = \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma$$

$$T_z(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma$$

$$T_{zz}(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma$$

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$T(P) = \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma$$

$$T_z(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma$$

$$T_{zz}(P) = \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma$$

Step 2: UDC from mean sphere to zero level (Poisson integral)

- Gridding data from the previous step needed (we loose time series!)
- Regular grid is more suitable for surface quadratures
- We have adopted two strategies:
  1. **Iterative approach (Xu, 2007) based on spherical Poisson kernel**
  2. Direct integration with reciprocal spherical Poisson kernel (Novak, 2002)
- Poisson integrals used read $F_k = K_{k,l} F_l$

$$
\begin{aligned}
T(P) &= \frac{R(r^2 - R^2)}{4\pi} \iint_\sigma \frac{T(Q)}{l^3} d\sigma \\
T_z(P) &= \frac{R(r^2 - R^2)}{4\pi} \frac{R}{r} \iint_\sigma \frac{T_z(Q)}{l^3} d\sigma \\
T_{zz}(P) &= \frac{R(r^2 - R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{T_{zz}(Q)}{l^3} d\sigma
\end{aligned}
$$

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint\limits_{\sigma} \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi}\frac{R^2}{r^2}\iint\limits_{\sigma}\frac{V_{ij}(Q)}{l^3}d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - ⇒ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint\limits_{\sigma} \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}:
  $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint_\sigma \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint\limits_{\sigma} \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - ⇒ many degrees of freedom!

Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}:
  $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint_{\sigma} \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

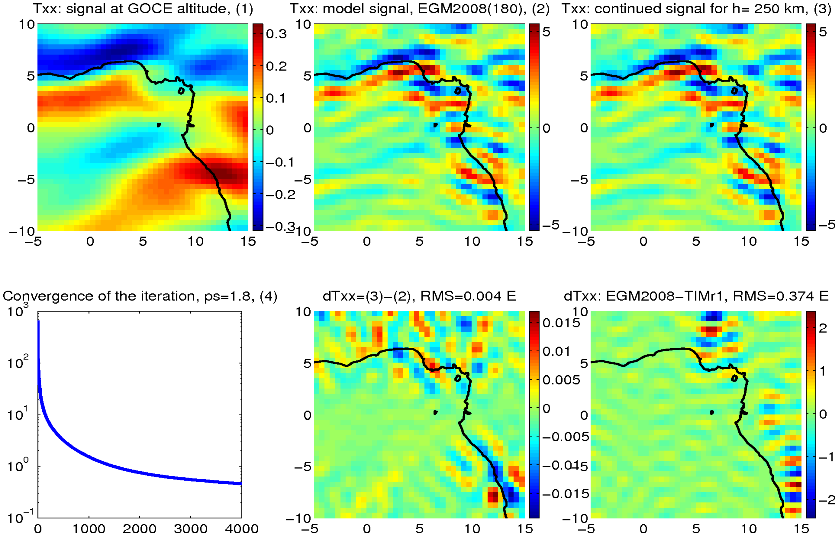
Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}:
  $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi} \frac{R^2}{r^2} \iint\limits_{\sigma} \frac{V_{ij}(Q)}{l^3} d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - “sensitivity” parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

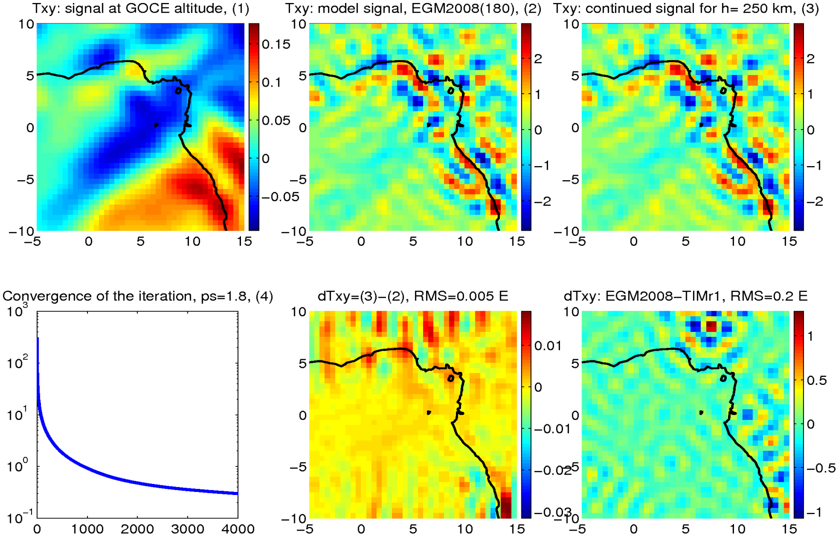
Iterative algorithm equipped by spherical Poisson integral

