
ESTIMATION OF UNCERTAINTIES OF FULL MOMENT TENSORS USING 1D AND 3D EARTH MODELS

Carl Tape

**University of Alaska Fairbanks
Geophysical Institute
2156 Koyukuk Dr.
Fairbanks, AK 99775**

20 September 2021

Final Report

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.



**AIR FORCE RESEARCH LABORATORY Space
Vehicles Directorate
3550 Aberdeen Ave SE
AIR FORCE MATERIEL COMMAND
KIRTLAND AIR FORCE BASE, NM 87117-5776**

DTIC COPY

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by AFMC/PA and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (<http://www.dtic.mil>).

AFRL-RV-PS-TR-2021-0101 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//SIGNED//

Dr. Frederick R. Schult
Program Manager, AFRL/RVBN

//SIGNED//

Erin N. Pettyjohn, Chief
AFRL Geospace Technologies Division

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 20-09-2021		2. REPORT TYPE Final Report		3. DATES COVERED (From - To) 01 Jul 2017 – 01 Jul 2021	
4. TITLE AND SUBTITLE Estimation of Uncertainties of Full Moment Tensors using 1D and 3D Earth Models				5a. CONTRACT NUMBER FA9453-17-C-0025	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER 62601F	
6. AUTHOR(S) Carl Tape				5d. PROJECT NUMBER 1010	
				5e. TASK NUMBER EF129470	
				5f. WORK UNIT NUMBER V133	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Alaska Fairbanks Geophysical Institute 2156 Koyukuk Dr. Fairbanks, AK 99775				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Space Vehicles Directorate 3550 Aberdeen Avenue SE Kirtland AFB, NM 87117-5776				10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/RVBN	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S) AFRL-RV-PS-TR-2021-0101	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited (AFRL-2021-4451 dtd 15 Dec 2021).					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT Uncertainty quantification is especially important for small events, such as chemical explosions, and for full moment tensors, for which the model parameter space is six-dimensional, rather than four-dimensional for a typical earthquake shear fault. In seismology we are often at the limit of signal-to-noise, and the quantification of uncertainties is necessary to allow one to make statistical-based interpretations. This project has three objectives: (1) to develop a modern computing framework for full moment tensor inversion and uncertainty quantification for 1D and 3D Earth models; (2) to understand uncertainty quantification of moment tensors in realistic, synthetic numerical experiments; and (3) to quantify the uncertainty in source types for Nevada Test Site explosions and earthquakes.					
15. SUBJECT TERMS Seismic moment tensor, seismic source mechanism, seismic discrimination					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			Dr. Frederick R. Schult
			Unlimited	58	19b. TELEPHONE NUMBER (include area code)

This page is intentionally left blank.

Table of Contents

List of Tables	i
List of Figures	ii
1 Summary	1
2 Introduction	2
2.1 The lune of eigenvalues for characterizing moment tensor source type	3
2.2 AxiSEM	3
2.3 SPECFEM3D	4
3 Technical approach	4
3.1 Target data sets	4
3.2 Proposed tasks	4
4 Results and discussion	7
4.1 Advances in understanding moment tensor source types	7
4.2 Moment tensor analyses using previous codes	7
4.3 The state of the MTUQ software package	8
4.4 Summary of activities for the proposed tasks	21
4.5 Presentations and dissemination of results	23
References	44

List of Tables

1	Codes for synthetic seismograms and moment tensor estimation	24
---	--	----

List of Figures

1	The eigenvalue lune for characterizing moment tensors	25
2	Probability density of random moment tensors	26
3	Illustration of subsets on the eigenvalue lune	27
4	Angular diameter of subsets on the lune	27
5	Uncertainty characterization for the 2017-09-03 North Korea test (with polarities) .	28
6	Uncertainty characterization for the 2017-09-03 North Korea test (without polarities)	29
7	Explosion, collapse, and earthquake events on the lune	31
8	Decompositions of full moment tensors for the six explosion events	32
9	Lune representation of Figure 8 for the six North Korea explosion events	33
10	Characterization of the source mechanisms of the six explosion events	34
11	Rayleigh wave time shift maps for the 2017-09-03 explosion event	35
12	Screenshot of the MTUQ repository	36
13	Screenshot of MTUQ documentation	37
14	Screenshot of MTUQ documentation	38
15	Screenshot of MTUQ documentation	39
16	Synthetic moment tensor inversion using MTUQ	40
17	Force inversion using MTUQ	41
18	Finite-element mesh for 3D seismic wavefield simulations	42
19	AGU poster conveying MTUQ	43

Note to reader: The pdf version of this document contains many hyperlinks that will direct the user to different parts of the document or to external websites. (If you jump to a different part of the pdf, you can return back by typing Command + Left Arrow.) Many of these websites are associated with the software repository MTUQ, which was developed under this contract. Given the nature of software development, we expect that the website urls will evolve with time, which will break the links in the pdf. The report is written to be comprehended without following any links, but the (functioning) links may be useful to some readers.

1 Summary

The moment tensor, a 3×3 symmetric matrix characterized by 6 model parameters, is a fundamental tool for characterizing earthquakes, nuclear explosions, mine collapses, volcanic events, hydraulic fracturing, and other events within Earth’s crust. However, even with dozens of stations in regional settings, it remains challenging to obtain reliable estimates of the moment tensor. Recent studies have demonstrated the importance of searching the full model parameter space. The studies have shown the large range of permissible solutions by plotting a waveform misfit function for realistic scenarios such as the explosions in North Korea (*Ford et al.*, 2010, 2012). Furthermore, they show that moment tensor uncertainties can have complex patterns and correlations.

Uncertainty quantification is especially important for small events, such as chemical explosions, and for full moment tensors, for which the model parameter space is six-dimensional, rather than four-dimensional for a typical earthquake shear fault. In seismology we are often at the limit of signal-to-noise, and the quantification of uncertainties is necessary to allow one to make statistical-based interpretations.

This project had three objectives:

- 1. to develop a modern computing framework for full moment tensor inversion and uncertainty quantification for 1D and 3D Earth models**
- 2. to understand uncertainty quantification of moment tensors in realistic, synthetic numerical experiments**
- 3. to quantify the uncertainty in source types for Nevada Test Site explosions and earthquakes**

The objectives build upon our recent work that established a sound theoretical basis for estimating uncertainties of full moment tensors (*Tape and Tape*, 2015, 2016; *Silwal and Tape*, 2016; *Alvizuri and Tape*, 2016). We now turn toward realistic synthetic numerical experiments and to real data sets in order to better understand moment tensor uncertainties.

Our project led to the establishment of the software package MTUQ (Moment Tensor Uncertainty Quantification), led by postdoc Ryan Modrak. This software package is the primary deliverable from the project. We also applied recent techniques of moment tensor uncertainty quantification to nuclear events recorded at the Nevada Test Site, as well as recent events from North Korea, including the largest of six events, which occurred in September 2017. Results were presented at professional meetings and in three publications (*Alvizuri et al.*, 2018; *Alvizuri and Tape*, 2018; *Tape and Tape*, 2019).

2 Introduction

The seismic moment tensor remains a fundamental component of source characterization in seismology (*Aki and Richards*, 1980). As a 3×3 matrix, it is represented by only six numbers that are estimated by using recorded seismic waveforms. For regional or global arrays with dozens of stations recording full waveforms on three-component seismograms, it may seem somewhat surprising that the uncertainties associated with the six moment tensor parameters are quite large. Many published studies or catalogs do not list uncertainties. And for regional or global earthquake catalogs, typically a constraint on the moment tensor is imposed, either that the moment tensor is deviatoric (e.g., *Dziewonski et al.*, 1981; *Sipkin*, 1986; *Dreger et al.*, 1998) or a double couple (e.g., *Zhu and Helmberger*, 1996).

There has been a growing number of observational studies of exotic events that have been modeled as full moment tensors (*Dreger et al.*, 2000; *Walter et al.*, 2009; *Šilený and Milev*, 2006; *Ford et al.*, 2009; *Baig and Urbancic*, 2010); see review in *Tape and Tape* (2013, Figures 14 and S14). Typically the objective of each study is to determine a moment tensor that generates a set of synthetic waveforms that is the closest match to a set of observed waveforms. Then the full moment tensor solution is compared to the best double couple solution to make a statistical inference about whether the source had an isotropic component or not.

Until recently, there had been limited progress on the quantification of moment tensor uncertainties, both for double couples (4 parameters) and for full moment tensors (6 parameters). There have been attempts to generate posterior samples of a solution or to represent a misfit function in a convenient space (*Wéber*, 2006; *Ford et al.*, 2009, 2010; *Walter et al.*, 2010; *Vavryčuk*, 2011; *Pesicek et al.*, 2012; *Dreger et al.*, 2012). We believe that our recent efforts *Tape and Tape* (2015, 2016) elucidate how to quantify moment tensor uncertainties, and they provide a foundation for our project.

Our project addresses the quantification of uncertainties for full moment tensors using 1D and 3D models. Moment tensor inversions using 3D models—requiring 3D wave propagation codes—have been performed in *Liu et al.* (2004); *Chen et al.* (2005); *Lee et al.* (2011). Synthetic tests to investigate the influence of 3D vs 1D structure on moment tensors have been done by *Hjörleifsdóttir and Ekström* (2010); *Covellone and Savage* (2012); *Stierle et al.* (2014). In our project we performed a detailed investigation of full moment tensor uncertainties that encompasses both real inversions and synthetic inversions.

In principle, moment tensor inversion should be straightforward, as there are only a few key steps: (1) specify a structural model, (2) generate Green’s functions for some model of Earth’s structure, (3) define a misfit function, (4) find the best-fitting moment tensor over model parameter space. Yet there are many, many more codes in use that address different subsets of data or use different misfit function; for example, there are multiple branches of the “cut-and-paste” (CAP) codes developed by Lupei Zhu and colleagues (*Zhao and Helmberger*, 1994; *Zhu and Helmberger*, 1996; *Tan et al.*, 2010; *Zhu and Ben-Zion*, 2013; *Chen et al.*, 2015; *Zhu and Zhou*, 2016). (We ourselves have adapted *Zhu and Ben-Zion* (2013) to handle uncertainties.)

A summary of codes related to this proposal is presented in Table 1. Descriptions of the two key wave propagation codes, AxiSEM and SPECFEM3D, are provided in Sections 2.2 and 2.3.

2.1 The lune of eigenvalues for characterizing moment tensor source type

Our recent theoretical work builds upon the simple concept that the three-dimensional space of eigenvalues of moment tensors is the most natural coordinate system for characterizing source types of moment tensors (*Tape and Tape*, 2012a,b,c, 2013), and it has also been used for discriminating between different source types (*Ford et al.*, 2020). The “fundamental lune” is a sector of this 3D eigenvalue space (with $\lambda_1 \geq \lambda_2 \geq \lambda_3$) that covers all possible moment tensor source types; an example is shown in Figure 1.

The lune is not merely a useful visual representation for source types; it also facilitates the exploration of uncertainties. For example, angles on the lune are identical to angles in matrix space between moment tensors having the same frame (or basis). In *Tape and Tape* (2015) we defined a parameterization that produces uniformly distributed moment tensors, which allowed us to develop a method for calculating a confidence parameter for moment tensors (*Tape and Tape*, 2016). As a first application, we calculated confidence parameters for double-couple moment tensors for 21 Alaska earthquakes *Silwal and Tape* (2016).

2.2 AxiSEM

Global wave propagation is still a computationally daunting task, and especially for high frequencies and inversion purposes, it will be difficult to apply to large datasets for years to come. However, a vast fraction of seismic data can be captured by Earth structures with first-order discontinuities. As such, all global tomography and source inversion models have been based on spherically symmetric earth models. We use the spectral-element solver AxiSEM (www.axisem.info) and its database descendant instaseis (www.instaseis.net) (*van Driel et al.*, 2015) by Collaborator T. Nissen-Meyer to devise a robust moment-tensor inversion framework based on global waveforms across the frequency band.

These full-wave propagation methods operate efficiently across the vast observable frequency spectrum of global waves reaching 1 Hz, and account for the full relevant range of physics in anisotropic, attenuative media. The benefit lies in the resultant dimensional collapse to two numerical dimensions for axisymmetric media, whereby the third azimuthal dimension is tackled analytically. For high-frequency wave propagation, this leads to 34 orders of magnitude speedup in computational cost compared to 3D domain discretizations. AxiSEM is a mature, robust, and benchmarked code, highly scalable anywhere between laptops and supercomputers, open-source and used by dozens of groups worldwide (*Nissen-Meyer et al.*, 2014).

Instaseis extracts waveforms from AxiSEM-based Green’s tensor databases. This instantaneous “once-and-for-all solution” relies on reciprocity and requires only two AxiSEM simulations to construct the databases, while allowing for arbitrary parameter changes (e.g. source-receiver geometry, moment-tensor, source-time function, processing) with negligible computational cost and storage requirements. The code is integrated with ObsPy (www.obspy.org) and contains a graphical user interface. Instaseis drives syngine, a novel IRIS DMC web-service for on-demand synthetics (<http://ds.iris.edu/ds/products/syngine>) (*Krischer et al.*, 2017). Synthetics are accessible for user-specified source-receiver combinations and a variety of models (PREM, IASP91, AK135f) with anisotropy and attenuation. AxiSEM is currently being generalized to local and regional wave propagation such that locally adapted databases (e.g. Europe, North America) will be feasible at higher frequencies. Another new development of AxiSEM incorporates 3D structures within a sparse context, such that wave propagation in smooth 3D tomographic models can be achieved

about two orders of magnitude faster than with 3D discretizations. This code is fully benchmarked against SPECFEM, showing excellent fits for 3D models such as S40RTS.

2.3 SPECFEM3D

For seismic wave propagation in 3D media we use SPECFEM3D, a spectral-element method code for solving the wave equation on unstructured, hexahedral finite-element meshes (*Komatitsch and Tromp, 2002a,b; Komatitsch et al., 2004; Peter et al., 2011*). SPECFEM3D, maintained by the Computational Infrastructure for Geodynamics (www.geodynamics.org), is a global-standard open-source code for seismic wave propagation. SPECFEM3D is widely used within seismology and has played a leading role in benchmarking clusters and new hardware, such as the NVIDIA Tesla GPUs (*Carrington et al., 2008; NVIDIA, 2012; NCSA, 2013*).

At UAF we use SPECFEM3D regularly for seismic wave simulations of earthquakes in Alaska, California, and New Zealand. SPECFEM3D has also been used within seismic imaging problems at local (*Tape et al., 2009, 2010*), regional (*Zhu et al., 2012*), and global (*Bozdağ et al., 2016*) scales.

3 Technical approach

Computational advances have afforded three powerful developments in seismology: (1) the processing of massive data sets of seismic waveforms, such as cross correlations of the ambient noise wavefield for crustal imaging (e.g., *Shapiro et al., 2005; Lin et al., 2012, 2013*); (2) the use of highly accurate simulations for calculating synthetic seismograms within forward models (*Komatitsch et al., 2002, 2004*) and inverse models (*Tape et al., 2009, 2010*); (3) Monte Carlo estimation of uncertainties for relatively small model parameter spaces (e.g., *Wéber, 2006; Shen et al., 2013*). Our project falls into the second and third categories.

3.1 Target data sets

We focus on three data sets that span a range of event types, magnitudes, and settings. These serve the basis for the synthetic inversions and for the real inversions, in particular, the quantification of uncertainty in source type of the moment tensor. The data sets are:

1. Nevada Test Site explosions and earthquakes (*Walter et al., 2006*).
2. 63 small (M_w 0.5–2.8), shallow events from Uturuncu volcano (*Alvizuri and Tape, 2016*)
3. 21 moderate (M_w 4–5.5) earthquakes from southern Alaska (*Silwal and Tape, 2016*)

Each of these data sets requires a different 1D model. For the Nevada Test Site we follow *Ford et al. (2009)* and use *Song et al. (1996)*, for the Uturuncu events we use a homogeneous model, and for the Alaska earthquakes we use a standard model for southern Alaska (scak: *Matumoto and Page, 1969; Lahr, 1975*).

3.2 Proposed tasks

Here we list the tasks originally proposed for this effort. In Section 4.4, we summarize our activities for each task.

Task 1: Uncertainty quantification of source type (theory)

Tape and Tape (2015, 2016) and *Silwal and Tape (2016)* established how to quantify confidence in a moment tensor solution. A key insight was that the fractional volume of moment tensor space is highly non-uniform as a function of matrix angle ω ranging from 0° to 180° . (In fact, the uniform distribution has the functional form $\sin^4 \omega$.) In the supplement of *Tape and Tape*

(2016), we presented equations for extending the quantification of uncertainties of moment tensors to quantification of source types. Source types are of great interest for distinguishing between different processes, such as explosions, earthquakes, tensile cracks, and more. In this task, we adapt our codes to quantify uncertainties in source types.

Task 2: Uncertainty quantification of source type for real data, I

We re-do all moment tensor inversions for all three data sets to investigate how uncertainties in source type relate to previously quantified uncertainties in moment tensors (*Silwal and Tape, 2016*). We expect that large uncertainties in moment tensors will generally result in large uncertainties in source type, but this needs to be verified.

Task 3: Establish python sequence of codes for moment tensor inversion

No current moment tensor code has the capabilities that we desire for maximizing productivity, reproducibility, and interoperability. However, all the capabilities do exist within different codes (Table 1). This task involves stitching all the components together into a python-based workflow for moment tensor inversions. The resulting software bridges ObsPy (*Krischer et al., 2015*) with AxiSEM (*Nissen-Meyer et al., 2014*), whereby ObsPy handles all seismogram fetching (from IRIS Data Management Center) and processing, and AxiSEM/syngine (Section 2.2) generates synthetic seismograms. ObsPy and AxiSEM are connected to a grid search moment tensor inversion code such as GRD_CMT3D by Qinya Liu (*Liu et al., 2004; Tape et al., 2009*) or something similar by the group of Tape or Nissen-Meyer.

Task 4: Implement our grid search and our uncertainty quantification into the python codes

Once we have tested the ObsPy + AxiSEM + moment tensor inversion workflow, we adapt the grid search and uncertainty quantification (*Silwal and Tape, 2016*) for the python codes. This is a code-translating task, as we already have the algorithms working in less-functional, less-user-friendly codes (C + Matlab + GMT + shell scripts). The investment in code improvements should facilitate better science, better analyses, and better results. Specific modifications are for the parameterization of model space (*Tape and Tape, 2015*) and ensuring that the misfit function is comparable to what we have previously used.

