

Sensitivity kernels for time-distance helioseismology

Damien Fournier, Chris Hanson, Laurent Gizon, Hélène Barucq

▶ **To cite this version:**

Damien Fournier, Chris Hanson, Laurent Gizon, Hélène Barucq. Sensitivity kernels for time-distance helioseismology: Efficient computation for spherically symmetric solar models. Astronomy and Astrophysics - A&A, 2018, 616, 10.1051/0004-6361/201833206 . hal-01938331

HAL Id: hal-01938331

https://inria.hal.science/hal-01938331

Submitted on 28 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A&A 616, A156 (2018)
https://doi.org/10.1051/0004-6361/201833206
© ESO 2018

Astronomy & Astrophysics

Sensitivity kernels for time-distance helioseismology

Efficient computation for spherically symmetric solar models

Damien Fournier[1], Chris S. Hanson[1], Laurent Gizon[1,2], and Hélène Barucq[3]

[1] Max-Planck-Institut für Sonnensystemforschung, Justus-von-Liebig-Weg 3, 37077 Göttingen, Germany
e-mail: fournier@mps.mpg.de
[2] Institut für Astrophysik, Georg-August-Universität Göttingen, Friedrich-Hund-Platz 1, 37077 Göttingen, Germany
[3] Magique-3D, Inria Bordeaux Sud-Ouest, Université de Pau et des Pays de l'Adour, 64013 Pau, France

Received 9 April 2018 / Accepted 17 May 2018

ABSTRACT

Context. The interpretation of helioseismic measurements, such as wave travel-time, is based on the computation of kernels that give the sensitivity of the measurements to localized changes in the solar interior. These kernels are computed using the ray or the Born approximation. The Born approximation is preferable as it takes finite-wavelength effects into account, although it can be computationally expensive.
Aims. We propose a fast algorithm to compute travel-time sensitivity kernels under the assumption that the background solar medium is spherically symmetric.
Methods. Kernels are typically expressed as products of Green's functions that depend upon depth, latitude, and longitude. Here, we compute the spherical harmonic decomposition of the kernels and show that the integrals in latitude and longitude can be performed analytically. In particular, the integrals of the product of three associated Legendre polynomials can be computed.
Results. The computations are fast and accurate and only require the knowledge of the Green's function where the source is at the pole. The computation time is reduced by two orders of magnitude compared to other recent computational frameworks.
Conclusions. This new method allows flexible and computationally efficient calculations of a large number of kernels, required in addressing key helioseismic problems. For example, the computation of all the kernels required for meridional flow inversion takes less than two hours on 100 cores.

Key words. Sun: helioseismology – Sun: oscillations – Sun: interior – methods: numerical

1. Introduction

The aim of time-distance helioseismology (Duvall et al. 1993) is to infer the subsurface structure of the Sun by measuring seismic wave travel times between any two points at the solar surface. The interpretation of these measurements requires understanding how waves propagate in the solar interior, i.e., solving the forward problem. Due to its simplicity, the ray approximation was initially used to invert flow velocities and sound-speed perturbations to a reference background model (Kosovichev 1996). It is still used today, for example to recover the meridional circulation (Rajaguru & Antia 2015). However, this approach is a high-frequency approximation that cannot be used to recover perturbations with sizes on the order of the local wavelength (Birch & Kosovichev 2000). Gizon & Birch (2002) derived a general framework for sensitivity kernels under the Born approximation and for random sources of excitation. Birch et al. (2004), Burston et al. (2015), and Böning et al. (2016) computed Born kernels using a normal-mode summation of the eigenfunctions in a solar-like stratified background. To treat axisymmetric background media (e.g., a background that includes large-scale differential rotation) and to include frequencies above the acoustic cutoff, Gizon et al. (2017) proposed solving the wave equation in frequency space by using a 2.5D finite-element solver. All these approaches are useful, but are computationally expensive. This limits their use in the interpretation of solar data as many kernels must be computed (and averaged). In some cases it is sufficient to consider perturbations to a steady spherically symmetric reference medium. The study of meridional circulation is one such application (see, e.g., Liang et al. 2017).

In this paper, we present a way to reduce the computational time of Born sensitivity kernels in a spherically symmetric background by treating the horizontal variables (the co-latitude θ and the longitude ϕ) analytically using the properties of the spherical harmonics. Here this approach is demonstrated using the scalar wave equation from Gizon et al. (2017), but it could be applied to the normal-mode summation method of Böning et al. (2016) or to solving the wave equation using a high-order finite-difference scheme (Mandal et al. 2017).

2. Born sensitivity kernels

2.1. Green's function in a spherically symmetric background

We follow the framework of Gizon et al. (2017), where the observable $\psi(\boldsymbol{r}, \omega)$ at spatial location $\boldsymbol{r} = (r, \theta, \phi)$ and frequency ω is linked to the divergence of the displacement: $\psi(\boldsymbol{r}, \omega) = c(\boldsymbol{r})\nabla \cdot \boldsymbol{\xi}(\boldsymbol{r}, \omega)$. This scalar quantity solves

$$L\psi(\boldsymbol{r}, \omega) = s(\boldsymbol{r}, \omega), \tag{1}$$

where L is the spatial wave operator at frequency ω,

$$L\psi := -(\omega^2 + 2\mathrm{i}\omega\gamma)\psi - 2\mathrm{i}\omega\boldsymbol{u} \cdot \nabla\psi - c\nabla \cdot \left(\frac{1}{\rho}\nabla(\rho c\psi)\right), \tag{2}$$