- Based on Xu et al. (2007, Geophysical prospecting) but on the sphere
- It applies the upward continuation until the residuals decrease $\delta = s_p(F_{\text{input}} - F_i)$
- It can be applied locally!
- We use the same kernel for all V_{ij}: $V_{ij}(P) = \frac{R(r^2-R^2)}{4\pi}\frac{R^2}{r^2}\iint\limits_{\sigma}\frac{V_{ij}(Q)}{l^3}d\sigma$
- Before doing so, set:
  - i - number of iterations
  - s_p - "sensitivity" parameter
  - tile/grid size wrt height
  - Algorithm can be split to ladder-like algorithm
  - $\Rightarrow$ many degrees of freedom!

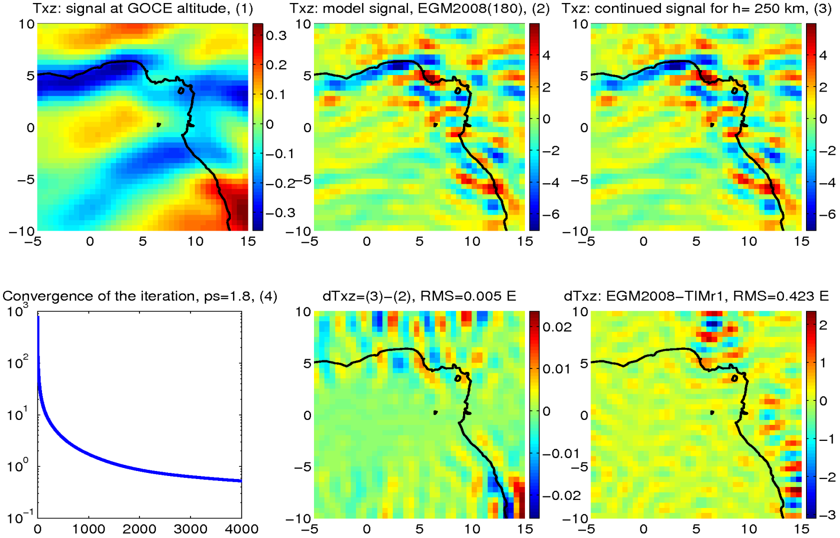
Iterative algorithm: noise-free example



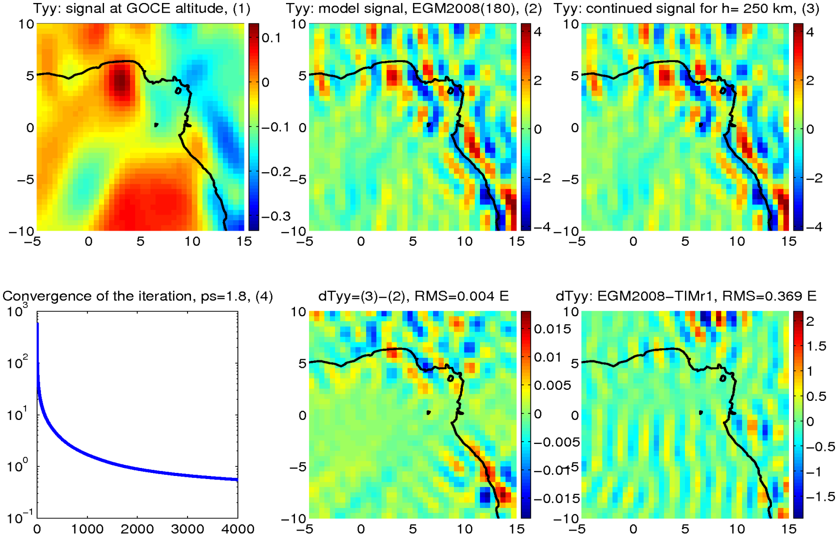
Iterative algorithm: noise-free example



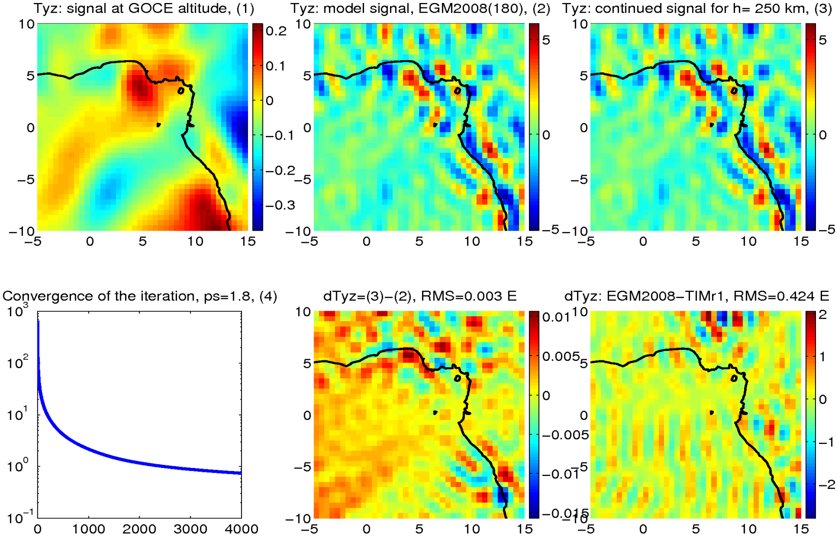
Iterative algorithm: noise-free example



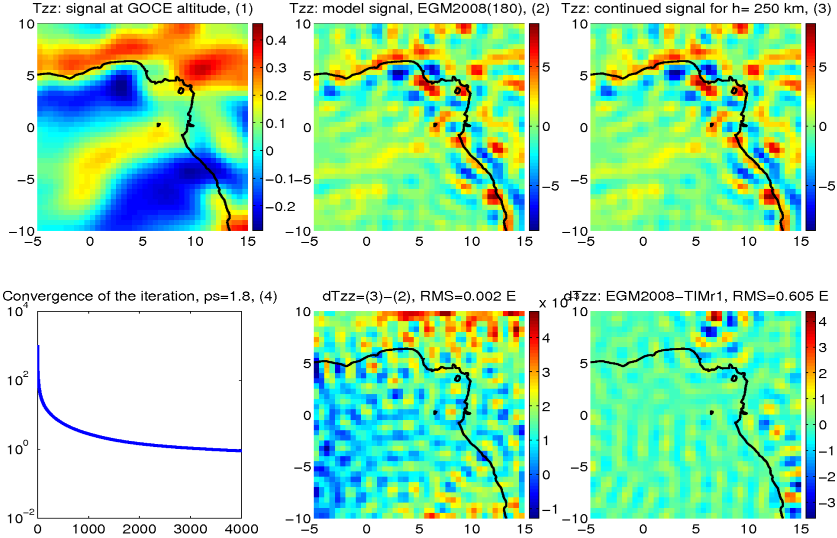