Task 5: Perform inversions using AxiSEM and real data

Perform inversions and uncertainty analysis with events in all three data sets using AxiSEM. The results should be similar to what we obtained in our ongoing project. But the new approach provides the advantage of spherical earth synthetics, as well as the option for radial anisotropy (i.e., transverse isotropy). Furthermore the code will be in better position for extensions to using 3D synthetic seismograms or to operational settings.

The combination of AxiSEM and instaseis/syngine is ideal for source inversion as it allows for immediate access to any changes in any relevant source parameters and already accommodates seven global Earth models stored at IRIS.

The AxiSEM wavefields that constitute the wavefield databases used for `instaseis` consist of the full Green tensor, i.e., separate solutions for each of the six independent moment tensor elements and three receiver components, for every earthquake depth and any receiver distance. This is accomplished by using reciprocity, such that the generation of the database boils down to a mere two AxiSEM simulations of point forces replicating the receiver components, and saving

the entire displacement field in 4D. The python wrapper `instaseis`, which is directly linked with Obspy (www.obspy.org) then computes the strain tensor upon the fourth-order polynomial basis of the AxiSEM spectral-element mesh, and performs tasks such as assembling the Green tensor to a specific moment tensor, frequency filters, time resampling.

Building a robust, standard tool for moment tensor inversion around this existing framework is not only feasible and timely to accomplish, but will also be of wide use among the growing user base of `instaseis`.

Task 6: Uncertainty quantification of source type for real data, II

This is repeat of Task 2, but here we use the new codes. All events from all three data sets—LLNL events (nuclear and earthquakes), Uturuncu events, and Alaska earthquakes—would be done.

Task 7: Perform inversions using AxiSEM and synthetic data, plus uncertainty quantification of source type

This is a repeat of Tasks 5 (inversions) and 6 (uncertainty in source type), but here we would perform synthetic inversions, whereby the “data” are generated using the best-fitting full moment tensor from Task 5. This would provide an interesting comparison between source type uncertainties for inversions with real data and source type uncertainties for inversions with synthetic data, whereby the source-station geometry would be the same for each case. The comparison would reveal the influence of real data (and real structure) on the estimation of source types.

The synthetic inversions will allow us to explore a range of effects on moment tensor uncertainties such as the influence of the number of stations, the geometry of stations, the choice of frequency band, the choice of body waves or surface waves, and the influence of idealized (or not) errors introduced into the synthetic data. These are all details of the misfit function that are generally difficult to assess within real inversions, where the misfit may be attributed to unknown Earth structure or unknown data errors. These are both controlled in synthetic inversions.

Task 8: Mesh and model wavefield for 3D velocity model

A pragmatic approach to investigating the influence of heterogeneities is to construct a 3D velocity model from the combination of a layered model (e.g., *Song et al.*, 1996) with stochastic heterogeneities added, similar to the approach of *Frankel and Clayton* (1986); *Baig et al.* (2003). Such an approach enables one to carefully control the amplitude and length scales of the heterogeneities in the model. A finite-element mesh will be constructed for the 3D model for seismic wavefield simulations with SPECFEM3D (Section 2.3) (*Casarotti et al.*, 2008; *Peter et al.*, 2011).

Task 9: Synthetic inversions in 1D and 3D models (with 3D synthetics)

Use the 3D wave propagation code (SPECFEM3D) to generate Green’s functions for the moment tensor inversions (e.g., *Liu et al.*, 2004). This task will involve significant computations: hundreds of hour-long simulations on hundreds of computing cores. However, once the catalog of Green’s functions are precalculated, we can extensively explore the moment tensor inversions at no additional computational cost.

We first test for the influence of 3D Earth structure by computing a set of synthetic seismograms using a 3D Earth model and the moment tensor solutions derived from Task 5. We will then perform the moment tensor inversion and uncertainty quantification using a 1D Earth model. Similar efforts have been previously attempted (*Covellone and Savage*, 2012), though not associated with full moment tensors (or uncertainties), as proposed here.

4 Results and discussion

Our results are presented in four parts. In Section 4.1, we summarize recent advances in representing moment tensor source types. In Section 4.2, we present applications and analyses that use older (or “legacy”) codes. A major component of the project was to establish a new software package (MTUQ) that contained the core capabilities of the previous codes but that also enabled expanded capabilities and development; MTUQ is summarized in Section 4.3. Finally, in Section 4.4 we summarize the specific outcomes for each of the proposed tasks (Section 3.2).

4.1 Advances in understanding moment tensor source types

The *vw* parameterization in *Tape and Tape* (2015) laid the foundation for calculating probability density functions for moment tensor source type. Once the parameterization was identified, we then did the following steps: (1) uniformly sample the coordinate domain (a “sample” is a 5-tuple that specifies a fixed-magnitude moment tensor), (2) evaluate the misfit function at each sample and convert it to a probability, (3) sum up the values in each *vw* rectangle and normalize to create a probability density function.

Figure 1 provides a stark reminder of the impact of the choice of moment tensor parameterizations. In each case, the 2D plot represents moment tensor source types. The functions show the distribution for a random set of moment tensors. An ideal parameterization would display a flat (constant) distribution. This is achieved with the *vw* rectangle. The lune exhibits a pattern that was previously only qualitatively known (*Hudson et al.*, 1989; *Ford et al.*, 2010; *Chapman and Leaney*, 2012); in *Tape and Tape* (2019) we derived the analytical expression. Practically speaking, it means that if you have a posterior distribution of moment tensors on the lune, you need to interpret it in the context of the random distribution, which is not flat.

The paper of *Tape and Tape* (2019) elaborates on the advantages of using the lune to characterize subsets—such as the subsets that are used to represent moment tensor uncertainties (e.g., *Mustać and Tkalčić*, 2016). Figure 3 makes the simple point that a subset on the lune near the isotropic point will have a much smaller angular radius than a subset—of the same angular width—that lies near the double couple. This point is reinforced in Figure 4, which shows the angular diameter of the set of moment tensors for each source type on the lune. As expected, this varies from 0° at the isotropic points to 180° at the double couple, at center.

4.2 Moment tensor analyses using previous codes

All publications from this project are listed in Section 4.5. Our JGR 2018 paper (*Alvizuri et al.*, 2018) was part of Alvarez’s PhD research (*Alvizuri*, 2016), which was supported by a number of projects, including our previous AFRL contract (FA9453-15-C-0063). The two major results from that paper were: (1) a new presentation of uncertainties for full moment tensors, showing the probability density function for moment tensor source types, and (2) a complete, revised presentation of the events in *Ford et al.* (2009), along with extensive documentation of results.

The foundational work in *Alvizuri et al.* (2018) enabled our investigation of the North Korea events, including the largest event, which occurred on 2017-09-03, two months into the project. Our results were published in the 2018 *Seismological Research Letters* special issue on the North Korea events, and we include several figures in this report. Figure 5 shows our primary display for

moment tensor estimation and uncertainty quantification for a single event, in this case, the largest North Korea nuclear explosion. It can be compared with Figure 6, which differs from Figure 5 in that no first-motion polarities were used. As discussed in *Alvizuri et al.* (2018), the inclusion of polarities reduces the uncertainties in moment tensor source type.

Figure 7 presents the best-fitting moment tensors for 9 events in the North Korea region: 6 nuclear explosions, 2 earthquakes, and 1 collapse. The presentation follows the tradition *Ford et al.* (2009) of displaying these three event types in the same plot of moment tensor source type.

Once a full moment tensor is estimated for a particular event, there are several options for how to proceed with interpreting a physical process. Some of the options that we advocate are not found in the literature, while other approaches that are common in the literature are not advocated by us. In Figure 8 we present two physically-based decompositions for each of the 6 North Korea events (ν - α and ϕ - ζ), and these decompositions are depicted graphically in Figure 9. One of these decompositions is the crack-plus-double-couple model, which is featured in Figure 10. Under the assumption of this two-process model, we find that the crack tensors tend to have their primary principal axis pointing near vertically; such a model had been previously advocated by *Patton and Taylor* (2011) and *Liu et al.* (2018).

A common challenge of estimating moment tensors is that the assumed Earth models are often overly simplified (such as 1D layered models), resulting in large and variable time shifts between data and synthetics. These time shifts pose problems of cycle skipping (misalignment of data and synthetic waveforms) that can impact the estimation of uncertainties for the source type (*Alvizuri et al.*, 2018). Figure 11 shows the Rayleigh wave time shifts associated with the best-fitting moment tensor for the largest North Korea event. These time shifts, and the hazards they pose for moment tensor estimation, provided motivation to establish the new moment tensor code with capabilities for interfacing with synthetic seismograms made from 3D seismic velocity models.

4.3 The state of the MTUQ software package

This section is augmented by screenshot examples from the MTUQ documentation (Figures 12–15).

Overview

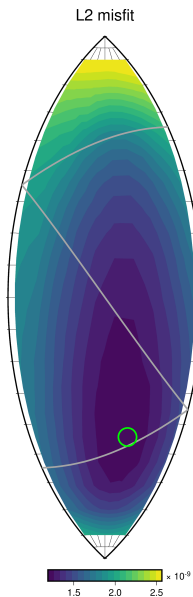
MTUQ provides *moment tensor* estimation and *uncertainty quantification* capabilities through a flexible Python toolbox.

Moment tensor estimation involves up to six independent moment tensor elements, plus additional hypocenter location and origin time parameters. Quantifying uncertainty over these high-dimensional spaces can be computationally expensive, so MTUQ aims to provide very fast misfit evaluation and model space exploration functions. In practice, these functions are fast enough that global searches or even detailed grid searches or are usually possible.

After searching over a parameter space, interpreting the resulting misfit or likelihood surfaces can be challenging, again because of high model space dimensionality. To draw meaningful interpretations from high-dimensional misfit surfaces, MTUQ provides a variety of visualization utilities. Many examples are provided in Section 4.3.

Full moment tensor inversion example

Perhaps the most common MTUQ use case is to carry out a search over full moment tensors (involving six independent parameters), and plot the results using the eigenvalue lune as follows:



For details about how the above figure was generated, and for instructions on how to perform uncertainty analyses for other seismic events, see Section 4.3.

Possible research applications

Through flexible design and modern programming practices, MTUQ tries to facilitate a wide range of research.

Possible applications include searches over full moment tensors, double couple moment tensors, and other constrained moment tensor spaces; searches over force orientation and magnitude; and searches over hypocenter, depth, origin time, and source-time function.

As a starting point for all these different types of uncertainty quantification, MTUQ includes an efficient implementation of the “cut and paste” misfit function popularized by *Zhu and Helmberger* (1996). Users can evaluate this misfit function over regularly- or randomly-spaced grids of moment tensors, forces, or origins. Alternatively, users can substitute their own custom misfit function or their own custom parameter grids.

In evaluating misfit, MTUQ compares observed seismic data with synthetic Green’s functions generated from an elastic wavefield solver. A number of elastic solvers are currently supported, including FK (*Zhu and Rivera*, 2002), AxiSEM (*Nissen-Meyer et al.*, 2014), SPECFEM3D (*Komatitsch et al.*, 2004) and SPECFEM3D_GLOBE (*Komatitsch and Tromp*, 2002a,b).

MTUQ misfit functions can be used as a starting point for Bayesian uncertainty analysis with *pymc* or other external MCMC libraries; however, MCMC libraries are not presently included in MTUQ itself.

Resources for users

An MTUQ user guide is hosted directly from the GitHub page. Included in the user guide are instructions for installing the package and for running a few representative examples.

The simplest of these examples is `SerialGridSearch.DoubleCouple.py`, which gives a sense for the basic data structures involved in an inversion.

By far the most involved example is `DetailedAnalysis.FullMomentTensor.py`, which generates a host of information, including separate data variance estimation of body waves, Rayleigh waves, and Love waves; misfit and variance reduction surfaces; marginal and maximum likelihood surfaces; geographic phase and amplitude error analysis; and much more.

Above, we have tried highlight features of MTUQ of that might be of interest to explosion monitoring users. A slightly broader and more technical overview, for a more general geophysics audience, is given at MTUQ’s GitHub README page. Experienced Python users may wish to jump directly to the online “Library Reference,” which gives the most complete, but also most technical, documentation of code capabilities.

In the following sections of the report, we borrow directly from the online documentation, trying to present code capabilities in a way that might be useful for prospective users.

To this end, we present topics in the order users are likely to encounter them, starting with data acquisition and Green’s function acquisition; continuing to data processing and misfit evaluation; and finishing with visualization and likelihood analysis.

Data acquisition

Acquiring and quality control checking seismic data can be one of the most challenging steps in an inversion. This page describes our preferred data client and MTUQ’s data and metadata format requirements.

Data acquisition clients

ObsPy FDSN clients provide a way of programmatically downloading waveforms from IRIS. PySEP is a wrapper around an ObsPy FDSN client and our preferred data acquisition tool.

Besides downloading waveforms through ObsPy calls, PySEP performs data processing and quality control, ensuring that the data processing and time discretization requirements below are satisfied.

Additionally, PySEP generates “weight files” that can optionally be used by `mtuq.process_data` to control the relative contribution of different stations to the data misfit function.

File format, metadata, and data processing requirements

SAC is currently the only file format supported by `mtuq.read`.

The following information must be present in the SAC metadata headers: *station latitude*, *station longitude*, *network code*, *station code*, *location code*

It is expected that the following data processing steps have been carried out prior to reading user-supplied waveforms by `mtuq.read`:

- instrument response has already been removed
- traces have already by rotated into vertical (Z), radial (R) and transverse (T) components
- all traces have the same sampling rate

Other file formats

- Besides SAC, many other file formats could be supported in principle. To make adding such support easier, `mtuq.read` follows the manner in which `obspy.read` works, by accepting a format

keyword argument that corresponds to a module in *mtuq.io.readers*.

Acquiring Green's functions

The response of a medium to an impulsive source is called a Green's function. This page describes the role of Green's functions in source inversions, the types of Green's functions supported by MTUQ, and how these types can be obtained.

Role of Green's functions

By combining Green's functions, it is possible to obtain synthetics from any moment tensor or force source. Synthetics can then be compared with data to determine a best-fitting source.

Generating synthetics through linear combinations of Green's functions is useful because it provides computational savings. It is more cost effective to compute Green's functions (an up-front expense, independent of the number of sources) than it is to perform an on-the-fly wavefield simulation for each new source.

GreensTensor objects

GreensTensor objects provide access to a set of Green's functions describing the medium response between a given station and origin.

Methods built into the *GreensTensor* class allow data processing and synthetics generation. In particular, the `get-synthetics` method accepts a *MomentTensor* or *Force* and returns traces representing the response at the given station.

Downloading precomputed Green's functions

An easy way to acquire Green's functions is to download them from *syngine*, a webservice that hosts Green's functions from the following 1D Earth models: PREM, AK135, and *iasp91*.

To download AK135 Green's functions and generate MTUQ *GreenTensor* objects:

```
from mtuq import download_greens_functions
greens_tensors = download_greens_functions(stations, origins, model="ak135f_2s")
```

A limitation of *syngine* is that Green's functions can be downloaded only on a station-by-station basis, not over an entire subset of Earth. An alternative for acquiring Green's functions, which avoids this limitation, is to perform your own Green's functions simulations.

Computing Green's functions from 1D Earth models

MTUQ supports the following 1D Green's functions formats: AxiSEM (preferred) and FK.

AxiSEM performs spectral element wave simulations in radially-symmetric Earth models. To generate synthetics in a format MTUQ natively supports, follow the instructions for in the AxiSEM user manual under "Output wavefields in NetCDF formatted needed by instaseis." AxiSEM NetCDF files can be used to retrieve vertical, radial, and transverse displacement in units of $\text{m} \cdot (\text{N} \cdot \text{m})^{-1}$.

To open an AxiSEM database client in MTUQ:

```
from mtuq import open_db
db = open_db(path_to_NetCDF_file, format="AxiSEM")
```

FK simulates wave propagation in horizontally-layered elastic media using a frequency-wavenumber method. FK simulations create SAC files in a directory tree organized by model, event depth, and

event distance. Each SAC file represents a vertical, radial, or transverse velocity time series in units of $10^{-20} \text{cm} \cdot (\text{dyne-cm})^{-1} \text{s}^{-1}$.

To open an FK database client in MTUQ:

```
from mtuq import open_db
db = open_db(path_to_FK_directory_tree, format="FK")
```

Once opened, an AxisEM or FK database client can be used to generate GreensTensor objects as follows:

```
greens_tensors = db.get_greens_tensors(stations, origin)
```

Computing Green's functions from 3D Earth models

MTUQ currently supports 3D Green's functions from SPECFEM3D. To generate a complete Green's function database for a given hypocenter and depth, six SPECFEM3D wavefield simulations are required. Output must be saved as SAC files at individual stations using the following filename convention, which comes from GRD-CMT3D. Place all SAC files corresponding to a single hypocenter and depth in the same directory as follows:

```
{basedir}/{event_id}/
  {net}.{sta}.{loc}.Z.Mrr.sac
  {net}.{sta}.{loc}.Z.Mtt.sac
  {net}.{sta}.{loc}.Z.Mpp.sac
  {net}.{sta}.{loc}.Z.Mrt.sac
  {net}.{sta}.{loc}.Z.Mrp.sac
  {net}.{sta}.{loc}.Z.Mtp.sac
  {net}.{sta}.{loc}.R.Mrr.sac
  {net}.{sta}.{loc}.R.Mtt.sac
  {net}.{sta}.{loc}.R.Mpp.sac
  {net}.{sta}.{loc}.R.Mrt.sac
  {net}.{sta}.{loc}.R.Mrp.sac
  {net}.{sta}.{loc}.R.Mtp.sac
  {net}.{sta}.{loc}.T.Mrr.sac
  {net}.{sta}.{loc}.T.Mtt.sac
  {net}.{sta}.{loc}.T.Mpp.sac
  {net}.{sta}.{loc}.T.Mrt.sac
  {net}.{sta}.{loc}.T.Mrp.sac
  {net}.{sta}.{loc}.T.Mtp.sac
```

To open an SPECFEM3D database client in MTUQ:

```
from mtuq import open_db
db = open_db(path_SPECFEM3D_output_directory, format="SPECFEM3D")
```

Once opened, a SPECFEM3D database client can be used to generate GreensTensor objects as follows:

```
greens_tensors = db.get_greens_tensors(stations, origin)
```

Green's function conventions

A variety of Green's function conventions exist. Figuring out which are used in a particular application can be challenging because it depends on

- the type of medium under consideration (for example, acoustic media require fewer independent Green's functions than elastic media)
- the symmetry of the medium (for example, 1D media require fewer independent Green's functions than 3D media)
- the choice of local Cartesian basis conventions (for example, some authors employ up-south-east, others north-east-down; see ObsPy documentation for more information)

A major goal is to avoid exposing users to unnecessary basis complexity. MTUQ accomplishes this by understanding external formats and converting to a common internal format that works for both 1D and 3D media.

