Multichannel OLA Inversion for Local Helioseismology

Jason Jackiewicz

Assistant Professor, New Mexico State University
Department of Astronomy

Laurent Gizon[1], **Aaron C. Birch**[2], and **Shravan Hanasoge**[1]

[1]Max Planck Institute for Solar System Research, Katlenburg-Lindau, Germany
[2]NWRA CoRA Division, Boulder, USA

1. INTRODUCTION

The inverse problem for helioseismology is to use measurements (e.g., frequency splittings, travel-time shifts, etc.) to infer sub-surface conditions. Typically, helioseismic inversions are linear, whereby small perturbations to a reference solar model are presumably linearly related to changes in the measurements. There are traditionally two main inversion approaches: Optimally Localized Averages (OLA) [1] and Regularized Least Squares (RLS) [2]. We will focus on OLA methods, which aim to provide a spatially-localized estimate of some solar interior quantity while simultaneously controlling the random noise.

Any small-amplitude perturbation δq to a solar reference model can be linearly related to a helioseismic measurement (e.g. travel-time shifts $\delta\tau$) according to

$$\delta\tau(\boldsymbol{r}_1,\boldsymbol{r}_2) = \sum_\alpha \int K^\alpha(\boldsymbol{r}_1,\boldsymbol{r}_2,\boldsymbol{r},z)\delta q^\alpha(\boldsymbol{r},z)\,\mathrm{d}\boldsymbol{r}\mathrm{d}z + \text{noise},$$

for all sub-surface perturbations α [3]. The points $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are two observation points taken at the solar surface between which the waves propagate. The K are sensitivity kernels. For any inversion, we would like to estimate δq^α given $\delta\tau$, K, and the noise. It turns out that for typical problems, the three-dimensional (3D) OLA approach is computationally impractical due to the size of the matrices involved [4].

Here we show some examples of a new Fourier OLA method for helioseismology (Jackiewicz et al., in preparation) that is much more efficient than its real-space counterpart. The reason for this efficiency is that (1) one need only invert many small matrices rather than several large ones, and (2) each matrix corresponds to a wavenumber, $\boldsymbol{k}$, that is independent from all other wavenumbers and can therefore be parallelized for computation. This method will be invaluable when we eventually will need to perform inversions considering all possible perturbations simultaneously, such as flows, sound speed, magnetic field, density, etc.

2. CROSS TALK IN INVERSIONS

Consider an inversion for the x-component of velocity of real solar flows v_x at a target depth z_0. Our inferred quantity, denoted u_x, will almost certainly be contaminated by the other two velocity components: this is known as "cross talk." To see how this occurs, first let the averaging kernel $\mathcal{K}$ be a convolution of the inversion weights w and the sensitivity kernels K:

$$\mathcal{K}^x(\boldsymbol{r},z;z_0) = \sum_{i,a} w_a(\boldsymbol{r}_i;z_0)K_a^x(\boldsymbol{r}-\boldsymbol{r}_i,z),$$

and the same for the y and z components too. An inversion procedure produces the inversion weights w, which ultimately are convolved with the travel times to obtain a final answer. Using the definition of the travel-time shifts from above, where a denotes a type of measurement and n_a is the noise in that measurement, one sees that

$$\begin{aligned}
u_x(\boldsymbol{r};z_0) &= \sum_{i,a=\text{oi,we,ns}} w_a(\boldsymbol{r}_i-\boldsymbol{r};z_0)\delta\tau_a(\boldsymbol{r}_i) \\
&= \sum_{i,a} w_a(\boldsymbol{r}_i-\boldsymbol{r};z_0)\int \mathrm{d}\boldsymbol{r}'\mathrm{d}z \sum_{\alpha=x,y,z} K_a^\alpha(\boldsymbol{r}'-\boldsymbol{r}_i,z)v_\alpha(\boldsymbol{r}',z) \\
&\quad + \sum_{i,a} w_a(\boldsymbol{r}_i-\boldsymbol{r};z_0)n_a(\boldsymbol{r}_i) \\
&= \int \mathrm{d}\boldsymbol{r}'\mathrm{d}z\ \big[\mathcal{K}^x(\boldsymbol{r}'-\boldsymbol{r},z;z_0)v_x(\boldsymbol{r}',z) \\
&\quad + \mathcal{K}^y(\boldsymbol{r}'-\boldsymbol{r},z;z_0)v_y)(\boldsymbol{r}',z) + \mathcal{K}^z(\boldsymbol{r}'-\boldsymbol{r},z;z_0)v_z(\boldsymbol{r}',z)\big] \\
&\quad + \sum_{i,\alpha} w_a(\boldsymbol{r}_i-\boldsymbol{r};z_0)n_a(\boldsymbol{r}_i).
\end{aligned}$$

