To The Graduate School:

The members of the Committee approve the thesis of Joel N. Duenow presented on May 13, 2003.

_____
Scott B. Smithson, Chairman

_____
Kenneth G. Dueker

_____
Robert F. Kubichek

_____
Igor B. Morozov

APPROVED:

_____
James I. Drever, Head, Department of Geology and Geophysics

_____
Don Roth, Dean, The Graduate School

Duenow, Joel N., Surface Scattering as a Source of Teleseismic Arrival Coda, M.S., Department of Geology and Geophysics, August, 2003.

Seismic codas of short-period Peaceful Nuclear Explosion (PNE) arrivals are strong and extensive in time. Some previous interpretations have described the teleseismic P_n coda as resulting from scattering occurring in the upper 100 km of the mantle, but these interpretations have minimized effects of crustal scattering and have not addressed codas of other arrivals. Because the crust is by far the most heterogeneous part of the Earth, it should be a key contributor to the scattered wavefields. To determine whether upper-crustal scattering explains seismic coda, a surface scattering model was created in which crustal heterogeneities are described as scattering points on the Earth's surface. 1-D reflectivity synthetics based on velocity models with different crustal and upper mantle complexities governed wave propagation. The seismic response was calculated using numerical integration in time and a 2-D surface integral over uniformly-distributed, Monte-Carlo-sampled surface points. Different crustal Q (Q_{crust}) values were used in creating the reflectivity synthetics to determine the observed coda Q (Q_{coda}) as a function of Q_{crust} for both teleseismic P and L_g events. These relationships were then used to estimate Q_{crust} for Quartz PNE data by measuring Q_{coda} values.

Using reflectivity synthetics based on velocity models of greater crustal complexity, like the Quartz-4 model by Morozova *et. al.* (1999) and a 5-layer crust model, led to greater coda strength in resulting receiver traces and stronger dependence of Q_{coda} on Q_{crust}. For the teleseismic P coda at 0.3 Hz, the Quartz-4 model gave a strong dependence of $Q_{coda} = 16 \cdot Q_{crust}^{0.47}$ at 0.3 Hz and $Q_{coda} = 28 \cdot Q_{crust}^{0.43}$ at 0.7 Hz. The IASP-91 model, because of its much simpler crustal structure and fewer crustal arrivals, gave a much weaker dependence of $Q_{coda} = 75 \cdot Q_{crust}^{0.06}$ at 0.3 Hz and $Q_{coda} = 75 \cdot Q_{crust}^{0.14}$ at 0.7 Hz. A model with IASP-91 crust and Quartz-4 mantle showed that the crust, not the mantle, played the most significant role in determining coda strength, since the arrivals in the synthetics were nearly identical to those in the IASP-91 model, and the relation, given by $Q_{coda} =$

$49 \cdot Q_{crust}^{0.14}$ at 0.3 Hz, is more similar to that of the IASP-91 model than the Quartz-4 model. Thus, velocity models with more complex crusts clearly result in higher-complexity seismic phase arrivals, stronger coda, and stronger Q_{coda}-Q_{crust} dependence.

The scattering potential per unit area, an empirical measure of scattering efficiency in terms of amplitude, was determined to be 1.25 times greater for L_g waves than for P waves for the Quartz-4 model. For the 5-layer crust model, the ratio was 4.25. These results suggest that L_g waves are an important component of the seismic coda.

An estimate of Q_{crust} for Quartz PNE data was made based on these estimates, though the frequency windows of PNE data and model data differed due to modeling limitations. For a teleseismic P Q_{coda} value measured from a PNE Quartz trace at 1.3 Hz, the Quartz-4 model gave a Q_{crust} value of 169. The Quartz-4 L_g parameterization at 1.3 Hz gave a Q_{crust} estimate of 23 using the PNE Quartz L_g coda. These are crustal values commensurate with others obtained previously in other regions.

SURFACE SCATTERING AS A SOURCE OF TELESEISMIC ARRIVAL CODA

by
Joel N. Duenow

A thesis submitted to the Department of Geology and Geophysics
and The Graduate School of The University of Wyoming
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE
in
GEOPHYSICS

Laramie, Wyoming
August, 2003

Acknowledgments

This work was sponsored by Defense Threat Reduction Agency Grants DTRA01-01-C-0057 and DTRA01-01-C-0081.

Table of Contents

List of Figures

List of Appendices

1 Introduction

The Comprehensive Nuclear-Test-Ban Treaty (CTBT) was adopted by the United Nations General Assembly in 1996 in an effort to restrict the development of new weapons and prevent countries without nuclear weapons from developing them. To date, 101 countries have ratified the treaty (CTBTO Preparatory Commission). CTBT monitoring technologies include land-based seismic, ocean hydroacoustic, infrasound, and radionuclide detectors. Seismic monitoring of the CTBT requires detailed knowledge of regional seismic phase propagation to allow for corrections of travel-times and amplitudes of regional seismic events because these seismic properties are highly sensitive to lithospheric heterogeneities (Morozov and Smithson 2001). Corrections for lithospheric structure obtained by seismic methods, for example, allow greater accuracy in source depth, leading to improved discernment between nuclear explosions and earthquakes.

Traditional approaches to the analysis of crustal structure using long-range (100-1000 km) seismic data include wide-angle reflection and refraction imaging. These methods, employed extensively worldwide, are primarily based on travel-time interpretation, utilizing only short segments of the seismic records following identified seismic events to constrain velocity heterogeneity and reflectivity. Other important physical properties of the crust such as attenuation have been studied inadequately and remain poorly understood. In the work presented here, I examine seismic attenuation by numerically modeling the seismic coda as resulting from scattering of seismic energy from near-surface crustal heterogeneities.

1.1 Types of Attenuation

Two types of attenuation govern seismic energy loss during propagation. Intrinsic attenuation describes energy loss due to absorption of energy within the material through which seismic waves propagate. Friction along grain boundaries and damping by fluids in cracks and pores are common absorption mechanisms. Since absorption reduces the amount of elastic energy

propagating, intrinsic attenuation decreases the amplitude of the seismic signal. Intrinsic attenuation is commonly described in terms of quality factor Q, defined by $\frac{2\pi E}{-\Delta E}$, where E is the energy of a seismic wave and ΔE is the energy loss (defined negative) per cycle (Aki and Richards 1980). Because it measures the ratio of energy propagating in a wave to energy dissipated in one period, Q is inversely proportional to the attenuation of the wave. P and S waves have separate intrinsic Q values (Q_p and Q_s) because of their differences in wavelength and propagation mechanism. Q_p generally has a value about twice that of Q_s (Anderson and Given 1982).

Elastic attenuation describes scattering of seismic energy away from its original propagation path. Typical scatterers include heterogeneities such as crustal faults, mountain belts, coastlines, basins, and surface topography. Elastic attenuation does not reduce the amount of energy propagating, but rather redirects it. Scattered energy arrives at a different time than energy propagating directly to a receiver because of the different path length traveled to and from scatterers. Therefore, in a region of scatterers, the energy of the scattered arrival is spread over a greater time than the direct arrival, with a coda following the arrival and decaying in amplitude with time.

1.2 Peaceful Nuclear Explosion (PNE) Data

Peaceful Nuclear Explosion (PNE) data gathered in the Former Soviet Union provide an unparalleled reference data set on which to perform amplitude measurements because of long offsets, dense receiver spacing, three-component recordings, and presence of both crustal and mantle seismic phases in the records. These profiles were the central part of the extensive Deep Seismic Sounding (DSS) investigation program from the 1960s through 1990s, using both PNEs and chemical explosions as seismic sources (Sultanov *et. al.* 1999). Several PNE profiles, recorded with analog, three-component seismographs, were gathered across the various geologic provinces of the

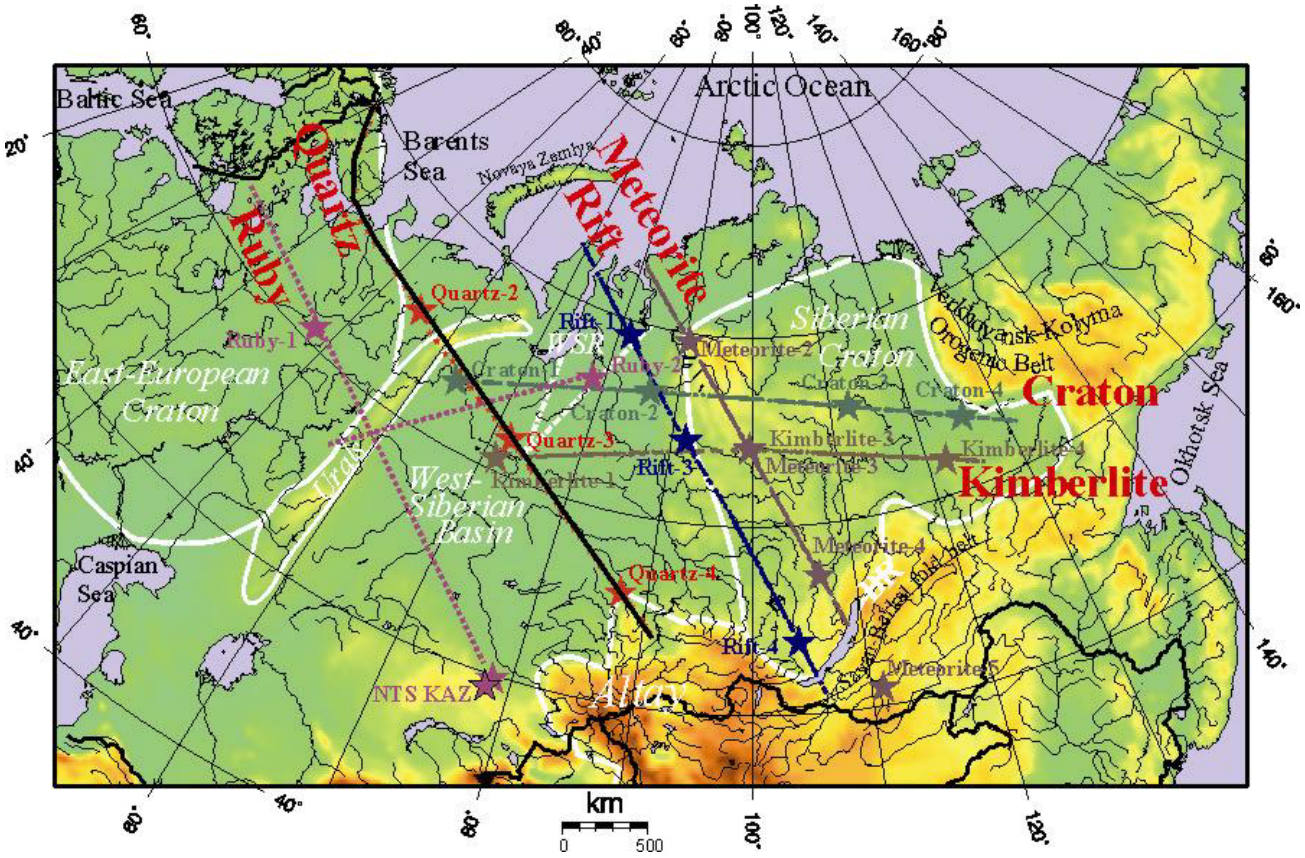


Figure 1.1: PNE profiles in the Former Soviet Union. Stars indicate nuclear explosion source locations. From Morozov and Smithson 2001.

Former Soviet Union (Figure 1.1). Some extended for over 3000 km with a nominal receiver spacing of 10 km, allowing for seismic observation to depths exceeding 800 km (Egorkin *et. al.* 1987; Ryaboy 1989; Kozlovsky 1990). The profiles typically included 3-4 PNE explosions and 50-80 chemical explosions along the seismic line. The Quartz PNE profile across the East European Craton, Ural Mountains, and West Siberian Basin has been extensively analyzed, though information is still being extracted from the data. In this work, I improve estimates of crustal attenuation for the Quartz profile. Several other profiles remain to be analyzed fully (Egorkin and Pavlenkova 1981; Mechie *et. al.* 1993, 1997; Priestley *et. al.* 1994; Pavlenkova 1996; Pavlenkova *et. al.* 1996; Ryberg *et. al.* 1996, 1998; Cipar and Priestley 1997; Egorkin 1997, 1998; Tittgemeyer *et. al.* 1997, 1999; Schueller *et. al.* 1997; Morozov *et. al.* 1998a,b; Nielsen *et. al.* 1999; Morozova *et. al.* 1999, 2000).

PNE records show several seismic phases (Figure 1.2; Figure 1.3). P_g is a crustal *P* phase typically propagating at velocities between 5.5 and 6.0 km/s. S_g, also a crustal phase, propagates at

approximately 3.2 to 3.5 km/s; beyond the critical point, it is called the L_g phase. L_g has a complex structure with no clear onset, but its maximum amplitude has a group velocity of 3.5 km/s and a coda trailing to group velocity of 2.9 km/s or less (Kennett 1986). The L_g phase may either be associated with a superposition of many higher modes of interfering surface waves trapped in the crust, or as interference of multiply-reflected *S* waves bouncing back and forth between the crust-mantle boundary, crustal interfaces, and the free surface (Kennett 1986). The P_n phase propagates in the upper few kilometers beneath the Moho as a refracted wave with velocity of approximately 8 km/s. S_n also propagates beneath the Moho, with a group velocity of 3.5-4.5 km/s. Reflections from mantle-transition-zone discontinuities at 410 km and 610 km depth are evident in PNE data as well.

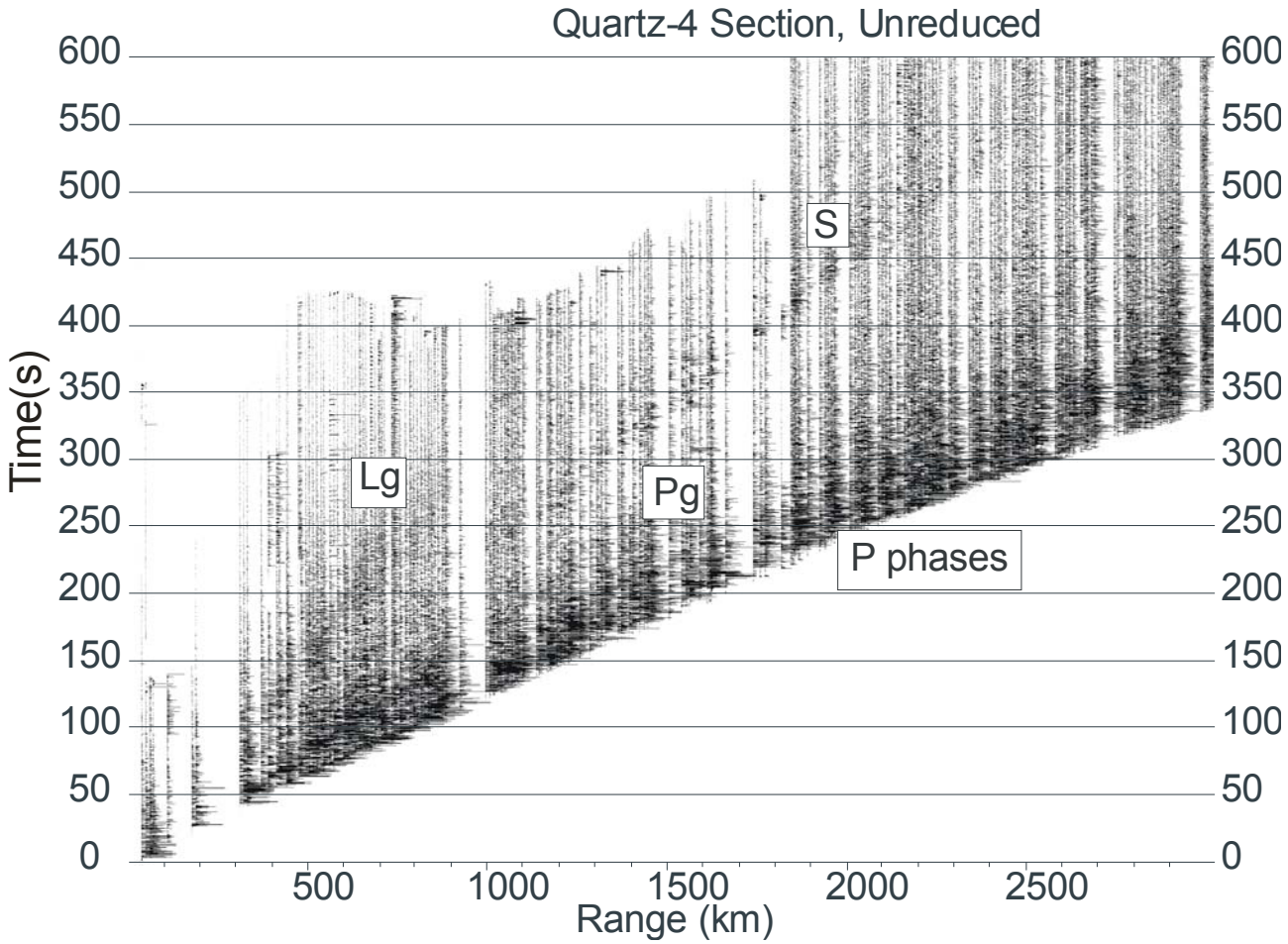


Figure 1.2: Quartz PNE section, shot number 4, unreduced time axis. Note strong L_g phase to approximately 1000 km offset and strong *P* phases.

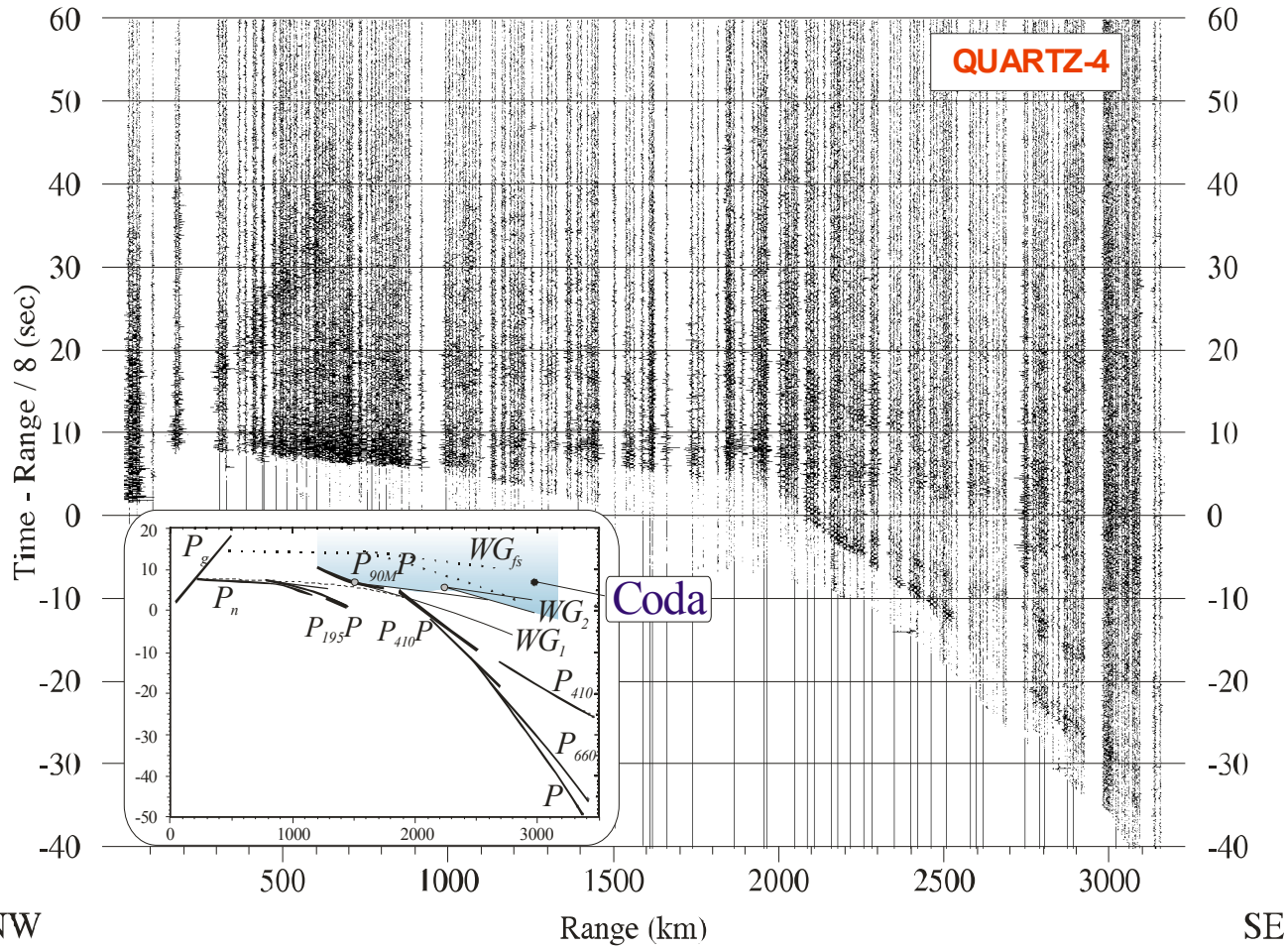


Figure 1.3: Quartz-4 section with time axis reduced at 8 km/s. Inset indicates seismic phase arrivals seen in the data. Whispering gallery (*WG*) phases indicate the teleseismic P_n. $P_{90M}P$, $P_{195}P$, and $P_{410}P$ are reflections from 90 km, 195 km, and 410 km depth, respectively. P_n, P_{410}, and P_{660} are refracted waves. The high-energy teleseismic *P* coda is shaded. From Morozov and Smithson 2000.

The teleseismic P_n, observed in both PNE and large conventional explosion data, propagates below the Moho to distances as great as 3000 km. It travels beneath both continents and oceans, interrupted only by major plate tectonic boundaries such as mid-ocean ridges, island arcs, and subduction zones (Molnar and Oliver 1969), with group velocity between 8.0 and 8.1 km/s and an additional branch at 8.5 km/s beyond 2700 km (Morozov *et. al.* 1998a). Note the strong, extensive coda seen at offsets beyond 1500 km (Figure 1.3), commonly thought to be due to scattering that delays the arrival of seismic energy, though the precise scattering mechanism and location remain in debate (Ryberg *et. al.* 1995, 2000; Tittgemeyer *et. al.* 1996, 2000; Ryberg and Wenzel 1999; Morozov *et. al.* 1998a; Morozov and Smithson 2000, 2001; Morozov 2001; Nielsen *et. al.* 2001).

Various wave-guide mechanisms have been proposed to describe teleseismic P_n propagation, including transmission of energy in a low-velocity zone beneath the Moho (Sutton and Walker 1972), guided waves in a high-velocity layer above a low-velocity layer (Mantovani *et. al.* 1977), fine-scale sub-Moho layering or sub-Moho scatterers (Figure 1.4, MS1) (Fuchs and Schulz 1976; Ryberg *et. al.* 1995; Tittgemeyer *et. al.* 1996, 2000; Ryberg and Wenzel 1999), scattering from partial-melt zones below 100 km depth (Figure 1.4, MS2) (Thybo and Perchuć 1997), and whispering-gallery modes of multiple sub-Moho refractions (Figure 1.4, MD) (Stephens and Isacks 1977; Menke and Richards 1980; Morozov *et. al.* 1998a).