Iterative algorithm: noise-free example



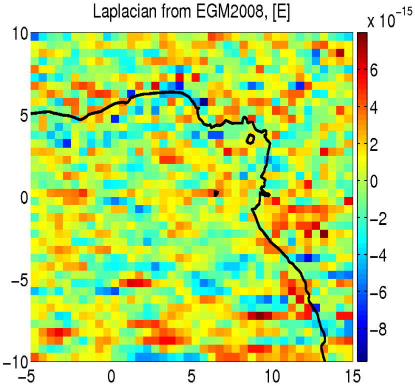
Iterative algorithm: noise-free example

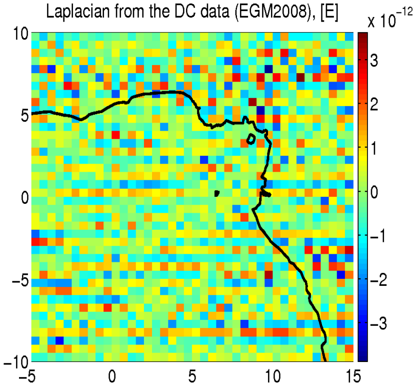


Iterative algorithm: noise-free example

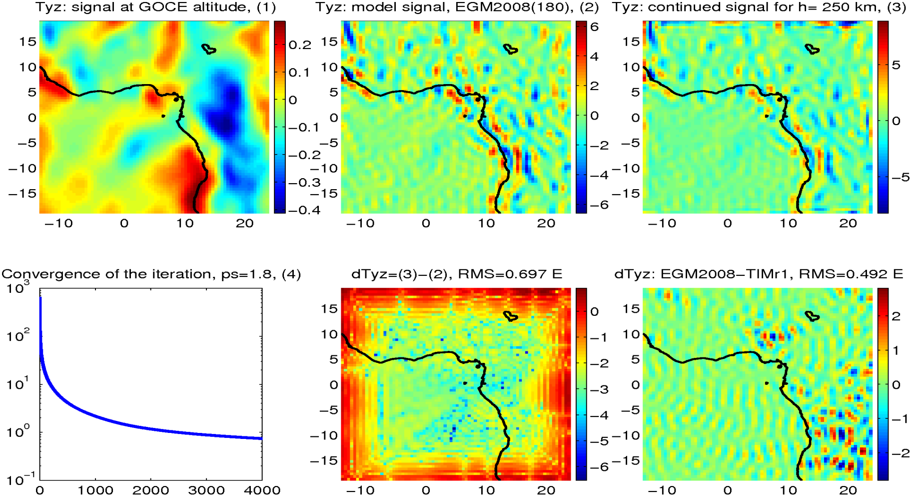


Iterative algorithm: Error amplification (check ΔT)



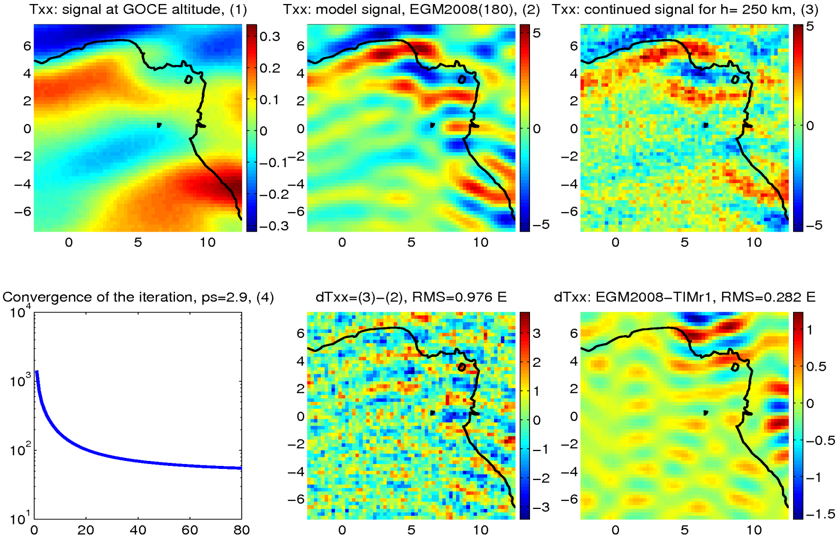


Iterative algorithm: noise-free example (with edge effect)