Note how the inferred quantity we compute, u_x, is a convolution of the x-component of the averaging kernel with the real solar flows v_x, **PLUS** contributions from the other two components and the noise. If the y and z components of the averaging kernels are nonzero, our answer is infected. In practice we constrain the other components to have a total integral of zero. It is up to the inversion procedure to minimize the damage from unwanted effects of other perturbations.

Figures 1 and 2 show examples of cross-talk in an OLA inversion (computed in Fourier space) for v_x and one for v_z.

Inversion for v_x

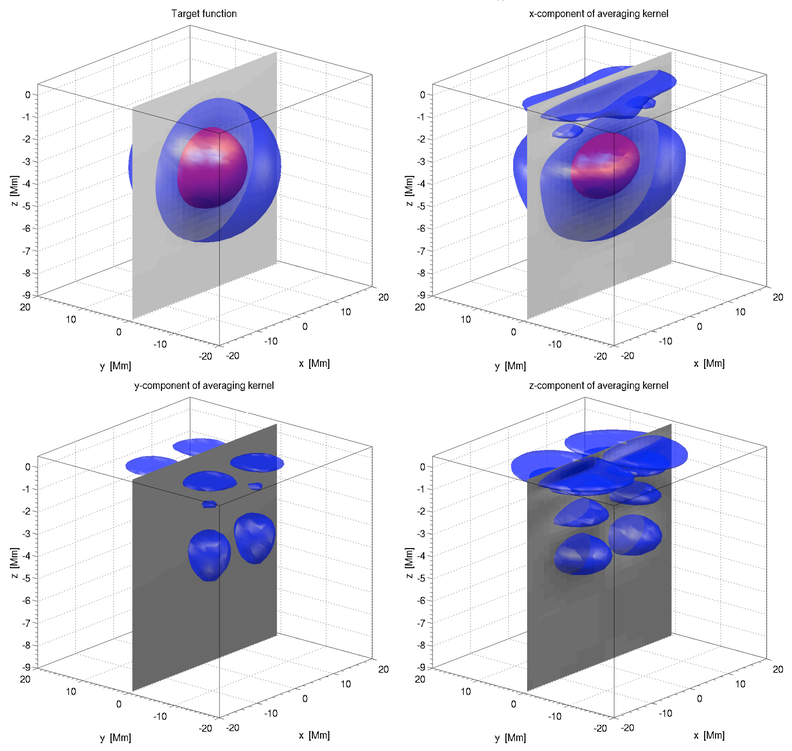

Figure 1: Example inversion for inferring the x-component of velocity, v_x, 3.5 Mm beneath the surface. Above are shown the three components of the averaging kernel, as well as the target function. One hopes to have a nice averaging kernel for the x component of velocity being inverted for (one which closely matches the target function), and null averaging kernels for the other two components y and z. That is almost the case here. The red color represents the half-maximum value of the target function (50%) whereas the blue is an isosurface of $\pm$5% of the maximum value. 72×3 kernels went into this inversion, as can be understood from the values in Table 1. The noise is about 30 m/s over 1 day.