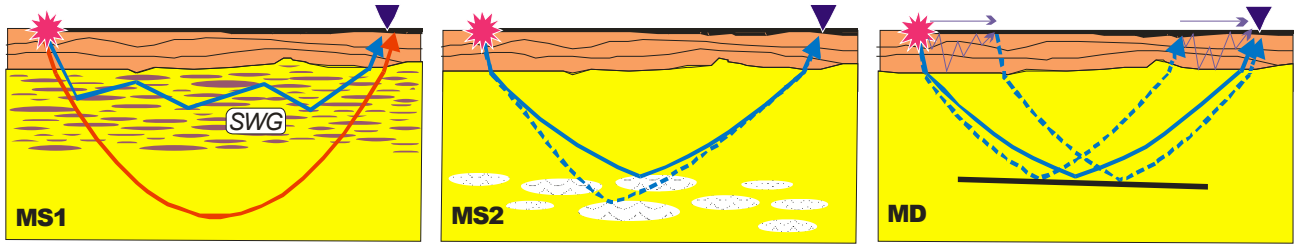


Figure 1.4: Three competing models for describing the source of coda energy by scattering. MS1: Strongly scattering mantle lid; high-frequency phases reach the receivers by a diffusive scattering-waveguide (SWG) propagation (blue ray); phases penetrating deeper lose their high-frequency energy in this upper layer (red ray) (Ryberg *et. al.* 1995). MS2: Upper 80 km of mantle is relatively transparent; single scattering occurs (possibly) on pockets of partial melt between 120-200 km; this scattering causes increased complexity of the first arrivals (Thybo and Perchuć, 1997). MD: Observed coda is mainly due to crustal scattering at both source and receiver ends (Morozov and Smithson 2000). (from Morozov and Smithson 2001)

Ryberg *et. al.* (1995) proposed a velocity model containing small-scale velocity fluctuations in the upper mantle to depths of approximately 100 km to describe a scattering waveguide along which the teleseismic P_n could propagate (Figure 1.4, MS1). Tittgemeyer *et. al.* (1996) and Ryberg and Wenzel (1999) performed reflectivity modeling to test similar models; Ryberg and Wenzel (1999) found that a scattering zone below the Moho of 75 km thickness with lamellae of 2 km average thickness and RMS velocity perturbation of 5% best matched the high-frequency (> 5 Hz) teleseismic P_n. These authors considered only the teleseismic P_n coda at frequencies greater than 5 Hz. This model neither describes the strong teleseismic P coda at frequencies below 5 Hz, shown by Morozov *et. al.* (1998a) to be stronger than that above 5 Hz, nor explains the coda of other seismic arrivals.

The crustal structure of the Ryberg and Wenzel (1999) and Tittgemeyer *et. al.* (1996, 2000) mantle scattering models was oversimplified. Tittgemeyer *et. al.* (2000) modeled the crust as a homogeneous layer with *P*-wave velocity (v_p) of 6.2 km/s to 30 km depth and a 5-km-thick transition zone with v_p = 7.5 km/s. The Moho was modeled as a first-order discontinuity with v_p contrast from 7.5 to 8.04 km/s. The simple, non-reflective crust of these models is unrealistic because the Earth's crust is the most structurally and compositionally heterogeneous part of the planet. Many authors have proposed that seismic coda can be explained by scattering within the crust (Greenfield 1971; Bannister *et. al.* 1990; Gupta *et. al.* 1991; Morozov *et. al.* 1998a; Morozov and Smithson 2000, 2001). Morozov and Smithson (2001) suggest that crustal scattering takes place both near the source and near the receiver (Figure 1.4, MD). This scattering within the crust can produce the necessary time delay of arrival energy to form a coda for all seismic events, not only the teleseismic P_n.

1.3 Crustal and Coda Attenuation

Two different *Q* values are important in this study. Intrinsic crustal attenuation, loss of seismic energy in the crust to internal friction, is described by an intrinsic crustal *Q* (Q_{crust}) value, where $Q_{crust} = \frac{2\pi E}{-\Delta E}$, *E* is energy, and ΔE is the energy loss per cycle. Coda *Q* (Q_{coda}) measures coda amplitude decay rate. It is determined by measuring the decay slope of amplitude *A* versus time *t* from a semi-log plot of the seismic trace envelope. Q_{coda} also depends on the dominant frequency of the signal *f* (Morozov and Smithson 2000):

$$A = Ao \exp\left(\frac{-\omega t}{2Q_{coda}}\right) = Ao \exp\left(\frac{-2\pi f t}{2Q_{coda}}\right) = Ao \exp\left(\frac{-\pi f t}{Q_{coda}}\right)$$

Therefore, taking the natural logarithm, $\ln A = \ln Ao + \frac{-\pi f t}{Q_{coda}}$, gives a straight line with slope $m = \frac{-\pi f}{Q_{coda}}$. Thus $\frac{Q_{coda}}{f} = \frac{-\pi}{slope}$, where slope is measured from the seismic trace envelope.

Values of Q_{crust} and Q_{coda} may differ. For example, Q_{coda} less than Q_{crust} indicates that seismic energy is leaking into the mantle and does not return to the surface where it may be recorded by a receiver. Q_{coda} greater than Q_{crust} indicates mantle phases are returning to the surface and contribute to the seismic energy recorded by the receiver. These mechanisms may be frequency-dependent, causing frequency dependence of the Q_{coda}-Q_{crust} relation. In this work, I determine Q_{crust} values for Quartz PNE data based on the Q_{coda} values measured from coda envelope slopes at different frequencies.

Morozov and Smithson (2000) performed seismic trace envelope measurements on Quartz PNE data for the *P* coda, which is composed of several *P* arrival codas, including *P*, $P_{410}P$, and two whispering-gallery (*WG*) phases describing the teleseismic P_n (Figure 1.5). These authors found

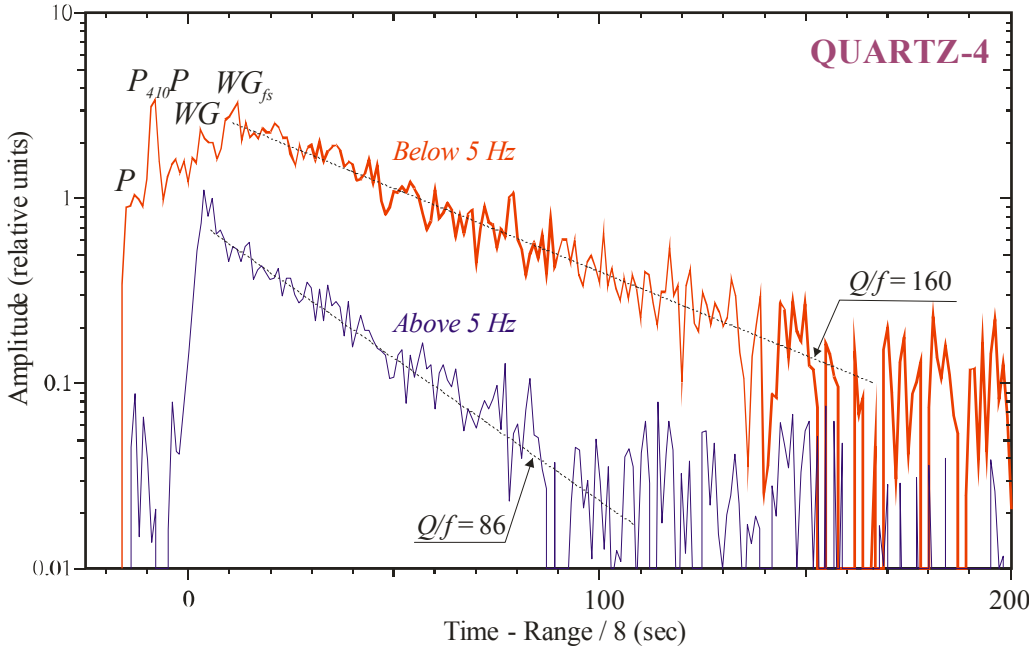


Figure 1.5: Amplitude of records filtered to high and low frequency ranges within the offset range 2500-2600 km from PNE Quartz-4. Seven three-component instantaneous trace-amplitude records were averaged within a 2s sliding time window and within the offset range. First arrivals (*P*), a reflection from the 410-km discontinuity ($P_{410}P$), and two whispering-gallery phases (*WG* and WG_{fs}) forming the teleseismic *P* are indicated. Slope measurement gives Q_{coda}/f value. From Morozov and Smithson 2000.

Q_{coda} = 430 at 5 Hz and Q_{coda} = 320 at 2 Hz. Clearly, coda attenuation depends on the dominant frequency. Though these authors had only two frequency values, they suggested the relation $Q_{coda} \approx 270 f^{0.3}$. These Q_{coda} estimates are consistent with crustal-average Q_s values and therefore support the authors' association of the coda with scattered L_g waves in the crust.

To fully address this crustal scattering problem, full 3-D modeling of crustal features, including faults, mountain ranges, basins, shorelines, and lithology would be required to account for 3-D wave propagation. However, full 3-D modeling is still not practical at this time due to limited computing resources and limited knowledge of scattering properties of the crust. The work I present here is a heuristic simulation of teleseismic *P* and L_g coda amplitude decay modeled as scattering from heterogeneities near the Earth's surface, using 1-D reflectivity synthetics to describe the wavefield and the Born approximation to describe scattering behavior. Since the Rayleigh wave (R_g) decays quickly in near-surface sediments, it is not viewed as a significant factor in coda formation (Dainty 1985); R_g is not seen beyond 200 km in PNE records, so it is not examined here. As this is not a 3-D model of crustal features, it cannot provide a completely realistic picture of resultant coda decay; however, it captures the key mechanism of coda generation and provides improved estimates of crustal attenuation.

2 Coda Model

2.1 Upper-Crustal Scattering Model

The scattering model represents crustal heterogeneities as scatterers distributed within the Earth's crust. The resultant scattered coda energy U recorded at a receiver at time t is an integral over volume V containing all scatterers,

$$U(\vec{r},t) = \int dt_s \iiint_V d^3\vec{r}_s \Psi(\vec{r}_s) U_{source}(\vec{r}_s,t_s) G(\vec{r}_s,t_s;\vec{r},t), \quad (1)$$

where t_s is the time of the direct arrival at the scatterer, $\vec{r}_s$ represents scatterer positions, Ψ is scattering potential, describing the amount of energy reflected at the scatterer, U_{source} is the seismic source function describing energy arriving at the scatterer from the source explosion, and G is the scattering Green's function, describing the propagation of scattered energy from scatterer to receiver. For simplicity, because the upper crust is assumed to be the primary contributor to the seismic coda, this volume integral is replaced with a surface integral (Figure 2.1):

$$U(\vec{r},t) = \int dt_s \iint_S d^2\vec{r}_s \Psi(\vec{r}_s) U_{source}(\vec{r}_s,t_s) G(\vec{r}_s,t_s;\vec{r},t). \quad (2)$$

The Green's function is approximated as translationally invariant in time and space:

$$G(\vec{r}_s,t_s;\vec{r},t) = \hat{G}(\vec{r}-\vec{r}_s,t-t_s). \quad (3)$$

This gives:

$$U(\vec{r},t) = \int dt_s \iint_S d^2\vec{r}_s \Psi(\vec{r}_s) U_{source}(\vec{r}_s,t_s) G(\vec{r}-\vec{r}_s,t-t_s). \quad (4)$$

Therefore, each scattering point acts as a secondary source for seismic waves.

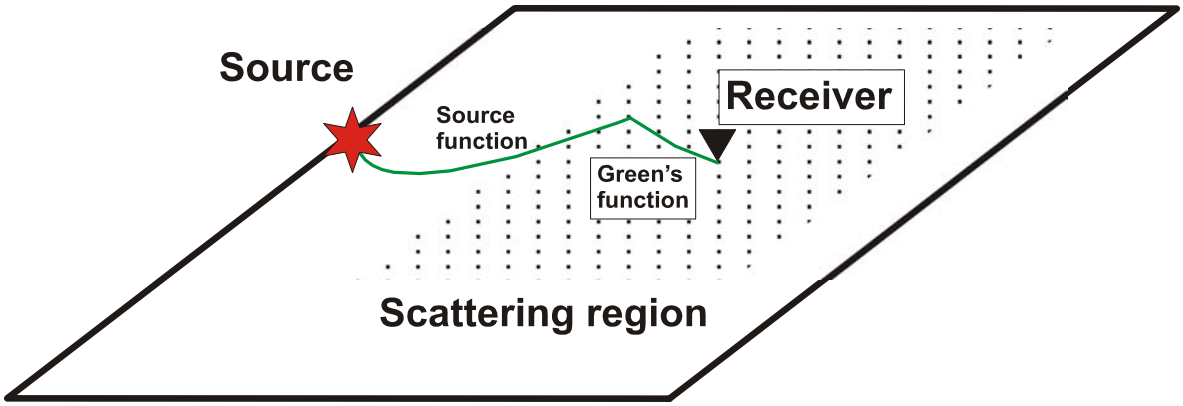


Figure 2.1: Surface scattering model. Seismic energy originates at source, scatters from uniformly-distributed Monte-Carlo-sampled surface points, and is detected at receiver. Source and Green's functions describe propagation of seismic energy.

2.2 Region of Constant Scattering Potential

For a given coda time t and velocity $v = \frac{|\vec{r} - \vec{r}_s|}{|t - t_s|}$ chosen for a particular event (e.g. scattered L_g with velocity v = 3.5 km/s), contributions of scattered energy originate from an elliptical ring surrounding the receiver (Figure 2.2) (Morozov and Smithson 2000). The scattering area within the ring increases with time due to its increasing radius; this increasing area compensates the energy

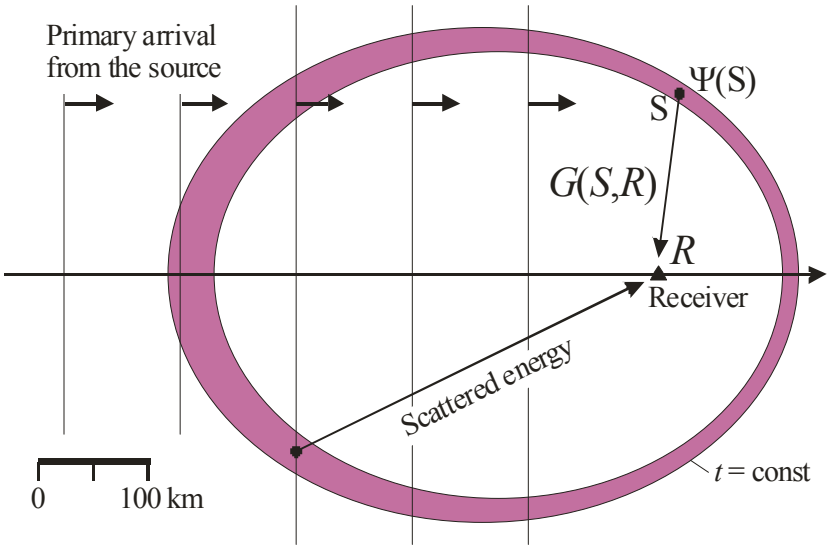


Figure 2.2: Surface area of ring contributes energy to coda of a seismic trace recorded at point *R* near time *t*. *S* is the position of a scatterer near the Earth's surface; *Ψ(S)* is the scattering potential, G(S,R) is the scattering Green's function. Since a dominant contribution to the coda comes from waves trapped within the crustal layers and propagating generally horizontally, their amplitude decay with distance is compensated by the increasing area of the scattering ellipse. From Morozov and Smithson 2000.

decay due to geometrical spreading, since the product of the ring's area and the *1/r* geometric spreading factor rapidly approaches a constant asymptote for increasing values of $|t - t_s|$. Therefore, for a plane of constant scattering potential and no intrinsic attenuation, coda energy would not decay.

Thus, in a region of constant scattering potential, observed coda decay Q_{coda} is indicative of crustal attenuation, denoted by its quality factor Q_{crust}. But in order to relate Q_{coda} and Q_{crust} exactly, relative levels of intrinsic and elastic (scattering) attenuation would need to be known. Therefore, an empirical relationship is sought:

$$\kappa_{coda} \equiv \frac{f}{Q_{coda}} = \kappa Q_{crust}^{\gamma}, \tag{5}$$

where f is the dominant frequency, κ_{coda} is the coda attenuation factor, and κ and γ are constants. A power law relation is used because it is general, asymptotes are well-behaved, and it has been commonly used when determining Q as a function of f in past studies.

Morozov and Smithson (2000) assumed κ values of 2 and 5 Hz and γ =-1 in Equation 5 when performing their coda measurements. In the work presented here, I perform numerical modeling to improve estimates of these parameters by using a range of Q_{crust} values in the reflectivity synthetics and filtering the synthetics into different frequency bands to select f.

2.3 Reflection Synthetic Seismic Sections

The source functions and scattering Green's functions used as input to the model were synthetic seismic sections created using the reflectivity method (Appendix A) (Fuchs and Müller 1971). Crustal and mantle layers are specified by their seismic *P* and *S* wave velocities, densities, and intrinsic Q_p and Q_s values, with discontinuities in these properties specified across chosen layers. I used four different velocity models to compute synthetic sections. The PNE Quartz-4 velocity model by Morozova *et. al.* (1999) (Figure 2.3a) contains a 3-layer crust overlain by a 3-km-thick sediment layer. Its mantle is complex, with low velocity zones at 110 and 210 km depths. The

IASP-91 model (Figure 2.3b) (Kennett 1991) contains a much simpler two-layer crust and a mantle without low-velocity zones. The IASP-Quartz model contains the IASP-91 crust and Morozova *et. al.* (1999) mantle (Figure 2.3c). The complex crust model has a 5-layer crust and the Morozova *et. al.* (1999) mantle (Figure 2.3d). Note that the Quartz-4 model and the complex crust model exhibit more complex waveforms (including crustal L_g phases in addition to the *P* and *S* phases) than the

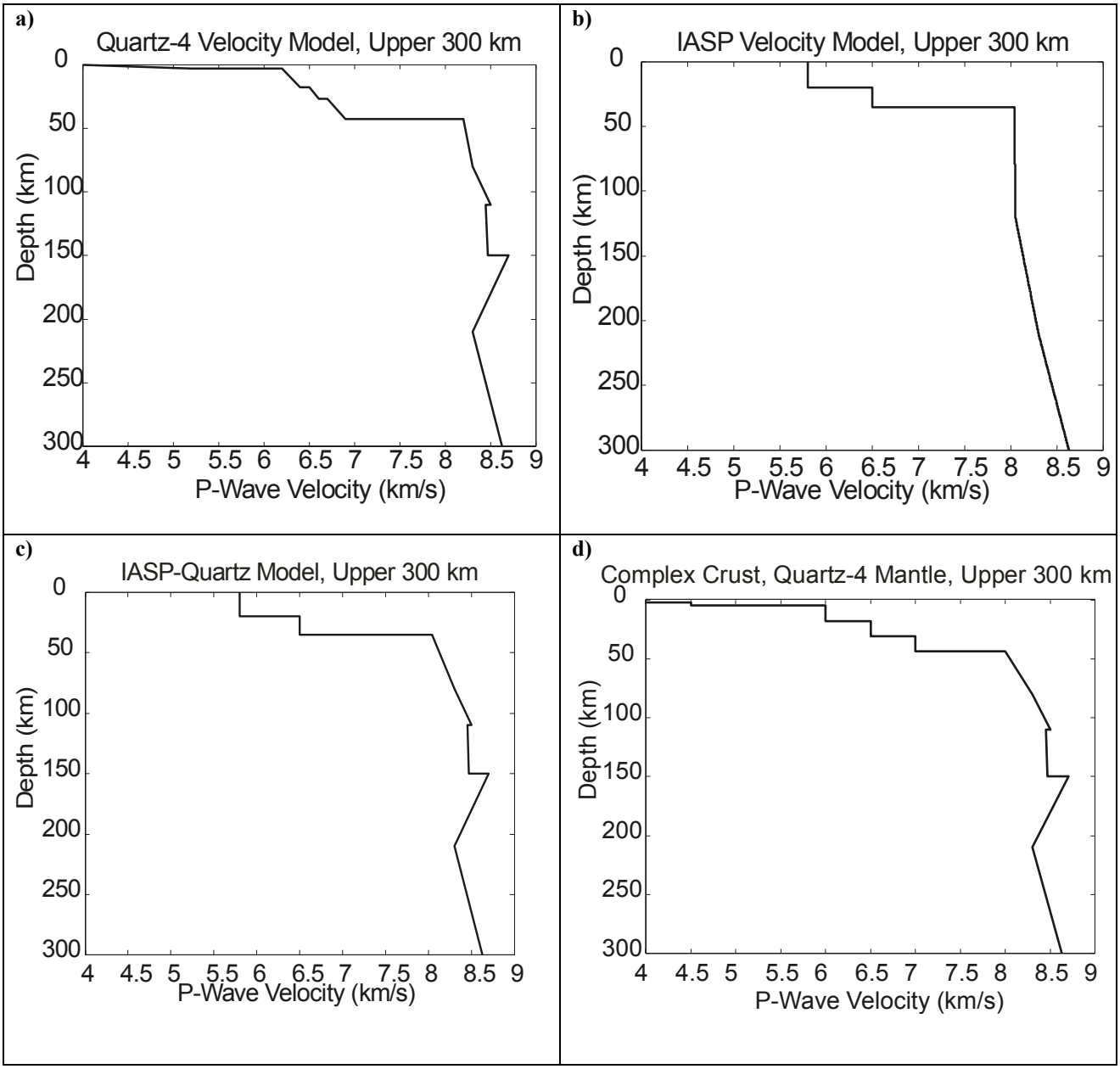


Figure 2.3: Upper 300 km of 1-D velocity models used in creation of reflection synthetic seismograms. a) PNE Quartz-4 model by Morozova *et. al.* (1999). b) IASP-91 model, with simpler crust and mantle than Quartz-4 model. c) IASP-Quartz model, with IASP-91 crust and Quartz-4 mantle. d) Complex 5-layer crust with Quartz-4 mantle.

IASP-91 and IASP-Quartz models with simpler crusts (Figure 2.4). These velocity models with different crustal and mantle complexities allow determination of which features have the greatest affects on amplitude and velocity of seismic events and therefore affect coda strength.