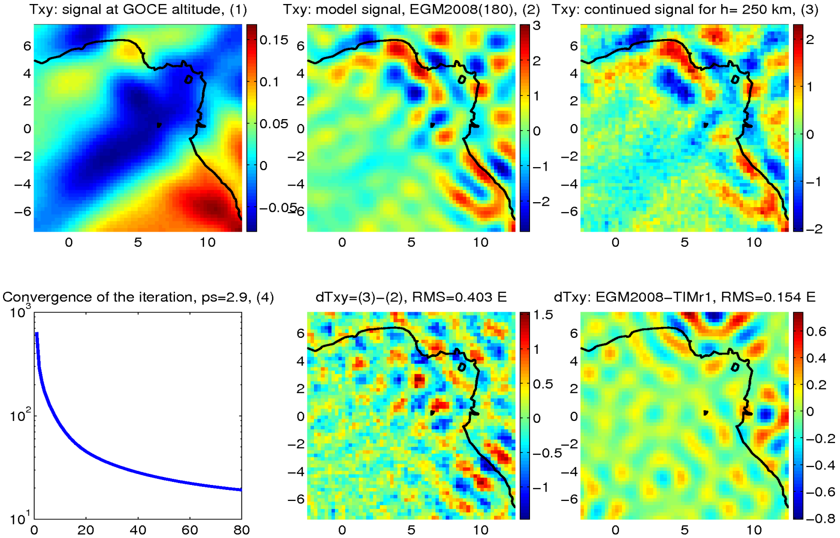
Iterative algorithm: real GOCE data

No filtering



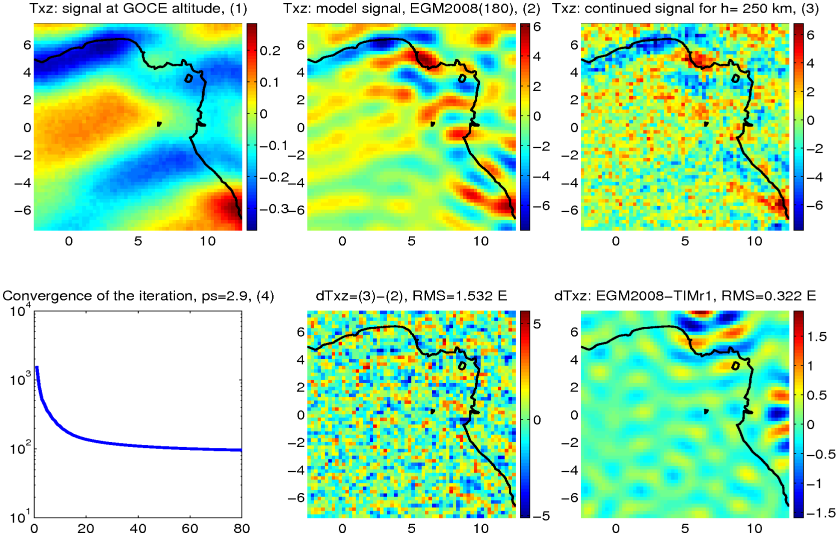
Iterative algorithm: real GOCE data

No filtering



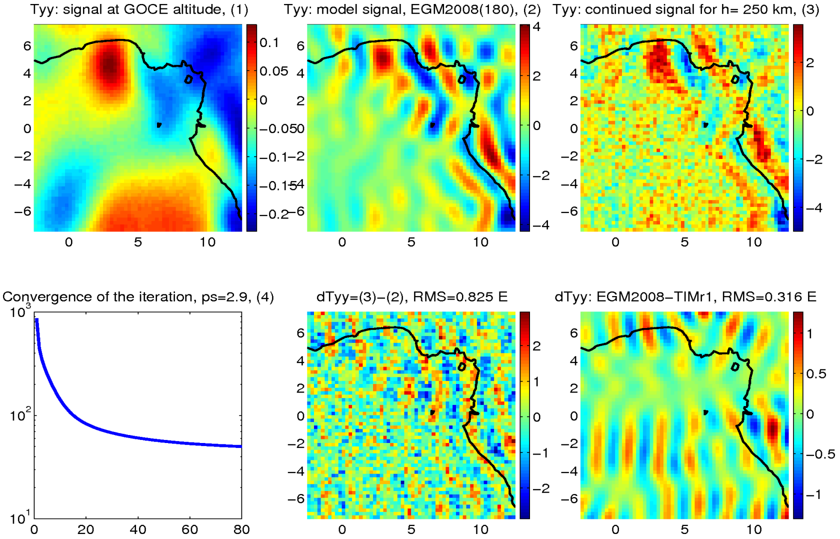
Iterative algorithm: real GOCE data

No filtering



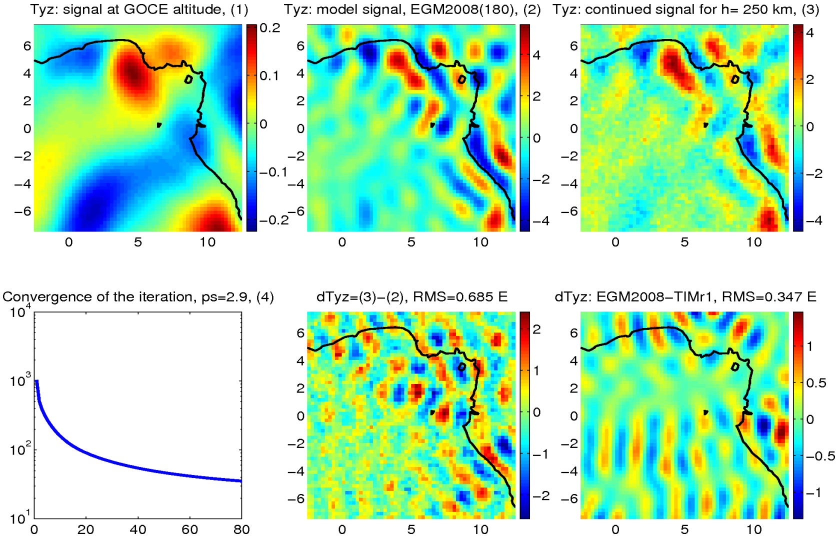