ρ and c are the solar density and sound speed from standard solar model S (Christensen-Dalsgaard et al. 1996), γ is the attenuation, $\boldsymbol{u}$ is a background flow, and s is a stochastic source term. We assume that the sources are spatially uncorrelated and depend only on depth and frequency such that the source covariance matrix is given by

$$M(\boldsymbol{r}, \boldsymbol{r}', \omega) := \mathbb{E}[s^*(\boldsymbol{r}, \omega) s(\boldsymbol{r}', \omega)] = A(r, \omega)\delta(\boldsymbol{r} - \boldsymbol{r}'), \tag{3}$$

where $A(r, \omega)$ is the radial profile of the source power. The wave field ψ can be obtained using

$$\psi(\boldsymbol{r}, \omega) = \int_\odot G(\boldsymbol{r}, \boldsymbol{r}', \omega) s(\boldsymbol{r}', \omega) \rho(\boldsymbol{r}') \mathrm{d}\boldsymbol{r}', \tag{4}$$

where G is the Green's function:

$$LG(\boldsymbol{r}, \boldsymbol{r}', \omega) = \frac{1}{\rho(\boldsymbol{r})}\delta(\boldsymbol{r} - \boldsymbol{r}'). \tag{5}$$

When the background is spherically symmetric (i.e., no flow and no heterogeneity), G can be written as

$$G(\boldsymbol{r}, \boldsymbol{r}', \omega) = \sum_{\ell=0}^{\ell_{\max}} \alpha_\ell G_\ell(r, \boldsymbol{r}', \omega) \sum_{m=-\ell}^{\ell} Y_\ell^{m*}(\theta', \phi') Y_\ell^m(\theta, \phi), \tag{6}$$

where $\boldsymbol{r} = (r, \theta, \phi)$, $\boldsymbol{r}' = (r', \theta', \phi')$, Y_ℓ^m are the normalized spherical harmonics, $\alpha_\ell = \sqrt{4\pi/(2\ell + 1)}$, and G_ℓ is the Legendre component of the Green's function

$$G_\ell(r, \boldsymbol{r}', \omega) = \int_0^{2\pi} \int_0^{\pi} G(\boldsymbol{r}, \boldsymbol{r}', \omega) P_\ell(\cos\theta) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{7}$$

Equation (6) can be simplified by using the addition theorem (DLMF 2017, Eq. (14.18.1)) and introducing the great-circle angle between $\boldsymbol{r}$ and $\boldsymbol{r}'$. However, Eq. (6) is the form required in the following sections.

2.2. Cross-covariance in a spherically symmetric background

In a spherically symmetric background the expectation value of the cross-covariance between an observation point $\boldsymbol{r}_1 = (r_0, \theta_1, \phi_1)$ and a point $\boldsymbol{r} = (r, \theta, \phi)$ is

$$\begin{aligned} C(\boldsymbol{r}_1, \boldsymbol{r}, \omega) &= \mathbb{E}[\psi^*(\boldsymbol{r}_1, \omega)\psi(\boldsymbol{r}, \omega)] \\ &= \sum_\ell \alpha_\ell^2 \sum_{m=-\ell}^{\ell} Y_\ell^{m*}(\theta_1, \phi_1) Y_\ell^m(\theta, \phi) C_\ell(\boldsymbol{r}_0, r, \omega), \end{aligned} \tag{8}$$

where

$$C_\ell(\boldsymbol{r}_0, r, \omega) = \int_0^{R_\odot} G_\ell(r', \boldsymbol{r}_0, \omega)^* G_\ell(r', \hat{\boldsymbol{r}}, \omega) A(r', \omega) \rho(r')^2 r'^2 \mathrm{d}r', \tag{9}$$

and $\boldsymbol{r}_0 = (r_0, 0, 0)$ and $\hat{\boldsymbol{r}} = (r, 0, 0)$ are on the polar axis. The radius r_0 is the observation radius, for example $\sim$150 km above the photosphere for SDO/HMI. In obtaining Eq. (8), we used the property that the cross-covariance depends only on the great-circle distance between the two points. To simplify the computations we place one point on the polar axis so that the Green's function is axisymmetric and only the mode $m = 0$ needs to be computed.

Using the convenient source of excitation introduced in Gizon et al. (2017), $C_\ell(\boldsymbol{r}_0, r, \omega)$ is directly linked to the imaginary part of the Green's function $G_\ell(r, \boldsymbol{r}_0, \omega)$, but this assumption is not mandatory in this paper. The important assumption concerns the covariance of the sources of excitation that needs to be of the form given by Eq. (3), such that the cross-covariance depends only on depths and the great-circle distance between the two points $\boldsymbol{r}_1$ and $\boldsymbol{r}$. This assumption is common in helioseismology and is generally used in forward modeling (e,g., Kosovichev et al. 2000; Böning et al. 2016; Mandal et al. 2017).

2.3. Born flow kernels

Recovering flows in the solar interior is a major goal for local helioseismology, and so we focus here on flow kernels. The method presented here can be applied to all other types of perturbations with respect to a spherically symmetric background. The Born sensitivity kernel $\boldsymbol{K} = (K_r, K_\theta, K_\phi)$ connects the travel-time perturbation $\delta\tau$ to the vector flow $\boldsymbol{u} = (u_r, u_\theta, u_\phi)$, such that

$$\delta\tau(\boldsymbol{r}_1, \boldsymbol{r}_2) = \int_\odot \boldsymbol{K}(\boldsymbol{r}, \boldsymbol{r}_1, \boldsymbol{r}_2) \cdot \boldsymbol{u}(\boldsymbol{r}) \, \mathrm{d}\boldsymbol{r}. \tag{10}$$