Q_p and Q_s values, remaining in the ratio 2:1, were specified for layers in the velocity model. Q_{crust} values of the reflectivity synthetics refer to the Q_s values specified for them. Q_{crust} values range from 50 to 1000 to encompass the common range of crustal Q values. Mantle attenuation in all four velocity models was specified with mantle P-wave intrinsic Q values obtained by Morozov *et. al.* (1998b) in their study of Quartz PNE data. For the Quartz-4 and complex crust models,

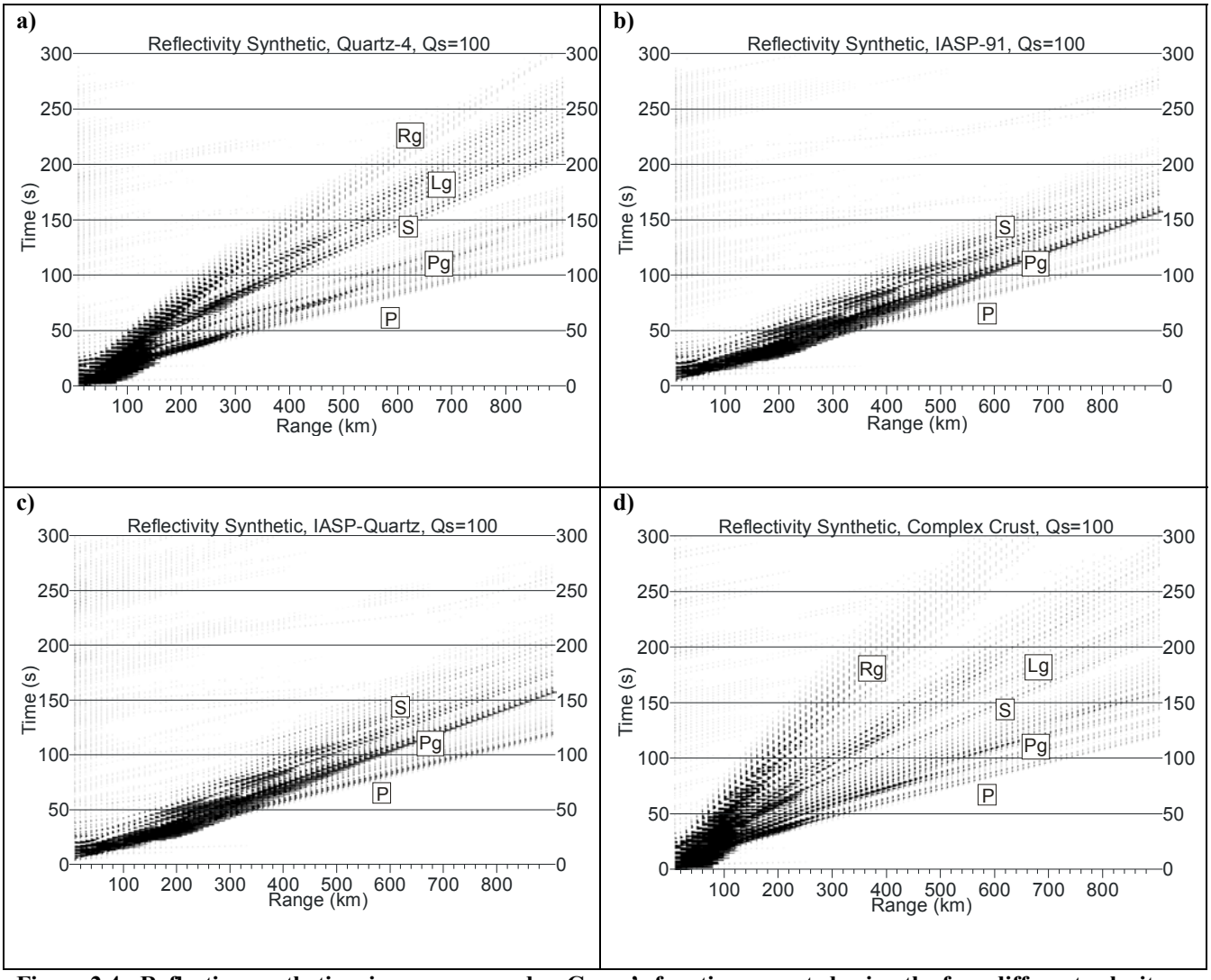


Figure 2.4: Reflection synthetic seismograms used as Green's functions, created using the four different velocity models a) Quartz-4 model. b) IASP-91 model. c) IASP-Quartz model; note similarity to *b.* d) Complex crust model; note similarity to *a*.

Q_p and Q_s values of the upper 3 km were set at 10 to attenuate R_g, because it is not seen beyond about 200 km in PNE data and is thought to be an insignificant factor in teleseismic P and L_g coda formation (Dainty 1985). A mute was applied to completely eliminate the R_g phase from the synthetics.

The relative amplitudes and energies of P and L_g arrivals are important in interpreting the behavior of seismic codas in this modeling. A trace envelope from the Quartz-4 model at 500 km offset (Figure 2.5a) shows that the L_g amplitude is greater than that of P and contains more energy

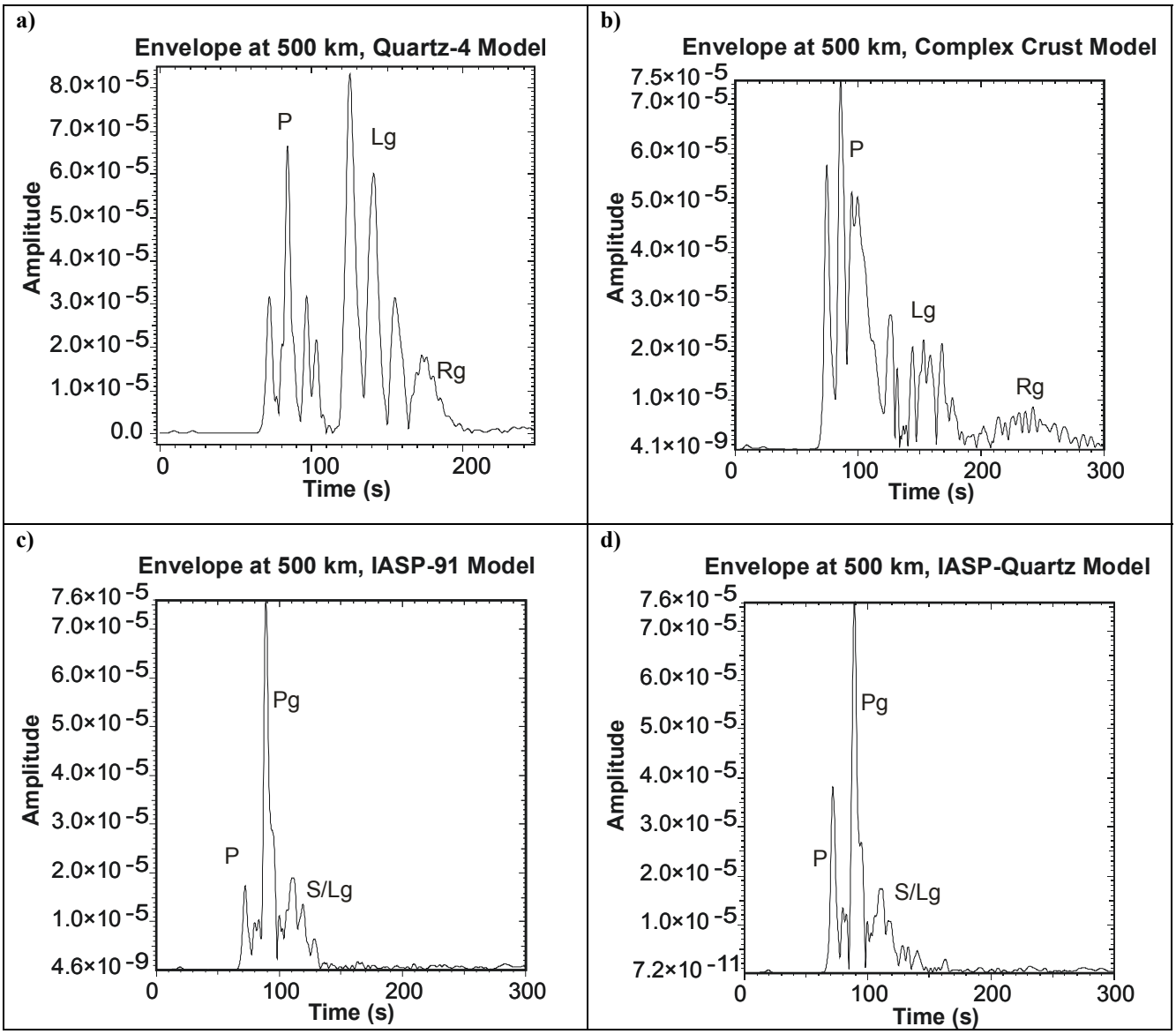


Figure 2.5: Seismic trace envelopes at 500 km offset, Q_{crust} = 100, 0.3 Hz. a) Quartz-4 model. b) Complex crust model. c) IASP-91 model. d) IASP-Quartz model.

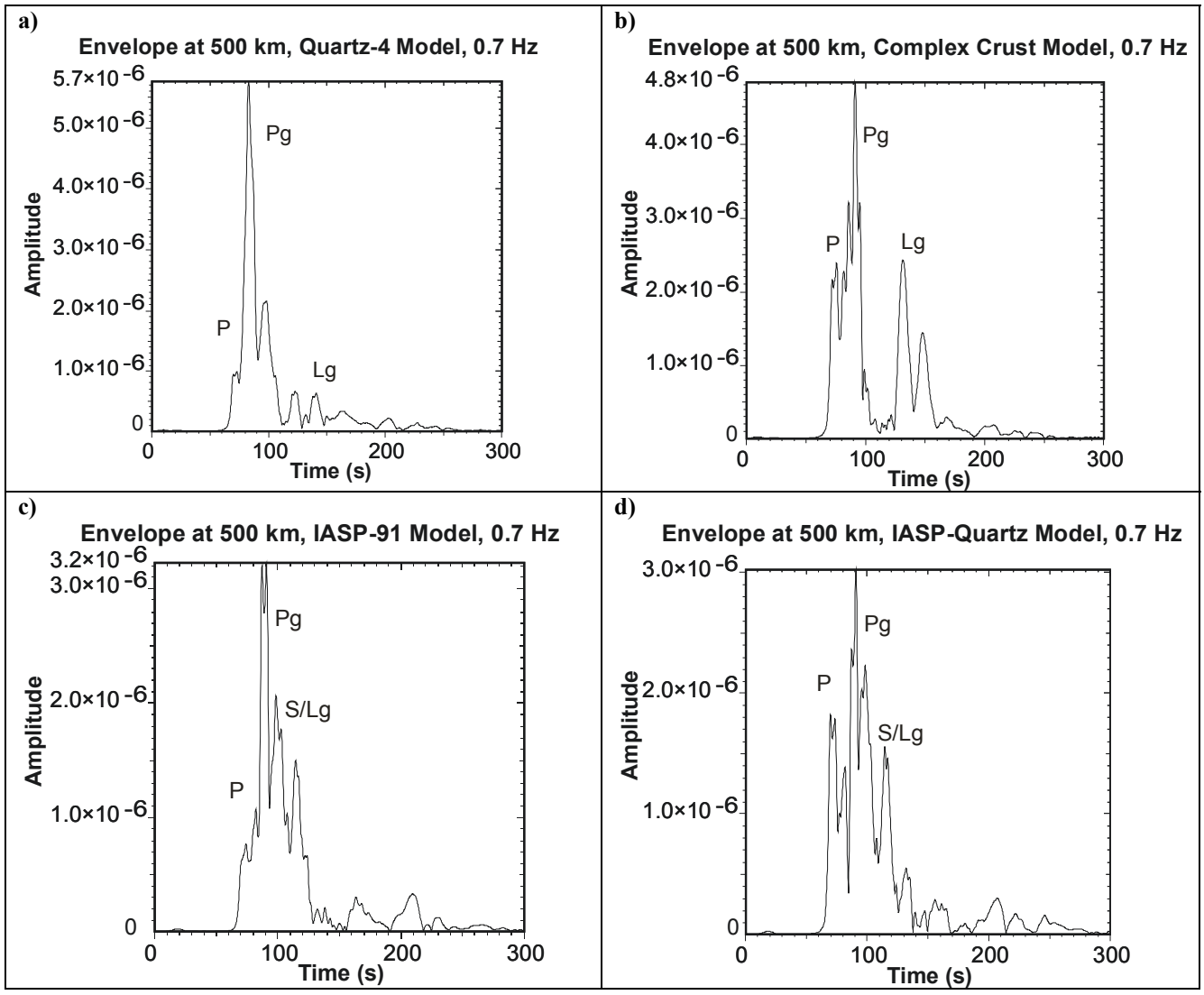


Figure 2.6: Seismic trace envelopes at 500 km offset, Q_{crust}=100, 0.7 Hz. a) Quartz-4 model. b) Complex crust model. c) IASP-91 model. d) IASP-Quartz model.

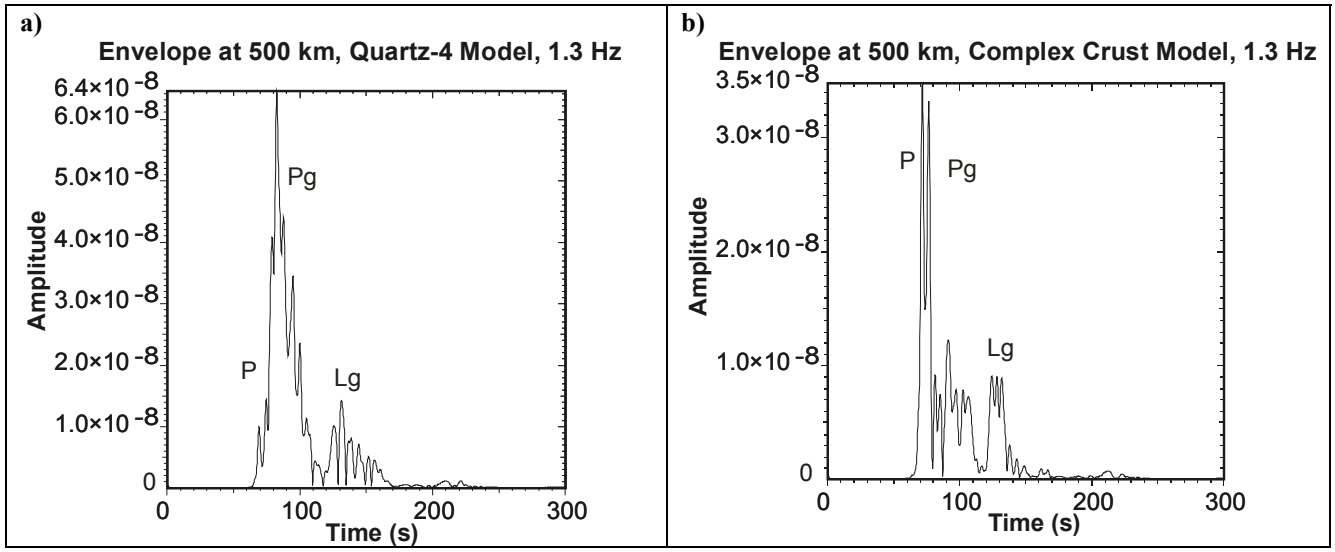


Figure 2.7: Seismic trace envelopes at 500 km offset, Q_{crust}=100, 1.3 Hz. a) Quartz-4 model. b) Complex crust model.

because of the greater amount of inscribed area beneath the L_g curve.

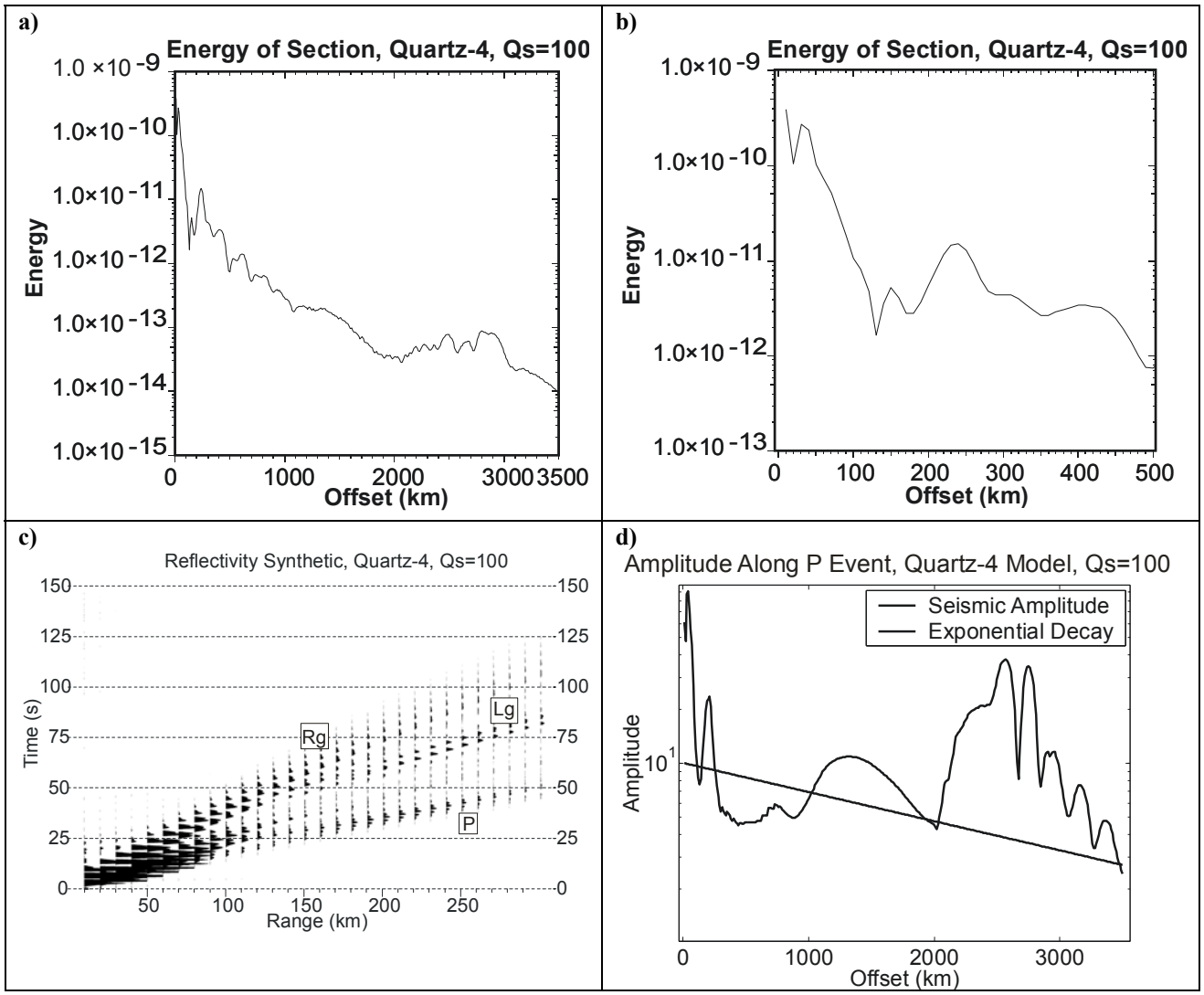
Envelopes at 0.7 Hz and 1.3 Hz appear in Figure 2.6 and Figure 2.7, respectively. For the Quartz-4 model at 0.3 Hz, the L_g arrivals appear stronger than the *P* arrivals. At 1.3 Hz and especially at 0.7 Hz, however, the *P* arrivals are stronger. For the complex crust model, the L_g strength relative to *P* appears greater at 0.7 Hz than at 0.3 Hz and only slightly weaker at 1.3 Hz. For the IASP-91 and IASP-Quartz models, the S/L_g arrivals are weak relative to *P* arrivals at 0.3 Hz but are more significant at 0.7 Hz.

2.4 Amplitude and Energy Decay Measurements

Reflection synthetic seismograms were examined for expected amplitude and energy decay. The energy of the entire section decays due to geometrical spreading (Figure 2.8a). A closer view of the offset range 0-300 km shows low amplitude between 100 and 200 km (Figure 2.8b); this is due to the small amplitude of P_n from 100 to 175 km and the arrival of an L_g event beyond 200 km (Figure 2.8c). Similar low amplitudes occur at small offsets for the other velocity models as well (Appendix B).

Body wave amplitude is expected to decay as the inverse of *offset*. To test this, a constant-width time interval through an event was selected from the time vs. offset plot of a seismic section. Amplitude was corrected for spreading by multiplying amplitude by the factor *offset*, then plotted vs. *offset* (Figure 2.8d-g). After this correction, no decay in amplitude with increasing offset is expected for infinite Q_{crust} value (no intrinsic crustal attenuation); greater decay is expected for lower Q_{crust} values, denoted by the superimposed $exp(-ft/Q_{crust})$ lines. Oscillations in the amplitude as a function of offset are not a concern, as they signify individual arrivals within the event package. The *P* amplitude at $Q_{crust} = 100$ appears to decay slightly less than expected (Figure 2.8d); for $Q_{crust} = 1000$, the decay appears as expected (Figure 2.8e). The increase in amplitude at 2500 km is due to a triplication at this offset between the teleseismic *P* and a *P* phase traveling deeper within the mantle.

Surface waves are expected to follow cylindrical spreading rules. The L_g event, corrected by the factor $\sqrt{offset}$ for cylindrical spreading, decays more quickly than expected at both Q_{crust} = 100 and Q_{crust} = 1000. No corrections were made, however, since the scattering Green's function, described by offsets less than 600 km, is not adversely affected at small offsets.



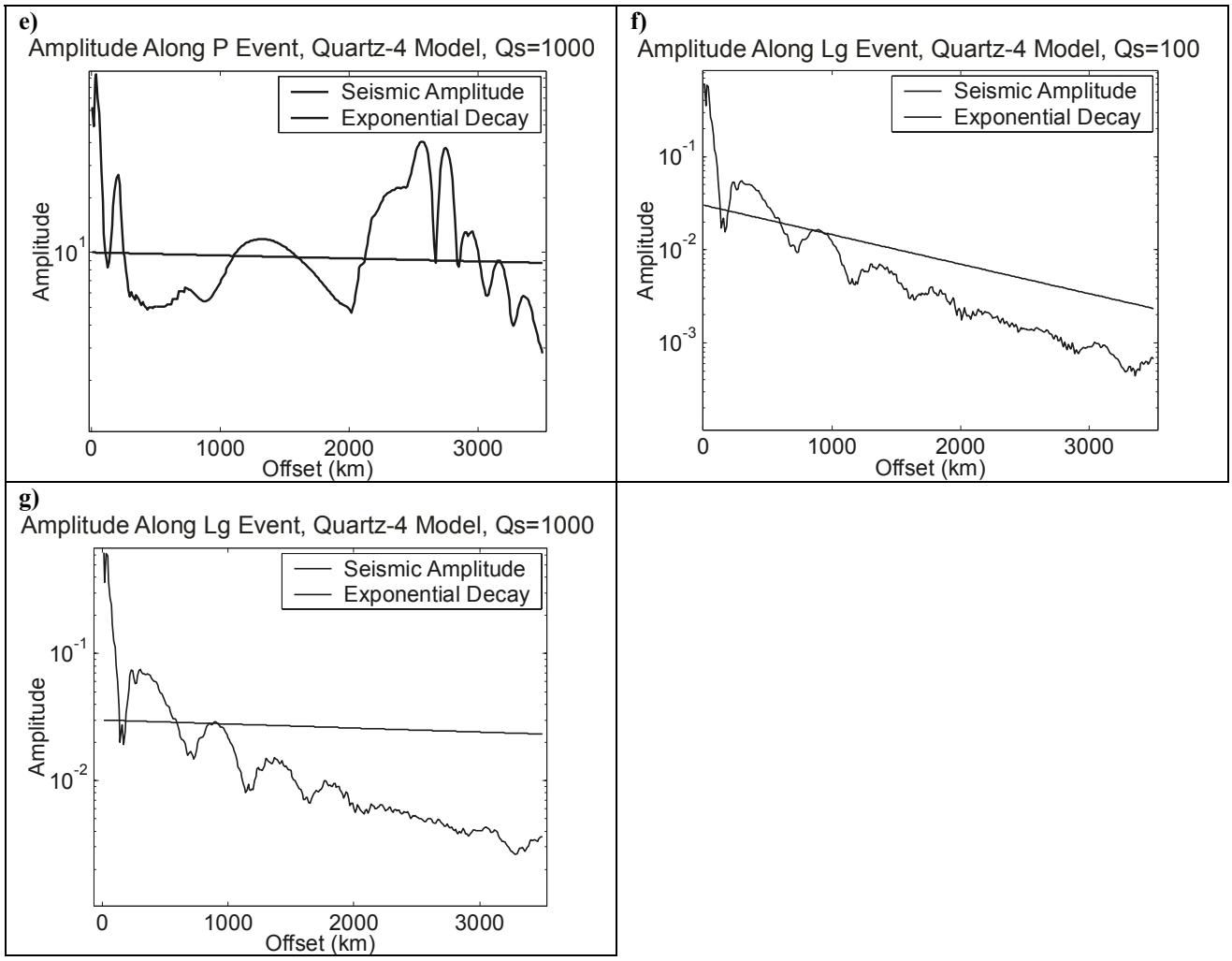


Figure 2.8: Energy and amplitude measurements of Quartz-4 reflectivity synthetics. a) Energy of entire trace plotted as a function of the trace's offset, Q_{crust}=100. Energy decays with offset due to geometrical spreading. b) Zoom highlighting decrease of energy at 100-200 km offset. c) Arrivals in reflectivity synthetic show small *P* amplitude between 100 km and 175 km offset and L_g event arrival at 200 km offset at 60s, causing low energy between 100 km and 200 km, increasing past 200 km offset.

d-g) Amplitude measured along a thin, 10-second strip for single seismic events. Amplitudes are corrected for spreading; thus for infinite Q_{crust}, amplitude would be horizontal. The exp(-$\pi ft/Q_{crust}$) lines, showing the expected slope due to crustal attenuation, are shown for f = 0.3 Hz and Q_{crust} = 100 and 1000. d) *P* event, corrected for spreading by $offset$, Q_{crust}=100. e) *P* event, Q_{crust}=1000. f) L_g, corrected for cylindrical spreading by $\sqrt{offset}$, Q_{crust}=100. g) L_g, Q_{crust}=1000.

