Flow Decomposition for Ocean Velocity Data Assimilation

Peter C Chu and Leonid Ivanov
Naval Postgraduate School
T.P. Korzhova, T.M. Margolina, O.V. Melnichenko
Marine Hydrophysical Institute
Sevastopol, Crimea, Ukraine
chu@nps.navy.mil

IUGG2003, Sapporo, Japan, June 30 – July 11, 2003

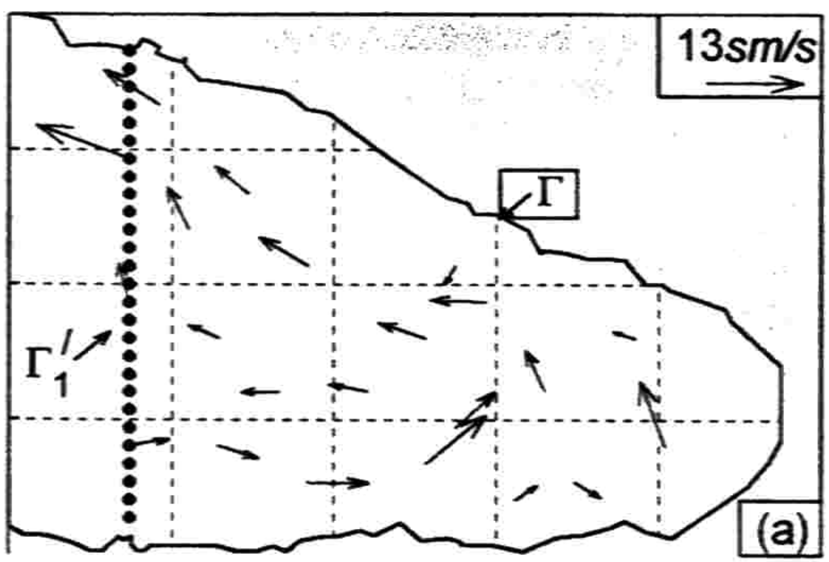
References

- Chu, P.C., L.M. Ivanov, T.P. Korzhova, T.M. Margolina, and O.M. Melnichenko, 2003: Analysis of sparse and noisy ocean current data using flow decomposition. Part 1: Theory. Journal of Atmospheric and Oceanic Technology, 20 (4), 478-491.

- Chu, P.C., L.M. Ivanov, T.P. Korzhova, T.M. Margolina, and O.M. Melnichenko, 2003: Analysis of sparse and noisy ocean current data using flow decomposition. Part 2: Application to Eulerian and Lagrangian data. Journal of Atmospheric and Oceanic Technology, 20 (4), 492-512.

Can we get the velocity signal from sparse and noisy data?

- Black Sea



- How can we assimilate sparse and noisy velocity data into numerical model?

Flow Decomposition

- 2 D Flow (Helmholtz)

  $u = - \partial \psi / \partial y + \partial \varphi / \partial x$

  $v = \partial \psi / \partial x + \partial \varphi / \partial y$

- 3D Flow (Toroidal & Poloidal): Very popular in astrophysics

- $$\mathbf{u} = \mathbf{r} \times \boldsymbol{\nabla} A_1 + \mathbf{r}\mathbf{A}_2 + \boldsymbol{\nabla} A_3$$

3D Incompressible Flow

- If Incompressible $\nabla \bullet \mathbf{u} = 0$

- We have

$$\mathbf{u} = \boldsymbol{\nabla} \times (\mathbf{r}\Psi) + \boldsymbol{\nabla} \times \boldsymbol{\nabla} \times (\mathbf{r}\Phi) \,.$$

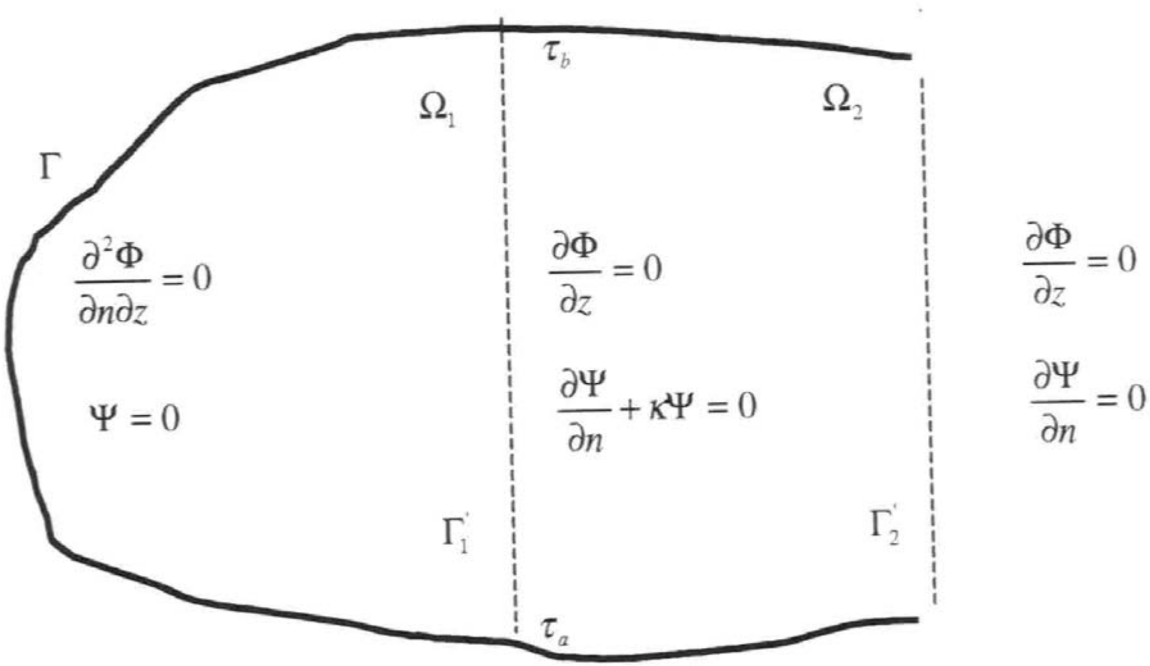
Flow Decomposition

$$u = \frac{\partial \Psi}{\partial y} + \frac{\partial^2 \Phi}{\partial x \partial z}, \qquad v = -\frac{\partial \Psi}{\partial x} + \frac{\partial^2 \Phi}{\partial y \partial z},$$