According to Gizon et al. (2017) we have

$$\begin{aligned} \boldsymbol{K}(\boldsymbol{r}, \boldsymbol{r}_1, \boldsymbol{r}_2) = 2\mathrm{i}\rho(r) \int_{-\infty}^{\infty} \mathrm{d}\omega \, \omega W^*(\boldsymbol{r}_1, \boldsymbol{r}_2, \omega) \\ \times \left[G(\boldsymbol{r}_2, \boldsymbol{r}, \omega) \nabla C(\boldsymbol{r}_1, \boldsymbol{r}, \omega) - G^*(\boldsymbol{r}_1, \boldsymbol{r}, \omega) \nabla C^*(\boldsymbol{r}_2, \boldsymbol{r}, \omega) \right], \end{aligned} \tag{11}$$

where W is a weighting function that relates a change in the cross-covariance to a change in travel-time (Gizon & Birch 2002) and $\nabla = (\partial_r, 1/r \, \partial_\theta, 1/(r \sin\theta) \, \partial_\phi)$ is the gradient operator with respect to the scattering location $\boldsymbol{r}$. We note that in a spherically symmetric background, seismic reciprocity implies $G(\boldsymbol{r}, \boldsymbol{r}', \omega) = G(\boldsymbol{r}', \boldsymbol{r}, \omega)$ for any $\boldsymbol{r}$ and $\boldsymbol{r}'$. The reference cross-covariance also satisfies $C(\boldsymbol{r}', \boldsymbol{r}, \omega) = C(\boldsymbol{r}, \boldsymbol{r}', \omega)$.

The expression for the kernel may differ when a different observable is chosen; however, the above integral will always involve the product of a Green's function with the cross-covariance. One approach to obtaining the flow kernels (Böning et al. 2016; Mandal et al. 2017) is to compute the 3D Green's function and the cross-covariance using its spherical harmonic decomposition using Eq. (6). A reference kernel is usually obtained for a fixed pair of observation points and later rotated to obtain kernels for other pairs of points. However, a fine resolution in θ and ϕ is required in order to perform this rotation accurately, which makes the computation expensive in both time and memory.

2.4. Spherical harmonic decomposition of Born flow kernels

In order to circumvent the disadvantages mentioned above (e.g., rotation) and improve accuracy, we propose a new approach based on the spherical harmonic decomposition of the kernel,

$$\boldsymbol{K}(\boldsymbol{r}, \boldsymbol{r}_1, \boldsymbol{r}_2) = \sum_{\bar{\ell}} \sum_{\bar{m}=-\bar{\ell}}^{\bar{\ell}} \boldsymbol{K}^{\bar{\ell}\bar{m}}(r, \boldsymbol{r}_1, \boldsymbol{r}_2) Y_{\bar{\ell}}^{\bar{m}}(\theta, \phi), \tag{12}$$

where

$$\boldsymbol{K}^{\bar{\ell}\bar{m}}(r, \boldsymbol{r}_1, \boldsymbol{r}_2) = \int_0^{2\pi} \int_0^{\pi} \boldsymbol{K}(\boldsymbol{r}, \boldsymbol{r}_1, \boldsymbol{r}_2) Y_{\bar{\ell}}^{\bar{m}*}(\theta, \phi) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{13}$$

Decomposing $G(\boldsymbol{r}_1, \boldsymbol{r}, \omega)$ and $C(\boldsymbol{r}_2, \boldsymbol{r}, \omega)$ into spherical harmonics, we can obtain the spherical harmonic coefficients of each kernel.

For the u_r kernel, we have

$$K_{\mathrm{r}}^{\bar{\ell}\bar{m}}(r, \boldsymbol{r}_1, \boldsymbol{r}_2) = \sum_{\ell,\ell'} \alpha_\ell \alpha_{\ell'} \sum_{m=-\ell}^{\ell} \sum_{m'=-\ell'}^{\ell'} I_r \times \Big(f^r_{\ell\ell'}(r) Y_\ell^{m*}(\theta_2, \phi_2) Y_{\ell'}^{m'*}(\theta_1, \phi_1) + g^r_{\ell\ell'}(r) Y_\ell^{m}(\theta_1, \phi_1) Y_{\ell'}^{m'}(\theta_2, \phi_2) \Big), \tag{14}$$

where

$$f^r_{\ell\ell'}(r) = 2\mathrm{i}\rho(r) \int_{-\infty}^{\infty} \omega W^*(\omega) G_\ell(r, \boldsymbol{r}_0, \omega) \partial_{\mathrm{r}} C_{\ell'}(\boldsymbol{r}_0, r, \omega) \mathrm{d}\omega, \tag{15}$$

$$g^r_{\ell\ell'}(r) = -2\mathrm{i}\rho(r) \int_{-\infty}^{\infty} \omega W^*(\omega) G^*_\ell(r, \boldsymbol{r}_0, \omega) \partial_{\mathrm{r}} C^*_{\ell'}(\boldsymbol{r}_0, r, \omega) \mathrm{d}\omega, \tag{16}$$

$$\text{and } I_r = \int_0^{2\pi} \int_0^{\pi} Y_\ell^m(\theta, \phi) Y_{\ell'}^{m'}(\theta, \phi) Y_{\bar{\ell}}^{\bar{m}*}(\theta, \phi) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{17}$$

The integral of three spherical harmonics over the unit sphere can be done analytically using the Gaunt formula (see, e.g., Edmonds 1960, Eq. (4.6.3))

$$I_{\mathrm{r}} = \frac{4\pi}{\alpha_\ell \alpha_{\ell'} \alpha_{\bar{\ell}}} (-1)^{\bar{m}} \begin{pmatrix} \ell & \ell' & \bar{\ell} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \ell & \ell' & \bar{\ell} \\ m & m' & -\bar{m} \end{pmatrix}, \tag{18}$$

where we have used the Wigner-3j symbols (see, e.g., Edmonds 1960, p. 45). The Wigner-3j symbol vanishes when $\bar{m} \neq m + m'$, which enables us to remove the sum over m' in the equation for K_{u_r}.