3 Numerical Modeling

3.1 Numerical Convolution Routine

The integral over time to compute the scattering model's seismic response at a receiver (Eq. 4) is performed using numerical convolution of the source function, describing the propagation of the incident wavefield of seismic energy to surface scatterers from the source explosion, and the scattering Green's function, describing the propagation from the scatterers to the receiver (Figure 2.1). I coded the routine in C++ within the framework of Dr. Igor Morozov's SIA system (Appendices C, D). The 2-D spatial integral is performed using uniformly-distributed Monte-Carlo sampling of surface points, which simulates constant scattering potential for the entire infinite plane. This geometry is important because using a plane of constant scattering potential should allow the quantitative dependence of Q_{coda} on Q_{crust} to be determined. To limit computing resource requirements, the region's size was limited in spatial extent but chosen to ensure that scattered energy from the region's edges would arrive after a reasonable time interval for measurement of coda slopes. Coda energy arrives from an elliptical ring, increasing in area with time, surrounding the receiver (Figure 2.2) (Morozov and Smithson 2000). While this ellipse remains within the scattering region, the decay of seismic energy due to geometrical spreading will be compensated by the increasing area of the scattering ring. Thus, the coda is expected to ring with attenuation governed by the Q_{crust} value until the ellipse begins exiting the scattering region. For measuring the teleseismic *P* coda, the receiver was placed at 2900 km offset from the source to allow a significant time interval between onset of *P* and L_g phases. Since only the *P* coda was measured at this offset, a time interval beginning at the teleseismic *P* arrival sufficient to measure the *P* coda was examined. Since L_g waves originating from the source (or L_g waves scattered near the source) travel too slowly to arrive within this time interval, the scattering region was assigned around the receiver only. The

region size was chosen as a square 1200 km on a side, allowing 600 km from the central receiver to the nearest edges (Figure 3.1a). This allows a 3 km/s L_g wave 200 seconds before scattered energy from the edge of the region arrives at the receiver, thus giving a 200-second window in which to

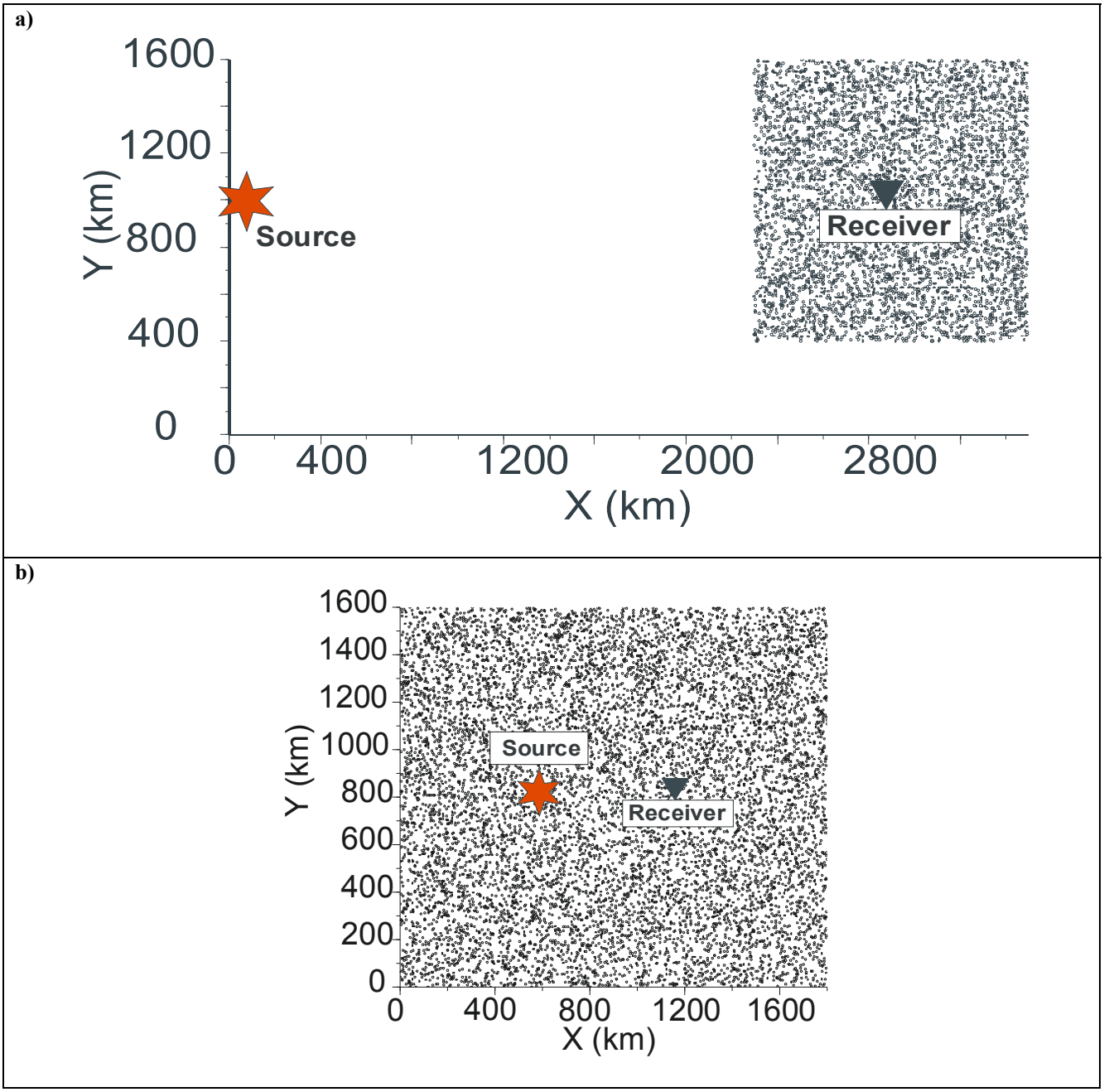


Figure 3.1: Scattering geometries. Scatterer locations within region are distributed uniformly, sampled in Monte-Carlo fashion. a) Geometry for teleseismic *P* coda, with long offset to achieve significant separation between teleseismic *P* and L_g events. Since the teleseismic *P* coda is being examined, the concern is with $P \rightarrow P$ and $P \rightarrow L_g$ scattering; thus the scattering region is required only around the receiver as L_g energy from the source will arrive after the measurement interval. b) Geometry for L_g coda, to include propagation of both teleseismic *P* and L_g source waves and all scattered waves.

reliably measure coda decay. Trials were also performed at 2200 km offset, but this did not allow a reliable measurement of *P*-wave coda before the arrival of the L_g coda since the coda amplitude builds slowly due to low amplitudes at small offsets in the synthetics (Figure 2.8b). As the Q_{crust} value in the reflectivity synthetics is constant throughout the section, and crustal scattering does not depend on the source-receiver offset, the receiver's offset value should not influence Q_{coda} values measured from the output traces.

For measuring the L_g coda, both source and receiver were located within the scattering region (Figure 3.1b) to include propagation of L_g waves both originating at the source and scattered, as both contribute to L_g coda.

3.2 τ-p Interpolation

To obtain seismic traces at appropriate source-scatterer and scatterer-receiver offset distances for use in the numerical convolution routine, I coded a τ-p interpolation routine (Yilmaz 1987) for seismic traces based on a seismic section (Appendix E). This interpolation code was first implemented and tested as a separate tool in Dr. Igor Morozov's SIA system before its use in the scattering routine. The code resamples the offset coordinate of an input seismic section to the user's specifications based on several parameters. The offset range, trace spacing, and interpolation width (which determines the number of traces used in the interpolation) are the offset parameters. The user may also choose the frequency range and slowness range in which to resample the section. For my work, the synthetic source and Green's functions used as input to the numerical integration routine were created with trace spacing of 10 km in order to conserve computer processing time and storage space.

Teleseismic *P* and L_g coda segments were chosen based on arrival times of direct waves from the source for these events. Single traces at 2900 km, the source-receiver offset distance for the

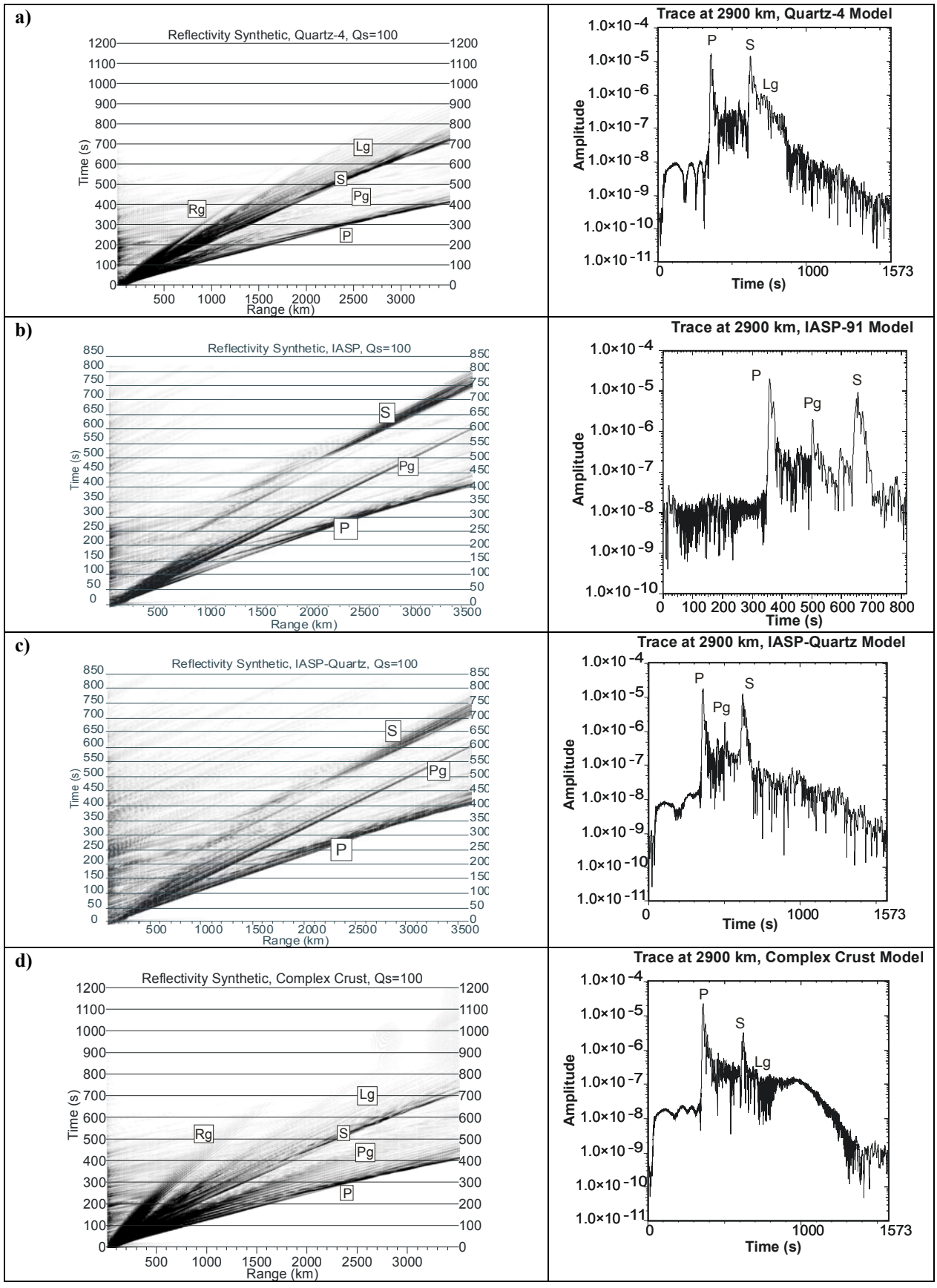


Figure 3.2: Reflectivity synthetics of the four models and trace envelopes at offset 2900 km representing the direct wave arrival. a) Quartz-4 model. b) IASP-91 model. c) IASP-Quartz model. d) Complex crust model.

teleseismic *P* coda, show onset times of the direct arrivals (Figure 3.2); coda segments follow these arrivals. The teleseismic *P* coda segment for the Quartz-4 model occurs at 350-600s. The IASP-91 model and IASP-Quartz model have a strong P_g phase, limiting the teleseismic *P* measurement interval to 350-500s.

3.3 Scaling of Scattered and Direct Arrivals

Realizations of the numerical integration routine each produced a single output trace

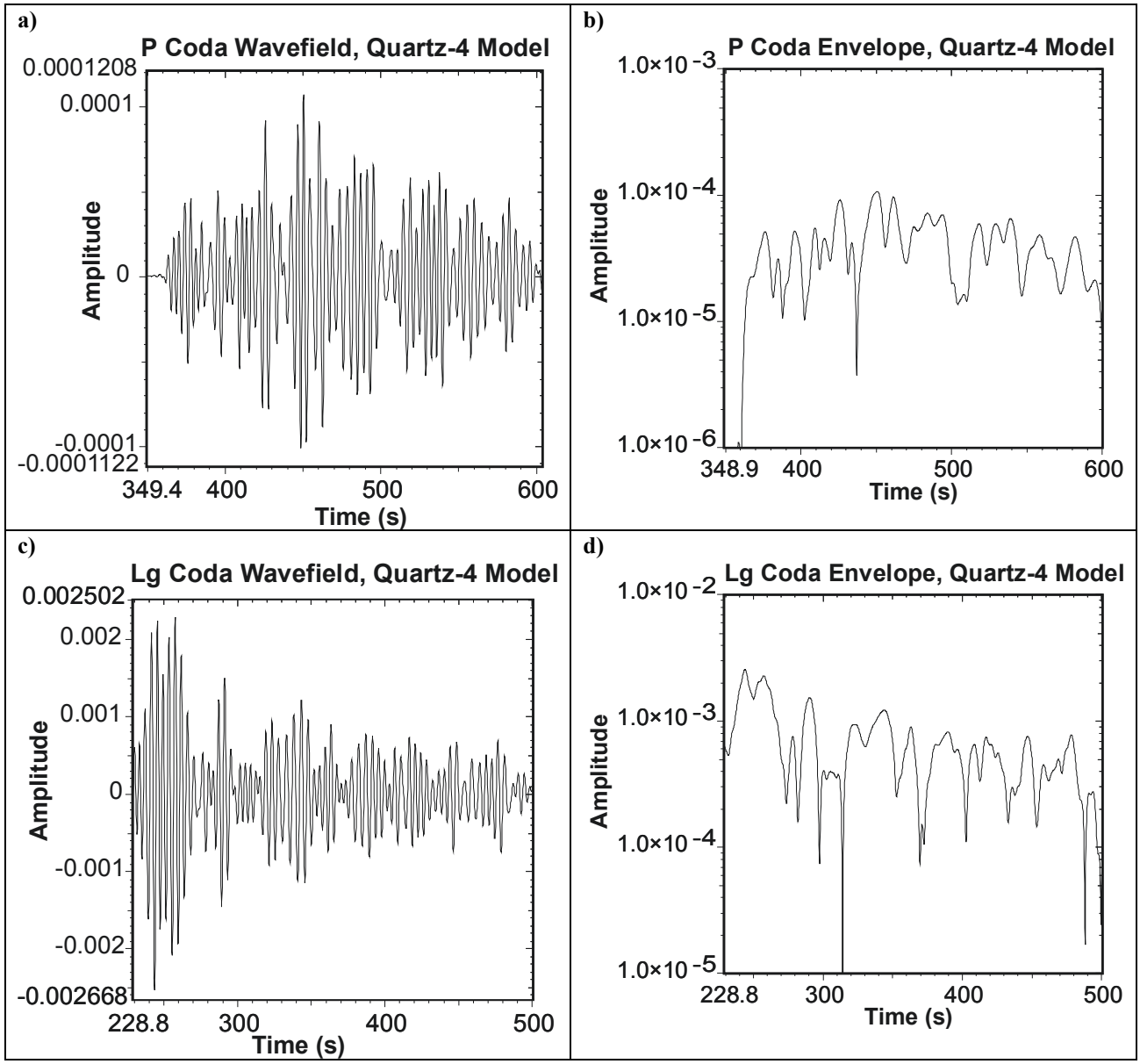


Figure 3.3: a) Teleseismic *P* coda wavefield output from scattering model, showing only the scattered response, Quartz-4 model. b) Semi-log plot of teleseismic *P* trace envelope; note coda slope amplitude appears linear after 450s, when coda has finished building. c) L_g coda wavefield. d) L_g coda envelope, semi-log scale.

(Figure 3.3). These scattered traces were scaled relative to the direct arrival from the source (Figure 3.2b) and summed with the direct arrival before coda measurement. Morozov and Smithson (2000) modeled coda power as a function of time by describing the primary event as a parabolic function with peak power P^o, duration τ, and relative coda amplitude parameter λ (Figure 3.4a), with the product $\lambda\tau$ giving the power relative to peak. Values determined were $\lambda = 0.22$ for all events and $\tau = 1.25$ for the teleseismic *P* event. Applying this to data in this study gives $\lambda\tau = .275$ for the relative power; taking the square root to apply this to amplitude gives a scattered amplitude to direct amplitude ratio of 0.524, so the scattered trace should have approximately half the amplitude of the direct wave at the time of coda onset. For the L_g event, $\tau = 2.5$ was used to account for the greater width of the arrival, implying a relative amplitude for L_g events of about 0.75. The scaling factor was found to differ between teleseismic *P* and L_g codas. Figure 3.4b shows the coda scaled appropriately for L_g at about 0.75 times the amplitude of the direct arrival; the teleseismic *P* coda appears to be less than half the amplitude of the peak of the *P* phases.

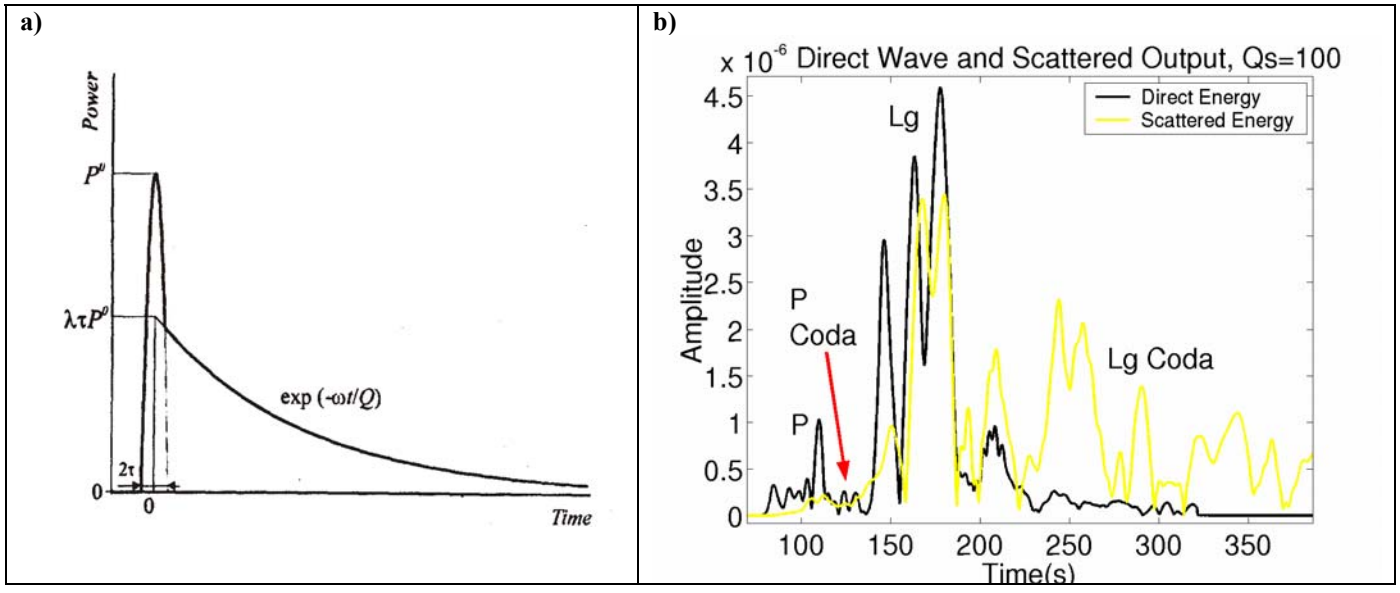


Figure 3.4: a) Morozov and Smithson (2000) model of coda amplitude relative to primary event. Relative coda amplitude at onset is given by $\lambda\tau$, where λ is the relative coda amplitude parameter and τ is the duration of the primary event. b) Scaling of L_g coda to 75% of the amplitude of L_g arrival. *P* coda appears less than half of the *P* arrival, implying scattering potential per unit area is different.

3.4 Scattering Potential

The relative amplitude was examined for a scattered trace and direct wave trace at each Q_{crust} value for all models to determine the necessary scaling factor for the *P* and L_g events (Figure 3.4b). From this scaling factor, the scattering potential Ψ per unit area is computed for *P* and L_g. Scattering potential per unit area is an effective measure of scattering efficiency in terms of the resulting amplitude; it is inversely proportional to the scaling factors computed and directly proportional to the scattering density. Scattering potential per unit area for the Quartz-4 model and the complex crust model are shown as a function of Q_{crust} (Figure 3.5). Note that the L_g coda has a scattering potential per unit area of about 1.25 times higher than that of the teleseismic *P* coda (Figure 3.5c). Because the teleseismic *P* coda is governed by $P \rightarrow P$ and $P \rightarrow L_g$ scattering and the L_g coda is also affected by $L_g \rightarrow P$ and $L_g \rightarrow L_g$ scattering, these extra scattering modes likely contribute to the greater scattering potential per unit area of L_g. For the complex crust model, the scattering potential per unit area was found to be about 4.25 times greater for L_g than for *P*. Again, the additional $L_g \rightarrow P$ and $L_g \rightarrow L_g$ scattering modes likely contribute to the greater scattering potential per unit area for L_g.