For internally storing moment tensors, forces, and Green's functions, MTUQ consistently uses an up-south-east Cartesian convention.

Data processing

Filtering, windowing and other signal processing operations are essential for meaningful comparisons between data synthetics.

To make data processing easier, `mtuq.ProcessData` is an attempt at a one-size-fits-all data processing class. While the approach below based on `mtuq.ProcessData` is very general, users are nevertheless free to apply their own custom data processing functions instead, with the only condition being that the requirements below end up being satisfied.

An approach based on *mtuq.ProcessData*

Data processing choices include filter type, window length, and many others. For detailed descriptions of all available parameters, see the library reference.

With *mtuq.ProcessData*, the choice of data processing parameters is clearly separated from the application of the data processing function to the seismic data, resulting in a two-step procedure. In the first step, the user supplies parameters to create a data processing function:

```
process_data = ProcessData(**parameters)
```

In the second step, an ObsPy stream is given as input and a processed stream returned as output:

```
processed_stream = process_data(stream)
```

Data processing can also be applied to an entire *Dataset* at once:

```
processed_data = mtuq.Dataset()
for stream in raw_data:
    processed_stream = process_data(stream)
    processed_data += processed_stream
```

Or, more succinctly:

```
processed_data = raw_data.map(process_data)
```

Requirements for processed data

For computational efficiency and software engineering reasons, processed traces must satisfy:

1. all traces must have the same sampling rate and number of samples
2. all traces common to the same station must have the same starttime and endtime

These conditions help MTUQ achieve faster memory access patterns, more efficient misfit evaluation, and more easily maintainable code. An easy way to ensure that they are satisfied is to simply use *mtuq.ProcessData* for data processing.

Of course, to make meaningful comparisons between data and synthetics, processed data must also pass basic quality control checks. For example, data must contain actual body wave or surface arrivals from the given event. MTUQ's waveform plotting functions can sometimes be helpful for these types of checks.

Misfit evaluation

Waveform difference and cross-correlation time-shift misfit evaluation on body-wave and surface-wave windows is implemented in C-accelerated Python. This misfit function has been tested against legacy Perl/C codes as well as previous published studies.

These misfit functions can be used with *mtuq.grid* search, which automatically partitions the grid over multiple MPI processes if invoked from an MPI environment.

Our misfit function quantifies fit between data and synthetics using time shifts followed by waveform differences. This approach, due to *Zhao and Helmberger (1994)* and *Zhu and Helmberger (1996)*, has become widely used in regional seismology.

Evaluating misfit is a two-step procedure:

```
function_handle = Misfit(**parameters)
values = function_handle(data, greens, sources)
```

In the first step, the user supplies parameters such as the type of norm (see below for detailed argument descriptions). In the second step, the user supplies data, Green's functions, and sources. Synthetics are then generated and compared with data, and a NumPy array of misfit values is returned of the same length as *sources*.

- **norm** (*str*)
 - 'L2': conventional L2 norm (fast)
 - 'L1': conventional L1 norm (slow)
 - 'hybrid': hybrid L1-L2 norm (much faster than L1 but still robust)
- **time_shift_groups** (*list*)
 - ['ZRT']: forces all three components to have the same time shift
 - ['ZR', 'T']: forces vertical and radial components to have the same time shift, while transverse time shift is allowed to vary independently
 - ['Z', 'R', 'T']: allows time shifts of all three components to vary independently
- **time_shift_min** (*float*): minimum allowable time shift (s)
- **time_shift_max** (*float*): maximum allowable time shift (s)

Because misfit evaluation is our most computationally expensive task, we have implemented three different versions:

- a readable pure Python version (`mtuq.misfit.level0`)
- a fast pure Python version (`mtuq.misfit.level1`)
- a very fast Python/C version (`mtuq.misfit.level2`)

While having exactly the same input argument syntax, these three versions differ in the following ways:

- `level0` provides a reference for understanding what the code is doing and for checking the correctness of the fast implementations
- `level1` is an optimized pure Python implementation which provides significant computational savings for $\text{len}(\text{sources}) > 100$. This version is the closest to the original C software (*Zhu and Helmberger, 1996*).
- `level2` is an optimized Python/C implementation, in which a Python wrapper is used to combine ObsPy traces into multidimensional arrays. These arrays are passed to a C extension module, which does the main computational work. Unlike the other two versions, this implementation requires that all ObsPy traces have the same time discretization.

Note that during installation, C extension modules are automatically compiled by `build_ext.sh` using compiler flags given in `setup.py`. For performance tuning or compiler troubleshooting, users may wish to modify the `get_compiler_args` function in `setup.py`.

Visualization galleries

As illustrated in the examples, MTUQ can be used to carry out grid searches over moment tensors, double couples, and forces.

The output of `mtuq.grid_search` in each of these cases is a data structure that can be passed directly to visualization functions. Examples are given below.

Plotting moment tensor results

Suppose we are running a moment tensor grid search:

```
ds = grid_search(data, greens, stations, origins, sources)
```

After the above command finishes, the data structure `ds` will hold all the moment tensors and corresponding misfit values. For a grid of regularly-spaced moment tensors, `ds` may look something like:

```
>>> print(ds)
```

Summary:

```
grid shape: (10, 10, 20, 10, 10, 10, 1)
grid size:  2000000
mean: 2.548e-09
std:  2.851e-10
min:  1.328e-09
```

max: 3.640e-09

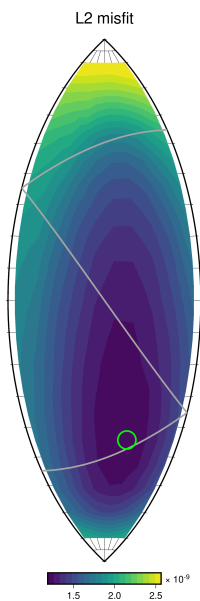
Coordinates:

```
* rho      (rho) float64 5.018e+15 5.418e+15 ... 9.272e+15 1.001e+16
* v        (v) float64 -0.3313 -0.2995 -0.2371 ... 0.2371 0.2995 0.3313
* w        (w) float64 -1.178 -1.176 -1.167 -1.141 ... 1.167 1.176 1.178
* kappa    (kappa) float64 18.0 54.0 90.0 126.0 ... 234.0 270.0 306.0 342.0
* sigma    (sigma) float64 -81.0 -63.0 -45.0 -27.0 ... 27.0 45.0 63.0 81.0
* h        (h) float64 0.05 0.15 0.25 0.35 0.45 0.55 0.65 0.75 0.85 0.95
* origin_idx (origin_idx) int64 0
```

Misfit values

To plot the misfit values returned by the grid search, we can pass `ds` to a plotting utility as follows:

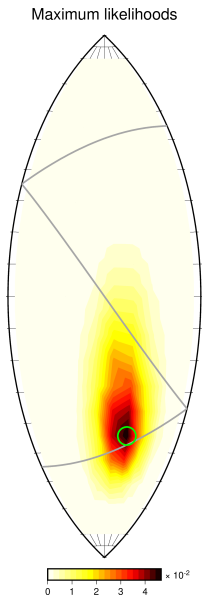
```
from mtuq.graphics import plot_misfit_lune
plot_misfit_lune(filename, ds)
```



Maximum likelihoods

If a data variance estimate *var* is available, then misfit values can be converted to likelihood values. In the following approach, a two-dimensional maximum likelihood surface is obtained by maximizing over orientation and magnitude parameters:

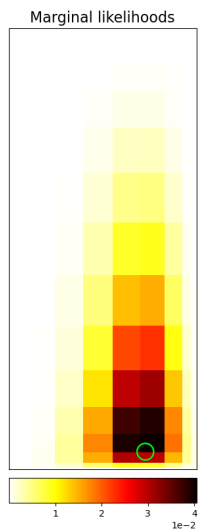
```
from mtuq.graphics import plot_likelihood_lune
plot_likelihood_lune(filename, ds, var)
```



Marginal likelihoods

An alternative for visualizing likelihood is integration over orientation and magnitude parameters, giving a two-dimensional marginal distribution. In this case, it is natural to use the so-called v,w parameterization, as follows:

```
from mtuq.graphics import plot_marginal_vw
plot_marginal_vw(filename, ds, var)
```

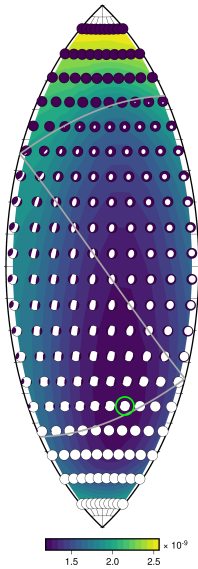


Tradeoffs between source type and orientation

To see how the orientation of the best-fitting moment tensor varies with respect to source type, we can add the `show_tradeoffs=True` option to any of the plotting utilities as follows:

```
from mtuq.graphics import plot_misfit_lune
plot_misfit_lune(filename, ds, show_tradeoffs=True)
```

Focal mechanism tradeoffs

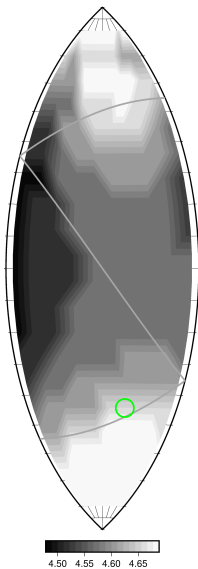


Tradeoffs between source type and magnitude

To see how the magnitude of the best-fitting moment tensor varies with respect to source type, we can use the following:

```
from mtuq.graphics import plot_magnitude_tradeoffs_lune
plot_magnitude_tradeoffs_lune(filename, ds)
```

Magnitude tradeoffs



Source code

[script to reproduce the 5 previous figures]

Users can run the script immediately after installing MTUQ, without any additional setup.

Plotting force results

Suppose we are running a force grid search:

```
ds = grid_search(data, greens, stations, origins, sources)
```

After the above command finishes, the data structure *ds* will contain all the forces and corresponding misfit values.

For a grid of regularly-spaced forces, *ds* may look something like:

```
>>> print(ds)
```

Summary:

```
grid shape: (200, 25, 25, 1)
grid size: 125000
mean: 2.984e-09
std: 1.428e-10
min: 2.605e-09
max: 3.339e-09
```

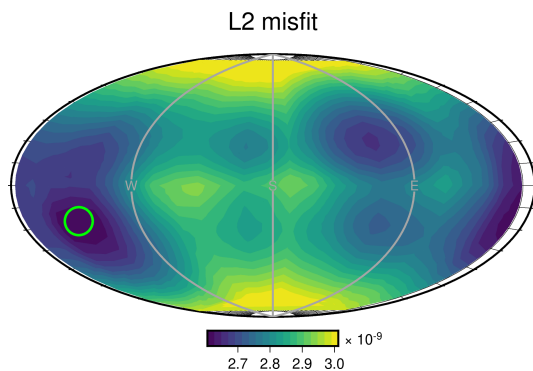
Coordinates:

```
* F0          (F0) float64 1e+11 1.012e+11 1.023e+11 ... 9.772e+11 9.886e+11
* phi         (phi) float64 7.2 21.6 36.0 50.4 ... 309.6 324.0 338.4 352.8
* h           (h) float64 -0.96 -0.88 -0.8 -0.72 -0.64 ... 0.72 0.8 0.88 0.96
* origin_idx  (origin_idx) int64 0
```

Misfit values

To plot the misfit values returned by the grid search, we can pass *ds* to a plotting utility as follows:

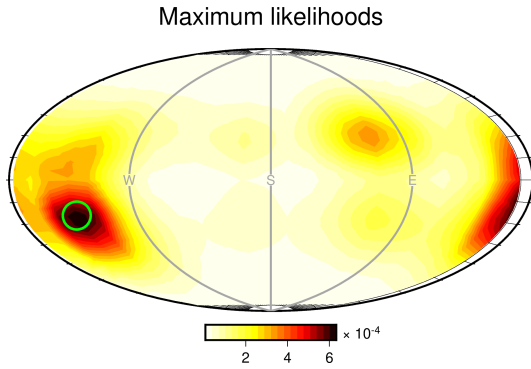
```
from mtuq.graphics import plot_misfit_force
plot_misfit_force(filename, ds)
```



Maximum likelihoods

If a data variance estimate *var* is available, then misfit values can be converted to likelihood values. In the following approach, a two-dimensional likelihood surface is obtained by maximizing over magnitude:

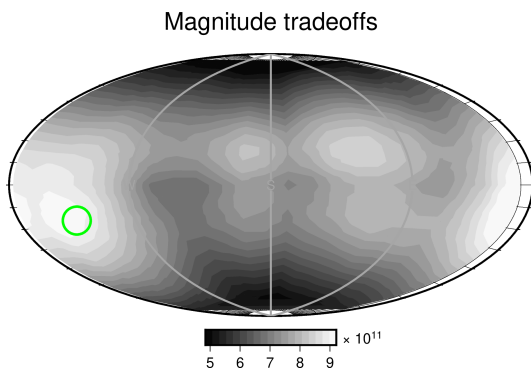
```
from mtuq.graphics import plot_likelihood_force
plot_likelihood_force(filename, ds, var)
```



Tradeoffs between force orientation and magnitude

To see how the magnitude of the best-fitting force varies with respect to orientation, we can use the following:

```
from mtuq.graphics import plot_magnitude_tradeoffs_force
plot_magnitude_tradeoffs_force(filename, ds)
```



Source code

[script to reproduce the three previous figures]

Users can run the script immediately after installing MTUQ, without any additional setup.

Getting Started

An easy way to get started is to try out these examples, which estimate source parameters for an event from *Silwal and Tape (2016)*.

Before running any of the scripts, users must first install MTUQ. After installation finishes, examples can be invoked within the *examples/* directory as follows:

```
python ./<name_of_example>
```

Note: The first time an example runs, Green's functions will be downloaded, which can take several minutes. Any subsequent runs will be much faster.

Serial example

The simplest example is `SerialGridSearch.DoubleCouple.py`. After running the script, users can check their output against this expected result.

A typical runtime is about 60 seconds. For faster results, `GridSearch.DoubleCouple.py` runs the same inversion in parallel.

Parallel examples

All the other examples include MPI parallelization for faster evaluation on parallel systems.

To see if MPI executables and libraries are present, start by checking if `mpirun` is on the system path. If necessary, ask a system administrator for assistance installing or configuring MPI.

If MPI is present, run the parallel examples as follows:

```
mpirun -n <nproc> python ./<name_of_example>
```

Note that the examples will still work if invoked without `mpirun`, but running times will be longer.

4.4 Summary of activities for the proposed tasks

Task 1: Uncertainty quantification of source type (theory)

We implemented a variation on an algorithm outlined in *Tape and Tape* (2016, Section S3.2). The procedure is straightforward: (1) generate N uniformly distributed moment tensors M_i ; (2) calculate the posterior probability $p_i = p(M_i)$ for all moment tensors (*Silwal and Tape*, 2016); (3) divide the vw rectangle into uniform cells, then sum up p_i within each cell. This procedure is described and presented in *Alvizuri et al.* (2018). See also Section 4.1.

Task 2: Uncertainty quantification of source type for real data, I

An example of the uncertainty in source type is shown in Figure 5c of *Alvizuri et al.* (2018). Another is shown below in Figure 5c (*Alvizuri and Tape*, 2018). See also Section 4.2

Task 3: Establish python sequence of codes for moment tensor inversion

This task is by far the largest of the contract, in terms of time commitment. The software we produced at part of this task is MTUQ (<https://github.com/uafgeotools/mtuq>), which is described in Section 4.3.

Task 4: Implement our grid search and our uncertainty quantification into the python codes

This task pertains to technical details of the implementation, with the goal of achieving performance of MTUQ that is comparable to the legacy code (`capuaf`). The grid search implementation is documented in Section 4.3. Much of the subsequent effort, following implementation, was to enhance the scalability and performance of the code. To make sure that the new python code scales well in a cluster environment, we compared various parallelization methods and adopted one that works for our application. To make sure the computational performance of the python code is acceptable, we identified and have started addressing bottlenecks through attention to numerical details and/or Cython acceleration. Code acceleration was accomplished using the Numpy/C API (<https://docs.scipy.org/doc/numpy/reference/c-api.html>).

Task 5: Perform inversions using AxiSEM and real data

The default moment tensor inversion in MTUQ uses AxiSEM synthetics (fetched via Syngine) and real data. One technical hurdle we faced was an issue in `instaseis`' Green's functions implementation that results in incorrect AxiSEM synthetics. The issue is described in detail at <https://github.com/krischer/instaseis/issues/66>. We were able to resolve this matter. We

also were able to calculate our AxiSEM Green’s functions ourselves, rather than using syngine to download AxiSEM Green’s functions. Because the number of velocity models supported by syngine is currently very limited (<http://ds.iris.edu/ds/products/syngine/#models>), this represents an important capability for future users.

Task 6: Uncertainty quantification of source type for real data, II

The core components for uncertainty quantification are documented in Section 4.3. These include visualization capabilities with the lune and vw -rectangle parameterizations.

Task 7: Perform inversions using AxiSEM and synthetic data, plus uncertainty quantification of source type

Examples of synthetic inversions using AxiSEM are shown in Figures 16 and 17 for force and moment tensor, respectively.

We benchmarked AxiSEM synthetics (based on spectral-element simulations with a spherical mesh) against CAP synthetics (based on FK simulations with a rectangular mesh). Results of the comparison, along with further methodological details, can be found at <https://github.com/geodynamics/axisem/issues/58>. The comparison shows that modeling errors resulting from curvature of the Earth become significant in CAP synthetics at distances greater than 500 km. Such errors can be avoided through the use of AxiSEM synthetics.

