Rapid wave model-based nearshore bathymetry inversion with UAS measurements


AGU Fall Meeting 2019

12/9/2019
Immediate understanding of bathymetry is crucial for coastal applications.

Several survey methods such as direct sampling and airborne Lidar are not always applicable.

Instead, easily measurable related quantities (e.g., imagery-based wave celerity) have been collected.

Then, physics-based model (e.g., STWAVE) can be used to relate indirect observations to bathymetry through inverse modeling/data assimilation.
Nearshore Bathymetry Estimation - Imagery Data Acquisition

Imagery data has been collected mostly from fixed tower-based platforms:

Recently, Unmanned Aircraft Systems (UAS) has been introduced (e.g., Holman et al., 2011):
UAS Survey on July 22, 2016 in Duck, NC

- UAS-derived imagery on a single flight along shoreline in the black box.
- CBathy and Structure-from-Motion (SfM) algorithms provide high-resolution wave celerity (blue dots) and beach topographic data (red dots).

---

1Brodie et al., 2019
We propose a flexible and fast bathymetry estimation framework utilizing:

1. low-cost commercial off-the-shelf UAS-based data acquisition
2. phase-averaging wave model: USACE's STWAVE
3. real-time batch-data inverse modeling approach, PCGA

Principal Component Geostatistical Approach, PCGA, performs scalable Hierarchical Bayesian inversion by approximating the covariance matrix with its dominant principal components.

---

1 Lee and Kitanidis, 2014, Lee et al, 2016, Lee et al., 2018
We propose a flexible and fast bathymetry estimation framework utilizing

1. low-cost commercial off-the-shelf UAS-based data acquisition
2. phase-averaging wave model: USACE’s STWAVE
3. real-time batch-data inverse modeling approach, PCGA¹

PCGA performs scalable Hierarchical Bayesian inversion:

- Modular (can be linked with any black-box nearshore models.)
- Jacobian-free
- Embarrassingly parallelizable
- Scalable: \( \mathcal{O}(100) \) model runs in total for \( > 10^7 \) unknowns/obs. through fast linear algebra/dimension reduction without much loss of accuracy.
- Insensitive to initial guess
- **Flexible prior assignment**: prior mean structure can be derived from parameteric models such as linear or Dean’s profiles.

¹Lee and Kitanidis, 2014, Lee et al, 2016, Lee et al., 2018
Public-domain Software for Reproducible Research

pyPCGA: Python interface for fast and scalable stochastic inversion¹

google pyPCGA!

Users can perform close-to-real-time bathymetry inversion on Jupyter notebook environment in two lines of code (after preprocessing steps, of course)

Several notebook examples combined with USACE’s STWAVE and AdH

¹https://github.com/jonghyunharrylee/pypcga
Results with Joint Inversion using Wave and Inland Data

- Compare the estimation result with direct bathymetry profiles surveyed near the UAS flight date.
- RMSE = 0.28 m within observation area (300 x 400 m)
- Converged in 3 iterations with \( \sim 150 \) STWAVE runs.
- 5 mins on a workstation equipped with 48 core Intel Xeon 8160 2.1 GHz.
Optimal measurement errors are determined through cross-validation/Bayesian hyperparameter estimation.

Wave celerity (via STWAVE-based inversion) and inland elevation (via Kriging interpolation) data were fitted well.
Estimated Bathymetry Profile along a Transect

Using wave celerity and inland elevation data

- Inversion results were not sensitive to initial guess assignments
- Direct surveyed profile is located within 95% credible interval.
Effect of Inland Elevation Data

(a) Estimate with wave celerity data alone

(b) with wave celerity and inland elevation

(c) Estimation uncertainty (std) with wave

(d) with wave celerity and inland elevation
Estimated Bathymetry Profiles with and without Inland Data

(a) Transect 1 with wave celerity
(b) with wave/inland data

(c) Transect 2 with wave celerity
(d) with wave/inland data

(e) Transect 3 with wave celerity
(f) with wave/inland data
Concluding Remarks

- With low-cost, multi-camera, multirotor UAS system, we expect close-to-real-time bathymetry imaging will be feasible in the near future.
- Our inversion method took only around 5 minutes on a modern workstation, within the UAS-based data collection duration.
- Estimated bathymetry profiles are remarkably close to the direct survey data (RMSE = 0.28 m) within the estimation credible interval due to the additional use of inland elevation data.
- We provide inversion software package for scientists and engineers.
- Future works:
  - will test with data sets with more severe weather conditions.
  - will implement with advanced wave models such as WaveWatch III and FunWAVE.

pyPCGA github link
Clone Me!
**AGU Fall Meeting 2019**

1. **EP43C-04** Thurs. 14:25 - 14:40 Moscone West - 3024, L3  
   Yizhou Qian: Applications of deep neural network to nearshore bathymetry with sparse measurements.

2. **EP53C-07** Fri. 13:40 - 15:40 Moscone South - eLightning Theater I  
   Mojtaba Forghani: Deep learning techniques for riverine bathymetry and flow velocity estimation bathymetry.