After determining their relative amplitudes, the direct and scattered wavefields were added together. Approximately 15 of these realizations were computed and compiled into a data set to improve stability of the measurements. This was required because the surface was uniformly sampled in a Monte-Carlo fashion to avoid unnatural coherency caused by equal scatterer-receiver distances for several scattering point locations that might result if the surface were sampled as a grid. These data sets were compiled for each of the Q_{crust} values, ranging from 50 to 1000 in steps of 50. Slope measurements were then performed on the trace envelopes in the manner of Morozov and Smithson (2000) (Figure 1.5) to obtain a Q_{coda}/f value for each Q_{crust} set. To obtain uncertainty estimates for each value, half of the traces from a set were chosen randomly and measurements were

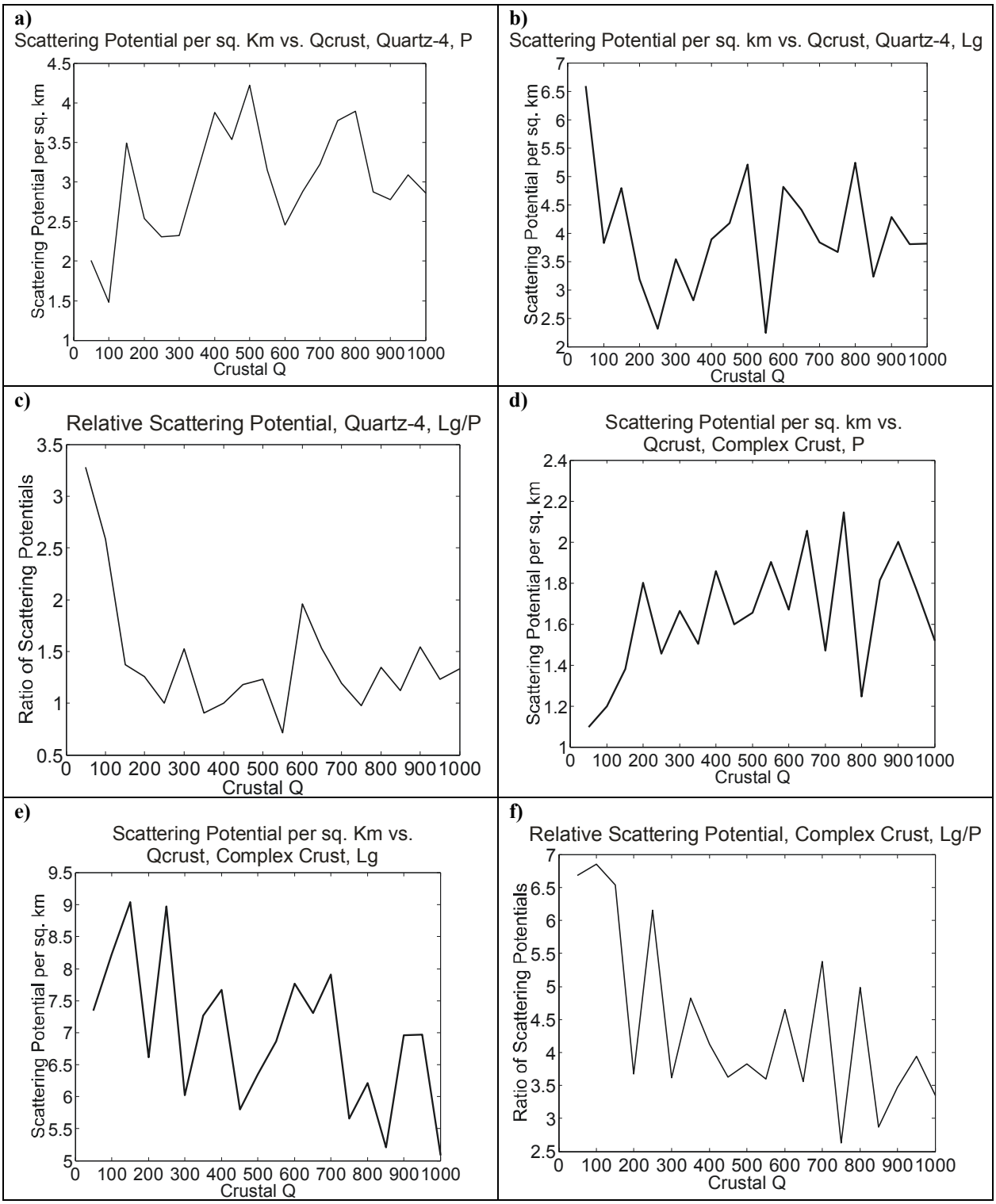


Figure 3.5: Determining scattering potential per unit area for Quartz-4 and complex crust velocity models. a) Scattering potential determined for *P*, Quartz-4 model. b) Scattering potential determined for L_g. c) Relative values of scattering potential for L_g and *P*. The L_g scattering potential is about 1.25 times higher than that of *P*. d) Scattering potential determined for *P*, complex crust model. b) Scattering potential determined for L_g. c) Relative values of scattering potential for and L_g and *P*. The L_g scattering potential is about 4.25 times higher than that of the *P*.

performed to obtain a Q_{coda}/f value; this was repeated several times with different randomly-chosen traces to obtain a range of values. The greatest and least of these Q_{coda}/f values for each Q_{crust} were chosen as error bounds.

3.5 L1 Regression Line Algorithm

I wrote Matlab code to fit the teleseismic *P* and L_g coda segments with both least-squares and L1 regression lines (Appendix F). The goal of measuring the teleseismic *P* and L_g coda slopes is best accomplished by fitting the general trend of these slopes and lightly weighting short, low-amplitude drops that occur intermittently within the coda waveform (Figure 3.3b). L1 fitting is preferred, therefore, because it iteratively downweights outliers.

I used the L1 matrix fitting algorithm described in Aster *et. al.* (2002). The L1 routine first uses the least-squares solution to the matrix equation $G_w m_0 = d_w$, shown here. The x_1-x_n values (Equation 6) are the independent variables, in this context the input Q_{crust} values, and the d_1-d_n values are the dependent variables, here the Q_{coda}/f values measured from the slopes of the scattered trace. Solving

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \dots & \dots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \dots \\ d_n \end{bmatrix} \tag{6}$$

gives $\mathrm{m} = \begin{bmatrix} m_1 \\ m_2 \end{bmatrix}$, the least-squares intercept and slope.

To iteratively reweight the data points so that outliers become less important, the L1 routine first calculates residual vector r_i:

$$r_i = \left| \left(d_w - G_w m_0 \right)_{(i)} \right| .$$

Next, diagonal matrix *R* is formed from the residual vector and used to reweight the solution:

$$R = diag(r)^{-1}.$$

The following matrix equation is then solved for m_{L1} to give a reweighted solution:

$$G_w^T R G_w \, m_{L1} = G_w^T R d_w .$$

Based on this new m_{L1} solution, r_i is recalculated and the algorithm is performed again. This iteration continues until the successive change in residuals becomes less than some small chosen value ε:

$$\frac{\left\| r - r_{previous} \right\|_1}{\left\| r_{previous} \right\|_1} < \varepsilon .$$

The goal of these measurements is to obtain values of κ and γ in Equation 5:

$$\kappa_{coda} \equiv \frac{f}{Q_{coda}} = \kappa Q_{crust}^{\gamma},$$

where Q_{coda}/f is the quantity measured from coda slopes and Q_{crust} values are specified in the reflectivity synthetics. Thus κ and γ may be obtained by taking the natural logarithm of this equation,

$$\log\left(\frac{f}{Q_{coda}} \right) = \log\left(\kappa Q_{crust}^{\gamma} \right) = \log \kappa + \gamma \log Q_{crust}$$

and plotting on a log-log scale to give a straight line with slope γ and intercept $\log \kappa$. These γ and κ values are used to compare the Q_{coda}-Q_{crust} relation obtained for the different velocity models.

4 Results

4.1 Near-Infinite Q_{crust} Value Behavior

For an infinite Q_{crust} value, a receiver trace inside the scattering region should experience no crustal attenuation. As expected, using a high Q_{crust} value of 10000 exhibits a trace envelope with negligible amplitude decay (Figure 4.1). This coda ringing is in accordance with Greenfield's (1971) suggestion that observed codas from underground nuclear explosions at Novaya Zemlya would persist with constant energy in the case of no crustal attenuation.

At lower Q_{crust} values, teleseismic P and L_g event codas become apparent, and their Q_{coda}/f values may be measured from the traces output by the scattering model.

I performed measurements on teleseismic P and L_g codas for each of the four velocity models. Traces were filtered for different f values, centered at 0.3, 0.7, and 1.3 Hz, to examine the coda as a function of frequency.

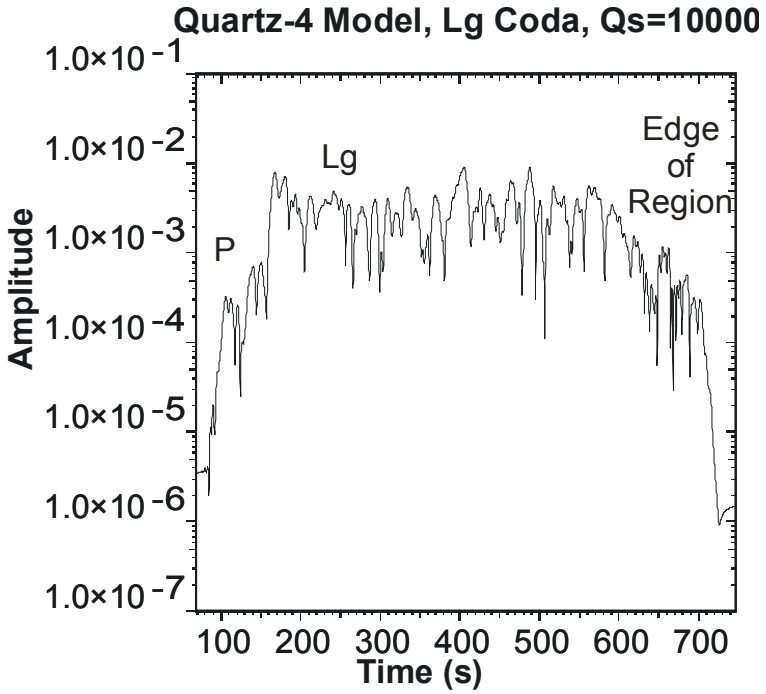


Figure 4.1: Scattered trace envelope output only (no direct response) from scattering model for receiver within scattering region, Quartz-4 velocity model, f=0.3 Hz, receiver located at 600 km offset from source. Note that amplitude does not decay with time because of the high Q_{crust} value of 10000 used.

4.2 Quartz-4 Model, Teleseismic P Coda, 0.3 Hz

After appropriate scaling of the teleseismic *P* coda with respect to the direct wave, the data sets of numerically modeled traces for the Quartz-4 velocity model at $f = 0.3$ Hz are shown for Q_{crust} values of 50 and 1000 (Figure 4.2). A delay in onset of coda amplitude is evident between the direct teleseismic *P* arrival and the strong part of the coda (Figure 4.2a,b) due to the low energy of arrivals at small offset in the Green's functions (Figure 2.8b). Just after the onset of the direct arrival, the scattering ellipse has small area; therefore, the energy of the Green's function synthetic section would need to approach infinity at zero offset to compensate for spreading at small coda times. Since the Green's function is of lower energy between 100 and 200 km (Figure 2.8b), the coda builds up only after the radius of the scattering ellipse expands beyond this distance.

Measurement of coda envelope slopes for the range of Q_{crust} values gives a γ value of -0.47 and a κ value of 0.019 for the teleseismic *P* coda (Figure 4.2c); thus the parameterization is $Q_{coda} = 16 \cdot Q_{crust}^{0.47}$. These values will be compared with those of the other models. Dotted lines E1 and E2 were drawn and γ and κ values computed to give an estimate of stability of the values as a function of slope; values appear on the plot.

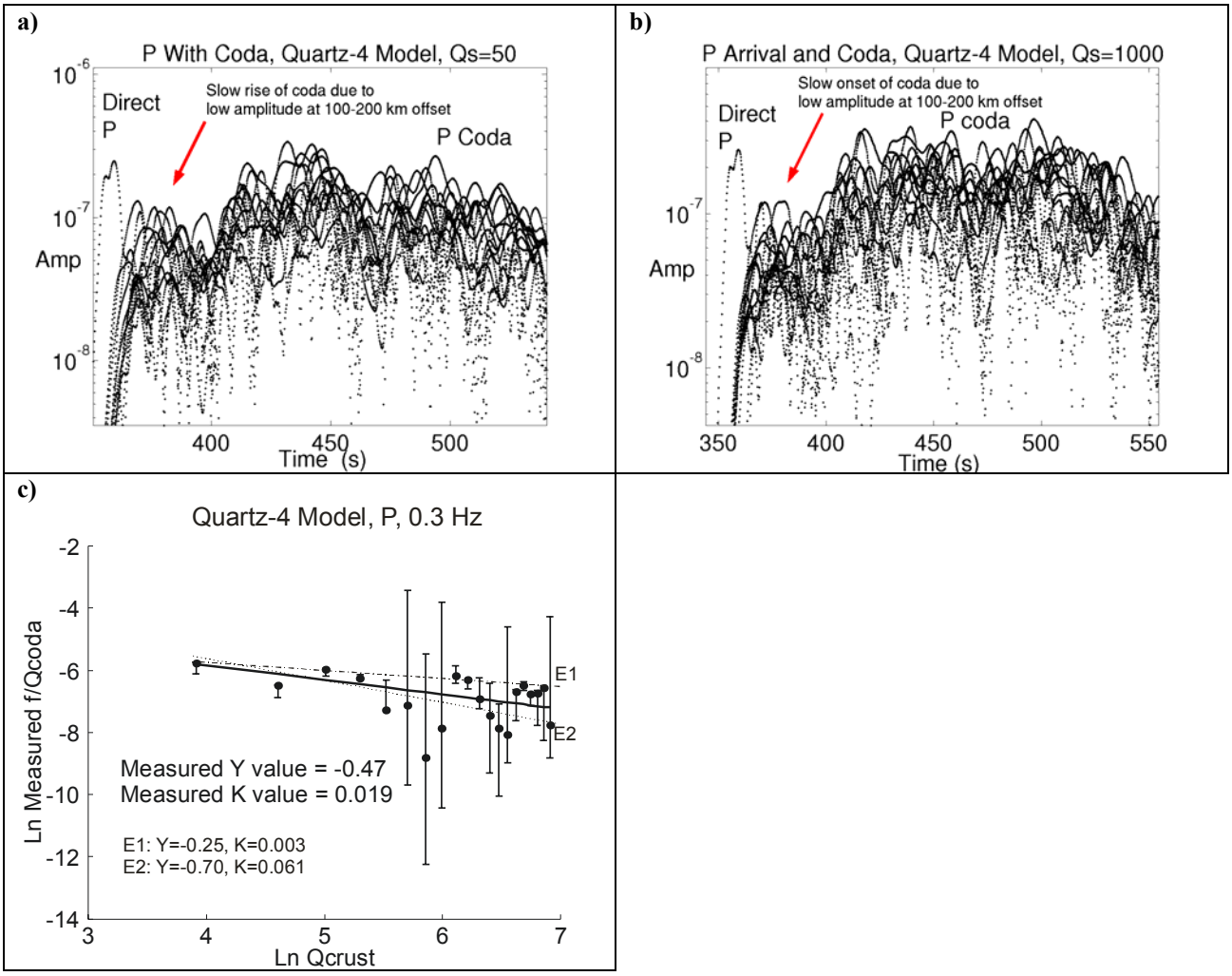


Figure 4.2: Quartz-4 model results at *f*=0.3 Hz. a) Teleseismic *P* direct arrival with coda for Q_{crust}=50. Note the slow onset of coda energy after the direct arrival. b) Teleseismic *P* arrival with coda for Q_{crust}=1000. c) Slope fitting of log(f/Q_{coda}) vs. log $(Qcrust)$ to determine parameters of Q_{coda}-Q_{crust} relation. Lines E1 and E2 give estimates of the variability of the parameters based on small changes in slope.

4.3 IASP-91 Model, Teleseismic P Coda, 0.3 Hz

The IASP-91 model amplitudes were scaled for teleseismic *P* coda and examined in a similar manner (Figure 4.3). A P_g phase follows the teleseismic *P* phase at 500s, limiting measurement to a time interval before 500s. The slow coda amplitude rise near the onset of the direct wave is also seen for this model (Figure 4.3a,b) because the amplitude of the reflectivity synthetics is low at less than 100 km offset (Appendix B). Measurement of the coda envelope slope gives a γ value of -0.06 and a κ value of 0.004 (Figure 4.3c) for the teleseismic *P* coda, indicating a much lesser dependence

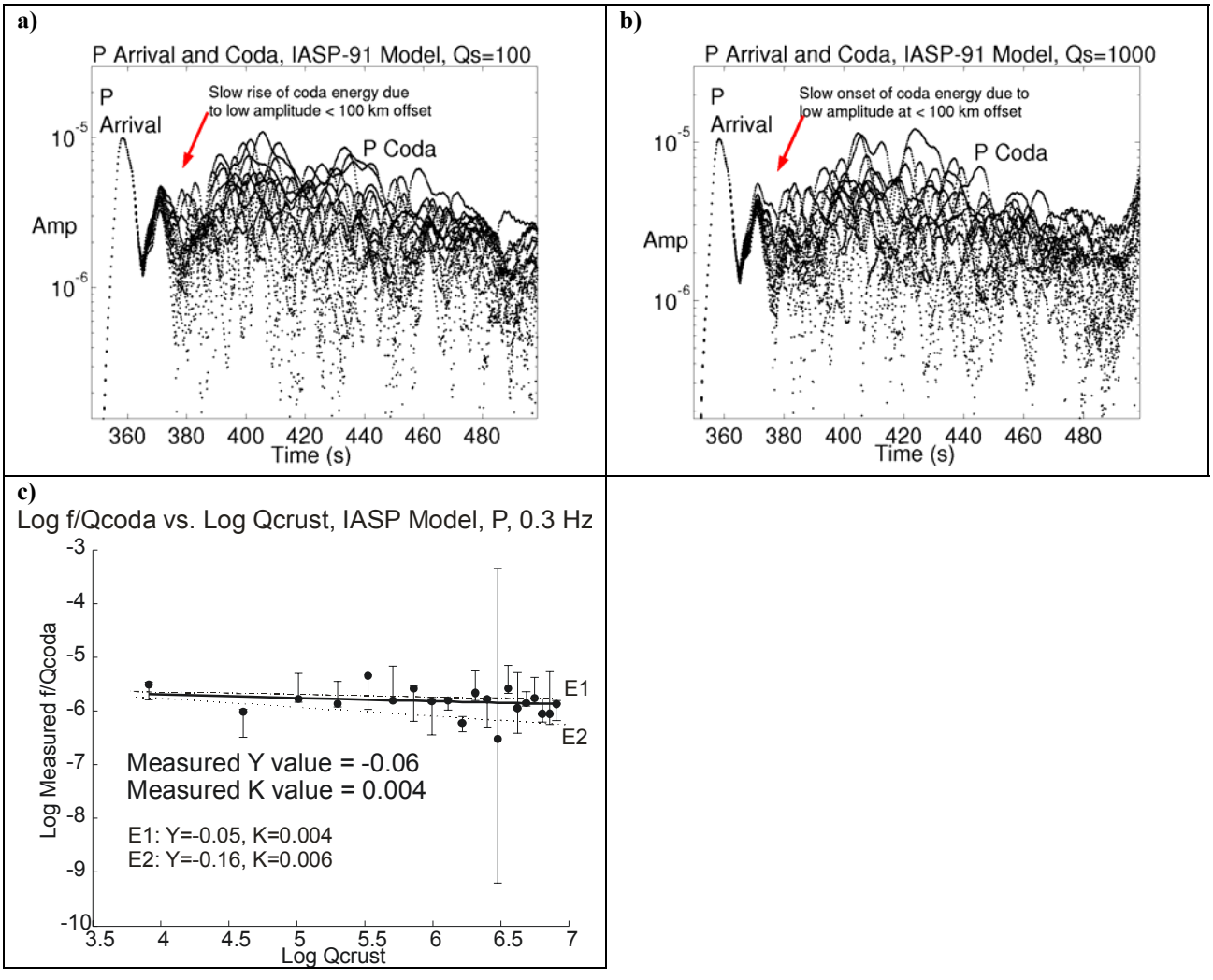


Figure 4.3: IASP-91 model results at 0.3 Hz. a) Teleseismic *P* direct arrival with coda at Q_{crust}=100; the slow rise of teleseismic *P* coda amplitude as seen in the Quartz-4 model is also evident here. b) Teleseismic *P* arrival and coda at Q_{crust}=1000. c) Determination of fitting parameters for Q_{coda}-Q_{crust} relation.

of Q_{coda} on Q_{crust} than for the Quartz-4 model; the relation is $Q_{coda} = 75 \cdot Q_{crust}^{0.06}$. This is expected because there is no strong crustal L_g phase in the IASP-91 synthetics as in the Quartz-4 synthetics.

4.4 IASP-Quartz Model, Teleseismic P Coda, 0.3 Hz

A similar procedure was also performed for the IASP-Quartz model (Figure 4.4). The arrivals are similar to those that resulted from the IASP-91 model with simple crust and mantle, with P_g following the teleseismic P. Measurement of the coda slope gives a γ value of -0.14 and a κ value of 0.007 (Figure 4.4c) for the teleseismic P coda, indicating again a weak dependence of Q_{coda} on Q_{crust} because of the absence of the L_g arrival. The relation is $Q_{coda} = 49 \cdot Q_{crust}^{0.14}$ for teleseismic P coda at 0.3 Hz.

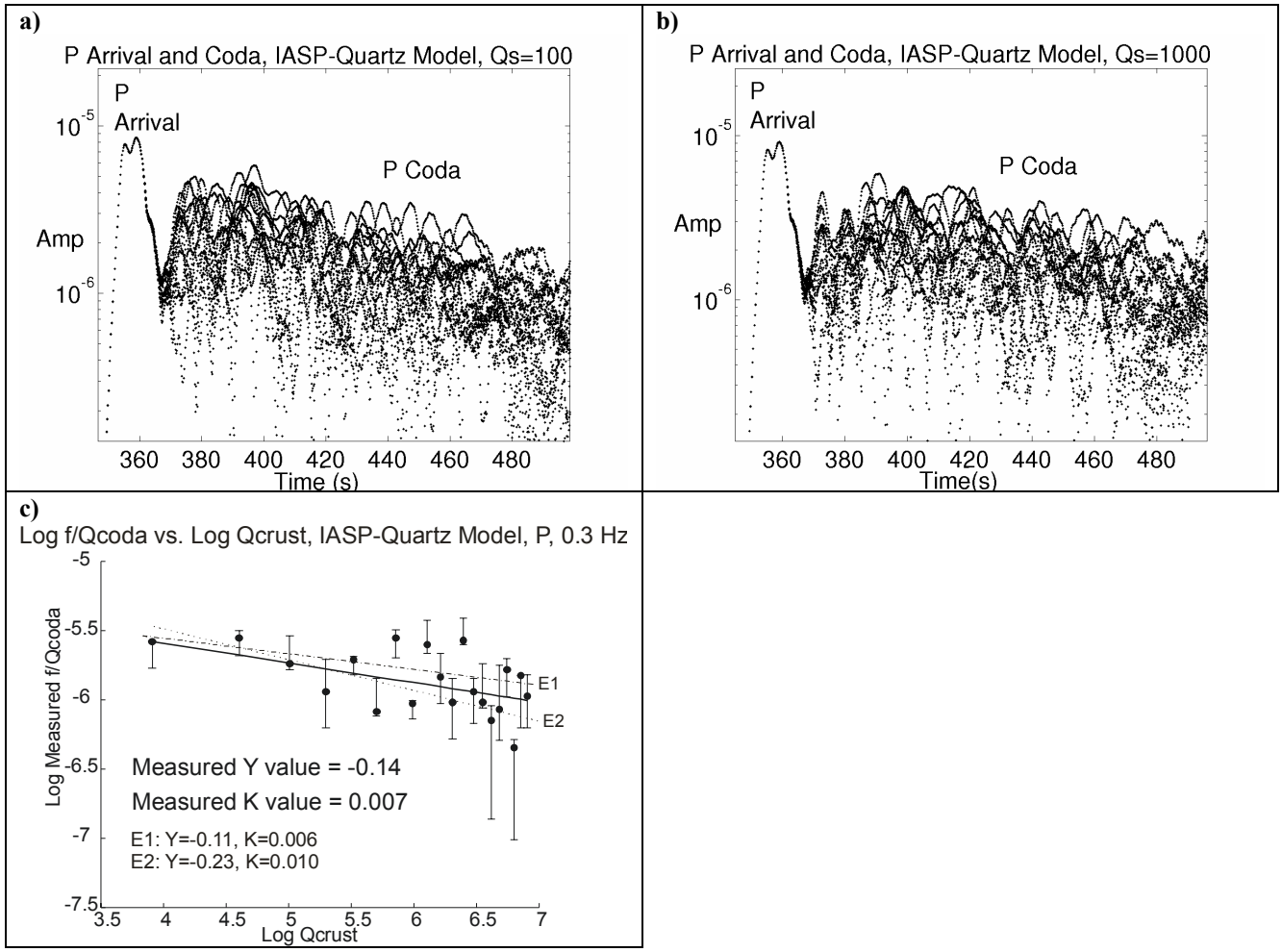


Figure 4.4: IASP-Quartz model at 0.3 Hz. a) Teleseismic *P* direct arrival with coda at Q_{crust}=100; the slow rise of teleseismic *P* coda amplitude is also evident here. b) Teleseismic *P* arrival and coda at Q_{crust}=1000. c) Determination of fitting parameters for Q_{coda}-Q_{crust} relation.