Iterative algorithm: real GOCE data

No filtering



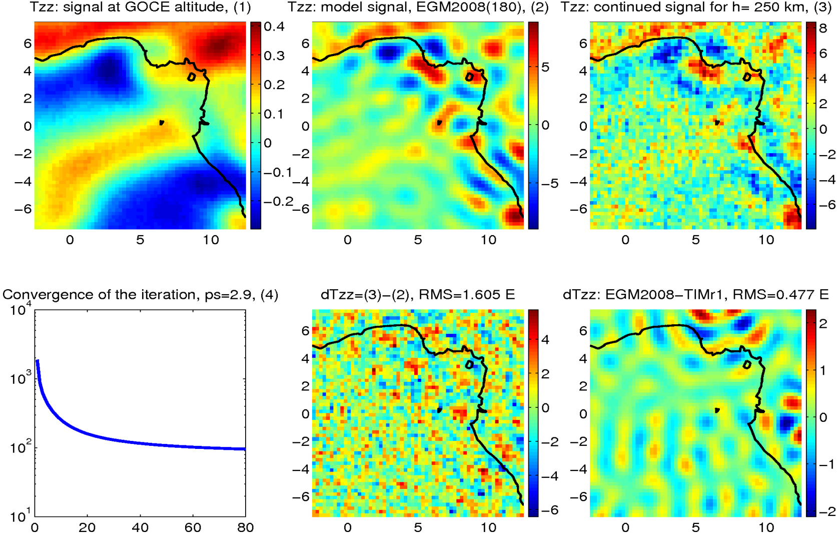
Iterative algorithm: real GOCE data

No filtering



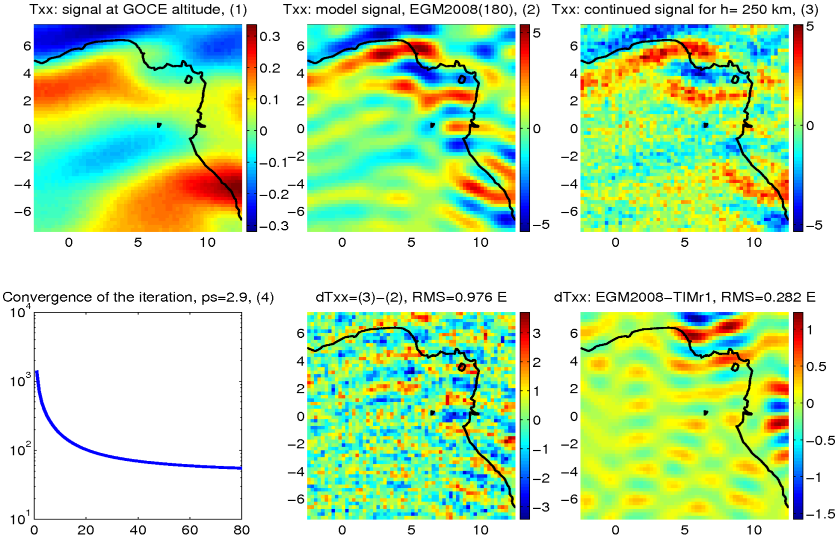
Iterative algorithm: real GOCE data

No filtering



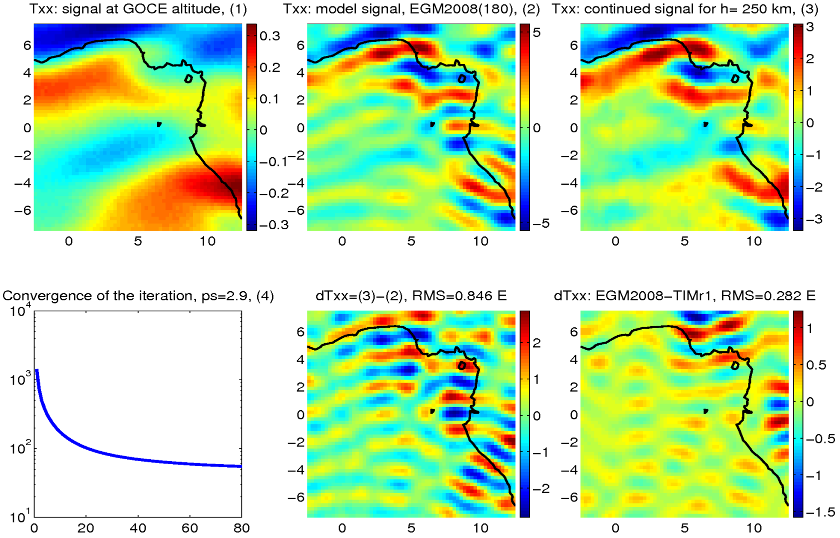