It can be shown that the expression for K_r, K_θ, and K_ϕ can be recast in the form

$$K_j^{\bar{\ell}\bar{m}}(r) = \sum_{\ell,\ell'} \alpha_\ell \alpha_{\ell'} \sum_{m=-L}^{L} \Big(I_j f^j_{\ell\ell'}(r) Y_\ell^{m*}(\theta_2, \phi_2) Y_{\ell'}^{\bar{m}-m*}(\theta_1, \phi_1) + I_j^* g^j_{\ell\ell'}(r) Y_\ell^{m*}(\theta_1, \phi_1) Y_{\ell'}^{\bar{m}-m*}(\theta_2, \phi_2) \Big), \tag{19}$$

where $j \in \{r, \theta, \phi\}$ and $L = \min(\ell, \ell')$.

Proceeding in a similar way, for K_r the kernel $K_\theta^{\bar{\ell}\bar{m}}$ depends on the functions f^θ and g^θ given by

$$f^\theta_{\ell\ell'}(r) = 2\mathrm{i}\rho(r) \int_{-\infty}^{\infty} \omega W^*(\omega) G_\ell(r, \boldsymbol{r}_0, \omega) C_{\ell'}(r, \boldsymbol{r}_0, \omega) \mathrm{d}\omega, \tag{20}$$

$$g^\theta_{\ell\ell'}(r) = -2\mathrm{i}\rho(r) \int_{-\infty}^{\infty} \omega W^*(\omega) G^*_\ell(r, \boldsymbol{r}_0, \omega) C^*_{\ell'}(r, \boldsymbol{r}_0, \omega) \mathrm{d}\omega. \tag{21}$$

The horizontal integral is

$$I_\theta = \frac{1}{r} \int_0^{2\pi} \int_0^{\pi} Y_\ell^m(\theta, \phi) \partial_\theta Y_{\ell'}^{m'}(\theta, \phi) Y_{\bar{\ell}}^{\bar{m}*}(\theta, \phi) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{22}$$

This integral I_θ is much more difficult to evaluate than I_r because of the θ derivative. In order to keep only associated Legendre polynomials in I_θ, we use

$$\frac{\mathrm{d}P_\ell^m(\cos\theta)}{\mathrm{d}\theta} = \frac{1}{2} \Big(\sqrt{(\ell + m)(\ell - m + 1)} P_\ell^{m-1}(\cos\theta) - \sqrt{(\ell + m + 1)(\ell - m)} P_\ell^{m+1}(\cos\theta) \Big), \tag{23}$$

where the P_l^m are the normalized associated Legendre polynomials. We use the convention that $P_\ell^{m\pm1} = 0$ if $|m \pm 1| > \ell$, so that Eq. (23) remains valid for $m = \pm\ell$. Integrating Eq. (22) over ϕ and using Eq. (23), I_θ becomes

$$I_\theta = \frac{1}{2\sqrt{2\pi}\, r} \Big(- \sqrt{(\ell' + m')(\ell' - m' + 1)} J^{m,m'-1,\bar{m}}_{\ell\ell'\bar{\ell}} + \sqrt{(\ell' + m' + 1)(\ell' - m')} J^{m,m'+1,\bar{m}}_{\ell\ell'\bar{\ell}} \Big), \tag{24}$$

where $m' = \bar{m} - m$ and

$$J^{mm'\bar{m}}_{\ell\ell'\bar{\ell}} = \int_0^{\pi} P_\ell^m(\cos\theta) P_{\ell'}^{m'}(\cos\theta) P_{\bar{\ell}}^{\bar{m}}(\cos\theta) \sin\theta \mathrm{d}\theta. \tag{25}$$

Fortunately, this integral can also be evaluated analytically. It involves a sum of products of Wigner-3j symbols (see Appendix A and Dong & Lemus 2002).

The derivation of u_ϕ is similar to u_θ and requires the evaluation of the horizontal integral

$$I_\phi = \frac{\mathrm{i}m'}{r} \int_0^{2\pi} \int_0^{\pi} \frac{1}{\sin\theta} Y_\ell^m(\theta, \phi) Y_{\ell'}^{m'}(\theta, \phi) Y_{\bar{\ell}}^{\bar{m}*}(\theta, \phi) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{26}$$

Using

$$\frac{P_\ell^m(\cos\theta)}{\sin\theta} = -\frac{1}{m} \left(\sqrt{\frac{(2\ell + 1)(\ell + m - 1)(\ell + m)}{2\ell - 1}} P_{\ell-1}^{m-1}(\cos\theta) + \sqrt{\frac{(2\ell + 1)(\ell - m - 1)(\ell - m)}{2\ell - 1}} P_{\ell-1}^{m+1}(\cos\theta) \right), \tag{27}$$

for $m \neq 0$, we obtain

$$I_\phi = \frac{\mathrm{i}}{2\sqrt{2\pi}\, r} \left(\sqrt{\frac{(2\ell' + 1)(\ell' + m' - 1)(\ell' + m')}{2\ell' - 1}} J^{m,\bar{m}-m-1,\bar{m}}_{\ell,\ell'-1,\bar{\ell}} + \sqrt{\frac{(2\ell' + 1)(\ell' - m' - 1)(\ell' - m')}{2\ell' - 1}} J^{m,\bar{m}-m+1,\bar{m}}_{\ell,\ell'-1,\bar{\ell}} \right). \tag{28}$$

Now that we have the equations for the kernels, let us summarize the algorithm used for the resolution:

1. Computation and storage of the Green's function $G_l(r, \boldsymbol{r}_0, \omega)$ with the source on the polar axis, as a function of depth and harmonic degree ℓ for all frequencies;
2. For each great-circle distance between $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$:
   - computation of the cross-covariance using Eq. (8). If the convenient source of excitation of Gizon et al. (2017) is used, the cross-covariance is directly obtained from the imaginary part of the Green's function;
   - computation of the weighting function W;
   - computation of the functions f^j and g^j;
3. Evaluation of the integrals I_j and computation of the kernel using Eq. (19).