4.5 Complex Crust Model, P Coda, 0.3 Hz

The complex crust model, with arrivals similar to the Quartz-4 model, gives a γ value of -0.98 and a κ value of 0.148 (Figure 4.5) for the teleseismic *P* coda, indicating a strong dependence of Q_{coda} on Q_{crust} because of the strong crustal L_g phase. The relation is $Q_{coda} = 2 \cdot Q_{crust}^{0.98}$ for teleseismic *P* coda at 0.3 Hz.

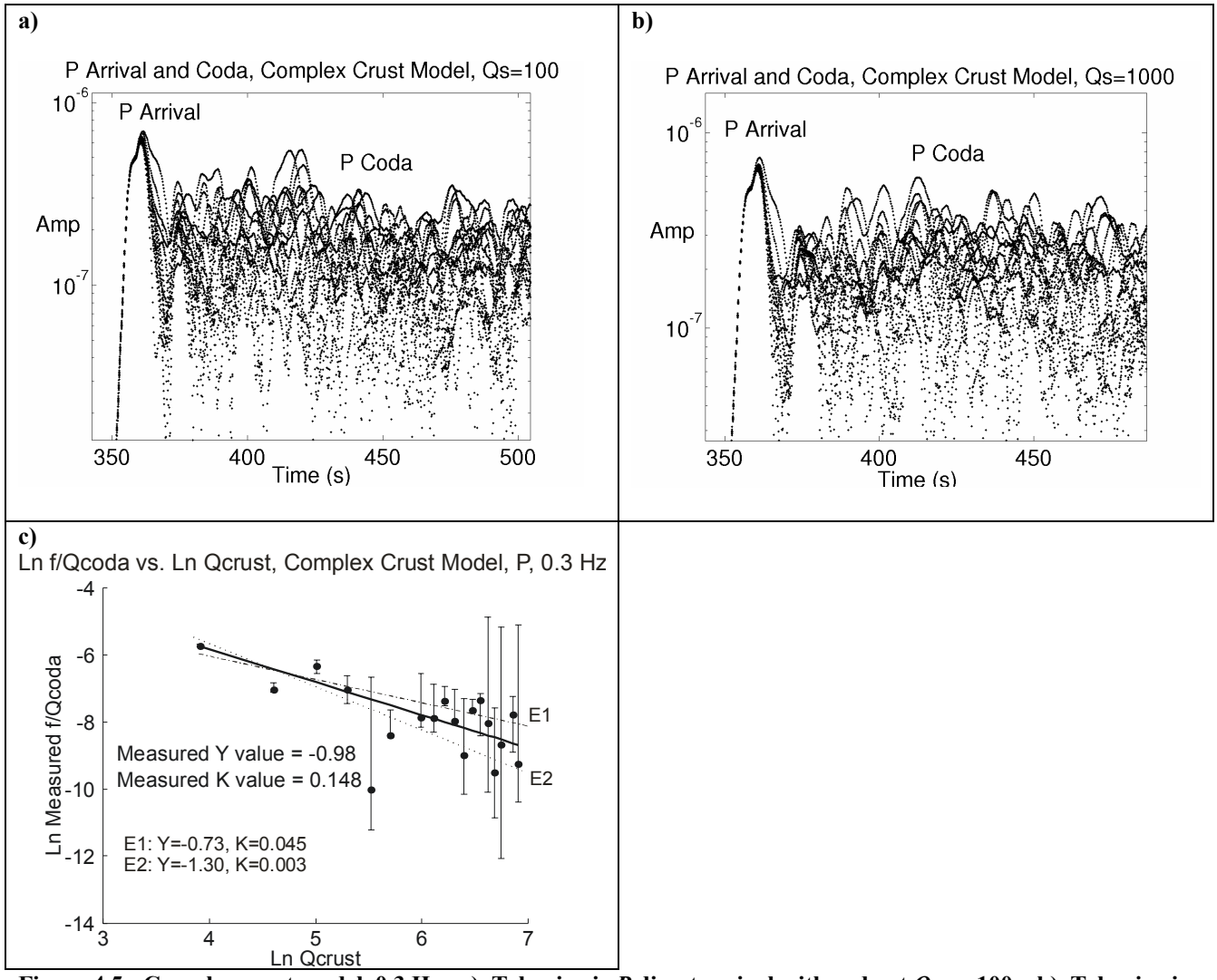


Figure 4.5: Complex crust model, 0.3 Hz. a) Teleseismic *P* direct arrival with coda at Q_{crust}=100. b) Teleseismic *P* arrival and coda at Q_{crust}=1000. c) Determination of fitting parameters for Q_{coda}-Q_{crust} relation.

4.6 *Quartz-4 Model, L_g Coda, 0.3 Hz*

The L_g coda was also scaled and examined for the Quartz-4 model at 0.3 Hz (Figure 4.6). The L_g coda at low Q_{crust} values may be interpreted easily, but at higher Q_{crust} values, coda appears to be nearly flat. The relation is $Q_{coda} = 18 \cdot Q_{crust}^{0.40}$ for L_g coda at 0.3 Hz.

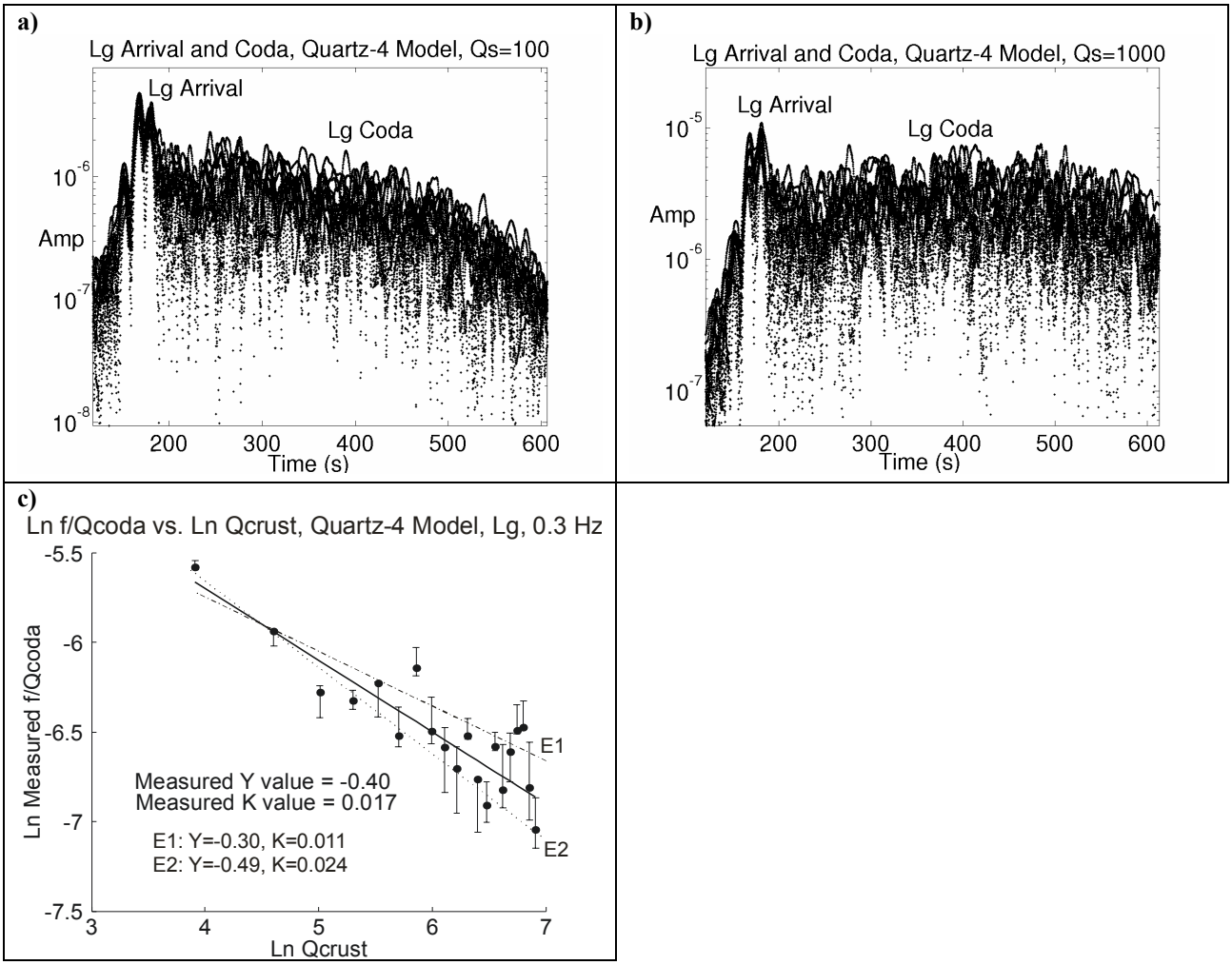


Figure 4.6: Quartz-4 model, L_g coda, 0.3 Hz. a) L_g coda envelope slope easily interpreted at low Q_{crust} values. b) amplitude appears flat at higher Q_{crust} values. c) Fit for parameters.

4.7 *Complex Crust Model, L_g Coda, 0.3 Hz*

The relation determined for the L_g coda at 0.3 Hz of the complex crust model is $Q_{coda} = 6 \cdot Q_{crust}^{0.78}$ (Figure 4.7).

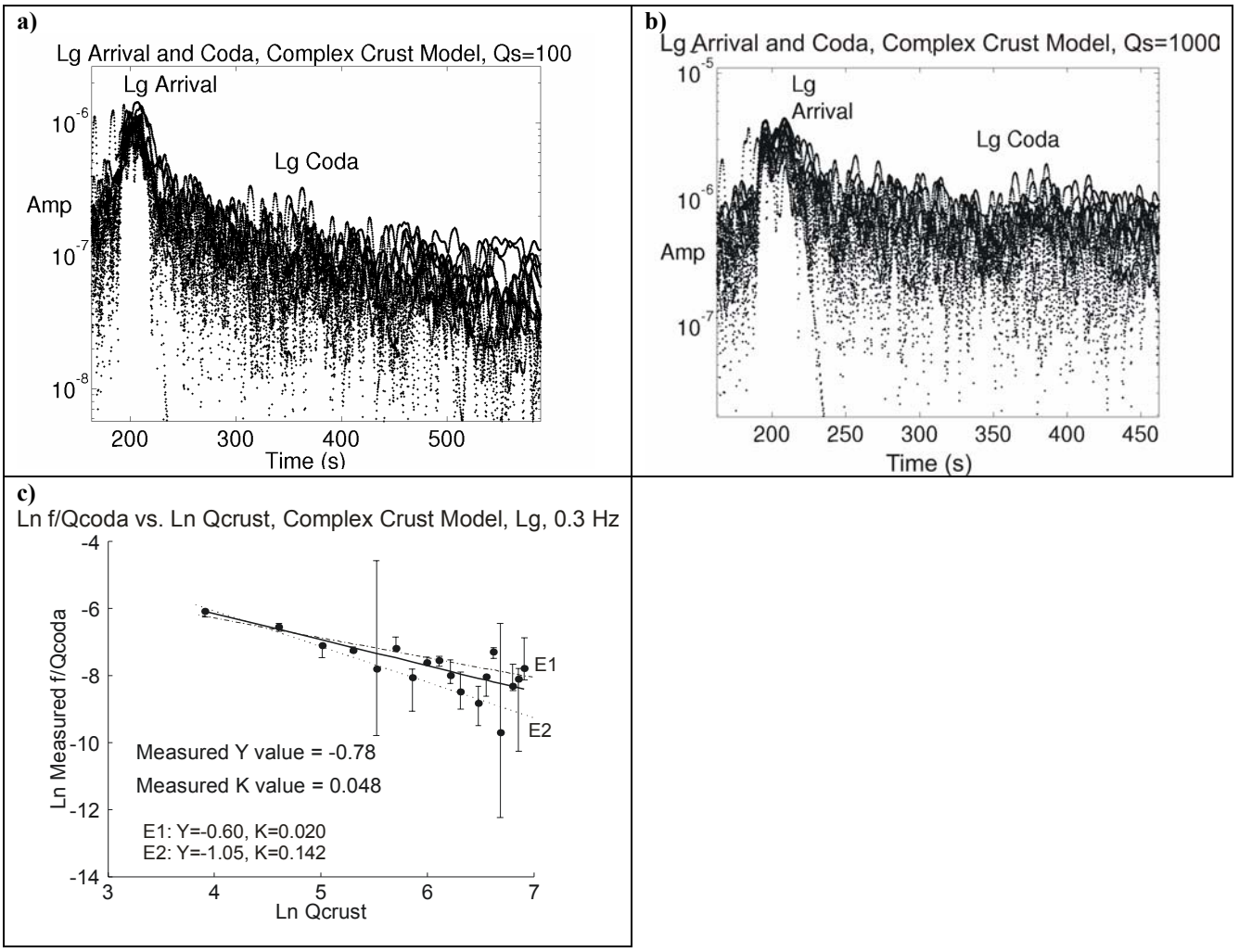


Figure 4.7: Complex crust model, L_g coda, 0.3 Hz. a) L_g arrival with coda at Q_{crust}=100. b) L_g arrival and coda at Q_{crust}=1000. c) Determination of fitting parameters for Q_{coda}-Q_{crust} relation.

4.8 Parameterizations at 0.7 Hz

To examine the effects of a higher frequency range, I filtered the reflectivity synthetics to a dominant frequency of 0.7 Hz. Results are shown in Figure 4.8 and tabulated in Table 1.

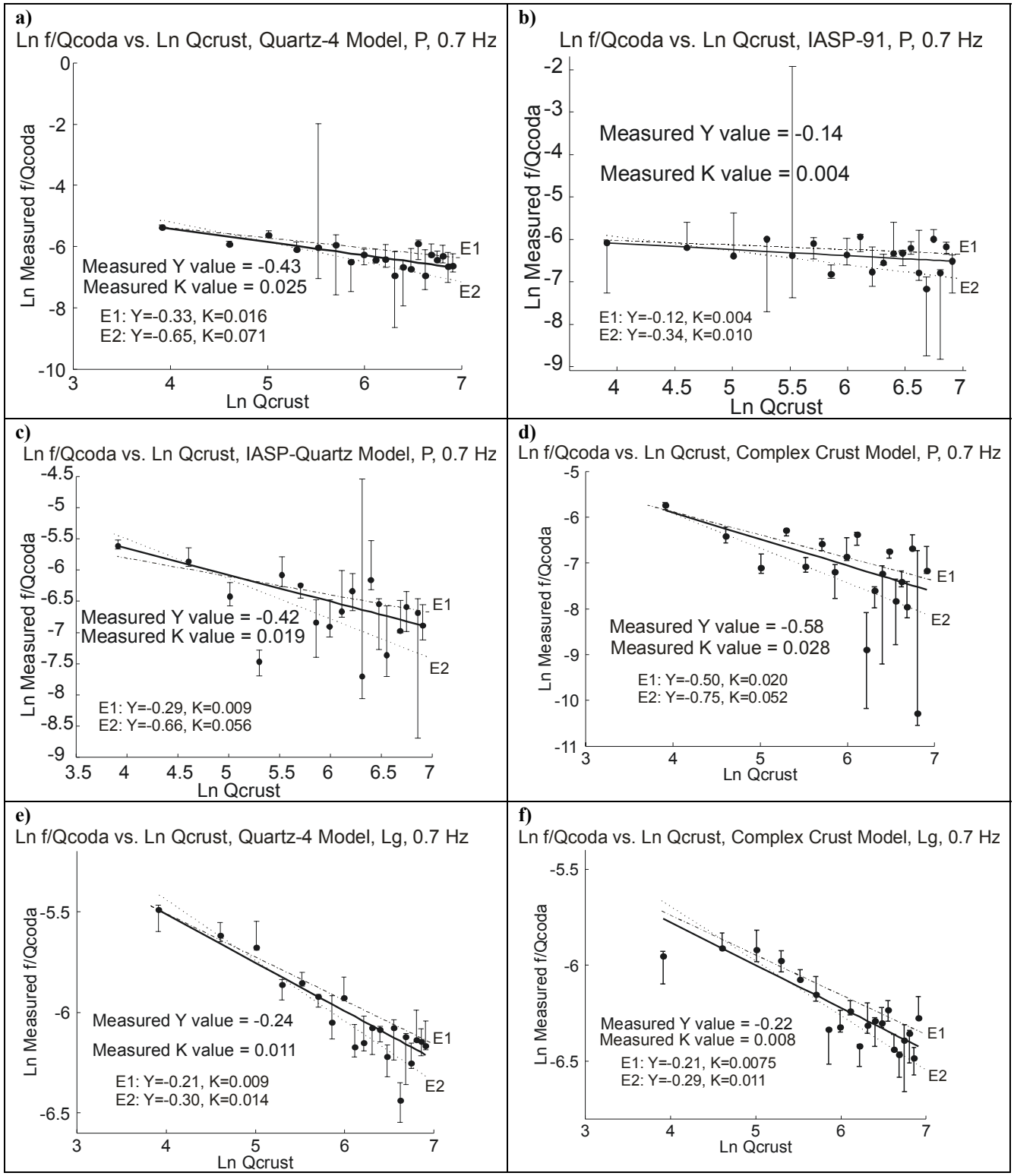


Figure 4.8: Parameterizations for Q_{coda}-Q_{crust} relation at 0.7 Hz. a) Quartz-4 model, *P* coda. b) IASP-91 model, *P* coda. c) IASP-Quartz model, *P* coda. d) Complex crust model, *P* coda. e) Quartz-4 model, L_g coda. f) Complex crust model, L_g coda.

4.9 Parameterizations at 1.3 Hz

The synthetics at 1.3 Hz were of poor quality because their strength at that frequency was several orders of magnitude smaller than at 0.7 Hz or 0.3 Hz. Therefore, teleseismic *P* coda measurements were unreliable. L_g coda measurements were performed for the Quartz-4 and complex crust models; results appear in Figure 4.9.

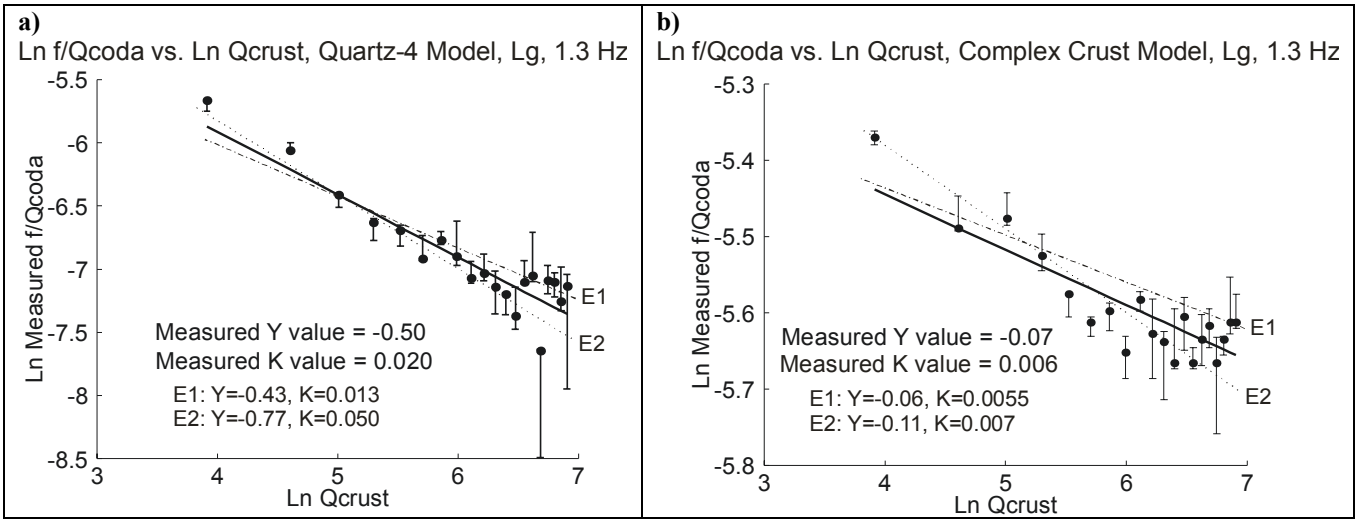


Figure 4.9: Parameterizations for Q_{coda}-Q_{crust} relation at 1.3 Hz. a) Quartz-4 model, L_g coda. b) Complex crust model, L_g coda.

4.10 Summary of Model Results

Results for the coefficients and exponents obtained from modeling results appear in Table 1.

Model, Coda	0.3 Hz		0.7 Hz		1.3 Hz	
	Coeff.	Exp.	Coeff.	Exp.	Coeff.	Exp.
Quartz-4, *P*	16	0.47	28	0.43	-	-
IASP-91, *P*	75	0.06	75	0.14	-	-
IASP-Quartz, *P*	49	0.14	37	0.42	-	-
Complex Crust, *P*	2	0.98	25	0.58	-	-
Quartz-4, L_g	18	0.40	63	0.24	65	0.50
Complex Crust, L_g	6	0.78	88	0.22	217	0.07

Table 1: Summary of modeling results.

4.11 Interpretation of Quartz PNE Data

Estimates of Q_{crust} for the Quartz PNE data were made using the inverse relation of the parameterization of Q_{coda} as a function of Q_{crust} for model data. The value for Q_{crust} of the Quartz data is given by:

$$Q_{crust} = \left(\frac{f}{\kappa Q_{coda}} \right)^{1/\gamma} = \left(\frac{\kappa_{coda}}{\kappa} \right)^{1/\gamma} \tag{7}$$

where values of γ and κ are from the modeling results and the inverse of coda attenuation parameter $\kappa_{coda} = \left(\frac{f}{Q_{coda}} \right)$ is measured from the coda envelope slope from the Quartz data. The primary frequencies of the Quartz data are between 1 Hz and 10 Hz, outside the strong range for the reflectivity synthetics here. The frequency range of the synthetics was limited by capabilities of the code used in their creation in that there was a tradeoff between trace length and sampling interval. Long traces, necessary when performing modeling at long offsets to achieve substantial separation between arrivals, required a greater sampling interval, and therefore the Nyquist frequency was limited to lower values than would have been possible with shorter traces.

To obtain an estimate of Q_{crust} for the teleseismic *P* coda of Quartz data, I selected a trace from the Quartz data at offset 2510 km and filtered it to a dominant frequency of 1.3 Hz (Figure 4.10a) to be as near as possible to the dominant frequency of the reflectivity synthetics used in modeling. The Quartz-4 model seems the best parameterization to use for a Q_{crust} estimate because this velocity model was obtained by travel-time modeling along the Quartz PNE line. Therefore, using κ_{coda} = 1/363 and parameter values γ = -0.43 and κ = 0.025 obtained at 0.7 Hz gives, using Eq. 7, a value of Q_{crust} = 169.