Task 8: Mesh and model wavefield for 3D velocity model

Over the course of the project, it became clear that the meshing capabilities that were internal to SPECFEM3D were preferred over capabilities of GEOCUBIT, which is external to SPECFEM3D and requires licensed software. Recent work featuring SPECFEM3D internal meshing for regional simulations can be found in *Chow et al. (2020)*.

We expanded capabilities for meshing 3D models by incorporating a third doubling layer in the finite-element meshes, allowing for greater computational efficiencies (Figure 18). We also resuscitated a computational software package (<http://krischer.github.io/wfdiff/>) that allows us to document the numerical accuracy (minimum resolvable period) of the synthetic seismograms; the updated version now resides at <https://github.com/uafgeotools/wfdiff>. This is a critical component, especially when the period of measurement of body waves is close to the minimum resolvable period of the synthetic seismograms.

Task 9: Synthetic inversions in 1D and 3D models (with 3D synthetics)

As described in Section 4.3, MTUQ interfaces with SPECFEM3D-generated Green’s functions for 3D velocity models.

Representing the impulse response of a 3D medium requires 18 independent time series (6 M_{ij} , 3 components). In contrast, representing the impulse response of a 1D medium requires only 10 independent time series, but FK and other codes employ redundant representations.

As described by <https://github.com/krischer/instaseis/issues/66>, there are other ways in which Green’s function conventions differ within the literature. To deal with this complexity, our approach is to store Green’s functions in MTUQ using a consistent 18 time series representation. Converting the FK and AxiSEM conventions to this new convention required some tedious algebra, but made it straightforward in the end to accommodate 3D Green’s synthetics.

We performed a number of tests and benchmarks in anticipation of other users trying 3D source inversions. One useful sanity check was to use a 3D solver in combination a 1D model, making

sure that the results match previous 1D solver results. Such tests are nontrivial because Green's conventions functions for 1D and 3D media differ significantly.

Force modeling and uncertainty quantification

Estimating point forces was not part of our proposed tasks, but it was a natural objective to pursue, given the capabilities of AxiSEM and also the utility of using forces instead of moment tensors to represent source processes (e.g., *Dammeier et al.*, 2015). Section 4.3 shows the capabilities for estimating point forces. As far as we know, these visualization capabilities have not appeared in previously published studies, and they could be useful in future work on sources such as landslides and possibly explosions.

4.5 Presentations and dissemination of results

Results from our project were presented in the following places:

- Peer-reviewed journal articles (*Alvizuri et al.*, 2018; *Alvizuri and Tape*, 2018; *Tape and Tape*, 2019)
- AGU 2020 presentation (*Modrak et al.*, 2020)
- AGU 2018 presentation (*Modrak et al.*, 2018) (see Figure 19)
- SSA 2018 presentations (*Tape and Tape*, 2018; *Tape et al.*, 2018)
- AGU 2017 presentations (*Tape et al.*, 2017; *Alvizuri and Tape*, 2017)
- code repository for MTUQ: <https://github.com/uafgeotools/mtuq>

Project results were also presented at AFRL Technical Interchange Meetings in Satellite Beach, Florida, in October 2017, February 2019, January 2020, and February 2021 (virtual).

Table 1: Codes relevant to this project for 3D synthetic seismograms and moment tensor inversions.

Green's functions	synthetics	moment tensor inversion	uncertainty quantification	references
fk	syn (outside cap)	cap	limited	<i>Zhu and Helmberger (1996)</i> <i>Zhu and Ben-Zion (2013)</i>
fk	syn (outside cap)	capuaf	unnamed	<i>Silwal and Tape (2016)</i> <i>Alvizuri and Tape (2016)</i>
axisem	axisem	(none)	(none)	<i>Nissen-Meyer et al. (2007, 2014)</i>
SPECFEM3D	SPECFEM3D	GRD_CMT3D or pycmt3d	limited	<i>Komatitsch et al. (2004)</i> <i>Liu et al. (2004)</i>
axisem, fk, or SPECFEM3D	axisem, fk, or SPECFEM3D	MTUQ	MTUQ	← THIS PROJECT

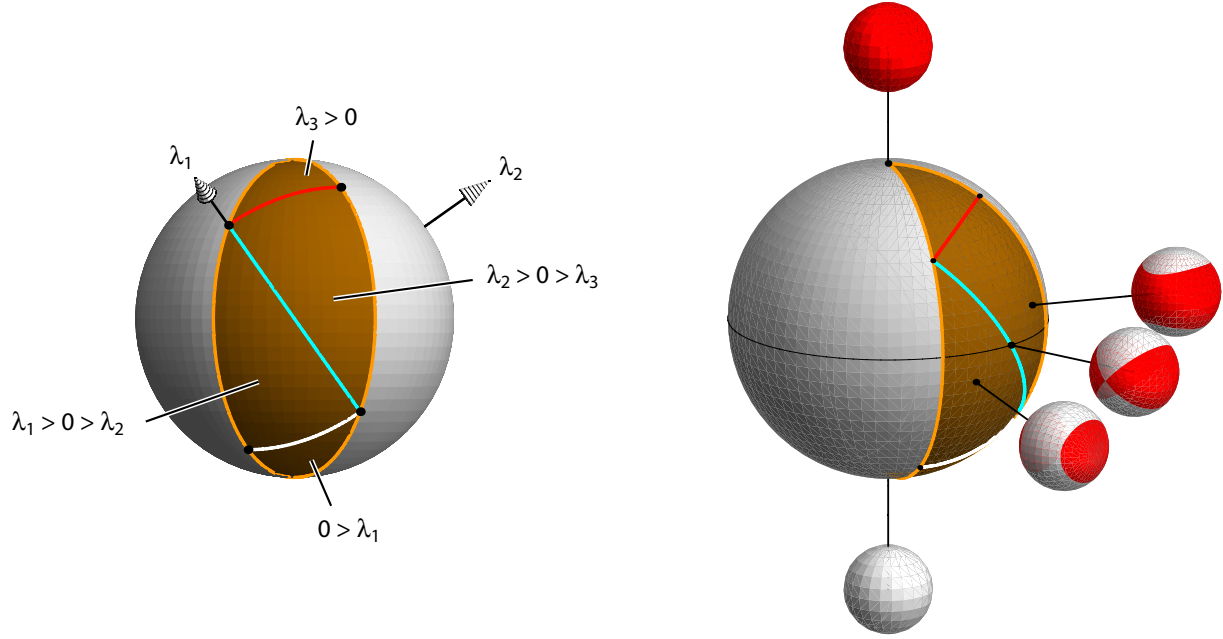
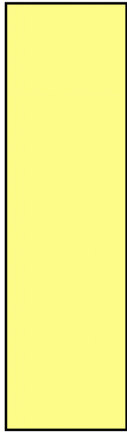


Figure 1: Four regimes for moment tensors, depicted on the fundamental lune (*Tape and Tape*, 2013, Fig. S1). The lune is the surface of normalized moment tensor eigenvalue triples $\mathbf{\Lambda} = (\lambda_1, \lambda_2, \lambda_3)$, with $\lambda_1 \geq \lambda_2 \geq \lambda_3$. The three arcs $\lambda_3 = 0$ (red), $\lambda_2 = 0$ (blue), and $\lambda_1 = 0$ (white) divide the lune into four regions. Above $\lambda_3 = 0$, beachballs are all red ($\lambda_3 > 0$). Above and to the right of the arc $\lambda_2 = 0$, moment tensor beachballs have red bands and white caps ($\lambda_2 > 0 > \lambda_3$). Below and to the left of it they have white bands and red caps ($\lambda_1 > 0 > \lambda_2$). Below the arc $\lambda_1 = 0$, beachballs are all white ($\lambda_1 < 0$). Double couple moment tensors are at the center of the lune, where $\mathbf{\Lambda} = (1, 0, -1)$. (Although points on the lune have unit norm, we often give them as unnormalized directions; this $(1, 0, -1)$ instead of $(1, 0, -1)/\sqrt{2}$.)

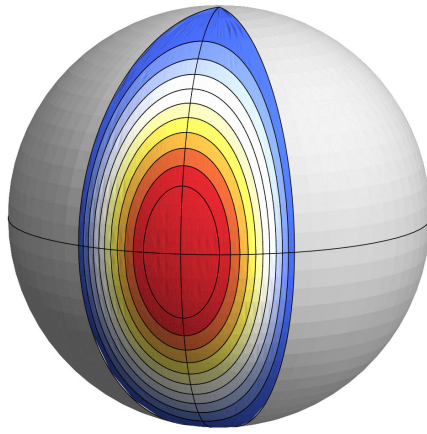
Probability density of random moment tensors

v - w rectangle



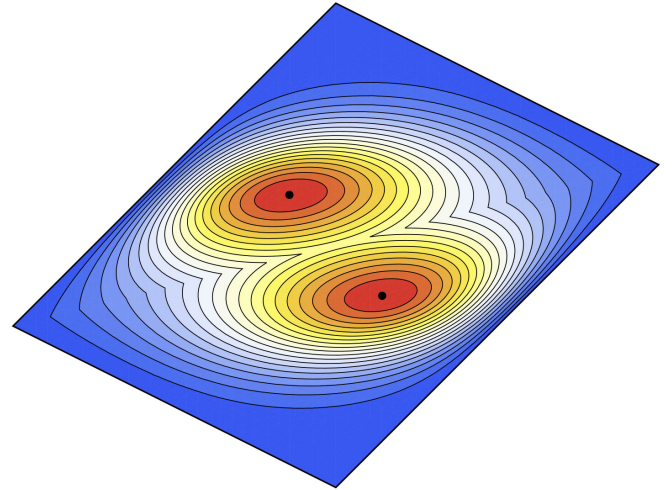
Tape and Tape (2015)
Alvizuri et al. (2018)

eigenvalue lune



Tape and Tape (2012)
Riedesel and Jordan (1989)

u - v parallelogram (T - k plot)



Hudson, Pearce, Rogers (1989)

Figure 2: Probability density of random moment tensors, displayed using three different source type plots: (1) the vw rectangle (Tape and Tape, 2015; Alvizuri et al., 2018), (2) the eigenvalue lune (Tape and Tape, 2012a; Riedesel and Jordan, 1989), and (3) the Tk plot of Hudson et al. (1989). The vw rectangle clearly conveys the notion of randomness better than the other two options. However, for some purposes, the eigenvalue lune has advantages, as explained in Tape and Tape (2019). The figure in the center appears in Figure 10 in Tape and Tape (2019). The figure at the right appears in Figure S2 in Tape and Tape (2019).

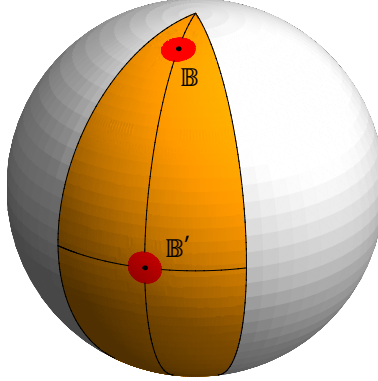


Figure 3: Fundamental lune (orange) and two of its subsets $\mathbb{B}$ and $\mathbb{B}'$ (both red) (*Tape and Tape* (2019), Figure 1). The lune is the set of all moment tensor source types. The sets $\mathbb{B}$ and $\mathbb{B}'$ are disks of angular radius 5° whose centers are on the central meridian of the lune at colatitude 20° and 90° , respectively. Although $\mathbb{B}$ and $\mathbb{B}'$ are congruent, the set $[\mathbb{B}']_{\mathbb{U}}$ of (unit) moment tensors with source type in $\mathbb{B}'$ turns out to be far larger than the analogous set $[\mathbb{B}]_{\mathbb{U}}$ of tensors with source type in $\mathbb{B}$. One implication: If M and M' are moment tensors known only to have their source types in $\mathbb{B}$ and $\mathbb{B}'$, respectively, then we can be much more certain about M than about M' .

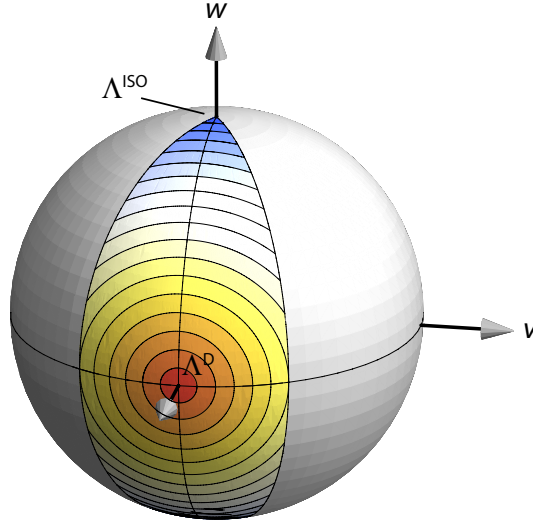


Figure 4: Contours of the function $\Lambda \rightarrow \delta([\Lambda]_{\mathbb{U}}) = \pi - 2\phi$ (*Tape and Tape* (2019), Figure 8). The function gives the angular diameter of the set of moment tensors that have source type Λ . The function varies from 0 at $\pm\Lambda^{\text{ISO}}$ to π at Λ^{D} . The contour interval is 10° .

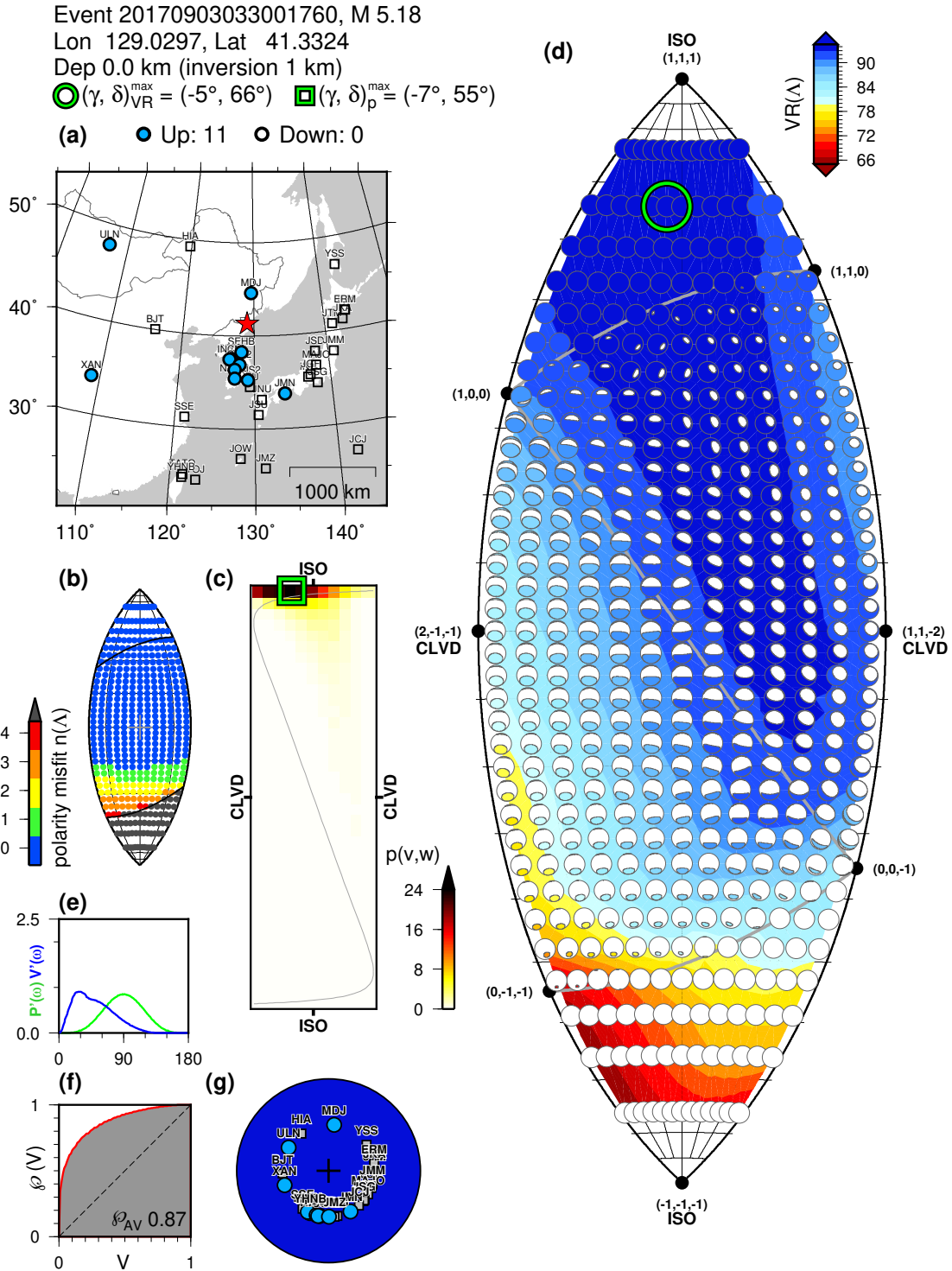


Figure 5: Uncertainty characterization for the 2017-09-03 North Korea nuclear test (Figure 4 of *Alvizuri et al. (2018)*), generated using legacy scripts that have not yet been fully rewritten for MTUQ. See caption of Figure 6 for details.