A summary of the different terms required to compute the different components of the flow kernels using Eq. (19) is given in Table 1.

We note that the algorithm presented above could of course be used to compute sensitivity kernels for the cross-covariance amplitude using the linear definition of Nagashima et al. (2017) and the appropriate choice of W. It is also possible to obtain kernels for the cross-covariance function at a given frequency by removing the weighting function. In this case, the functions g^j are just the complex conjugates of f^j.

Table 1. Terms required to compute the spherical harmonic coefficients of the flow sensitivity kernels $K_{\rm j}^{\bar{\ell}\bar{m}}$ using Eq. (19)

j	$f^j_{\ell\ell'}(r)$	I_j
r	$2\mathrm{i}\rho(r)\int_{-\infty}^{\infty}\omega W^*(\omega)G_\ell(r,\boldsymbol{r}_0,\omega)\partial_{\rm r}C_{\ell'}(\boldsymbol{r}_0,r,\omega)\mathrm{d}\omega$	$\frac{4\pi}{\alpha_\ell\alpha_{\ell'}\alpha_{\bar{\ell}}}\begin{pmatrix}\ell & \ell' & \bar{\ell}\\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix}\ell & \ell' & \bar{\ell}\\ m & \bar{m}-m & -\bar{m}\end{pmatrix}$
θ	$2\mathrm{i}\rho(r)\int_{-\infty}^{\infty}\omega W^*(\omega)G_\ell(r,\boldsymbol{r}_0,\omega)C_{\ell'}(\boldsymbol{r}_0,r,\omega)\mathrm{d}\omega$	$\frac{1}{2\sqrt{2\pi}r}\Big(-\sqrt{(\ell'+\bar{m}-m)(\ell'-\bar{m}+m+1)}\,J^{m,\bar{m}-m-1,\bar{m}}_{\ell\ell'\bar{\ell}}$ $+\sqrt{(\ell'+\bar{m}-m+1)(\ell'-\bar{m}+m)}\,J^{m,\bar{m}-m+1,\bar{m}}_{\ell\ell'\bar{\ell}}\Big)$
ϕ	$f^\phi_{\ell\ell'}(r)=f^\theta_{\ell\ell'}(r)$	$\frac{\mathrm{i}}{2\sqrt{2\pi}r}\Big(\sqrt{\frac{(2\ell'+1)(\ell'-\bar{m}+m-1)(\ell'-\bar{m}+m)}{2\ell'-1}}\,J^{m,\bar{m}-m+1,\bar{m}}_{\ell,\ell'-1,\bar{\ell}}$ $+\sqrt{\frac{(2\ell'+1)(\ell'+\bar{m}-m-1)(\ell'+\bar{m}-m)}{2\ell'-1}}\,J^{m,\bar{m}-m-1,\bar{m}}_{\ell,\ell'-1,\bar{\ell}}\Big)$

Notes. The integrals $J^{mm'\bar{m}}_{\ell\ell'\bar{\ell}}$ depend only on Wigner-3j symbols and can be computed using Eq. (A.2) corresponding to the algorithm of Dong & Lemus (2002).

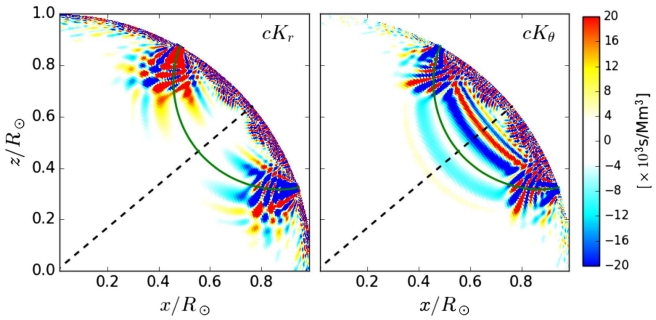


Fig. 1. Slices along a constant meridian of the point-to-point 3D travel-time difference kernel for $u_{\rm r}$ (*left*) and u_θ (*right*). The 3D kernel for u_ϕ is zero along this slice. The kernel is computed with $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ separated by 42°, with mean latitude 40°. The green line is the ray path between the two points and the dashed black line shows the image plane of Fig. 2.

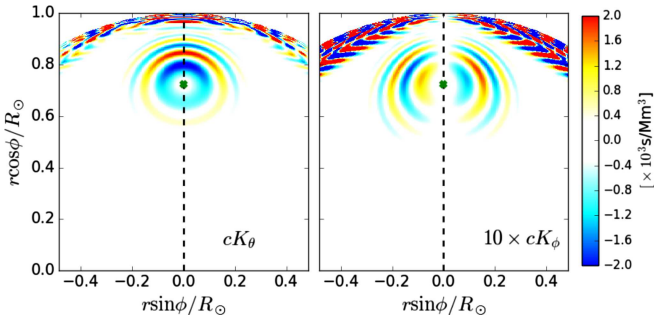


Fig. 2. Slices of the point-to-point 3D travel-time difference kernel for u_θ (*left*) and u_ϕ (*right*) along the plane indicated in Fig. 1. The 3D kernel for $u_{\rm r}$ is mostly zero within this plane. The kernel is computed with $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ separated by 42°, with mean latitude 40°. The green cross indicates the intersection of the ray path and the image plane. The dashed black line shows the image plane of Fig. 1.