$\nabla^2 \Psi$ = - ζ, ζ is relative vorticity

$\nabla^2 \Phi$ = - w

Boundary Conditions



Basis Functions

$$\Psi(x, y, z, t^\circ) = \sum_{k=1}^{\infty} a_k(z, t^\circ)\Psi_k(x, y, z, \kappa^\circ),$$

$$\frac{\partial\Phi(x, y, z, t^\circ)}{\partial z} = \sum_{m=1}^{\infty} b_m(z, t^\circ)\Phi_m(x, y, z),$$

Determination of Basis Functions

- Poisson Equations
- Γ - Rigid Boundary
- Γ' – Open Boundary
- $\{\lambda_k\}$, $\{\mu_m\}$ are Eigenvalues.

- Basis Functions are predetermined.

$$\Delta \Psi_k = -\lambda_k \Psi_k,$$

$$\Delta \Phi_m = -\mu_m \Phi_m,$$

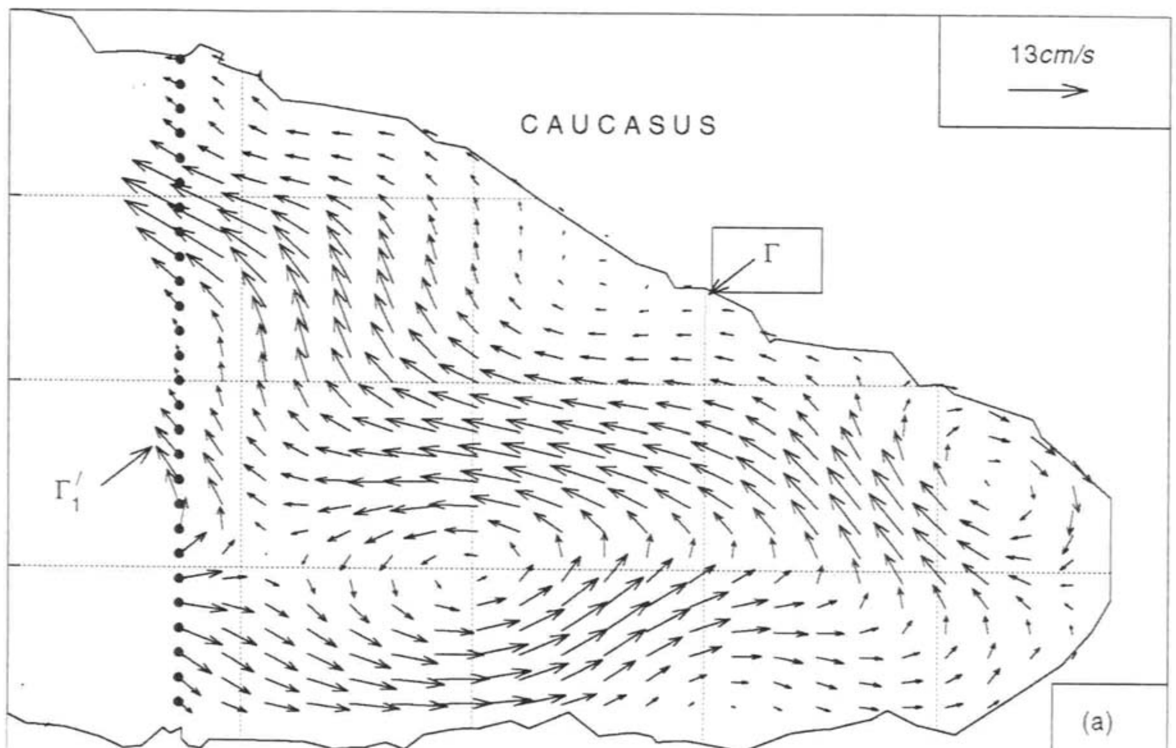
$$\Psi_k|_\Gamma = 0, \quad \frac{\partial \Phi_m}{\partial n}|_\Gamma = 0,$$

$$\left[\frac{\partial \Psi_k}{\partial n} + \kappa(\tau)\Psi_k\right]|_{\Gamma'} = 0, \quad \Phi_m|_{\Gamma'} = 0.$$

Flow Reconstruction

$$
\begin{aligned}
u_{KM} &= \sum_{k=1}^{K} a_k(z,t^\circ)\frac{\partial \Psi_k(x,y,z,\kappa^\circ)}{\partial y} + \sum_{m=1}^{M} b_m(z,t^\circ)\frac{\partial \Phi_m(x,y,z)}{\partial x}, \\
v_{KM} &= -\sum_{k=1}^{K} a_k(z,t^\circ)\frac{\partial \Psi_k(x,y,z,\kappa^\circ)}{\partial x} + \sum_{m=1}^{M} b_m(z,t^\circ)\frac{\partial \Phi_m(x,y,z)}{\partial y}
\end{aligned}
$$

Reconstructed Circulation



Conclusions

- Reconstruction is a useful tool for processing real-time velocity data with short duration and limited-area sampling.
- The scheme can handle highly noisy data.
- The scheme is model independent.
- The scheme can be used for assimilating sparse velocity data