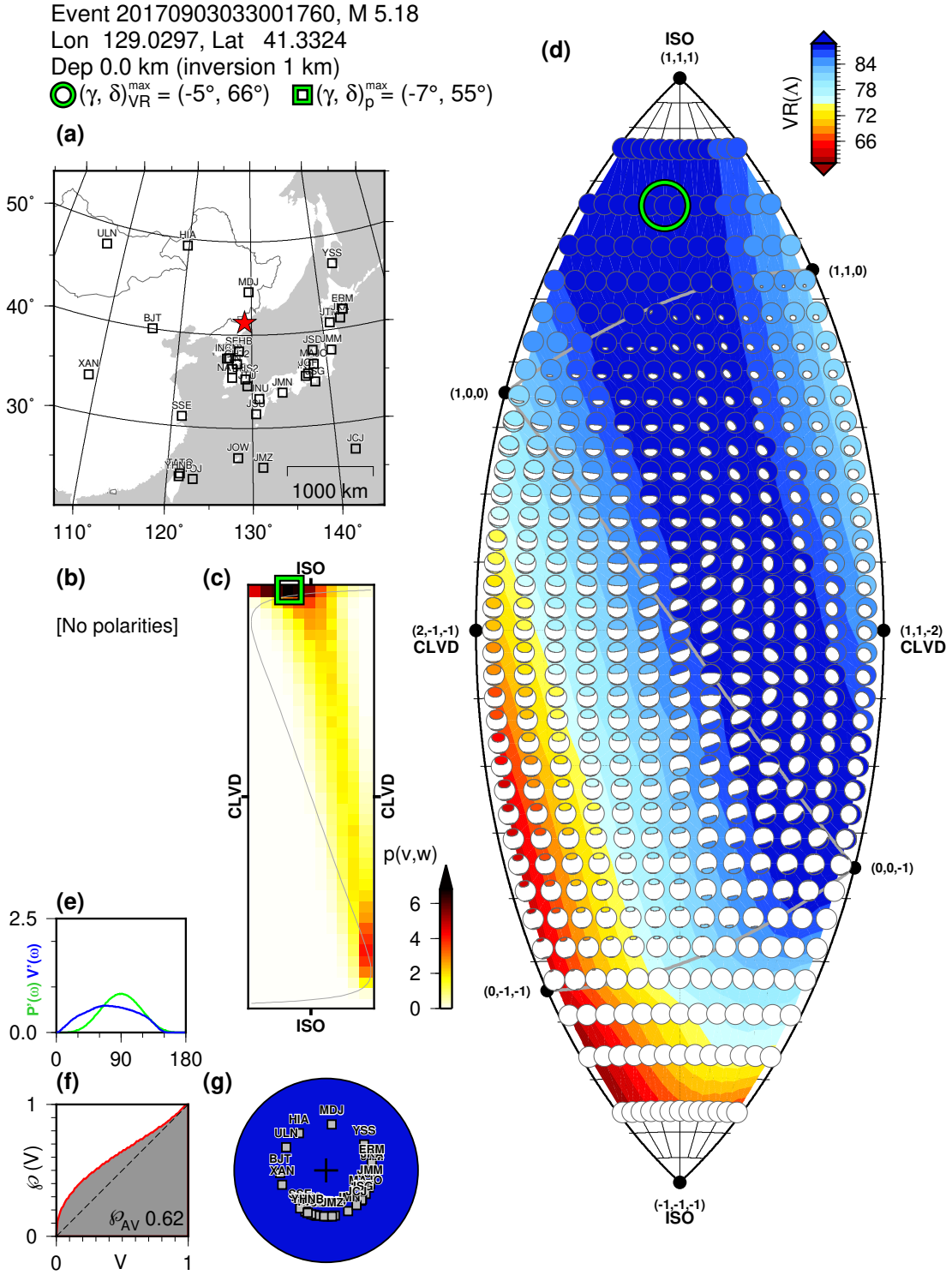


Figure 6: See caption on next page.

[Figure 6, previous page] Full moment tensor uncertainty summary for the 2017-09-03 M_w 5.18 explosion event (Figure S5 of *Alvizuri and Tape* (2018)). The only difference between this figure and Figure 5 is that here, polarities were not used. This representation follows *Alvizuri et al.* (2018) and is shown for other events, in *Alvizuri* (2018). (a) Map of source location (red star) and stations used in the inversion for this event. The station is colored blue if the observed first-motion preference polarity on the vertical component is up (compression) and white if it is down (dilatation). (b) Contour plot of the polarity misfit on the lune. (c) Source type probability density $p(v, w)$ in the vw rectangle *Tape and Tape* (2015). A green square indicates the location of the point (v_x, w_x) where p is maximum; this point is apt to differ from the source type (v_0, w_0) of M_0 . (d) Contour plot of the variance reduction $VR(\mathbf{\Lambda})$. At each point $\mathbf{\Lambda}$, the variance reduction $VR(\mathbf{\Lambda})$ is the maximum variance reduction $VR(M)$ for moment tensors M that have source type $\mathbf{\Lambda}$. Large values (blue) of VR represent better fit between observed and synthetic waveforms. Of the beachballs $M(\mathbf{\Lambda})$, our solution M_0 (green circle) is the one with largest VR . The gray arcs on the lune are the great circle arcs $\lambda_1 = 0$, $\lambda_2 = 0$, and $\lambda_3 = 0$. Selected eigenvalue triples (black dots) on the boundary of the lune are indicated, with the understanding that the triples need to be normalized. The positive isotropic source $(1, 1, 1)$ is at the top, the negative isotropic source $(-1, -1, -1)$ is at the bottom, and the double couple $(1, 0, -1)$, not shown, would be at the center of the lune. (e) The curves $\hat{V}'(\omega)$ and $\hat{P}'(\omega)$ that are used to construct the confidence curve $\mathcal{P}(V)$ in (f), as explained in *Silwal and Tape* (2016) in the context of double couples. For full moment tensors, as here, $\hat{V}'(\omega) \propto \sin^4 \omega$. (f) The confidence curve $\mathcal{P}(V)$ for M_0 . The more the curve resembles the shape of a capital gamma (Γ), the better. The shaded area is the average confidence $\mathcal{P}_{AV}$. (g) The moment tensor M_0 , plotted in a lower-hemisphere projection. The location of the piercing point for each station depends on the station azimuth, epicentral distance, and the assumed layered reference model.

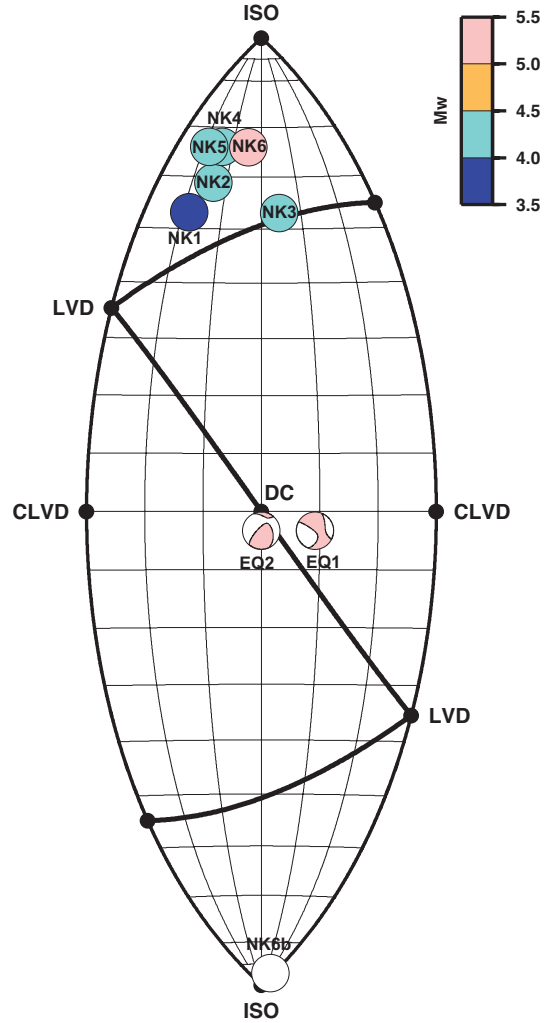


Figure 7: Best-fitting moment tensors for six explosion events (top), two earthquakes (middle), and the cavity collapse event (NK6b) (Figure 6 of *Alvizuri and Tape (2018)*). The solid regions of each moment tensor are colored by magnitude; the magnitude of the collapse event is M_w 4.26. See Figure 1 for context.

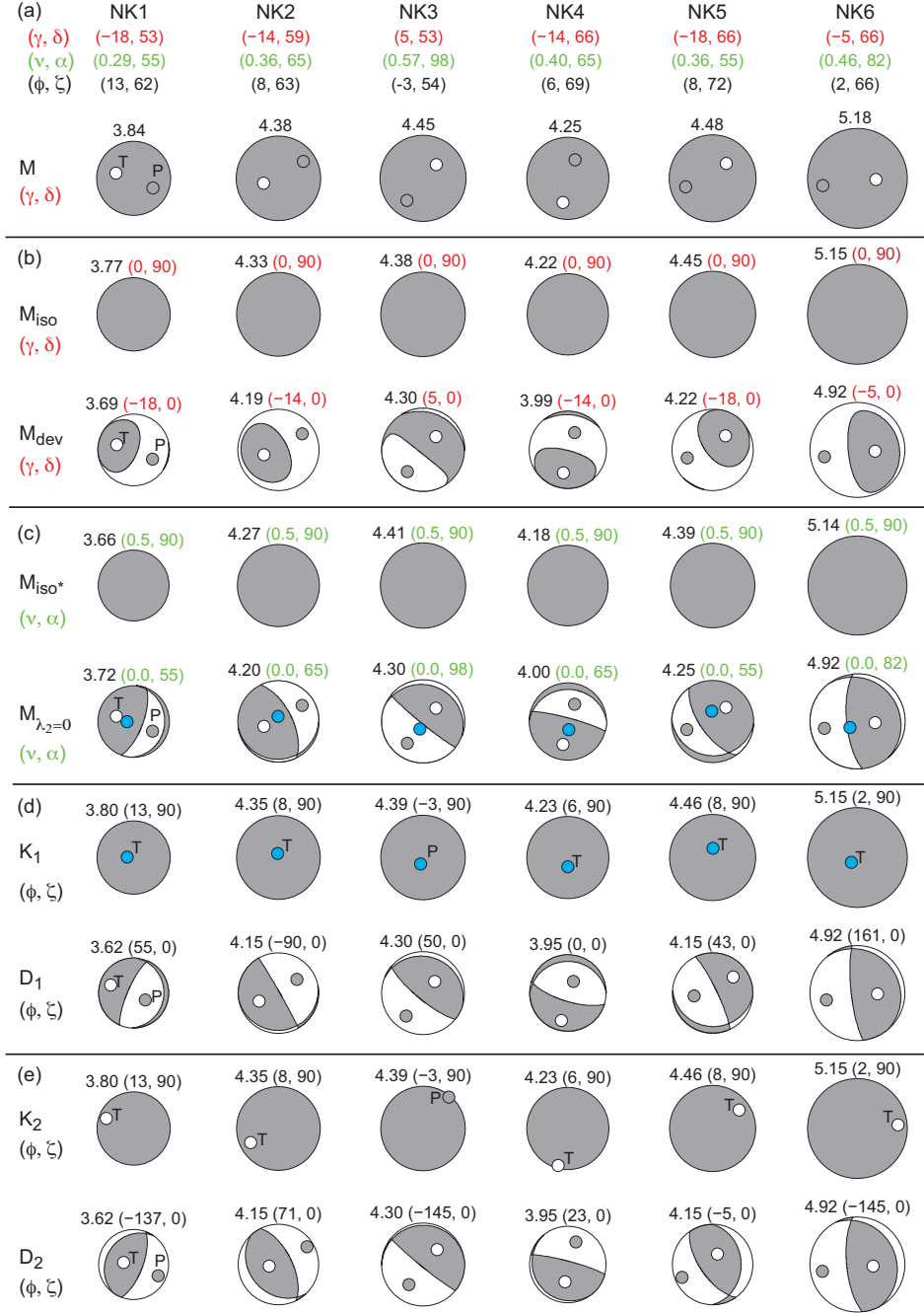
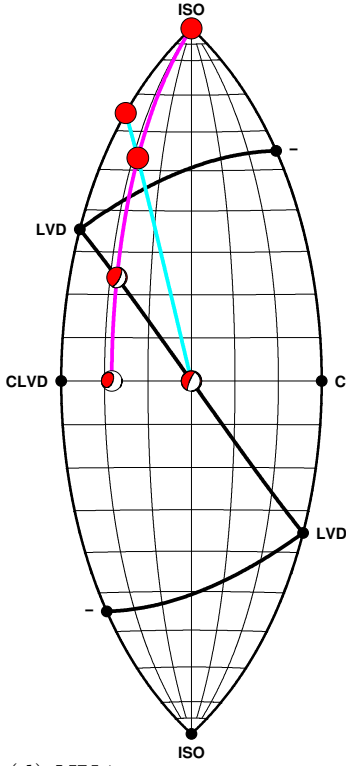
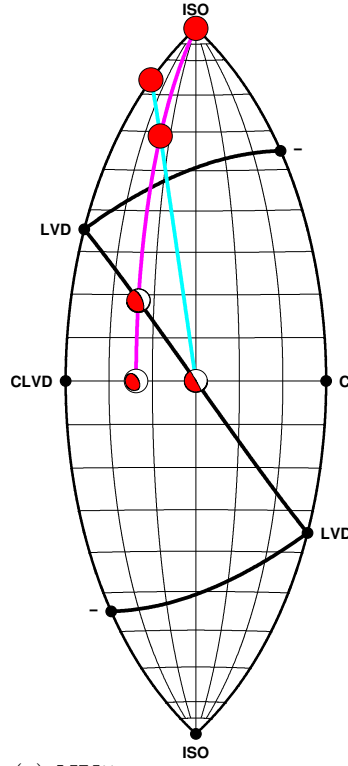


Figure 8: Decompositions of full moment tensors for the six explosion events (Figure 7 of *Alvizuri and Tape* (2018)); see Discussion for details and also Figure 9. The magnitude is labeled above each beachball. (a) The six best-fitting full moment tensors for events NK1, ..., NK6. The colored labels correspond to (γ, δ) (red), (ν, α) (green), (ϕ, ζ) (black). (b) $M = M_{\text{dev}} + M_{\text{iso}}$, where $M_{\text{iso}} = \frac{1}{3} \text{tr}(M)I$. (c) $M = M_{\lambda_2=0} + M_{\text{iso}*}$, where $M_{\text{iso}*} = \sqrt{3} \lambda_2 I$. The label (ν, α) above is the Poisson parameter ν and the angle α between fault planes *Aki and Richards* (1980); *Tape and Tape* (2013). The blue dot is the normal vector to the near-horiztonal fault plane. (d) $M = K_1 + D_1$: one of two decompositions for the crack-plus-double-couple model, where K_1 is the crack tensor and D_1 is the double couple moment tensor *Minson et al.* (2007); *Tape and Tape* (2013). The label (ϕ, ζ) above each beachball is the lune azimuth to K and ζ is the crack fraction of M . (e) $M = K_2 + D_2$: the second possibility for the crack-plus-double-couple model.

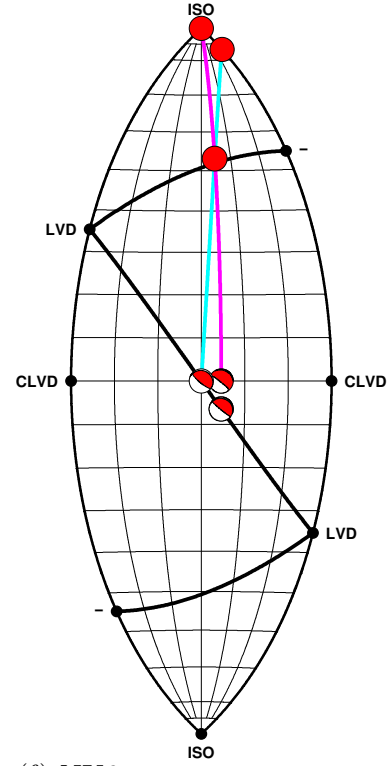
(a) NK1



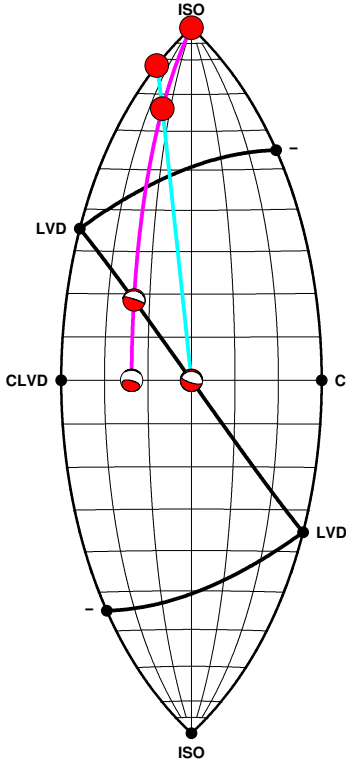
(b) NK2



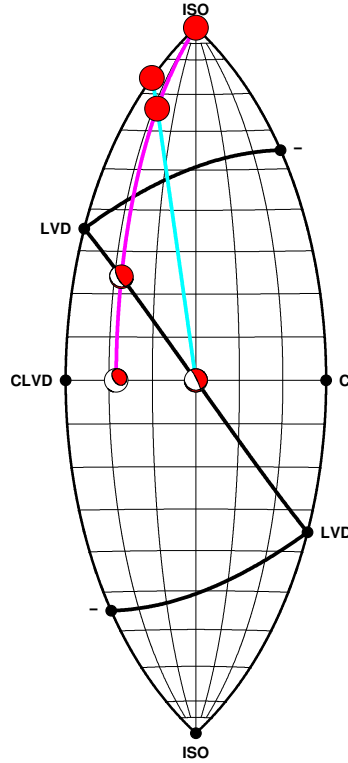
(c) NK3



(d) NK4



(e) NK5



(f) NK6

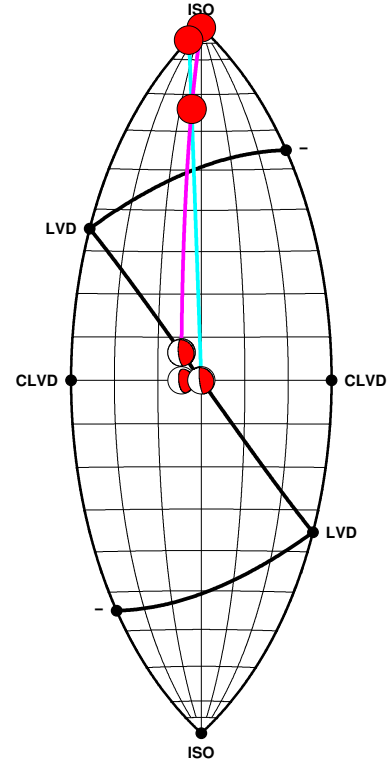


Figure 9: Lune representation of Figure 8 for the six North Korea explosion events (Figure S6 of *Alvizuri and Tape* (2018)). The estimated moment tensor is at the intersection of the cyan and magenta arcs. The cyan arc represents the CDC decomposition $M = K + D$, where the beachballs are shown for K_1 , at the lune boundary, and D_1 , at the lune center. The magenta arc, with fixed lune longitude, shows the isotropic and deviatoric parts of M , as well as $M_{\lambda_2=0}$, which shows the fault planes of the classical model of *Aki and Richards* (1980).