Inversion for v_z

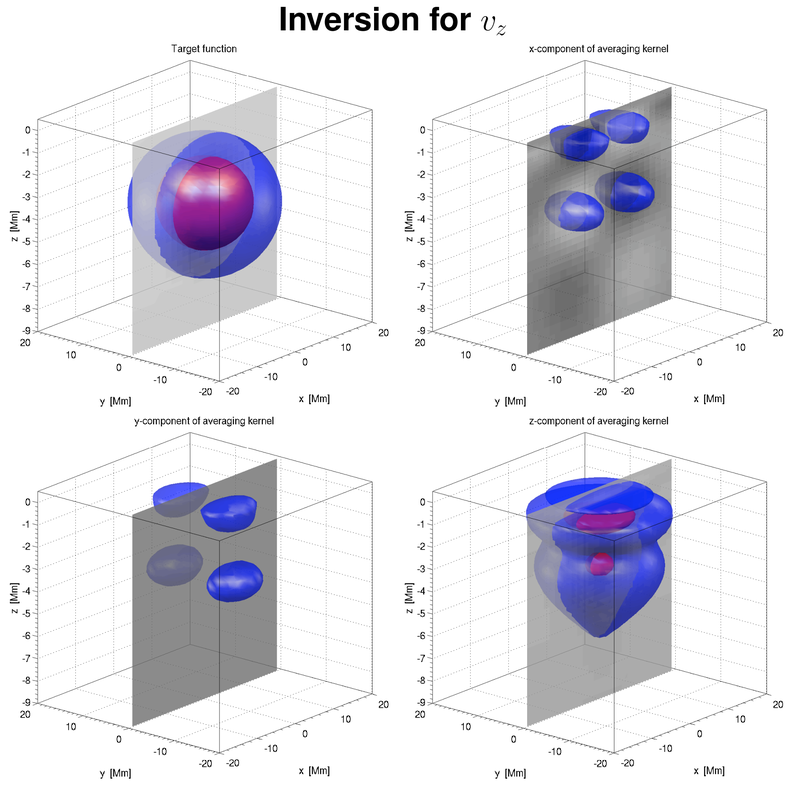

Figure 2: Similar to Figure 1, except now we attempt an inversion for inferring the z-component of velocity, v_z, 3.5 Mm beneath the surface. Above are shown the three components of the averaging kernel, as well as the target function. One hopes to have a nice averaging kernel for the z component of velocity being inverted for (one which closely matches the target function), and null averaging kernels for the other two components x and y. This case is not as successful as the first case - there is more structure near the surface in the averaging kernel, but the cross-talk is again minimal. The red color represents the half-maximum value of the target function (50%) whereas the blue is an isosurface of $\pm$5% of the maximum value. The noise is about 30 m/s over 1 day.

Table 1: Ridges, distances, and geometries used in the example flow inversions in Figures 1 and 2. For each velocity component, 72 kernels are used for a total of 216.

Ridges	Distances $\triangle$ (Mm)	Geometries
	7.3	
f	10.2	Out-In
	13.1	
	16.0	
p_1	19.0	West-East
	21.9	
p_2	24.8	North-South
	27.7	

3. EXAMPLE 3D SOUND-SPEED INVERSION COMPARISON

We now show a comparison of a real-space OLA inversion with its Fourier-space counterpart in a simple toy example. An inversion for sound speed is computed considering only two measurements and thus two sensitivity kernels (point-to-point, shown in Figure 4). Trade-off, or "L" curves and computation times of the two inversions are shown in Figure 3. The inversions agree only when the convolution grid for the real-space inversion is of maximum size (see Figure 3). The convolution is unnecessary in Fourier space - it is simply a multiplication. Note the dramatic speed-up for the Fourier inversion (due to smaller matrices in this case), which is even greater for larger problems since, as stated in the introduction, each wavenumber $\boldsymbol{k}$ is independent from the others and leads to complete parallelization.