**Recent papers about bathymetry**


2. Lee et al., Riverine bathymetry imaging with indirect observations, *Water Resources Research*, 2018
References

- Ghorbanidehno, Lee, Farthing, Hesser, Kitanidis, and Darve, Efficient data assimilation algorithm for bathymetry application, *Journal of Atmospheric and Oceanic Technology*, 2019
- Lee and Kitanidis, Large-scale hydraulic tomography and joint inversion of head and tracer data using the principal component geostatistical approach (PCGA), *Water Resources Research*, 50(7), 2014
Inverse Problem in Hierarchical Bayesian Framework

Consider the measurement equation

\[ y_t = h(s_t) + v_t \quad v_t \sim \mathcal{N}(0, R_t) \]

\[ y_t := n_{\text{obs}} \times 1 \text{ noisy measurements} \]
\[ h := \text{forward model} \]
\[ s_t := n_{\text{unknowns}} \times 1 \text{ bathymetry} \]
\[ v_t := \text{measurement and model error} \]

- Need to account for the uncertainty in model and data
- Treat parameters as random variables
- Hierarchical Bayesian\(^1\) Geostatistical Approach\(^2\)

---

\(^1\)Gelman, Calin, and Stern, 2013; Kitanidis, 2010
\(^2\)Kitanidis, 1995
Geostatistical Approach

The posterior estimate $\hat{s}$ and covariance $\Gamma_{\text{post}}$:

$$\arg\min_{s,\beta} \frac{1}{2} \| y - h(s) \|^2_{\Gamma_{\text{noise}}^{-1}} + \frac{1}{2} \| s - X\beta \|^2_{\Gamma_{\text{prior}}^{-1}}$$

Algorithm  Bayesian geostatistical approach

1: while Not converged do
2: Solve the system of equations,

$$\begin{pmatrix}
J_k \Gamma_{\text{prior}} J_k^T + \Gamma_{\text{noise}} & J_k X \\
(J_k X)^T & 0
\end{pmatrix}
\begin{pmatrix}
\xi_{k+1} \\
\beta_{k+1}
\end{pmatrix} =
\begin{pmatrix}
y - h(s_k) + J_k s_k \\
0
\end{pmatrix}$$

where, the Jacobian $J = \frac{\partial h}{\partial s}|_{s=s_k}$

3: The update $s_{k+1} = X\beta_{k+1} + \Gamma_{\text{prior}} J_k^T \xi_{k+1}$
4: end while

5: $\Gamma_{\text{post}} = \Gamma_{\text{prior}} - \left( \Gamma_{\text{prior}} J_k^T X \right) \left( \begin{pmatrix}
J_k \Gamma_{\text{prior}} J_k^T + \Gamma_{\text{noise}} & J_k X \\
(J_k X)^T & 0
\end{pmatrix} \right) \left( \begin{pmatrix}
J_k \Gamma_{\text{prior}} \\
X
\end{pmatrix} \right)$
**Principal Component Geostatistical Approach (PCGA)**

<table>
<thead>
<tr>
<th>Method</th>
<th>Adjoint-based method</th>
<th>PCGA</th>
</tr>
</thead>
<tbody>
<tr>
<td># of simulation runs</td>
<td>$n_{\text{obs}} + 1$</td>
<td>$\kappa + 1$</td>
</tr>
<tr>
<td>matrix multiplication</td>
<td>$\mathcal{O}(n_{\text{obs}} n_{\text{unknowns}})$</td>
<td>$\mathcal{O}(n_{\text{unknowns}} \kappa)$</td>
</tr>
<tr>
<td>storage</td>
<td>$\mathcal{O}(n_{\text{obs}} n_{\text{unknowns}})$</td>
<td>$\mathcal{O}(n_{\text{obs}} \kappa)$</td>
</tr>
</tbody>
</table>

- $\kappa + 1$ simulation runs in each iteration
- $\kappa \sim O(100)$ or less for many problems in earth science
- Can handle large measurements (e.g., $10^7$ measurements)
- Easy to implement; treat multi-physics models as a “blackbox” like Ensemble-based methods
- Parallel executions

---

1Lee and Kitanidis, 2014
Computational Challenges

1. Computing and storing Covariance matrices are expensive!
   \[ \Gamma_{\text{prior}} J^\top, \quad J\Gamma_{\text{prior}} J^\top \]
   - \( n_{\text{unknowns}} + 1 \) number of forward model executions in each iteration
   - \( \mathcal{O}(n_{\text{unknowns}}^2) \) storage

2. Computing and storing the Jacobian and its products are expensive (e.g., \( n_{\text{obs}} \gg 10^6 \)).
   - \( n_{\text{observation}} + 1 \) number of forward model executions in each iteration
Summary: we employed

1. $O(n)$ fast linear algebra (e.g., $\mathcal{H}$-matrices and FMM) for decomposition of the prior covariance matrix.

2. Generalized Eigenvalue Decomposition to construct exact preconditioner of saddle-point matrix.