Iterative algorithm: real GOCE data

No filtering



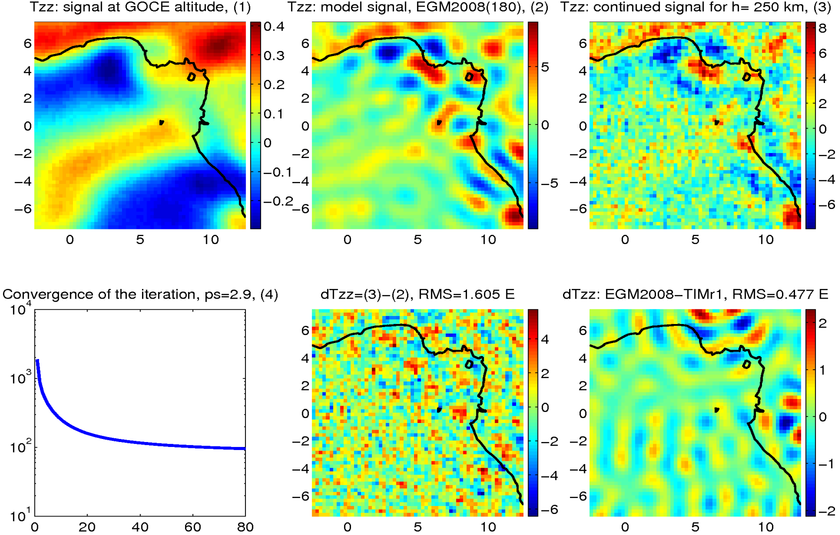
Iterative algorithm: real GOCE data

Filtered (anisotropic filter)



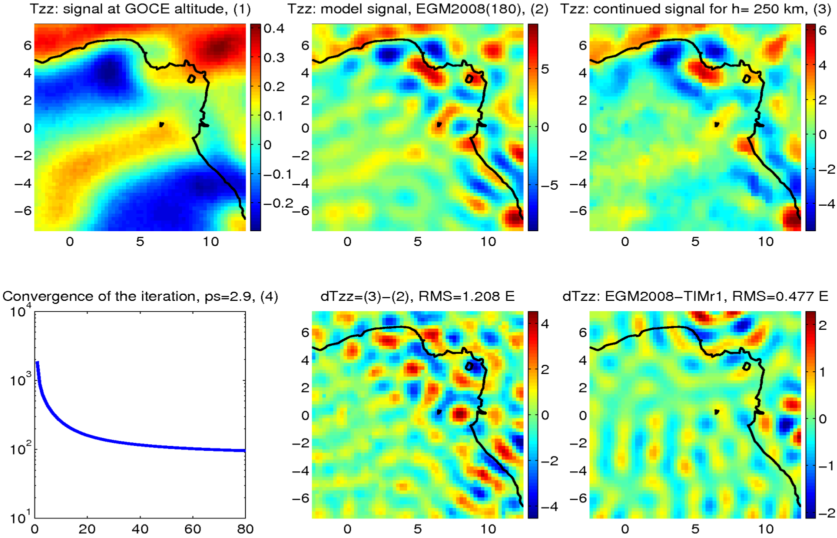
Iterative algorithm: real GOCE data

No filtering



Iterative algorithm: real GOCE data

Filtered (anisotropic filter)