2.5. Numerical validation

To evaluate the flow kernels in this framework, the only ingredient to prescribe is the Green's function as a function of the spherical harmonic degree ℓ and depth for a source located at the pole. We compute it using 1D finite elements with the solver Montjoie (Chabassier & Duruflé 2016; Fournier et al. 2017). The Green's functions are computed with a high enough frequency resolution to resolve the modes using the mode linewidths (5671 frequencies corresponding to 4 days of observations at 60 s cadence) and $\ell_{\rm max}=400$.

Representations of the different components of the flow kernels between two points $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ centered at 40° and separated by 42° is shown in Figs. 1 and 2. They exhibit the classical banana-doughnut shape with zero sensitivity along the ray path. Small-scale structures are visible close to the surface as we kept values of ℓ up to 400. Visually, there is no difference between the kernels computed with this new approach, the ones from Gizon et al. (2017), or the ones obtained by rotation so only one is shown here.

To allow a more quantitative approach, Fig. 3 compares kernels computed using our new approach to the approach presented by Gizon et al. (2017) where the background is axisymmetric and the Green's function is computed for each azimuthal degree m on a 2D grid. These kernels K_r and K_θ are averaged over longitudes ($\bar{m}=0$) where $\boldsymbol{r}_1$ is located at the pole and $\boldsymbol{r}_2$ is at a co-latitude of 42° (similar to Fig. (17) in Gizon et al. 2017). The results show good agreement, validating the method presented here. We note slight differences in the structure of K_θ at a depth of 500 km, and attribute this to the numerics of the 2D FEM solver differing. Specifically, the 2D FEM has an inherent difficulty in computing the real part of the Green's function close to the Dirac source location (see for details Chabassier & Duruflé 2016). In order to ensure that these small differences do not affect the interpretation of the data, we compute the travel times induced by the meridional flow model from Gizon et al. (2017). We decompose the flow in Legendre polynomial $\boldsymbol{u}^{\bar{\ell}}$ akin to Eq. (7) and compute a travel time for each $\bar{\ell}$ according to

$$\delta\tau_{\bar{\ell}}=\int_0^{R_\odot}\boldsymbol{K}^{\bar{\ell},\bar{m}=0}(r)\cdot\boldsymbol{u}^{\bar{\ell}}(r)\,r^2\mathrm{d}r. \tag{29}$$

The bottom panel of Fig. 3 shows that this travel time as a function of $\bar{\ell}$ is nearly indistinguishable from the travel times in Gizon et al. (2017); the differences are less than 0.5 ms.

As the computational burden of this new method depends on the maximum harmonic degree of the Green's function $\ell_{\rm max}$ and the number of $\bar{\ell}$ required for the kernel, we illustrate the efficiency of the method on two problems of interest: kernels for meridional flow inversions and kernels for supergranulation inversions.

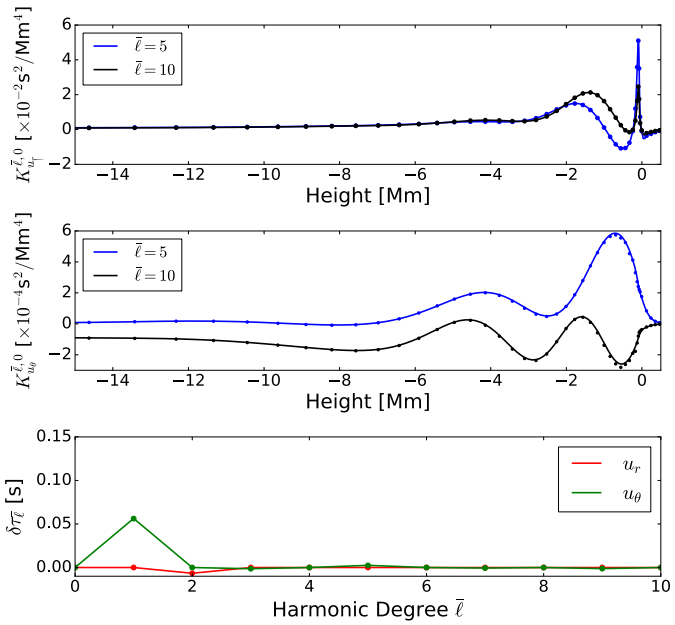


Fig. 3. *Top* and *middle panels*: comparison of the kernels for $\bar{\ell} = 5$ and 10 (blue and black, respectively) computed using the method here (solid lines) and the method of Gizon et al. (2017; dots). *Bottom panel*: travel times $\delta\tau_{\bar{\ell}}$ due to the radial (red) and the latitudinal (green) components of the flow for each $\bar{\ell}$ of the kernels presented here (solid line) and those of Gizon et al. (2017; dots).

2.6. Computation of kernels for meridional flow

As a first test, we compute the kernels that are required to interpret meridional flow measurements. As the flow varies slowly with latitude, we can limit the number of spherical harmonic coefficients of the kernels to $\bar{\ell} \leq 10$ (see Fig. 3). As for the observations, a low-pass filter with $\ell_{\max} = 300$ is applied to the Green's function. The method can be parallelized in $\bar{\ell}$ so we use 11 cores to compute kernels up to $\bar{\ell} = 10$.

For meridional flow measurements, the separation distance between the source and the receiver is generally prescribed for different values of the mean latitude (see, e.g., Liang et al. 2017). For a given separation distance, we compute 15 kernels corresponding to 15 different latitudes. We then vary the separation distance in order to probe different depths.