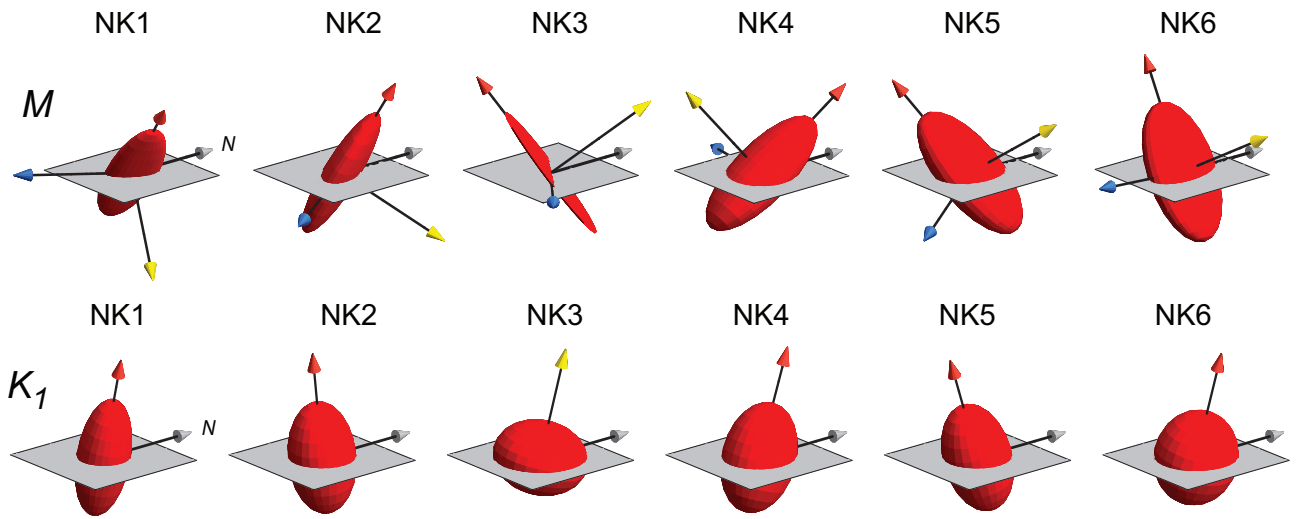


Figure 10: Ellipsoids depicting the moment tensors for the six explosion events (NK1, $\dots$, NK6) (Figure 8 of *Alvizuri and Tape* (2018)). The ellipsoid can be thought of as enclosing the all-outward vector fields associated with each moment tensor. (top) Moment tensors M depicted in Figure 8a. The principal axes are colored as T (red), null (blue), and P (yellow). The north arrow is black and labeled N. (bottom) Crack moment tensors K_1 depicted in Figure 8d-top. Only the primary principal axis is plotted.

Event 20170903033001760
 Model MDJ2, inv depth 1 km
 Freq 0.02 - 0.06 Hz (16.67 - 50.00 sec)
 Surface vertical timeshifts (sec)

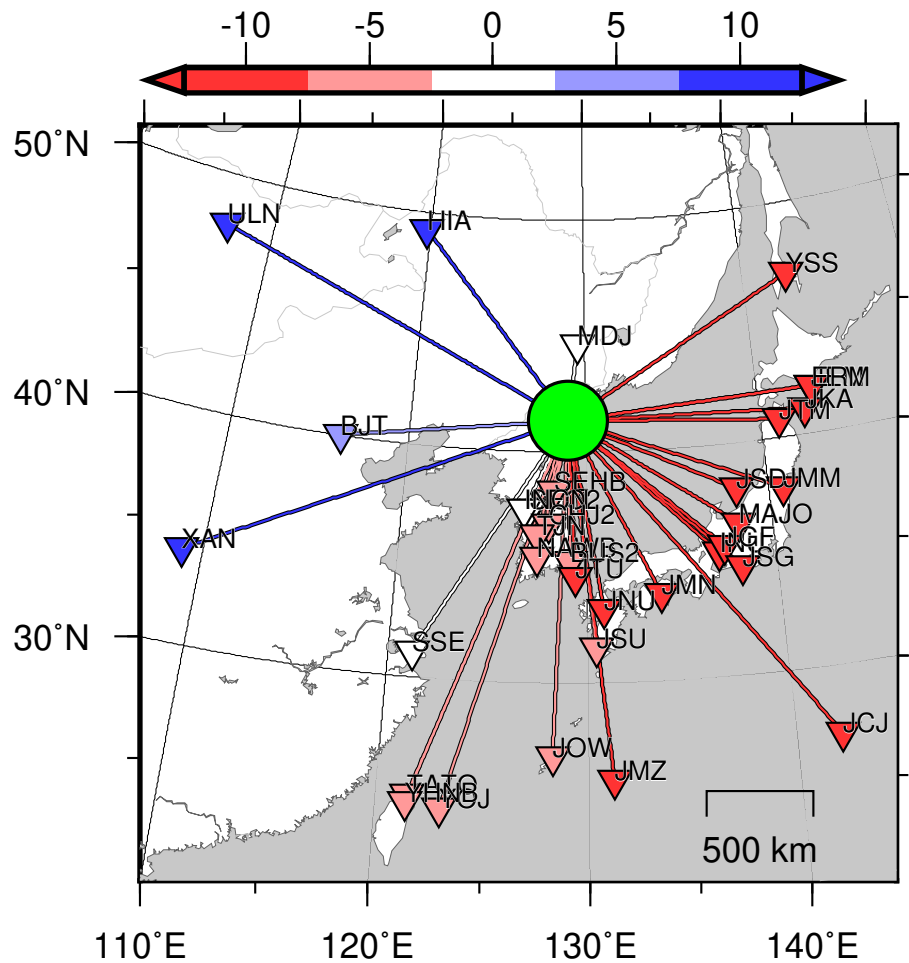


Figure 11: Rayleigh wave time shift maps for the 2017-09-03 explosion event (NK6) (Figure 9 of *Alvizuri and Tape* (2018)). The time shifts correspond to the waveform fits for periods 17–50 s. The 1D model we use is calibrated for the path to MDJ, to the north, for which the time shift is 0.7 s.

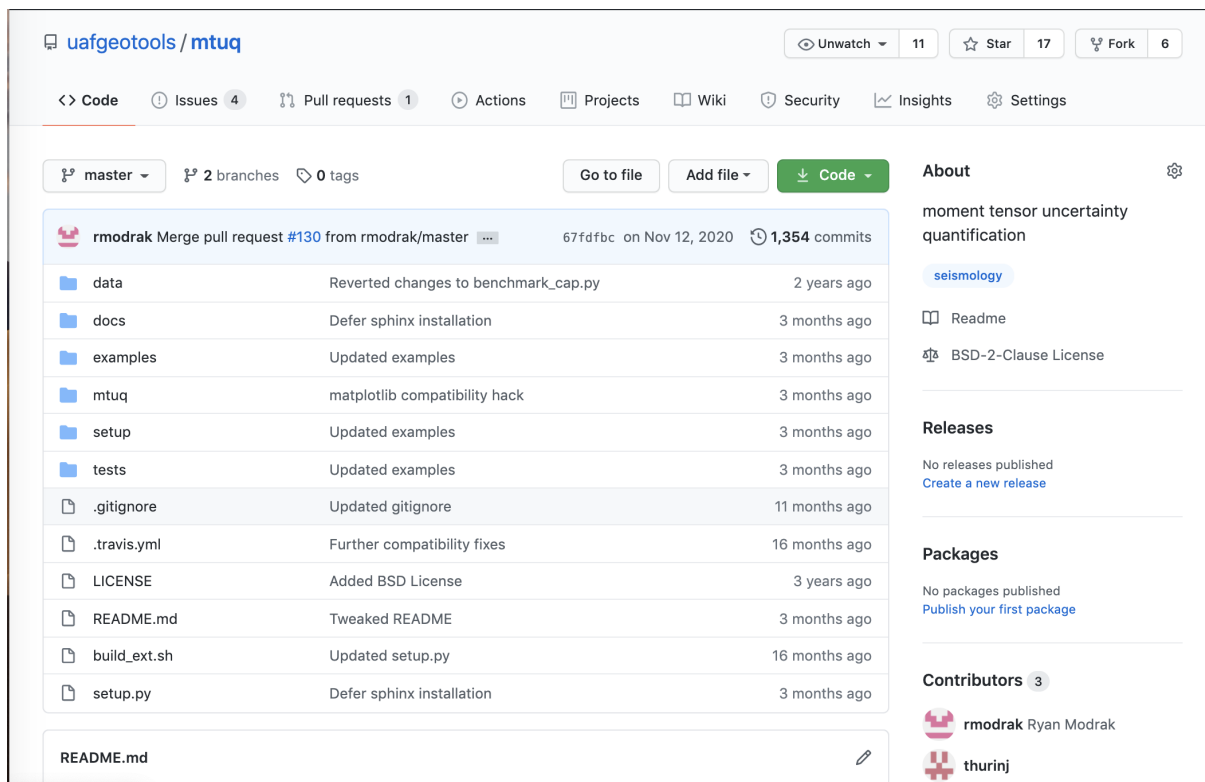


Figure 12: MTUQ code repository, available at <https://github.com/uafgeotools/mtuq>. This code is the primary product of this contract; it is described in Section 4.3.

MTUQ

Site ▾

Page ▾

« Data processing

Seismic data structures

mtuq.Dataset

Container for observed or synthetic seismic data

mtuq.GreenTensor

Green's tensor object

mtuq.GreenTensorList

Container for Green's tensor objects

Data I/O

mtuq.download_greens_tensors

Downloads Green's tensors from syngine

mtuq.open_db

Opens Green's tensor database from disk

mtuq.read

Reads seismic data from disk

Data processing and inversion

mtuq.ProcessData

Data processing function constructor

mtuq.Misfit

Misfit function constructor

mtuq.grid_search

Evaluates misfit over grids

mtuq.MTUQDataArray

Data structure for storing misfit values on regularly-spaced grids

mtuq.MTUQDataFrame

Data structure for storing misfit values on irregularly-spaced grids

Station and event structures

mtuq.Station

Station metadata object

mtuq.Origin

Origin metadata object

mtuq.Wavelet

Source wavelet or source-time function object

mtuq.Force

Force source

mtuq.MomentTensor

Moment tensor source

Figure 13: Example of MTUQ documentation (see also Section 4.3).

mtuq.GreensTensor

class `mtuq.GreensTensor` (`traces=None`, `station=None`, `origin=None`, `id=None`, `tags=[]`, `include_mt=True`, `include_force=False`) [\[source\]](#)

Bases: `obspy.core.stream.Stream`

Holds multiple time series corresponding to the independent elements of an elastic Green's tensor

Note

Besides the methods below, *GreensTensor* includes many useful data processing methods inherited from `obspy.core.Stream`. See [ObsPy documentation](#) for more information.

Public Methods

<code>convolve</code>	Convolve time series with given wavelet
<code>get_synthetic</code>	Generates synthetic through a linear combination of time series
<code>select</code>	Selects time series that match the supplied metadata criteria

Private Methods

Warning

Private methods are mainly for internal/developer use and their API might change without notice.

<code>_allocate_stream</code>	Allocates ObsPy stream used by <code>get_synthetic</code>
<code>_get_shape</code>	Returns shape of NumPy array used by <code>get_synthetic</code>
<code>_preallocate</code>	Preallocates structures used by <code>get_synthetic</code>
<code>_precompute</code>	Precomputes NumPy array used by <code>get_synthetic</code>
<code>_set_components</code>	Gets called before or during <code>get_synthetic</code> to specify which components are returned

Figure 14: Example of MTUQ documentation, showing the `GreensTensor` functionality (see also Section 4.3).

Computing Green's functions from 3D Earth models

MTUQ currently supports 3D Green's functions from SPECfEM3D.

To generate a complete Green's function database for a given hypocenter and depth, six SPECfEM3D wavefield simulations are required. Output must be saved as SAC files at individual stations using the following filename convention, which comes from [GRD_CMT3D](#). Place all SAC files corresponding to a single hypocenter and depth in the same directory as follows:

```
{basedir}/{event_id}/
{net}.{sta}.{loc}.Z.Mrr.sac
{net}.{sta}.{loc}.Z.Mtt.sac
{net}.{sta}.{loc}.Z.Mpp.sac
{net}.{sta}.{loc}.Z.Mrt.sac
{net}.{sta}.{loc}.Z.Mrp.sac
{net}.{sta}.{loc}.Z.Mtp.sac
{net}.{sta}.{loc}.R.Mrr.sac
{net}.{sta}.{loc}.R.Mtt.sac
{net}.{sta}.{loc}.R.Mpp.sac
{net}.{sta}.{loc}.R.Mrt.sac
{net}.{sta}.{loc}.R.Mrp.sac
{net}.{sta}.{loc}.R.Mtp.sac
{net}.{sta}.{loc}.T.Mrr.sac
{net}.{sta}.{loc}.T.Mtt.sac
{net}.{sta}.{loc}.T.Mpp.sac
{net}.{sta}.{loc}.T.Mrt.sac
{net}.{sta}.{loc}.T.Mrp.sac
{net}.{sta}.{loc}.T.Mtp.sac
```

To open an SPECfEM3D database client in MTUQ:

```
from mtuq import open_db
db = open_db(path_SPECfEM3D_output_directory, format="SPECfEM3D")
```

Once opened, a SPECfEM3D database client can be used to generate [GreensTensor](#) objects as follows:

```
greens_tensors = db.get_greens_tensors(stations, origin)
```

Figure 15: Example of MTUQ documentation, showing the file naming convention for calculating Green's functions in 3D Earth models (see also Section 4.3).

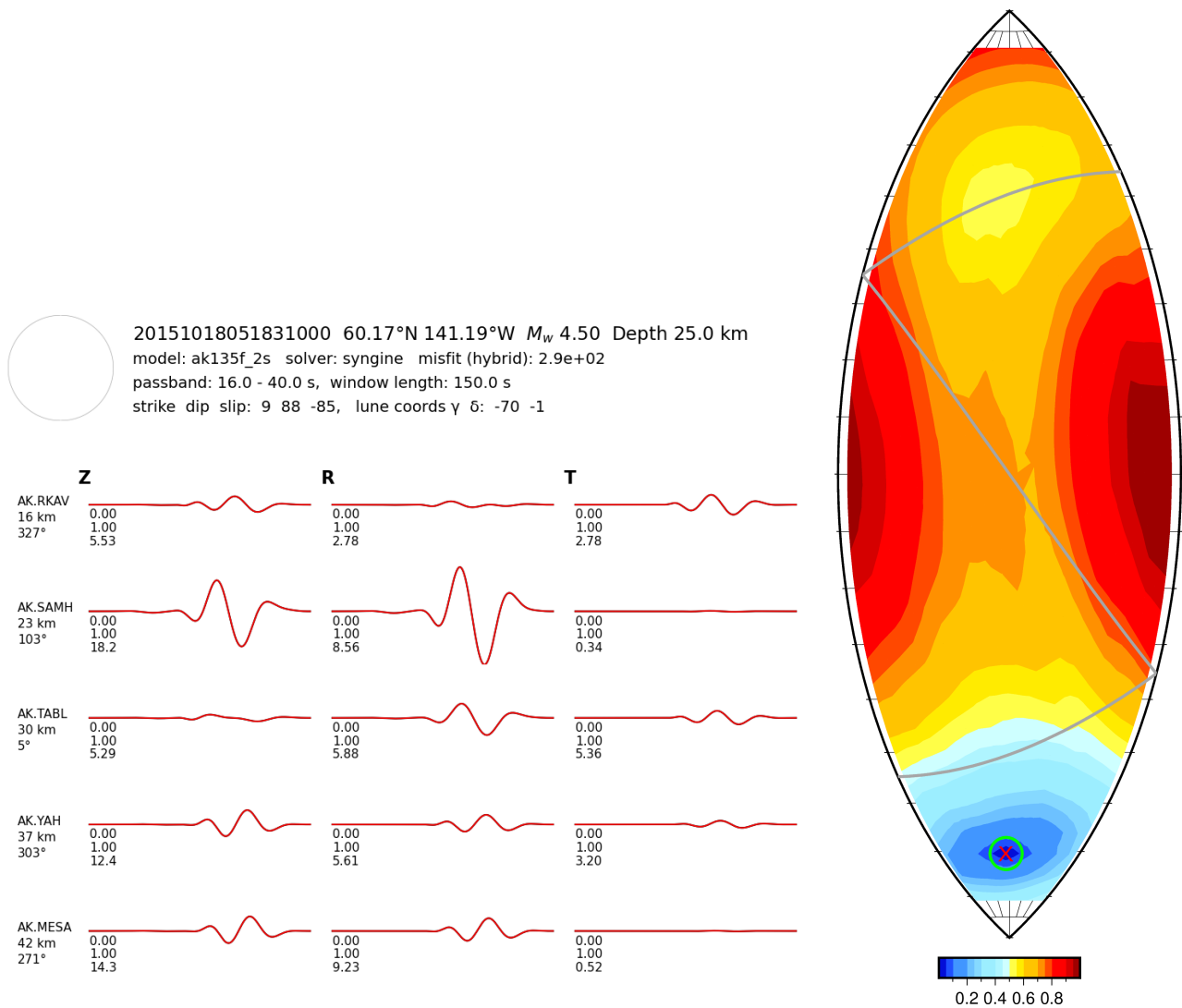


Figure 16: Synthetic moment tensor inversion with zero data errors added, resulting in perfect recovery of the target moment tensor (in this case, a negative isotropic moment tensor), as expected.