Concluding remarks on downward continuation of GOCE data (1. From real to mean sphere)

- GOCE orbit: $\delta r \in \langle -16, 16 \rangle$ km
- Simple gradient method can be applied to data in any frame (GRF, LNOF)
- $\frac{dV_{ij}}{dr}$ can be computed from existing models
- Accuracy of GOCE data should “survive” this step
- Might less than 9-point stencil would also be satisfactory

Concluding remarks on downward continuation of GOCE data
(1. From real to mean sphere)

- GOCE orbit: $\delta r \in \langle -16, 16 \rangle$ km
- Simple gradient method can be applied to data in any frame (GRF, LNOF)
- $\frac{dV_{ij}}{dr}$ can be computed from existing models
- Accuracy of GOCE data should "survive" this step
- Might less than 9-point stencil would also be satisfactory

Concluding remarks on downward continuation of GOCE data (1. From real to mean sphere)

- GOCE orbit: $\delta r \in \langle -16, 16 \rangle$ km
- Simple gradient method can be applied to data in any frame (GRF, LNOF)
- $\frac{dV_{ij}}{dr}$ can be computed from existing models
- Accuracy of GOCE data should “survive” this step
- Might less than 9-point stencil would also be satisfactory

Concluding remarks on downward continuation of GOCE data
(1. From real to mean sphere)

- GOCE orbit: $\delta r \in \langle -16, 16 \rangle$ km
- Simple gradient method can be applied to data in any frame (GRF, LNOF)
- $\frac{dV_{ij}}{dr}$ can be computed from existing models
- Accuracy of GOCE data should "survive" this step
- Might less than 9-point stencil would also be satisfactory

Concluding remarks on downward continuation of GOCE data (1. From real to mean sphere)

- GOCE orbit: $\delta r \in \langle -16, 16 \rangle$ km
- Simple gradient method can be applied to data in any frame (GRF, LNOF)
- $\frac{dV_{ij}}{dr}$ can be computed from existing models
- Accuracy of GOCE data should “survive” this step
- Might less than 9-point stencil would also be satisfactory

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
– Required: gridding and LNOF (we loose time series!)
? Optimal DC empirical parameters must be found for a given tile and spacing
• Anisotropic filtering recommended
? **Still an open question until use of reprocessed/complete data**

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
– Required: gridding and LNOF (we loose time series!)
? Optimal DC empirical parameters must be found for a given tile and spacing
• Anisotropic filtering recommended
? **Still an open question until use of reprocessed/complete data**

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
– Required: gridding and LNOF (we loose time series!)
? Optimal DC empirical parameters must be found for a given tile and spacing
- Anisotropic filtering recommended

? **Still an open question until use of reprocessed/complete data**

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

- \+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
- \+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
- – Required: gridding and LNOF (we loose time series!)
- ? Optimal DC empirical parameters must be found for a given tile and spacing
- • Anisotropic filtering recommended
- ? **Still an open question until use of reprocessed/complete data**

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
– Required: gridding and LNOF (we loose time series!)
? Optimal DC empirical parameters must be found for a given tile and spacing
• Anisotropic filtering recommended
? Still an open question until use of reprocessed/complete data

Concluding remarks on downward continuation of GOCE data (2. From mean sphere to ground level)

+ DC of the GGs for $h = 250$ km is possible! Basic features recovered, for $N = 180$ we got an agreement at 0.5 - 1 E level (best RMS achieved for $N = 130 - 150$)
+ Algorithm performs even at mE level for noise-free data (tested with EGM2008)
– Required: gridding and LNOF (we loose time series!)
? Optimal DC empirical parameters must be found for a given tile and spacing
• Anisotropic filtering recommended
? **Still an open question until use of reprocessed/complete data**

Thank you!