Table 2 shows the computational times and memory requirements of the different steps of the algorithm. The computation of the Green's function for a source located at the pole can be done once and for all and stored as it is necessary to do it for every kernel. Therefore, the computation is very fast (3 seconds per frequency for $\ell_{\max} = 300$) and embarrassingly parallel in frequency so it could also be recomputed every time. The computation of the frequency integrals (f^j and g^j) consists in loading the Green's function and the weighting function W and summing over frequencies. The reading of the Green's function files for all frequencies takes most of the time. The spatial integrals I_r, I_θ, and I_ϕ can be computed once and stored for future use. However, the computational time is small compared to the full computation of the kernel, so we decide to recompute I_j every time as the reading time can depend upon file system *I/O* load. The computations are parallelized in $\bar{\ell}$ and hence need 11 cores for each step since $0 \leq \bar{\ell} \leq 10$. The computation of the kernel for u_r is faster since the computation of I_r requires the evaluation of only two Wigner-3j symbols, unlike I_θ. However, the major difference in computational time between K_r and K_θ comes from the sum in ℓ' in Eq. (19). For K_r, the sum covers the range from $\ell - \bar{\ell}$ to $\ell + \bar{\ell}$, since $I_r = 0$ for other values due to the properties of the Wigner-3j. On the contrary, the sum in ℓ' for the computation of K_θ must be computed for the full range from 0 to ℓ.

Even though the computational burden of K_θ is greater than K_r, the total burden remains significantly lower than for other methods (see Table 3). All the kernels required to perform a meridional flow inversion can be computed within 2 h with 100 cores, and the memory requirements do not exceed 1 GB. The approach mentioned in Sect. 2.3, where the full 3D kernel is computed and rotated to obtain different latitudes, would take 11 days using 1000 cores with very significant memory requirements. In the axisymmetric approach of Gizon et al. (2017) the computational time would be about 40 days on 1000 cores for all the same set of kernels.

The computational times presented here are for point-to-point measurements; however, this framework can easily be extended to geometric averaging such as arc-to-arc measurements often performed for meridional flow measurements (e.g., Liang et al. 2017). It is only necessary to replace the product of the two spherical harmonics in Eq. (19) by a sum over all the points of the arc.

Table 2. Computational time of the different steps to obtain 15 flow kernels (15 latitudes) for a given separation distance with $\bar{\ell} \leq 10$ and $\ell_{\max} = 300$ using 11 cores.

Computation steps	Time [11 cores]	Memory
Green's function	13 min	100 MB
f^r (or f^θ)	9 min	1 GB
I_r	0.5 s	1 GB
I_θ	6 min	1 GB
Sum of terms in $K_r^{\bar{\ell}m}$	3 min	1 GB
Sum of terms in $K_\theta^{\bar{\ell}m}$	32 min	1 GB

Table 3. Comparison of the computational time and memory requirements to evaluate 225 kernels for the meridional flow.

Method	Time [cpu hours]	Memory [GB]
This paper	170	1
Rotation of 3D kernels	2.7×10^5	40
Gizon et al. (2017)	10^6	8

Notes. For the rotation of the 3D kernels as in Böning et al. (2016), and Mandal et al. (2017), we used a horizontal grid sampled with $N_\theta = 1001$ and $N_\phi = 2001$ points.

2.7. Computation of kernels for supergranulation

Resolving smaller scale flows such as supergranules requires a high spatial resolution, and thus the Green's function needs to include much higher harmonic degrees than for meridional flow Green's function. For example, Duvall & Hanasoge (2013) considered measurements up to $\ell_{\max} = 700$, but even higher $\ell_{\max}$ values may be required. Furthermore, supergranulation flows have maximum power around $\ell = 120$, thus the kernels should at least be computed up to $\bar{\ell} = 300$, or higher depending on the power distribution of the flow at large ℓ. The computational burden for these kernels is summarized in Table 4. The computation

Table 4. Computational time of the different steps to obtain 15 flow kernels for a given separation distance with $\overline{\ell} \leq 300$ and $\ell_{\max} = 700$ using 100 cores.

Computation steps	Time [100 cores]	Memory
Green's function	11 min	100 MB
f^r (or f^θ)	6 min	1 GB
I_r	2 min	1 GB
I_θ	270 min	1 GB
Sum of terms in $K_r^{\overline{\ell},\overline{m}}$	15 h	1 GB
Sum of terms in $K_\theta^{\overline{\ell},\overline{m}}$	39 h	1 GB

of the $m = 0$ component of the Green's function now takes about 23 second per frequency, and the loading of the files to compute f^j around 6 min. The computation of the Wigner symbols is computationally more challenging as ℓ increases, since the number of loops scales as $\ell_{\max}^3$ due to loops in ℓ, ℓ', and m. While the computation of I_r is still fast, the evaluation of I_θ now takes 270 min on 100 cores. Computing a set of 200 kernels would take 3 days with 1000 cores, which is significantly longer than for the meridional flow kernels, but still one order of magnitude faster than the approach of Gizon et al. (2017) and with a smaller memory requirement.

3. Conclusions

We presented a technique that is faster than previous approaches used to compute travel-time kernels under the assumption that the background medium is spherically symmetric. This technique does not rely on the numerical computation of kernel rotations and thus the memory requirements are small. Instead, the spatial integrals are performed analytically, which also leads to higher accuracy. For example, for meridional circulation applications, the kernels can be computed one thousand times faster than with previous methods, using a tenth of the memory requirement.

Acknowledgements. The computer infrastructure was provided by the German Data Center for SDO funded by the German Aerospace Center (DLR) and by the Ministry of Science of the State of Lower Saxony, Germany.