For an estimate from L_g coda, I selected an L_g coda segment from a Quartz trace at 650 km offset, as the L_g phase is most recognizable there (Figure 4.10b). The trace was filtered to 1.3 Hz.

Using the parameter values of γ = -0.50 and κ = 0.020 obtained at 1.3 Hz gives, using Eq. 7, a value of Q_{crust} = 23.

Q_{coda} and Q_{crust} values are expected to differ based on tectonic regime. Cratons and inactive regions are expected to have the highest values, while active regions, basins, and rocks with fluid-filled cracks and pores are expected to have lower values. Thick accumulations of sediment and severe velocity gradients at the crust-mantle transition also may decrease values (Mitchell and Cong 1998). These authors also suggest that Q_{coda} values increase with time elapsed from the most recent tectonic activity because of loss of crustal fluids and permeability with time.

Cong and Mitchell (1998) reported Q_{crust} values of 63 for the Turkish and Iranian Plateaus, 71 near the Black and Caspian Seas, and 201 for the Arabian Peninsula. The value I obtain for the Quartz crust from the teleseismic *P* coda, recorded in the East European Craton, is commensurate with these values. The L_g coda trace, recorded in the West Siberian Basin, which is overlain by 2-4 km of sediment, indicates a substantially lower Q_{crust} value than the teleseismic *P* coda trace, as expected.

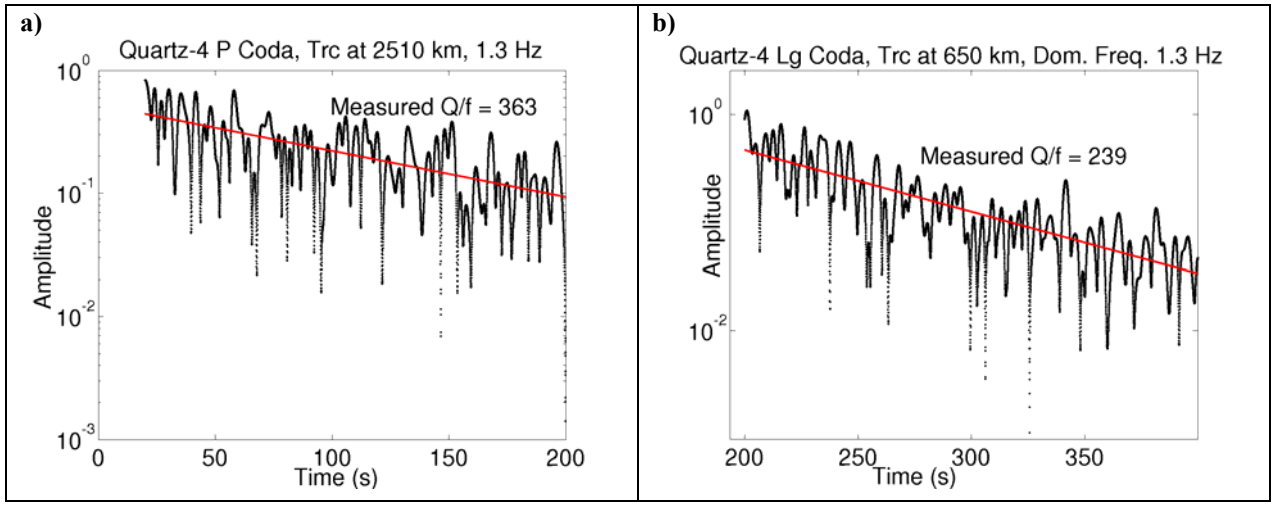


Figure 4.10: Coda segments from Quartz data. a) Teleseismic *P* coda at 1.3 Hz. b) L_g coda at 1.3 Hz.

5 Discussion

Comparing the crustal complexity of the velocity models and the resulting crustal phases in the reflectivity synthetics clarifies the differing Q_{coda}-Q_{crust} relations obtained for the models. The Quartz-4 model has a 3-km-thick sediment layer overlying a 3-layer crust with velocity discontinuities at 18 and 27 km depth and Moho at 43 km (Figure 2.3a). In addition, the Quartz-4 mantle is complex, with low-velocity zones at 110 and 210 km depth. Strong L_g and P phases are evident in the reflectivity synthetics (Figure 2.4a). The Quartz-4 model shows strong ringing and slow attenuation of both teleseismic P and L_g codas, with a dependence of $Q_{coda} = 16 \cdot Q_{crust}^{0.47}$ at 0.3 Hz for the P coda and $Q_{coda} = 18 \cdot Q_{crust}^{0.40}$ for the L_g coda (Table 1). The complex crust model contains a 5-layer crust and the Quartz-4 mantle (Figure 2.3d), which also produces strong L_g and P phases and strong relations of $Q_{coda} = 2 \cdot Q_{crust}^{0.98}$ for the P coda and $Q_{coda} = 6 \cdot Q_{crust}^{0.78}$ for the L_g coda at 0.3 Hz.

Both models using the simple IASP-91 crust give much smaller exponent values than these. The IASP-91 reflectivity synthetics, based on a 2-layer crust and upper mantle velocity model without low velocity zones (Figure 2.3b), show no strong L_g phases, indicating much less crustal propagation for this model than for the more crustally-complex models. The teleseismic P coda is found to decay more rapidly than in the previous cases. Additionally, the Q_{coda} values are not strongly dependent on Q_{crust}, with $Q_{coda} = 75 \cdot Q_{crust}^{0.06}$ at 0.3 Hz. Evidently, simple crust, simple mantle, or both contribute to the weak dependence. Examining the IASP-Quartz model, with simple IASP-91 crust and complex Quartz-4 mantle (Figure 2.3c), shows that complex crust contributes most to complex reflectivity synthetics and resulting strong coda. The Quartz-4 and complex crust models contain many more velocity discontinuities and gradient changes (Figure 2.3) than the other two models, leading to strong crustal and shallow mantle phases (Figure 2.4). The reflectivity synthetics of the IASP-Quartz model (Figure 2.4c) appear nearly identical in terms of seismic phases

to the IASP-91 model, with simple crust and mantle, therefore the crust of the IASP-Quartz model has a greater effect than the mantle. The resulting dependence appears similar to the IASP-91 model as well, with $Q_{coda} = 43 \cdot Q_{crust}^{0.14}$ at 0.3 Hz, only slightly greater than for the IASP-91 model. Thus, velocity models with more complex crusts clearly result in higher-complexity crustal seismic phase arrivals, stronger teleseismic *P* coda, and stronger Q_{coda}-Q_{crust} dependence.

These findings provide strong support for the argument of Morozov and Smithson (2000, 2001) that crustal scattering should be a key contributor to the scattered wavefield. In theory, modeling using a uniform crust may be consistent with the PNE data (Ryberg *et. al.* 1995; Ryberg and Wenzel 1999; Tittgemeyer *et. al.* 1996, 2000), but the crust is known to be highly heterogeneous, providing a foundation for crustal scattering; therefore, modeling that excludes crustal scattering is unrealistic. Morozov and Smithson (2000) suggest that strong coda results from crustal scattering of seismic waves incident from the mantle and propagating at high apparent velocities (Figure 1.4, MD). In addition, crustal guided waves such as P_g, L_g, S_g, and R_g likely produce secondary phases propagating though the mantle which also contribute to the observed coda (Dainty 1985, 1990). Thus, crustal scattering occurs both near the source and near the receiver. Morozov and Smithson (2000) obtain estimates (Figure 1.5) of $Q_{coda} = 380$ at 2 Hz and $Q_{coda} = 430$ at 5 Hz. These correspond to crustal values in Eurasia; Mitchell *et. al.* (1997) noted Q_{coda} values to range from less than 350 to greater than 800. This supports the Morozov and Smithson (2000) suggestion that this coda is primarily due to L_g energy.

To examine the Q_{coda}-Q_{crust} relation as a function of frequency, I compared the parameter values obtained at 0.3, 0.7, and 1.3 Hz for the models. Only the L_g codas were examined at 1.3 Hz because of the weak signal present after synthetics had been filtered to 1.3 Hz.

The Quartz-4 *P* coda parameterization showed a slight decline in exponent value, from 0.47 to 0.43, between 0.3 and 0.7 Hz (Table 1). This is likely due to the decrease in amplitude of the L_g events relative to *P* events from 0.3 to 0.7 Hz (Figure 2.5; Figure 2.6). The complex crust *P* coda

dependence decreased from 0.98 to 0.58 from 0.3 to 0.7 Hz. Amplitude was actually greater at 0.7 Hz than at 0.3 Hz, but energy remained comparable. For the IASP-91 model, the exponent increased slightly, from 0.06 to 0.14, in accordance with the relative increase of S/L_g amplitude relative to P amplitude between 0.3 and 0.7 Hz. The IASP-Quartz model showed a significant increase in exponent value between 0.3 and 0.7 Hz, concordant with the significant relative increase of S/L_g amplitude relative to P amplitude.

The Quartz-4 model's L_g coda gave exponent values of 0.40, 0.24, and 0.50, at 0.3 Hz, 0.7 Hz, and 1.3 Hz, respectively. The L_g amplitude at 1.3 Hz is higher relative to P than that at 0.7 Hz, consistent with the higher exponent value, though the amplitude is not as great as that at 0.3 Hz. The complex crust model shows a progressive downward trend in exponent value as a function of frequency, with values of 0.78, 0.22, and 0.07 at 0.3 Hz, 0.7 Hz, and 1.3 Hz, respectively. This is consistent with the lower relative amplitude of L_g relative to P at 1.3 Hz than at lower frequency values.

Estimates of Q_{crust} values for the region covered by Quartz PNE data were made using the modeling results. A Q_{crust} value of 169 was obtained for the teleseismic P coda recorded in the East European Craton. This is certainly a crustal, rather than mantle, value, supporting the assertions of crustal scattering made by Morozov and Smithson (2000, 2001). Mitchell and Cong (1998) found a range of Q_{crust} values in the Middle East from 63 to 201, though the Middle East has been tectonically active more recently than Northern Eurasia, and more recently tectonic areas are expected to have lower Q_{crust} values. Mitchell and Cong (1998) stated that Q_{crust} values up to about 1000 are possible for stable regions. The Quartz-4 model's L_g coda parameterization gave a Q_{crust} value of 23 for the Quartz L_g coda at 650 km offset from the source. This value is expected to be lower because the trace was recorded in the West Siberian Basin, overlain by 2-4 km of sediment, which can greatly reduce Q_{crust} values.

6 Conclusions

In this work, the seismic codas of teleseismic *P* and L_g events were simulated as seismic scattering from heterogeneities near the Earth's surface. Reflectivity synthetics based on velocity models of different crustal and upper mantle properties governed wave propagation in the model. The seismic response was calculated using numerical integration in time and a 2-D surface integral over uniformly-distributed Monte-Carlo-sampled surface points. Q_{coda} values were measured on modeled traces based on reflectivity synthetics with a range of intrinsic Q_{crust} values to determine Q_{coda} as a function of Q_{crust}. This relation was inverted to obtain Q_{crust} estimates for Quartz PNE data by measuring Q_{coda} of these data.

Using reflectivity synthetics based on velocity models of greater crustal complexity, like the Quartz-4 and complex crust models, led to greater coda strength in resulting receiver traces and stronger Q_{coda}-Q_{crust} relation. For the teleseismic *P* coda at 0.3 Hz, the Quartz-4 model gave a strong dependence of $Q_{coda} = 16 \cdot Q_{crust}^{0.47}$ at 0.3 Hz and $Q_{coda} = 28 \cdot Q_{crust}^{0.43}$ at 0.7 Hz. The IASP-91 model gave a much weaker dependence, with $Q_{coda} = 75 \cdot Q_{crust}^{0.06}$ at 0.3 Hz and $Q_{coda} = 75 \cdot Q_{crust}^{0.14}$ at 0.7 Hz, because of its much simpler crustal structure and fewer crustal arrivals. The IASP-Quartz model showed that the crust, not the mantle, played the most significant role in determining coda strength, with the relation given by $Q_{coda} = 49 \cdot Q_{crust}^{0.14}$ at 0.3 Hz. Therefore, crustal scattering has proven to be a viable mechanism to describe seismic coda.

An estimate of Q_{crust} for Quartz PNE data was made based on these estimates, though the frequency windows of Quartz data and model data differed due to modeling limitations. For a teleseismic *P* Q_{coda} value measured from a PNE Quartz trace at 1.3 Hz, the Quartz-4 model gave a Q_{crust} value of 169 using the parameterization at 0.7 Hz. The Quartz-4 L_g parameterization at 0.3 Hz gave a Q_{crust} estimate of 23 using the PNE Quartz L_g coda. These are both crustal values commensurate with others obtained previously in other regions.

References

- Aki, K., Richards, P., 1980. *Quantitative Seismology: Theory and Methods*, W.H. Freeman, San Francisco, CA.
- Anderson, D.L., Given, J.W., 1982. Absorption band *Q* model of the Earth, J. Geophys. Res. 87, 3893-3904.
- Aster, R., Borchers, B., Thurber, C., 2002. *Parameter Estimation and Inverse Problems*, preliminary edition.
- Bannister, S.G., Husebye, E.S., Ruud, B.O., 1990. Teleseismic *P* coda analyzed by three-component and array techniques: deterministic location of topographic *P*-to-R_g scattering near the NORESS array, Bull. Seis. Soc. Am. 80, 1969-1986.
- Cipar, J., Priestley, K., 1997. Central Siberia upper mantle cross-section from Deep Seismic Sounding explosions, in *Upper Mantle Heterogeneities from Active and Passive Seismology*, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Cong, L., Mitchell, B.J., 1998. Seismic Velocity and *Q* Structure of the Middle Eastern Crust and Upper Mantle from Surface-Wave Dispersion and Attenuation, Pure Appl. Geophys. 153, 503-538.
- CTBTO (Comprehensive Nuclear-Test-Ban Treaty Organisation) Preparatory Commission, *Monitoring Technologies*, online, accessed 29 May 2003, <http://www.ctbto.org>.
- Dainty, A.M., 1985. Air Force Geophysical Laboratory Report, AFGL-TF-86-0218.
- Dainty, A.M., 1990. Studies of coda using array and three-component processing, Pure App. Geophys. 132, 221-244.
- Egorkin, A.V., Pavlenkova, N.I., 1981. Studies of mantle structure of U.S.S.R. territory on long-range seismic profiles, Phys. Earth Planet. Int. 25, 12-26.
- Egorkin, A.V., Zyuganov, S.K., Pavlenkova, N.I., Chernishov, N.M., 1987. Results of lithospheric studies from long-range profiles in Siberia, Tectonophysics 140, 29-47.
- Egorkin, A.V., 1997. Evidence for 520-km discontinuity, in *Upper Mantle Heterogeneities from Active and Passive Seismology*, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Egorkin, A.V., 1998. Velocity structure, composition and discrimination of crustal provinces in the former Soviet Union, Tectonophysics 298, 395-404.
- Fuchs, K., Müller, G., 1971. Computation of Synthetic Seismograms with the Reflectivity Method and Comparison with Observations, Geophys. J. R. Astr. Soc. 23, 417-433.
- Fuchs, K., Schulz, K., 1976. Tunneling of low-frequency waves through the subcrustal lithosphere, J. Geophys. 42, 175-190.
- Greenfield, R.J., 1971. Short-period *P*-wave generation by Rayleigh-wave scattering at Novaya Zemlya, J. Geophys. Res. 76, 7988-8002.
- Gupta, I.N., McElfresh, T.W., Wagner, R.A., 1991. Near-source scattering of Rayleigh to *P* in teleseismic arrivals from Pahute Mesa (NTS) shots, in *Explosion Source Phenomenology*, Ed. S.R. Taylor, H.J. Patton, and P.G. Richards, American Geophysical Monograph 65, 151-160.
- Kennett, B.L.N., 1986. L_g Waves and Structural Boundaries, Bull. Seis. Soc. Am. 76, 1133-1141.
- Kennett, B.L.N., 1991. *IASPEI 1991 Seismological Tables*, Research School of Earth Sciences, Australia National University, Canberra.
- Kozlovsky, Y.A., 1990. The USSR Integrated Program of Continental Crust Investigations

and Studies of the Earth's Deep Structure under the Globus Project, in *Super-Deep Continental Drilling and Deep Geophysical Sounding*, ed. K. Fuchs, Y.A. Kozlovsky, A.I. Krivtsov, M.D. Zoback, Springer, Berlin, 90-103.

- Mantovani, E., Schwab, F., Liao, H., Knopoff, L., 1977. Teleseismic S_n: a guided wave in the mantle, Geophys. J. R. Astr. Soc. 51, 709-726.
- Mechie, J., Egorkin, A.V., Fuchs, K., Ryberg, T., Solodilov, L., Wenzel, F., 1993. *P*-wave mantle velocity structure beneath northern Eurasia from long-range recording along the profile Quartz, Phys. Earth Planet. Int. 79, 269-286.
- Mechie, J., Egorkin, A.V., Solodilov, L., Fuchs, K., Lorenz, F., Wenzel, F., 1997. Major features of the mantle velocity structure beneath northern Eurasia from long-range seismic recordings of Peaceful Nuclear Explosions, in *Upper Mantle Heterogeneities from Active and Passive Seismology*, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Menke, W.H., Richards, P.G., 1980. Crust-mantle whispering gallery phases: a deterministic model of teleseismic P_n wave propagation, J. Geophys. Res. 85, 5416-5422.
- Molnar, P., Oliver, J., 1969. Lateral Variations of attenuation in the upper mantle and discontinuities in the lithosphere, J. Geophys. Res. 74, 2648-2682.
- Mitchell, B.J., Pan, Y., Xie, J., Cong, L., 1997. L_g coda Q variation across Eurasia and its relation to crustal evolution, J. Geophys. Res. 102, 22767-22779.
- Mitchell, B.J., Cong, L., 1998. L_g Coda Q and its Relation to the Structure and Evolution of Contintents: A Global Perspective, Pure Appl. Geophys. 153, 655-663.
- Morozov, I.B., Morozova, E.A., Smithson, S.B., 1998a. On the nature of the teleseismic P_n phase observed in the recordings from the ultra-long profile "Quartz", Russia, Bull. Seis. Soc. Am. 88, 62-73.
- Morozov, I.B., Morozova, E.A., Smithson, S.B., Solodilov, L.N., 1998b. 2-D Image of Seismic Attenuation beneath the Deep Seismic Sounding Profile "Quartz," Russia, Pure Appl. Geophys. 153, 311-343.
- Morozov, I.B., Smithson, S.B., 2000. Coda of Long-Range Arrivals from Nuclear Explosions, Bull. Seis. Soc. Am. 90, 929-939.
- Morozov, I.B., Smithson, S.B., 2001. Amplitude Analysis and Modeling of Regional Phases in PNE Profiles in Northern Eurasia and Seismic Regionalization, Defense Threat Reduction Agency Program Research and Development Announcement DRTA01-PRDA-01-01.
- Morozov, I.B., 2001. High-frequency wave propagation in the uppermost mantle; discussion, J. Geophys. Res. 106, 30715-30718.
- Morozova, E.A., Morozov, I.B., Smithson, S.B., Solodilov, L.N., 1999. Heterogeneity of the uppermost mantle beneath Russian Eurasia from the ultra-long profile QUARTZ, J. Geophys. Res. 104, no. B9, 20329-20348.
- Morozova, E.A., Morozov, I.B., Smithson, S.B., Solodilov, L., 2000. Lithospheric boundaries and upper mantle heterogeneity beneath Russian Eurasia: evidence from the DSS profile QUARTZ, Tectonophysics 329, 333-344.
- Nielsen, L., Thybo, H., Solodilov, L., 1999. Seismic tomographic inversion of Russian PNE data along profile Kraton, Geophys. Res. Lett. 26, 3413-3416.
- Nielsen, L., Thybo, H., Egorkin, A.V., 2001. Constraints on reflective bodies below the 8° discontinuity from reflectivity modeling, Geophys. J. Int. 145, 759-770.
- Pavlenkova, N.I., 1996. General features of the uppermost mantle stratification from long-range seismic profiles, Tectonophysics 264, 261-278.
- Pavlenkova, N.I., Pavlenkova, G.A., Solodilov, L.N., 1996. High velocities in the uppermost mantle of the Siberian craton, Tectonophysics 262, 51-65.
- Priestley, K., Cipar, J., Egorkin, A., Pavlenkova, N., 1994. Upper-mantle velocity structure

beneath the Siberian platform, Geophys. J. Int. 118, 369-378.
- Ryaboy, V., 1989. *Upper mantle structure studies by explosion seismology in the USSR*, Delphic Associates, Arlington, VA.
- Ryberg, T., Fuchs, K., Egorkin, A.V., Solodilov, L., 1995. Observation of high-frequency teleseismic P_n on the long-range Quartz profile across northern Eurasia, J. Geophys. Res. 100, 18151-18163.
- Ryberg, T., Wenzel, F., Mechie, J., Egorkin, A., Fuchs, K., Solodilov, L., 1996. Two-Dimensional Velocity Structure beneath Northern Eurasia Derived from the Super Long-Range Seismic Profile Quartz, Bull. Seis. Soc. Am. 86, 857-867.
- Ryberg, T., Wenzel, F., Egorkin, A.V., Solodilov, L., 1998. Properties of the mantle transition zone in northern Eurasia, J. Geophys. Res. 103, 811-822.
- Ryberg, T., Wenzel, F., 1999. High-frequency wave propagation in the uppermost mantle, J. Geophys. Res. 104, 10655-10666.
- Ryberg, T., Tittgemeyer, M., Wenzel, F., 2000. Finite difference modeling of *P*-wave scattering in the upper mantle, Geophys. J. Int. 2000, 787-800.
- Schueller, W., Morozov, I.B., Smithson, S.B., 1997. Crustal and Uppermost Mantle Velocity Structure of Northern Eurasia along the Profile Quartz, Bull. Seis. Soc. Am. 87, 414-426.
- Stephens, C., Isacks, B.L., 1977. Toward an understanding of S_n: normal modes of Love waves in an oceanic structure, Bull. Seism. Soc. Am. 67, 69-78.
- Sultanov, D.D., Murphy, J.R., Rubinstein, Kh.D., 1999. A Seismic Source Summary for Soviet Peaceful Nuclear Explosions, Bull. Seis. Soc. Am. 89, 640-647.
- Sutton, G.H., Walker, D.A., 1972. Oceanic mantle phases recorded on seismograms in the northwestern Pacific at distances between 7° and 40°, Bull. Seism. Soc. Am. 62, 631-655.
- Thybo, H., Perchuć, E., 1997. A partially molten zone beneath the global 8° discontinuity at approximately 100 km depth, in *Upper Mantle Heterogeneities from Active and Passive Seismology*, ed. Karl Fuchs, Kluwer Academic Publishers, Dordrect, Netherlands.
- Tittgemeyer, M., Wenzel, F., Fuchs, K., Ryberg, T., 1996. Wave propagation in a multiple-scattering upper mantle: observations and modeling, Geophys. J. Int. 127, 492-502.
- Tittgemeyer, M., Ryberg, T., Fuchs, K., Wenzel, F., 1997. Observation of teleseismic P_n/S_n on super long-range profiles in northern Eurasia and their implications for the structure of the lithosphere, in *Upper Mantle Heterogeneities from Active and Passive Seismology*, ed. Karl Fuchs, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Tittgemeyer, M., Wenzel, F., Ryberg, T., Fuchs, K., 1999. Scales of Heterogeneities in the Continental Crust and Upper Mantle, Pure Appl. Geophys. 156, 29-52.
- Tittgemeyer, M., Wenzel, F., Fuchs, K., 2000. On the Nature of P_n, J. Geophys. Res. 105, 16173-16180.
- Yilmaz, O., 1987. *Seismic Data Processing*, ed. S.M. Doherty, Society of Exploration Geophysicists, Tulsa, OK.

