SoftwareX 5 (2016) 252–259

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

ASKI: A modular toolbox for scattering-integral-based seismic full waveform inversion and sensitivity analysis utilizing external forward codes

Florian Schumacher*, Wolfgang Friederich

Institute of Geology, Mineralogy and Geophysics, Ruhr-University Bochum, Germany

ARTICLE INFO

Article history:
Received 7 July 2016
Accepted 31 October 2016

Keywords:
ASKI
Seismic full waveform inversion
Waveform sensitivity kernels
Object-oriented programming

ABSTRACT

Due to increasing computational resources, the development of new numerically demanding methods and software for imaging Earth's interior remains of high interest in Earth sciences. Here, we give a description from a user's and programmer's perspective of the highly modular, flexible and extendable software package ASKI – Analysis of Sensitivity and Kernel Inversion – recently developed for iterative scattering-integral-based seismic full waveform inversion. In ASKI, the three fundamental steps of solving the seismic forward problem, computing waveform sensitivity kernels and deriving a model update are solved by independent software programs that interact via file output/input only. Furthermore, the spatial discretizations of the model space used for solving the seismic forward problem and for deriving model updates, respectively, are kept completely independent. For this reason, ASKI does not contain a specific forward solver but instead provides a general interface to established community wave propagation codes. Moreover, the third fundamental step of deriving a model update can be repeated at relatively low costs applying different kinds of model regularization or re-selecting/weighting the inverted dataset without need to re-solve the forward problem or re-compute the kernels. Additionally, ASKI offers the user sensitivity and resolution analysis tools based on the full sensitivity matrix and allows to compose customized workflows in a consistent computational environment. ASKI is written in modern Fortran and Python, it is well documented and freely available under terms of the GNU General Public License (http://www.rub.de/aski).

1. Motivation and significance

In the context of Earth sciences, researchers as well as industrial companies have a natural interest in more accurate imaging methods which can be applied to the increasing amounts of available seismic data. Software implementing such new methods have an increased demand of computational resources on high-performance computing systems which, however, become available more easily nowadays.

The imaging method of seismic full waveform inversion (FWI) aims at utilizing the complete information content of measured seismic waveforms for deriving an earth model. Established methods iteratively derive a series of models $\mathbf{m}^1, \mathbf{m}^2, \ldots, \mathbf{m}^n, \ldots$ converging to the solution of the inverse problem by minimizing a waveform misfit criterion. Starting off with an initial model $\mathbf{m}^0$ of sufficient quality, in each iteration $n \geq 1$ first the seismic forward problem is solved, i.e. seismic wave propagation is simulated with respect to model $\mathbf{m}^{n-1}$ assuming the mechanisms of the involved seismic sources as known (or inverting for source properties jointly). On the basis of the observed residual between the measured seismic waveforms and the synthetic ones computed with respect to model $\mathbf{m}^{n-1}$, then a model $\mathbf{m}^n$ is derived which best reduces the misfit criterion in use. One group of currently used methods are based on the (pre-conditioned) conjugate gradient of the misfit functional with respect to the model parameters [1–4]. Another group of currently used methods minimize the misfit criterion by Newton-like [5–7] or Gauss–Newton methods [8–11] which utilize (approximations of) higher order derivatives of the misfit functional with respect to the model parameters for deriving a model update. These generally have faster convergence properties than gradient-based methods but can be subject to higher computational costs. Established FWI codes (for gradient-based as well as Newton-like or Gauss–Newton methods) infer derivatives of the misfit criterion by combination of the wavefield

* Corresponding author.
E-mail address: florian.schumacher@rub.de (F. Schumacher).

http://dx.doi.org/10.1016/j.softx.2016.10.005

originating from the seismic source with the wavefield of backpropagated residuals originating from the receiver positions. Thus, solving the forward problem, i.e. simulating seismic wave propagation, is strongly interwoven with computing the derivatives and is usually implemented in the same code which thereby has a rather monolithic character.

Seismic FWI is a complex problem that requires demanding numerical computations as well as handling of large amounts of data on high-performance computing systems. Thereby, complex workflows arise that need to be handled by researchers in a consistent and flexible way. From a geophysical point of view, FWI applications may have a wide range in terms of scale (from global to ultra-sonic), considered wave types and frequencies. Hence, it is desirable to have modular and extendable, thus efficient, solutions to FWI. Nowadays, new developments follow this approach and try to establish the above stated inversion strategies within integrated systems or toolboxes providing flexibility in choosing inversion methods and in general follow modularized approaches to solving the seismic inverse problem [12–16].