20151018051831000 60.17°N 141.19°W F 1.00e+10 N Depth 0 m
 model: ak135f_2s solver: syngine misfit (hybrid): 5.6e+01
 passband: 5.0 - 50.0 s, window length: 100.0 s
 ϕ θ : 13 38

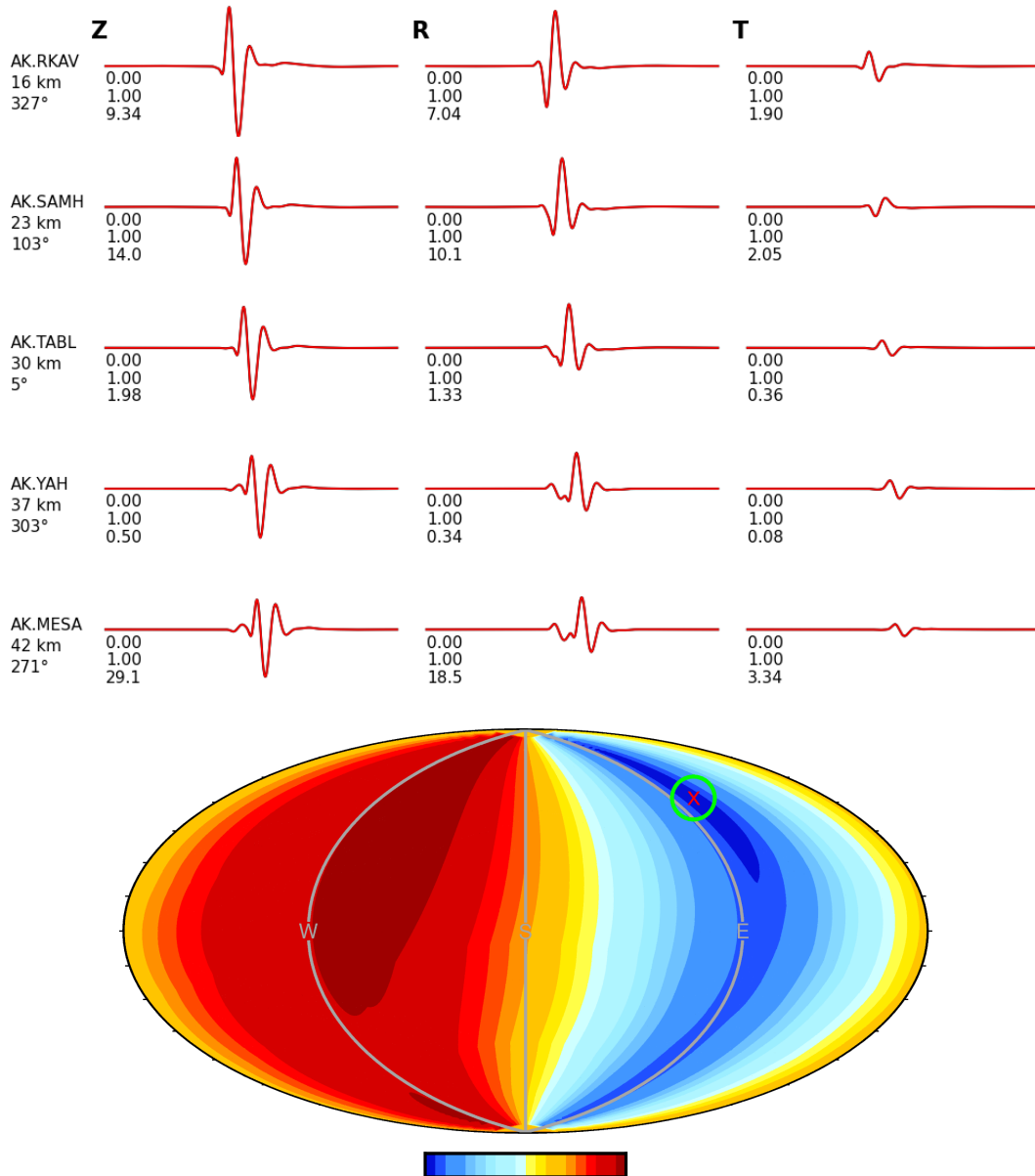
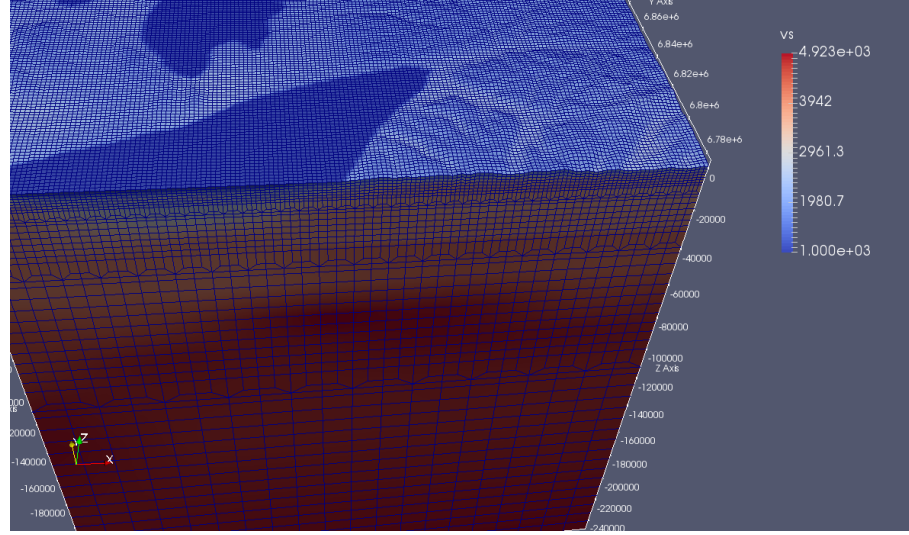


Figure 17: Synthetic inversion for a force pointing up and to the east. As expected for a synthetic inversion, the recovery is perfect. The top plot shows the variations of the waveform misfit function in the 2D space of the force direction.

(a)



(b)

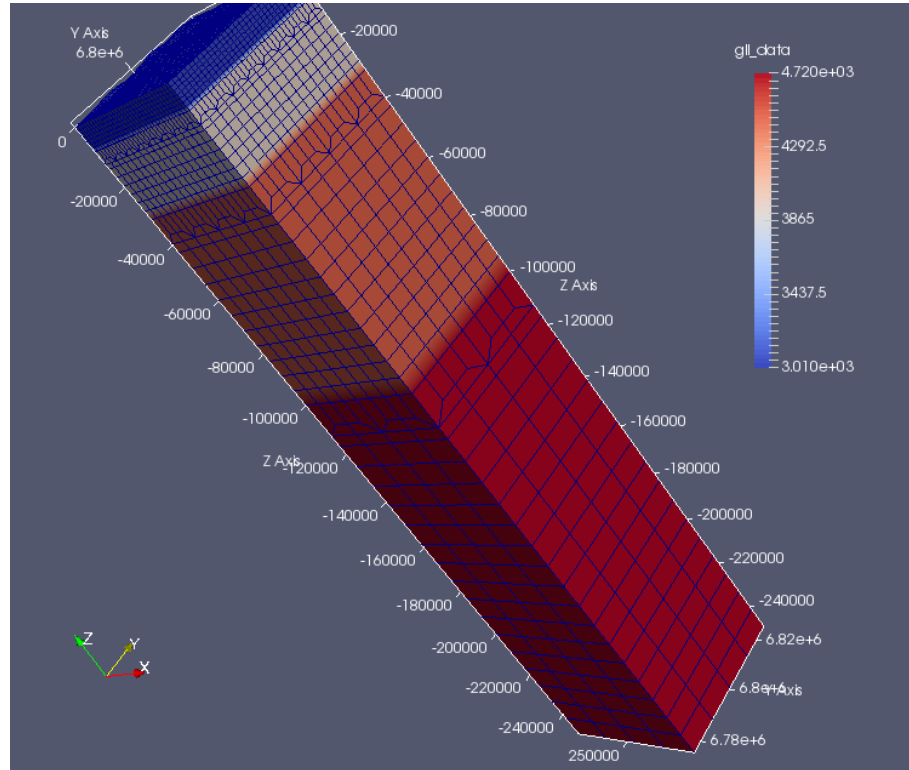


Figure 18: Examples of hexahedral finite-element meshes used for 3D seismic wavefield simulations with SPECFEM3D. The capability of adding a third mesh-doubling layer (whereby elements get twice as large as you go deeper) is illustrated in both images. This reduces the computational cost of the simulations without sacrificing numerical accuracy. These meshes were made motivated by the need to perform moment tensor inversions using 3D Earth models instead of 1D Earth models.



MTUQ—moment tensor uncertainty quantification based on obspy, instaseis and pymc

(github.com/uafseismo/mtuq)



Ryan Modrak, Celso Alvizuri, Vipul Silwal, Carl Tape

Abstract

Characterizing the source process for nuclear explosions, earthquakes, volcanic and geothermal events, induced events, landslides, and related phenomena is of fundamental importance to seismology. This characterization may include origin time, hypocenter, magnitude, orientation, full moment tensor, force, and source time function. Inverting for these parameters involves strong tradeoffs that motivate the use of robust uncertainty quantification methods. We present validation tests and early results from MTUQ, a new software package designed for such use cases (github.com/uafseismo/mtuq). MTUQ provides distributed parallel grid search capabilities based on mpi4py and Monte Carlo inversion capabilities based on pymc. Obspy is used for basic data processing capabilities. The parallel grid search routines have been well tested, and the MCMC methods are currently under development. After establishing these capabilities, we intend to apply MTUQ to a variety of regional seismology and nuclear monitoring problems.

Motivation

- Enable new research
 - high-dimensional source inversion
 - source-waveform estimation
 - differentiating between force monopoles, dipoles
- Address limitations of legacy codes
 - observers
 - flexibility
- Assist operational seismologists
 - nuclear treaty monitoring
 - earthquake monitoring
- Leverage new parameterizations
 - linear parameterizations for source-type classification
 - "uniform" parameterizations for uncertainty analysis

Selected applications

Global seismology	Nuclear monitoring
Application exploration	NDC, RL, SL, UG
Nuclear monitoring	NDC, RL, SL, UG
Seismic hazard	NDC, RL, SL, UG
Volcano seismology	NDC, RL, SL, UG
NDC = Non-double couple source	
RL = Rigid fault location	
SL = Spatially distributed source	
UG = Source-time function	
UG = Uniform uncertainty quantification	

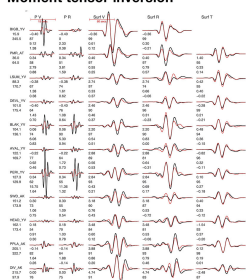
A wide diversity of applications creates a demand for flexible source inversion software. Common to all applications is a need for robust uncertainty quantification.

Python ecosystem



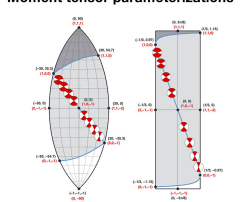
Python provides a rich collection of open-source scientific packages. Applications written using NumPy run highly optimized compiled code under the hood, and mpi4py allows parallelization across cores and nodes.

Moment tensor inversion



A function that quantifies the fit between data and synthetics is needed for uncertainty analysis. For the North Korea examples at right, we applied an L1 waveform-difference functional to bandpassed body- and surface-wave components. As shown above, 2 body-wave and 3 surface-wave components were considered, all weighted equally. Time-shift corrections based on cross-correlation lag were made prior to taking waveform differences. Such a shift function is included in MTUQ; other choices can also be implemented.

Moment tensor parameterizations



A goal of MTUQ is to leverage new parameterizations based on geometric formulations of moment tensor space. The line parameterization at left results from the plotting moment tensor eigenvalues [2], while the "uniform" parameterization at right arises naturally in uncertainty analysis.

Planned software components

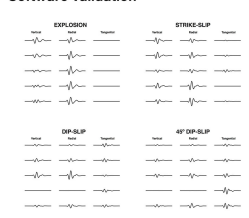
massively parallel grid search	Metropolis-Hastings MCMC	Hamiltonian MCMC
based on mpi4py	based on pymc	based on pymc
< 8 parameters	< 12 parameters	> 12 parameters (?)
high scalability	efficient sampling	efficient sampling

For different problem types, MTUQ will provide different probabilistic inversion methods. Grid searches are embarrassingly parallel and run effectively on large clusters. Markov chain Monte Carlo (MCMC) methods are not easily parallelized and may require manual tuning, but they sample model space very efficiently.

FK	AxiSEM/Instaseis	SYNGINE	SPECTFEM3D_GLOBE
frequency-wavenumber modeling	spectral element modeling	spectral element modeling	spectral element modeling
layered media	isotropic media	isotropic media	fully 3D media
SAC output	reCAP output	reCAP output	SAC/SOF output

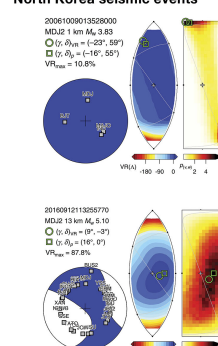
Interfaces for the first three solvers [4,5] currently exist in MTUQ. Green's functions from each are read and stored as obspy streams overlaid with linear combination machinery for generating synthetics. SPECTFEM3D [6] is easily added in the existing framework, though the far greater cost of working in 3D may require trades to be computed on-the-fly rather than read from precomputed databases.

Software validation

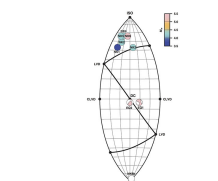


The response of a layered or radially symmetric medium to a moment tensor source can be expressed as a linear combination of traces from four "fundamental sources." MTUQ, engine, and the legacy inversion software CAP all employ this type of representation. Benchmark comparisons between all four packages are included in the MTUQ tests directory.

Full moment tensor analysis of North Korea seismic events



Top: Uncertainty analysis for NK nuclear explosion. Bottom: Uncertainty analysis for nearby NK earthquakes. Detailed information about the uncertainty plots is provided in [7], from which these figures were modified.



Best-fitting moment tensors for 6 NK explosion events (fall near top of above lune plot), 2 nearby earthquakes (falls in middle of plot), and 1 cavity collapse event (falls near bottom). Modified from [7].

Working example

```

import sys
import os
import numpy as np
import obspy
import instaseis
import pymc
import mtuq

# Define the source parameters
def define_source_parameters():
    # Define the source parameters
    # ... (code continues) ...

```

References

- [1] Silwal et al., 2016, Seismic moment tensors and estimated uncertainties in southern Alaska, *JGR*, doi:10.1002/2015JB012588
- [2] Tape and Tape, 2012, A geometric setting for moment tensors, *Geophys. Res. Lett.*, doi:10.1029/2011GL048111
- [3] Salvendy et al., 2015, Probabilistic Programming in Python using PyMC, *arXiv:1507.08050*
- [4] Nissen-Meyer et al., 2014, AxiSEM: broadband 3-D seismic wavefields in axisymmetric media
- [5] IRIS DMC, 2015, Data Services Products: Synthetics Engine, <https://doi.org/10.17611/DP/SYNTHESIS>
- [6] Komratich and Tromp, 2002, Spectral-element simulations of global seismic wave propagation, *Geophys. Res. Lett.*, doi:10.1029/2001GL014545
- [7] Alvizuri and Tape, 2018, Full moment tensor analysis of North Korea nuclear explosions, *JGR*, doi:10.1029/2017JB015848

This work was supported by AFRL contract FA9453-17-C-0025.

Figure 19: Image of 2018 AGU poster featuring MTUQ (Modrak et al., 2018). This publication is not accessible publicly, which is why it is included here. The details can be examined if the reader zooms into the figures displayed in the pdf. The scientific content is addressed elsewhere in the report.

References

- Aki, K., and P. G. Richards (1980), *Quantitative Seismology, Theory and Methods*, W. H. Freeman, San Francisco, Calif., USA.
- Alvizuri, C. (2016), Estimation of Full Moment Tensors, Including Uncertainties, for Earthquakes, Volcanic Events, and Nuclear Explosions, Ph.D. thesis, University of Alaska Fairbanks, USA.
- Alvizuri, C. (2018), Seismic moment tensor results for nine events in the North Korea region, ScholarWorks@UA at <http://hdl.handle.net/11122/8441>: descriptor file with figures, text files of catalog, and input weight files.
- Alvizuri, C., and C. Tape (2016), Full moment tensors for small events ($M_w < 3$) at Uturuncu volcano, Bolivia, *Geophys. J. Int.*, *206*, 1761–1783, doi:10.1093/gji/ggw247.
- Alvizuri, C., and C. Tape (2018), Full moment tensor analysis of nuclear explosions in North Korea, *Seismol. Res. Lett.*, *89*(6), 2139–2151, doi:10.1785/0220180158.
- Alvizuri, C., V. Silwal, L. Krischer, and C. Tape (2018), Estimation of full moment tensors, including uncertainties, for nuclear explosions, volcanic events, and earthquakes, *J. Geophys. Res. Solid Earth*, *123*, 5099–5119, doi:10.1029/2017JB015325.
- Alvizuri, C. R., and C. Tape (2017), Full moment tensors with uncertainties for the 2017 North Korea declared nuclear test and for a collocated, subsequent event, Abstract S43H-2965 presented at 2017 Fall Meeting, AGU, New Orleans, La., 11-15 Dec.
- Baig, A., and T. Urbancic (2010), Microseismic moment tensors: A path to understanding frac growth, *Leading Edge*, *29*, 320–324, doi:10.1190/1.3353729.
- Baig, A. M., F. A. Dahlen, and S.-H. Hung (2003), Traveltimes of waves in three-dimensional random media, *Geophys. J. Int.*, *153*, 467–482.
- Bozdağ, E., D. Peter, M. Lefebvre, D. Komatitsch, J. Tromp, J. Hill, N. Podhorszki, and D. Pugmire (2016), Global adjoint tomography: first-generation model, *Geophys. J. Int.*, *207*, 1739–1766, doi:10.1093/gji/ggw356.
- Carrington, L., D. Komatitsch, M. Laurenzano, M. M. Tikir, D. Michéa, N. Le Goff, A. Snively, and J. Tromp (2008), High-frequency simulations of global seismic wave propagation using SPECFEM3D_GLOBE on 62K processors, in *International Conference for High Performance Computing, Storage, and Analysis (SC 2008)*, IEEE, doi:10.1109/SC.2008.5215501.
- Casarotti, E., M. Stupazzini, S. J. Lee, D. Komatitsch, A. Piersanti, and J. Tromp (2008), CUBIT and seismic wave propagation based upon the spectral-element method: An advanced unstructured mesher for complex 3D geological media, in *Proceedings of the 16th International Meshing Roundtable*, edited by M. L. Brewer and D. Marcum, pp. 579–597, Springer, Berlin.
- Chapman, C. H., and W. S. Leaney (2012), A new moment-tensor decomposition for seismic events in anisotropic media, *Geophys. J. Int.*, *188*, 343–370, doi:10.1111/j.1365-246X.2011.05265.x.