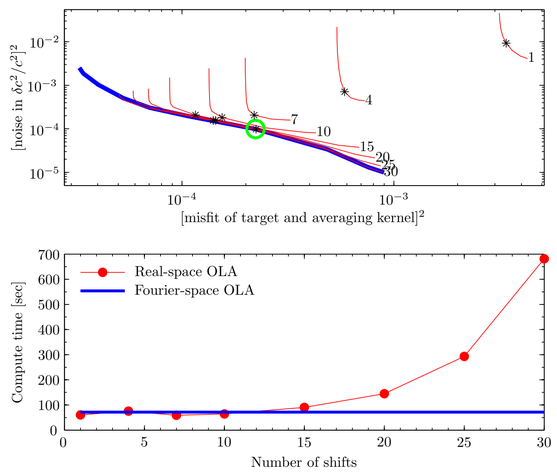

Figure 3: Trade-off curves (top panel) for a toy inversion for sound speed using only two measurements (kernels). The red curves are for multiple real-space inversions using a larger and larger convolution grid (from top right to bottom left). The blue curve shows the Fourier inversion, where no convolution is necessary. For the maximum size convolution grid, the two solutions agree as expected. The bottom panel shows the computation time for each run. To get the same answer in both methods, the real-space inversion takes almost an order of magnitude longer in this example. See Figure 4 for the kernels and convolution grid.

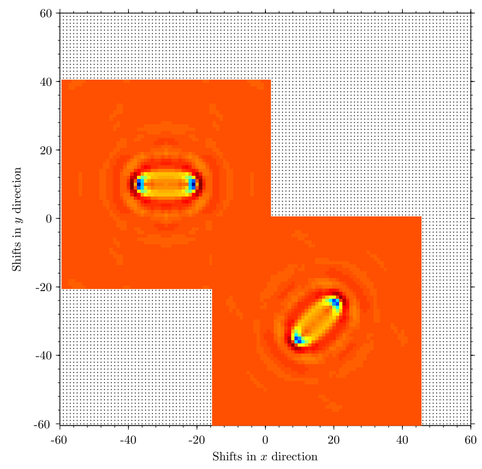

Figure 4: Illustration of a snapshot of the "convolution" grid necessary for the real-space OLA inversion. This example shows two sound-speed kernels used in the example in Figure 3. The more shifts used, the closer the answer arrives to the Fourier-space inversion - but the greater the computation time as well.

4. Conclusions

An example of a time-distance helioseismic [5] flow inversion demonstrated how cross talk comes into play, and an example sound-speed inversion showed how a large speed-up is obtained for OLA inversions carried out in Fourier space. Both of these factors are important for upcoming helioseismic studies (e.g., with SDO data) because (1) it is expected that large-scale inversions will include many or all possible perturbations and it will be necessary to sort out the cross-talk among them, and (2) these large-scale inversions will be computationally demanding and require very efficient algorithms. This will likely be accomplished with a Fourier-based scheme as presented here.

References

[1] G.E. Backus and J.F. Gilbert. The resolving power of gross earth data. *Geophys. J. R. Astr. Soc.*, 16:169–205, 1968. doi: 10.1111/j.1365-246X.1968.tb00216.x.

[2] J. Zhao, A. G. Kosovichev, and T. L. Duvall, Jr. Investigation of Mass Flows beneath a Sunspot by Time-Distance Helioseismology. *Astrophys. J.*, 557:384–388, August 2001. doi: 10.1086/321491.

[3] L. Gizon and A. C. Birch. Time-Distance Helioseismology: Noise Estimation. *Astrophys. J.*, 614:472–489, October 2004. doi: 10.1086/423367.

[4] J. Jackiewicz, L. Gizon, and A. C. Birch. High-Resolution Mapping of Flows in the Solar Interior: Fully Consistent OLA Inversion of Helioseismic Travel Times. *Solar Phys.*, 251:381–415, September 2008. doi: 10.1007/s11207-008-9158-z.

[5] T. L. Duvall, Jr., S. M. Jefferies, J. W. Harvey, and M. A. Pomerantz. Time-distance helioseismology. *Nature*, 362:430–432, April 1993. doi: 10.1038/362430a0.

AAS Solar Physics Division Meeting, Boulder, CO, June 14 - June 18, 2009.