As one variety of Gauss–Newton FWI, the scattering-integral (SI) method [9,17] is particularly suitable for modularization, but is not considered by one of the above stated modular approaches. The fundamental steps of solving the forward problem and deriving a model update can be naturally decoupled, since the computation of the involved derivatives of waveform data with respect to the model parameters (called *waveform kernels*) is done by combination of the wavefield originating from the seismic source with Green's functions originating from the receiver positions that are independent of actual measured seismograms. Green's functions can be re-used in different source–receiver combinations serving as generalized backpropagations. This motivates to pre-compute the required wavefields and store them to hard disk before computing the waveform kernels. As a consequence, it becomes possible to solve the forward problem independently using established wave propagation codes, which are connected to the inversion algorithm by a suitable interface. Hence, this approach allows to independently develop inversion concepts and regularization methods on one hand, and to develop the in general demanding forward codes, e.g. with the objective of computational performance, on the other hand.

Furthermore, the general separation of solving the forward problem and computing the waveform kernels/deriving a model update, strongly suggest to introduce independent spatial model descriptions for solving the forward problem and for approaching the inversion step, as this is highly beneficial for the overall regularization of the inverse problem and hence the convergence of the iterative solution (also compare [18, sec. 3.2]). In Schumacher et al. [11] we chose this novel approach also in order to make scattering-integral-based FWI more computationally feasible. Naturally, a very modular inversion process arises that we implemented in the software package ASKI in an accordingly modular object-oriented fashion. ASKI stands for *Analysis of Sensitivity and Kernel Inversion* and offers the user a platform to solve various seismic FWI problems as well as resolution and sensitivity analysis within a modular, internally consistent, flexible and extendable computational environment.

In this paper, we describe the functionalities that ASKI offers, how these are implemented, how a researcher may use and possibly extend ASKI, and which benefits and challenges arise from the modular structure of ASKI for both, users and developers.

2. ASKI in general

ASKI is a toolbox for sensitivity and resolution analysis as well as for solving FWI problems in an iterative fashion by the SI method based on waveform sensitivity kernels. These kernels constitute a connection between waveform data samples and model values by quantifying how a certain data sample changes if a certain model parameter value is perturbed. For more details on the waveform sensitivity kernels used by ASKI and formulae how to compute them, we refer to Schumacher et al. [11, esp. appx. A2]. The computation of the kernels requires spectral wavefields originating from seismic sources and, independently, Green's functions originating from the receiver components. The scattering-integral-based waveform inversion implemented by ASKI is conceptually of very modular nature due to a very strict organizational separation of the three basic steps of solving the forward problem (called "stage I" in [11, sec. 3]), computing waveform sensitivity kernels ("stage II") and deriving a model update ("stage III"). Based on the sensitivity kernels computed at stage II, any sensitivity and resolution analysis can be conducted, having the full sensitivity matrix at hand. These three stages are illustrated in Fig. 1.

ASKI does not solve the seismic forward problem internally, but instead provides interfaces to existing forward codes to compute the required wavefields. Supported forward codes are, at the moment, the 1D semi-analytical code Gemini [21] and the 3D spectral-element code SPECFEM3D [22] for both, Cartesian and spherical framework, as well as the 3D nodal discontinuous-Galerkin code NEXD [23]. Extension to other forward codes is planned.

In order to make scattering-integral-based waveform inversion computationally more feasible and to approach the inverse problem in a more natural way based on the resolving power of the inverted seismic data, ASKI uses a volumetric spatial representation of the model space (called *inversion grid* in ASKI) that is independent of the model description for solving the forward problem, which is assumed by ASKI to be a point grid and is called *wavefield points* (cp. [11, sec. 3.1]). Very different kinds of inversion grids are provided by ASKI, accounting for complexity and geometrical scale of the particular inverse problem to solve. Additionally, we suggest in Schumacher et al. [11, sec. 3.2] to do the inversion step in the frequency domain, which is why ASKI computes frequency-domain sensitivity kernels from spectral wavefields provided by the forward codes.

At stage III, the inversion procedure allows to account for regularization terms of the misfit criterion to be optimized and to discard particular data samples of the data set or apply a specific weight to each datum. It is even possible to alter the misfit criterion as a whole, at this stage. ASKI, therefore, provides options to apply any regularization conditions to the inversion step that are representable as linear equations of the model update values, in particular smoothing and damping. At relatively low costs the computation of a model update can be repeated applying different regularization or data weighting/selection.

3. ASKI from a user's perspective

Fig. 2 shows a simplified workflow of main ASKI operations. The software package ASKI consists of numerous independent executables and scripts that communicate by input/output of files and can be composed to customized workflows of iterative FWI as well as sensitivity and resolution analysis. ASKI is controlled by input parameter files and operated by calling the executables.

For a particular workflow of FWI or sensitivity/resolution analysis, a user must set a parameter file that specifies all general information that will not change throughout iterations of full waveform inversion (if there are any) and from which locations of all files and directories used by the workflow can be inferred. Therefore, it is called the main parameter file (Fig. 2, Ⓐ) and it is required as input to almost all ASKI executables. Along with some conventions on nomenclature, all files required by an executable