- Chen, P., T. H. Jordan, and L. Zhao (2005), Finite-moment tensor of the 3 September 2002 Yorba Linda earthquake, *Bull. Seismol. Soc. Am.*, *95*(3), 1170–1180, doi:10.1785/0120040094.
- Chen, W., S. Ni, H. Kanamori, S. Wei, Z. Jia, and L. Zhu (2015), CAPjoint, A computer software package for joint inversion of moderate earthquake source parameters with local and teleseismic waveforms, *Seismol. Res. Lett.*, *86*(2A), 432–441, doi:10.1785/0220140167.
- Chow, B., Y. Kaneko, C. Tape, R. Modrak, and J. Townend (2020), An automated workflow for adjoint tomography—waveform misfits and synthetic inversions for the North Island, New Zealand, *Geophys. J. Int.*, *223*, 1461–1480, doi:10.1093/gji/ggaa381.
- Covellone, B. M., and B. Savage (2012), A quantitative comparison between 1D and 3D source inversion methodologies: Application to the Middle East, *Bull. Seismol. Soc. Am.*, *102*(5), 2189–2199, doi:10.1785/0120110278.
- Dammeier, F., A. guilhem, J. R. Moore, F. Haslinger, and S. Loew (2015), Moment tensor analysis of rockslide seismic signals, *Bull. Seismol. Soc. Am.*, *105*(6), 3001–3014, doi:10.1785/0120150094.
- Dreger, D., R. Uhrhammer, M. Pasyanos, J. Franck, and B. Romanowicz (1998), Regional and far-regional earthquake locations and source parameters using sparse broadband networks: A test on the Ridgecrest sequence, *Bull. Seismol. Soc. Am.*, *88*(6), 1353–1362.
- Dreger, D. S., H. Tkalčić, and M. Johnson (2000), Dilational processes accompanying earthquakes in the Long Valley Caldera, *Science*, *288*, 122–125.
- Dreger, D. S., A. Chiang, S. R. Ford, and W. R. Walter (2012), Refinement of regional distance seismic moment tensor and uncertainty analysis for source-type identification, in *Proceedings of the 2012 Monitoring Research Review: Ground-Based Nuclear Explosion Monitoring Technologies*, pp. 377–386, National Nuclear Security Administration.
- Dziewonski, A., T.-A. Chou, and J. H. Woodhouse (1981), Determination of earthquake source parameters from waveform data for studies of global and regional seismicity, *J. Geophys. Res.*, *86*(B4), 2825–2852, doi:10.1029/JB086iB04p02825.
- Ford, S. R., D. S. Dreger, and W. R. Walter (2009), Identifying isotropic events using a regional moment tensor inversion, *J. Geophys. Res.*, *114*, B01306, doi:10.1029/2008JB005743.
- Ford, S. R., D. S. Dreger, and W. R. Walter (2010), Network sensitivity solutions for regional moment-tensor inversions, *Bull. Seismol. Soc. Am.*, *100*(5A), 1962–1970, doi:10.1785/0120090140.
- Ford, S. R., W. R. Walter, and D. S. Dreger (2012), Event discrimination using regional moment tensors with teleseismic-*P* constraints, *Bull. Seismol. Soc. Am.*, *102*(2), 867–872, doi:10.1785/0120110227.
- Ford, S. R., G. D. Kraft, and G. A. Ichinose (2020), Seismic moment tensor event screening, *Geophys. J. Int.*, *221*, 77–88, doi:10.1093/gji/ggz578.

- Frankel, A., and R. W. Clayton (1986), Finite difference simulations of seismic scattering: Implications for the propagation of short-period seismic waves in the crust and models of crustal heterogeneity, *J. Geophys. Res.*, *91*(B6), 6465–6489.
- Hjörleifsdóttir, V., and G. Ekström (2010), Effects of three-dimensional Earth structure on CMT earthquake parameters, *Phys. Earth Planet. Inter.*, *179*, 178–190.
- Hudson, J. A., R. G. Pearce, and R. M. Rogers (1989), Source type plot for inversion of the moment tensor, *J. Geophys. Res.*, *94*(B1), 765–774.
- Komatitsch, D., and J. Tromp (2002a), Spectral-element simulations of global seismic wave propagation—I. Validation, *Geophys. J. Int.*, *149*, 390–412.
- Komatitsch, D., and J. Tromp (2002b), Spectral-element simulations of global seismic wave propagation—II. Three-dimensional models, oceans, rotation and self-gravitation, *Geophys. J. Int.*, *150*, 308–318.
- Komatitsch, D., J. Ritsema, and J. Tromp (2002), The spectral-element method, Beowulf computing, and global seismology, *Science*, *298*, 1737–1742.
- Komatitsch, D., Q. Liu, J. Tromp, P. Süß, C. Stidham, and J. H. Shaw (2004), Simulations of ground motion in the Los Angeles basin based upon the spectral-element method, *Bull. Seismol. Soc. Am.*, *94*(1), 187–206, doi:10.1785/0120030077.
- Krischer, L., T. Mengies, R. Barsch, M. Beyreuther, T. Lecocq, C. Caudron, and J. Wassermann (2015), ObsPy: a bridge for seismology into the scientific Python ecosystem, *Computational Science & Discovery*, *8*(1), 014003, doi:10.1088/1749-4699/8/1/014003.
- Krischer, L., A. R. Hutko, M. van Driel, S. Stähler, M. Bahavar, C. Trabant, and T. Nissen-Meyer (2017), On-demand custom broadband synthetic seismograms, *Seismol. Res. Lett.*, *88*(4), 1127–1140, doi:10.1785/0220160210.
- Lahr, J. C. (1975), Detailed Seismic Investigation of Pacific–North American Plate Interaction in Southern Alaska, Ph.D. thesis, Columbia University, New York City.
- Lee, E.-J., P. Chen, T. H. Jordan, and L. Wang (2011), Rapid full-wave centroid moment tensor (CMT) inversion in a three-dimensional earth structure model for earthquakes in Southern California, *Geophys. J. Int.*, *186*, 311–330, doi:10.1111/j.1365-246X.2011.05031.x.
- Lin, F.-C., B. Schmandt, and V. C. Tsai (2012), Joint inversion of Rayleigh wave phase velocity and ellipticity using USArray: Constraining velocity and density structure in the upper crust, *Geophys. Res. Lett.*, *39*, L12303, doi:10.1029/2012GL052196.
- Lin, F.-C., D. Li, R. W. Clayton, and D. Hollis (2013), High-resolution 3D shallow crustal structure in Long Beach, California: Application of ambient noise tomography on a dense seismic array, *Geophysics*, *78*(4), Q45–Q56.
- Liu, J., L. Li, J. Zahradník, E. Sokos, C. Liu, and X. Tian (2018), North Korea’s 2017 test and its nontectonic aftershock, *Geophys. Res. Lett.*, *45*, 3017–3025, doi:10.1002/2018GL077095.

- Liu, Q., J. Polet, D. Komatitsch, and J. Tromp (2004), Spectral-element moment tensor inversions for earthquakes in southern California, *Bull. Seismol. Soc. Am.*, *94*(5), 1748–1761, doi:10.1785/012004038.
- Matumoto, T., and R. A. Page (1969), Microaftershocks following the Alaska earthquake of March 28, 1964: Determination of hypocenters and crustal velocities in the Kenai Peninsula-Prince William Sound area, in *The Prince William Sound, Alaska, Earthquake of 1964 and Aftershocks*, vol. II, Parts B and C, edited by L. E. Leipold and F. J. Wood, pp. 157–173, U.S. Department of Commerce, Environmental Science Services Administration, Washington, D.C., U.S. Coast and Geodetic Survey Publication 10-3.
- Minson, S. E., D. S. Dreger, R. Bürgmann, H. Kanamori, and K. M. Larson (2007), Seismically and geodetically determined nondouble-couple source mechanisms from the 2000 Miyakejima volcanic earthquake swarm, *J. Geophys. Res.*, *112*, B10308, doi:10.1029/2006JB004847.
- Modrak, R., V. Silwal, C. Alvuzuri, and C. Tape (2018), Moment tensor inversion and uncertainty quantification using mtuq, instaseis, obspy and pymc, Abstract S53E-0458 presented at 2018 Fall Meeting, AGU, Washington, D.C., 10-14 Dec.
- Modrak, R., C. Tape, C. R. Alvizuri, V. Silwal, J. Thurin, and M. Cleveland (2020), Force and moment tensor uncertainty quantification with 1D and 3D Earth models, Abstract S045-0004 presented at 2020 Fall Meeting, AGU, 1-17 Dec.
- Mustać, M., and H. Tkalčić (2016), Point source moment tensor inversion through a Bayesian hierarchical model, *Geophys. J. Int.*, *204*, 311–323, doi:10.1093/gji/ggv458.
- NCSA (2013), 4 Applications Sustain 1 Petaflop on BlueWaters, (Retrieved 16 Jan 2015).
- Nissen-Meyer, T., A. Fournier, and F. A. Dahlen (2007), A two-dimensional spectral-element method for computing spherical-earth seismograms – I. Moment-tensor source, *Geophys. J. Int.*, *168*, 1067–1092, doi:10.1111/j.1365-246X.2006.03121.x.
- Nissen-Meyer, T., M. van Driel, S. C. Stähler, K. Hosseini, S. Hempel, L. Auer, A. Colombi, and A. Fournier (2014), AxiSEM: broadband 3-D seismic wavefields in axisymmetric media, *Solid Earth*, *5*, 425–445, doi:10.5194/se-5-425-2014.
- NVIDIA (2012), NVIDIA Tesla K20-K20X GPU Accelerators – Benchmarks, application Performance Technical Brief, 8 p.
- Patton, H. J., and S. R. Taylor (2011), The apparent explosion moment: Inferences of volumetric moment due to source medium damage by underground nuclear explosions, *J. Geophys. Res.*, *116*, B03310, doi:10.1029/2010JB007937.
- Pesicek, J. D., J. Sileny, S. G. Prejean, and C. H. Thurber (2012), Determination and uncertainty of moment tensors for microearthquakes at Okmok Volcano, Alaska, *Geophys. J. Int.*, *190*, 1689–1709, doi:10.1111/j.1365-246X.2012.05574.x.
- Peter, D., et al. (2011), Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes, *Geophys. J. Int.*, *186*, 721–739.

- Riedesel, M. A., and T. H. Jordan (1989), Display and assessment of seismic moment tensors, *Bull. Seismol. Soc. Am.*, *79*(1), 85–100.
- Shapiro, N. M., M. Campillo, L. Stehly, and M. H. Ritzwoller (2005), High-resolution surface-wave tomography from ambient seismic noise, *Science*, *307*, 1615–1618.
- Shen, W., M. H. Ritzwoller, V. Schulte-Pelkum, and F.-C. Lin (2013), Joint inversion of surface wave dispersion and receiver functions: a Bayesian Monte-Carlo approach, *Geophys. J. Int.*, *192*, 807–836.
- Šílený, J., and A. Milev (2006), Seismic moment tensor resolution on a local scale: simulated rock-burst and mine-induced seismic events in the Kopanang gold mine, South Africa, *Pure App. Geophys.*, *163*, 1495–1513, doi:10.1007/s00024-006-0089-z.
- Silwal, V., and C. Tape (2016), Seismic moment tensors and estimated uncertainties in southern Alaska, *J. Geophys. Res. Solid Earth*, *121*, 2772–2797, doi:10.1002/2015JB012588.
- Sipkin, S. A. (1986), Estimation of earthquake source parameters by the inversion of waveform data: global seismicity, 1981–1983, *Bull. Seismol. Soc. Am.*, *76*(6), 1515–1541.
- Song, X. J., D. V. Helmberger, and L. Zhao (1996), Broad-band modelling of regional seismograms: the Basin and Range crustal structure, *Geophys. J. Int.*, *125*, 15–29.
- Stierle, E., V. Vavryčuk, J. Šílený, and M. Bohnhoff (2014), Resolution of non-double-couple components in the seismic moment tensor using regional networks—I: a synthetic case study, *Geophys. J. Int.*, *196*, 1869–1877, doi:10.1093/gji/ggt502.
- Tan, Y., A. Song, S. Wei, and D. Helmberger (2010), Surface wave path corrections and source inversions in southern California, *Bull. Seismol. Soc. Am.*, *100*(6), 2891–2904, doi:10.1785/0120090063.
- Tape, C., and W. Tape (2018), The eigenvalue lune as a window on moment tensors, Abstract presented at 2018 SSA Annual Meeting, Miami, Fl., 14-17 May.
- Tape, C., Q. Liu, A. Maggi, and J. Tromp (2009), Adjoint tomography of the southern California crust, *Science*, *325*, 988–992, doi:10.1126/science.1175298.
- Tape, C., Q. Liu, A. Maggi, and J. Tromp (2010), Seismic tomography of the southern California crust based on spectral-element and adjoint methods, *Geophys. J. Int.*, *180*, 433–462, doi:10.1111/j.1365-246X.2009.04429.x.
- Tape, C., C. Alvizuri, V. Silwal, and W. Tape (2017), Uncertainties for seismic moment tensors and applications to nuclear explosions, volcanic events, and earthquakes, Abstract S44B-01 presented at 2017 Fall Meeting, AGU, New Orleans, La., 11-15 Dec.
- Tape, C., C. R. Alvizuri, and W. Tape (2018), Uncertainty estimation of moment tensor source types, Abstract presented at 2018 SSA Annual Meeting, Miami, Fl., 14-17 May.
- Tape, W., and C. Tape (2012a), A geometric setting for moment tensors, *Geophys. J. Int.*, *190*, 476–498, doi:10.1111/j.1365-246X.2012.05491.x.

- Tape, W., and C. Tape (2012b), A geometric comparison of source-type plots for moment tensors, *Geophys. J. Int.*, *190*, 499–510, doi:10.1111/j.1365-246X.2012.05490.x.
- Tape, W., and C. Tape (2012c), Angle between principal axes triples, *Geophys. J. Int.*, *191*, 813–831, doi:10.1111/j.1365-246X.2012.05658.x.
- Tape, W., and C. Tape (2013), The classical model for moment tensors, *Geophys. J. Int.*, *195*, 1701–1720, doi:10.1093/gji/ggt302.
- Tape, W., and C. Tape (2015), A uniform parameterization of moment tensors, *Geophys. J. Int.*, *202*, 2074–2081, doi:10.1093/gji/ggv262.
- Tape, W., and C. Tape (2016), A confidence parameter for seismic moment tensors, *Geophys. J. Int.*, *205*, 938–953, doi:10.1093/gji/ggw057.
- Tape, W., and C. Tape (2019), The eigenvalue lune as a window on moment tensors, *Geophys. J. Int.*, *216*, 19–33, doi:10.1093/gji/ggy373.
- van Driel, M., L. Krischer, S. C. Stähler, K. Hosseini, and T. Nissen-Meyer (2015), Instaseis: instant global seismograms based on a broadband waveform database, *Solid Earth*, *6*, 701–717, doi:10.5194/se-6-701-2015.
- Vavryčuk, V. (2011), Tensile earthquakes: Theory, modeling, and inversion, *J. Geophys. Res.*, *116*, B12320, doi:10.1029/2011JB008770.
- Walter, F., J. F. Clinton, N. Deichmann, D. S. Dreger, S. E. Minson, and M. Funk (2009), Moment tensor inversions of icequakes on Gornergletscher, Switzerland, *Bull. Seismol. Soc. Am.*, *99*(2A), 852–870, doi:10.1785/0120080110.
- Walter, F., D. S. Dreger, J. F. Clinton, N. Deichmann, and M. Funk (2010), Evidence for near-horizontal tensile faulting at the base of Gornergletscher, a Swiss Alpine Glacier, *Bull. Seismol. Soc. Am.*, *100*(2), 458–472, doi:10.1785/0120090083.
- Walter, W. R., K. D. Smith, J. L. O’Boyle, T. F. Hauk, F. Ryall, S. D. Ruppert, S. C. Myers, R. Abbot, and D. A. Dodge (2006), An assembled western United States dataset for regional seismic analysis, ISSO 9660 CD, LLNL release UCRL-MI-222502, available via download request at <https://ds.iris.edu/mda/18-001>.
- Wéber, Z. (2006), Probabilistic local waveform inversion for moment tensor and hypocentral location, *Geophys. J. Int.*, *165*, 607–621.
- Zhao, L.-S., and D. V. Helmberger (1994), Source estimation from broadband regional seismograms, *Bull. Seismol. Soc. Am.*, *84*(1), 91–104.
- Zhu, H., E. Bozdag, D. Peter, and J. Tromp (2012), Structure of the European upper mantle revealed by adjoint tomography, *Nature Geoscience*, *5*, 493–498.
- Zhu, L., and Y. Ben-Zion (2013), Parameterization of general seismic potency and moment tensors for source inversion of seismic waveform data, *Geophys. J. Int.*, *194*, 839–843, doi:10.1093/gji/ggt137.

- Zhu, L., and D. Helmberger (1996), Advancement in source estimation techniques using broadband regional seismograms, *Bull. Seismol. Soc. Am.*, 86(5), 1634–1641.
- Zhu, L., and L. A. Rivera (2002), A note on the dynamic and static displacements from a point source in multilayered media, *Geophys. J. Int.*, 148, 619–627, doi:10.1046/j.1365-246X.2002.01610.x.
- Zhu, L., and X. Zhou (2016), Seismic moment tensor inversion using 3D velocity model and its application to the 2013 Lushan earthquake sequence, *Physics and Chemistry of the Earth*, 95, 10–18, doi:10.1016/j.pce.2016.01.002.

DISTRIBUTION LIST

DTIC/OCF	
8725 John J. Kingman Rd, Suite 0944	
Ft Belvoir, VA 22060-6218	1 cy
AFRL/RVIL	
Kirtland AFB, NM 87117-5776	1 cy
Official Record Copy	
AFRL/RVB/Dr. Frederick R. Schult	1 cy

This page is intentionally left blank.