References

Birch, A. C., & Kosovichev, A. G. 2000, Sol. Phys., 192, 193
Birch, A. C., Kosovichev, A. G., & Duvall, Jr., T. L. 2004, ApJ, 608, 580
Böning, V. G., Roth, M., Zima, W., Birch, A. C., & Gizon, L. 2016, ApJ, 824, 14
Burston, R., Gizon, L., & Birch, A. C., 2015, Space Sci. Rev., 196, 201
Chabassier, J., & Duruflé, M. 2016, High Order Finite Element Method for Solving Convected Helmholtz Equation in Radial and Axisymmetric Domains. Application to Helioseismology, INRIA, Research Report RR-8893
Christensen-Dalsgaard, J., Dappen, W., Ajukov, S. V., et al. 1996, Science, 272, 1286
DLMF. 2017, NIST Digital Library of Mathematical Functions, http://dlmf.nist.gov/, Release 1.0.16 of 2017-09-18, eds. f. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, et al.
Dong, S.-h., & Lemus, R. 2002, Appl. Math. Lett., 15, 541
Duvall, T., Jeffferies, S., Harvey, J., & Pomerantz, M. 1993, Nature, 362, 430
Duvall, T. L., & Hanasoge, S. M. 2013, Sol. Phys., 287, 71
Edmonds, A. R. 1960, Angular Momentum in Quantum Mechanics (Princeton University Press)
Fournier, D., Leguèbe, M., Hanson, C. S., et al. 2017, A&A, 608, A109
Gizon, L., & Birch, A. 2002, ApJ, 571, 966
Gizon, L., Barucq, H., Duruflé, M., et al. 2017, A&A, 600, A35
Kosovichev, A. G. 1996, ApJ, 461, L55
Kosovichev, A. G., Duvall, Jr., T. L., & Scherrer, P. H. 2000, Sol. Phys., 192, 159
Liang, Z.-C., Birch, A. C., Duvall, Jr., T. L., Gizon, L., & Schou, J. 2017, A&A, 601, A46
Mandal, K., Bhattacharya, J., Halder, S., & Hanasoge, S. M. 2017, ApJ, 842, 89
Nagashima, K., Fournier, D., Birch, A. C., & Gizon, L. 2017, A&A, 599, A111
Rajaguru, S., & Antia, H. 2015, ApJ, 813, 114

Appendix A: Algorithm for the integral of three associated Legendre polynomials

For the sake of completeness, we summarize the algorithm of Dong & Lemus (2002) adapted to this study. The integral of three associated Legendre polynomials,

$$J^{mm'\overline{m}}_{\ell\ell'\overline{\ell}} = \int_0^\pi P_\ell^m(\cos\theta) P_{\ell'}^{m'}(\cos\theta) P_{\overline{\ell}}^{\overline{m}}(\cos\theta)\sin\theta \mathrm{d}\theta, \tag{A.1}$$

can be computed analytically in terms of sums of products of Wigner-3j symbols

$$J^{mm'\overline{m}}_{\ell\ell'\overline{\ell}} = \frac{(-1)^{\overline{m}}(2\pi)^{3/2}}{\alpha_\ell \alpha_{\ell'} \alpha_{\overline{\ell}}} \sum_{\ell_{12}=\min(|\ell-\ell'|,m_{12})}^{\ell+\ell'} Q_{12} \times \sum_{\ell_{123}=\min(|\ell_{12}-\overline{\ell}|,m_{123})}^{\ell_{12}+\overline{\ell}} Q_{123} \sqrt{\frac{(\ell_{123}-m_{123})!}{(\ell_{123}+m_{123})!}} J(\ell_{123}, m_{123}), \tag{A.2}$$

where the indices $m_{12} = m + m'$ and $m_{123} = m + m' + \overline{m}$ represent sums over the azimuthal degrees. The quantities Q_{12} and Q_{123} must be evaluated for various values of ℓ_{12} and ℓ_{123} as defined under the sums in Eq. (A.2). They depend on the Wigner-3j symbols:

$$Q_{12} = (2\ell_{12}+1) \begin{pmatrix} \ell & \ell' & \ell_{12} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \ell & \ell' & \ell_{12} \\ m & m' & -m_{12} \end{pmatrix},$$

$$Q_{123} = (2\ell_{123}+1) \begin{pmatrix} \ell_{12} & \overline{\ell} & \ell_{123} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \ell_{12} & \overline{\ell} & \ell_{123} \\ m_{12} & \overline{m} & -m_{123} \end{pmatrix}.$$

The value of Q_{12} (resp. Q_{123}) is non-zero only if $\ell_{12} + \ell + \ell'$ (resp. $\ell_{12} + \overline{\ell} + \ell_{123}$) is even. The last term $J(\ell_{123}, m_{123})$ is the integral

$$J(\ell_{123}, m_{123}) = \int_{-1}^{1} P_{\ell_{123}}^{m_{123}}(x)\mathrm{d}x, \tag{A.3}$$

which can be evaluated analytically. In this paper, we only need this value for $m_{123} = \pm 1$. As this integral is zero for odd values of ℓ_{123}, due to the parity of the associated Legendre polynomials, we set $\ell_{123} = 2p + 1$. Then, for a given m_{123}, the value of $J(\ell_{123}, m_{123})$ can be evaluated recursively using

$$J(2n+1, 1) = \frac{(2n+1)(2n-1)}{4n(n+1)} J(2n-1, 1) \tag{A.4}$$

$$\text{and} \quad J(2n+1, -1) = \frac{(2n-1)^2}{4(n+1)^2} J(2n-1, -1), \tag{A.5}$$

where $n = 1, 2, \ldots p$, together with the initial conditions

$$J(1,1) = -\frac{\pi}{2} \quad \text{and} \quad J(1,-1) = \frac{\pi}{4}. \tag{A.6}$$