Appendix A: Reflection Synthetic Creation SIA Job

This script for Dr. Igor Morozov's SIA system calls code that creates reflection synthetic seismic sections (Fuchs and Müller 1971) with the parameters given here. In this example, a section to 3500 km offset, sampling interval 200 ms, trace length 1572s, and Q_{crust} = 100, in the frequency range 0.01 to 2.5 Hz, is created.

For the Quartz-4 and complex crust models, the R_g phase was attenuated by specifying a low Q_{crust} value of 10 within the upper 3 km of the velocity model. Q_{crust} values are specified by Q_p and Q_s; mantle Q_p values were obtained from Morozov et. al. (1998b), model B.

```
######################################################################
#
#  R  E  F  L  E  C  T  I  V  I  T  Y
#
#  Creating Quartz-4 reflectivity synthetic
#
#  0-3500 km
#
#  Crustal Qs and Qp values specified here; Mantle Q values are
#     taken from Morozov et. al. 1998b, model B
######################################################################


##########################################################
#   I  N  P  U  T     P  A  R  A  M  E  T  E  R  S
##########################################################
*define

#run name
job    Morozova-qs100

#Q values; P and S
qp     200
qs     100

#trace parameters
TLEN   1572864          #trace length (ms)
TSAMP  200              #sample interval (ms)

#frequency parameters - corner frequencies (Hz)
flow   0.01
flow1  0.3
fhigh  2
fhigh1 2.5
swidth 5.               #spike width (ms)
wvtyp  fuchs            #wave type
gainv  0.4              #gain value

#receiver positions: 10 km spacing to offset 3500 km
rdepth 0.               #receiver depth (km)
rclose 10               #closest receiver position (km)
rspace 10               #receiver spacing (km)
rfar   3500.            #furthest receiver position (km)

#output files
outdir /transient6/jnduenow/synthetics
outfile{outdir}/{job}-t{TLEN}-s{TSAMP}.sia      #sia file for input to scattering jobs
logf   {outdir}/{job}-t{TLEN}-s{TSAMP}-param.txt

*setup
noapp                   #no appending to previous files (overwrite instead)
```

```
*info  edit    time
Creating REFLECTIVITY synthetics for syncoda scattering
################################################
# Reflection synthetic parameters
#      Qs values default to Vp/sqrt(3)
#      dabove, dbelow are density (g/cm^3)
################################################
*call  reflect displ   TSAMP   TLEN            x       comp
modes  full
fwin   2       flow    flow1   fhigh   fhigh1  0.      #frequency window
phvwin 2       2.      3.      80.     120.    1200    #phase velocity window
source expl    0.8     wvtyp   0.      swidth  #exp. depth 0.8 km
receiv rdepth  rclose  rspace  rfar
reflect        0                               #depth at which reflectivity starts
layers flat    0.02    conv                    #flat layers, min vel contrast 0.02 km/s
#depth Vpabove Vpbelow Vsabove Vsbelow Qpabove Qpbelow Qsabove Qsbelow dabove  dbelow
0              4                       10      10      10      10      2.8
3      5.2     6.2                     10      qp      10      qs      2.8     2.8  1
18     6.4     6.5                     qp      qp      qs      qs      2.8     2.8  1
27     6.6     6.7                     qp      qp      qs      qs      2.8     2.8  1
43     6.9     8.2                     qp      1500    qs      750     2.8     3.2
80     8.3     8.3                     1500    800     750     400     3.2
110    8.5     8.45                    800     800     400     400     3.2
150    8.47    8.7                     800     400     400     200     3.2
210    8.3     8.3                     400     880     200     440     3.2
410    9.03    9.36                    880     1200    440     600     3.2
660    10.2    10.79                   1200    1200    600     600     3.2
871    11.2506 11.2506                 1200    1200    600     600     3.2

*info  eof     time

#####################################
# convert trace headers from km to m
#####################################
*call  hdrmath
set    x
1000                   x

#####################################
# write to .SIA file
#####################################
*call  write           outfile
Reflectivity synthetics in layered model
```

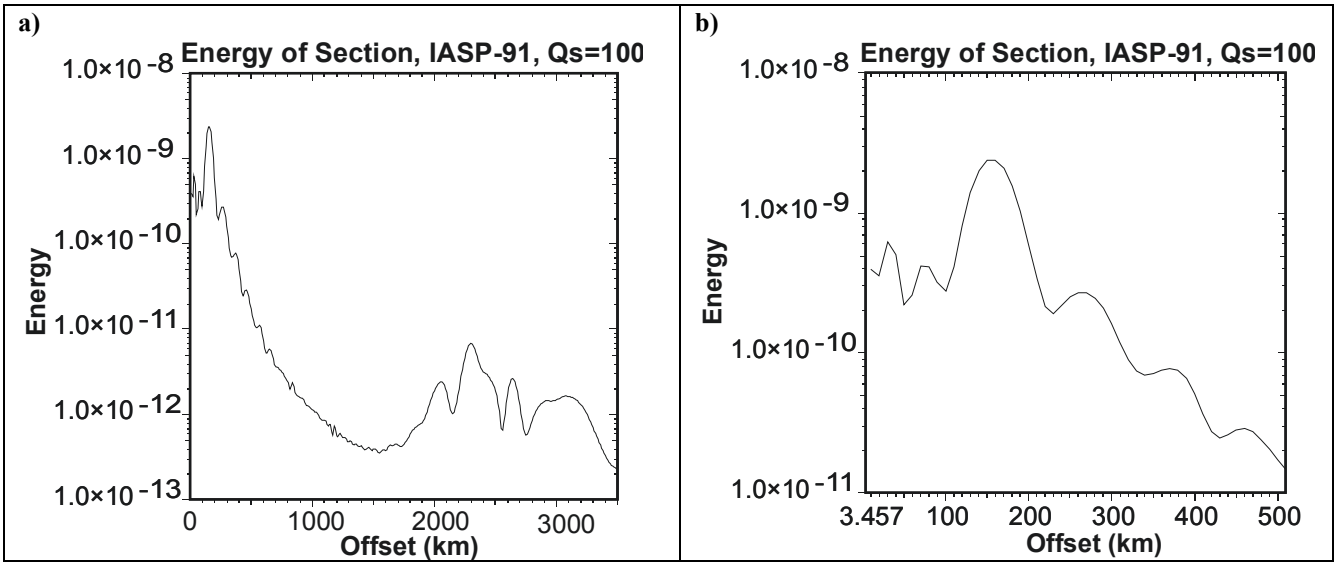
Appendix B: Trace Amplitude and Energy Measurements

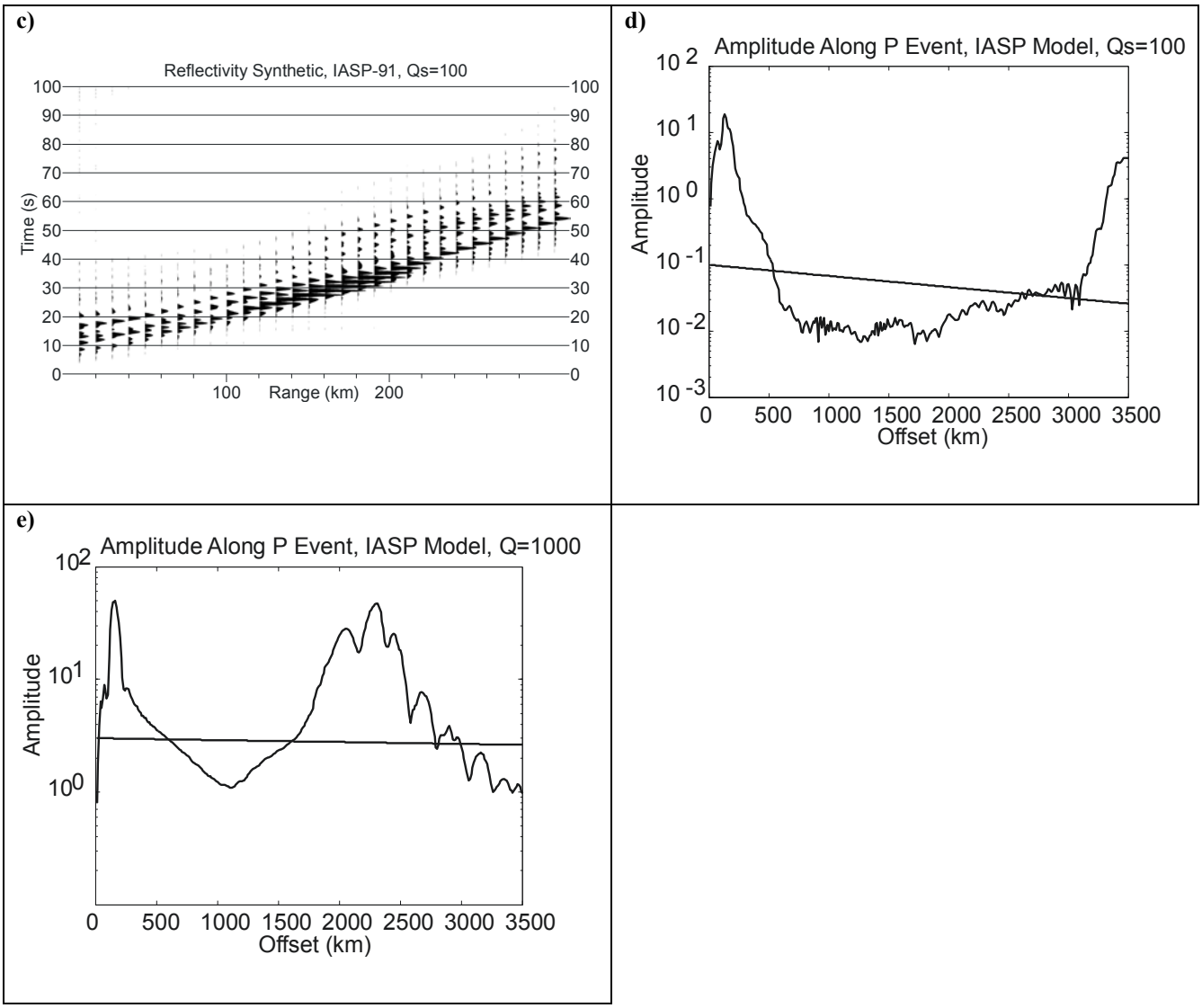
1) Energy and amplitude measurements of reflectivity synthetics based on IASP-91 velocity model.

a) Energy of trace vs. offset of the trace, Q_{crust} = 100. Energy decays due to geometrical spreading.
b) Zoom of energy at small offset, showing low energy between 0 and 100 km.
c) Arrivals in reflectivity synthetic show reason for low energy at 0-100 km offset. Note strong first arrival between 110 km and 210 km offset and increased amplitude after first arrival beyond 160 km, causing increase.

d,e) Amplitude measured along a thin, 10s strip of single seismic events. Amplitudes are corrected for spreading. Expected slope, given by $exp(-f_{dom}t/Q_{crust})$ for effects of crustal attenuation, is shown as a line.
d) Body wave *P*, Q_{crust} = 100, corrected for spreading by *offset.*
e) Body wave *P*, Q_{crust} = 1000, corrected for spreading by *offset*.

The high-amplitude portions at small offset occur because a large number of seismic events converge near the source, amplitudes of these other events contribute here. The peak in *d* beyond 3000 km is due to noise; the amplitude decays as expected between 1000 km and 2000 km. In *e*, there is a triplication near 2500 km from the teleseismic *P* to a refracted phase propagating at a greater depth in the mantle, giving the peak at large offset.



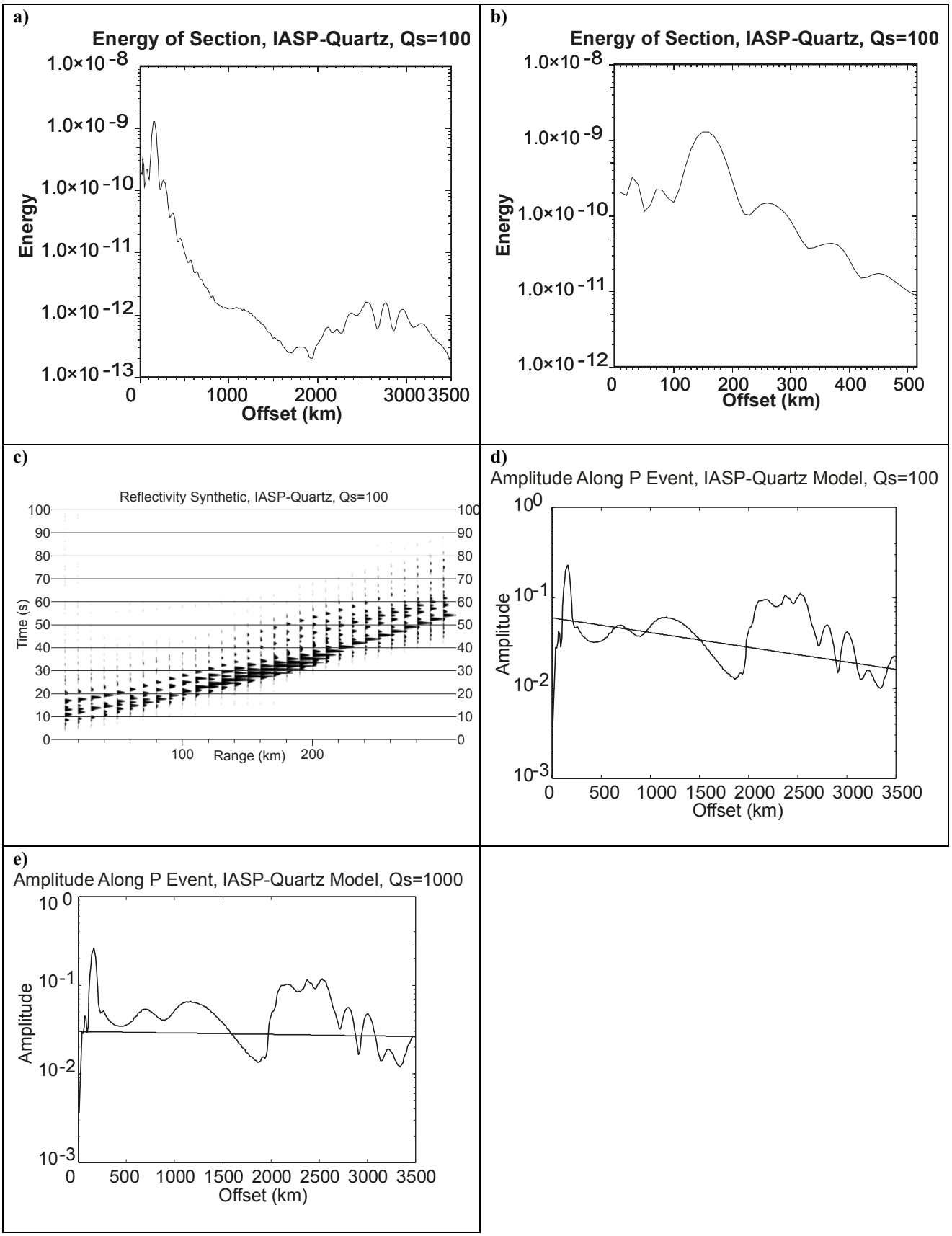


2) Energy and amplitude measurements of reflectivity synthetics based on IASP-Quartz velocity model.

a) Energy of trace vs. offset of the trace, Q_{crust} = 100. Energy decays due to geometrical spreading.
b) Zoom of energy at small offset, showing low energy between 0 and 100 km.
c) Arrivals in reflectivity synthetic show reason for low energy at 0-100 km offset. Note strong first arrival between 110 km and 210 km offset and increased amplitude after first arrival beyond 160 km, causing increase.

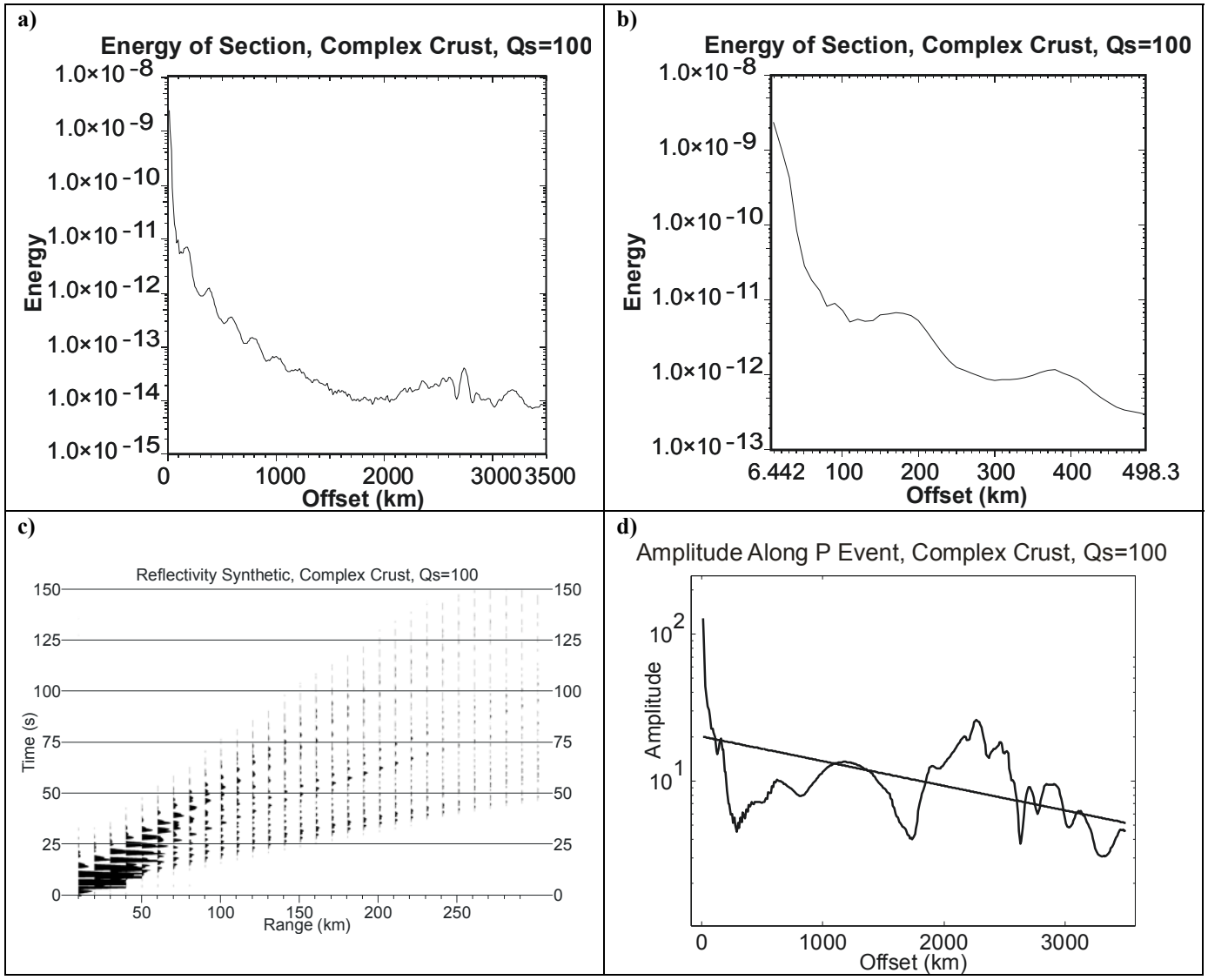
d,e) Amplitude measured along a thin, 10s strip of single seismic events. Amplitudes are corrected for spreading. Expected slope, given by $exp(-f_{dom}t/Q_{crust})$ for effects of crustal attenuation, is shown as a line.
d) Body wave *P*, Q_{crust} = 100, corrected for spreading by *offset.*
e) Body wave *P*, Q_{crust} = 1000, corrected for spreading by *offset*.

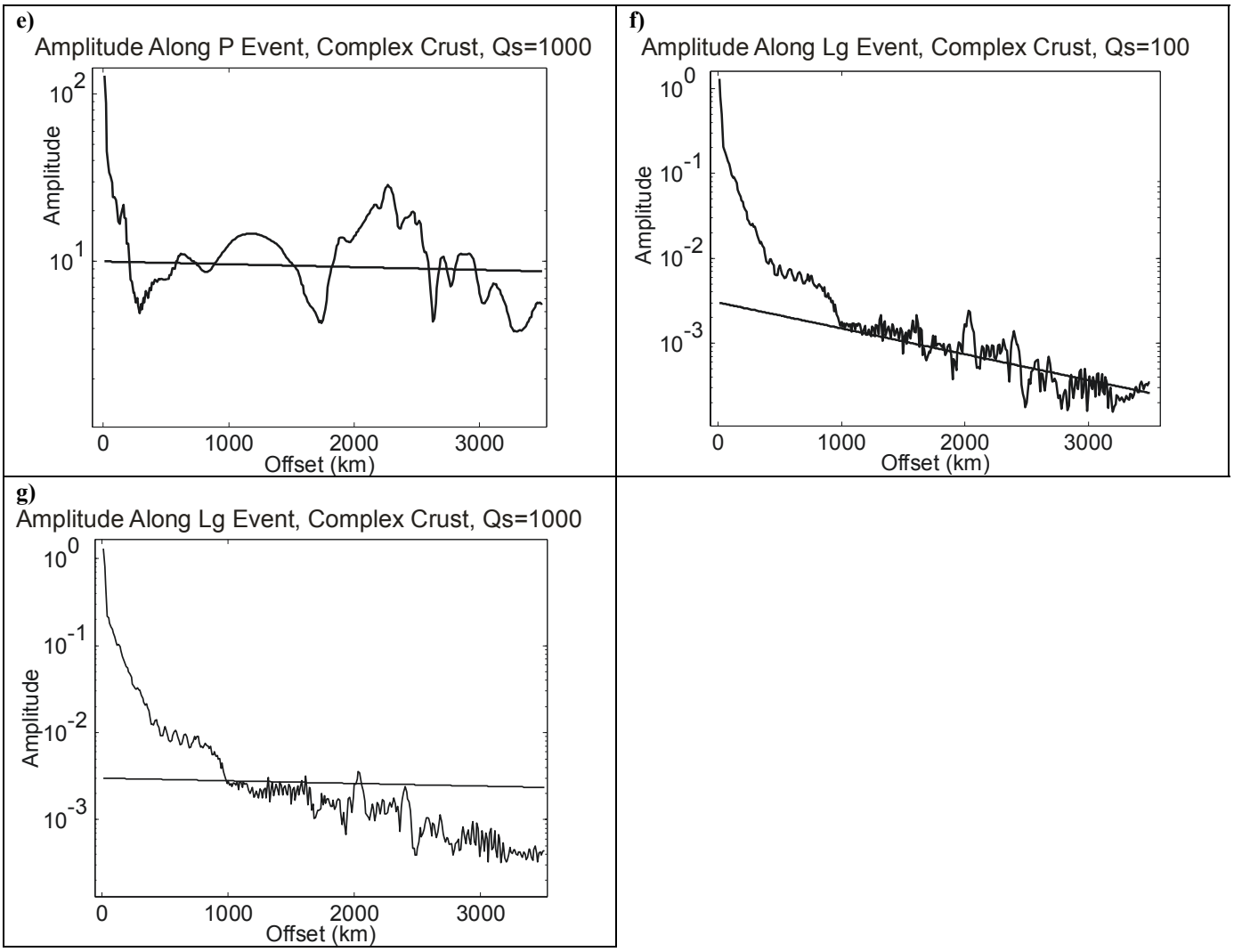
Amplitudes appear to decay as expected.


a)
Energy of Section, IASP-Quartz, Qs=100
Energy
Offset (km)
b)
Energy of Section, IASP-Quartz, Qs=100
Energy
Offset (km)
c)
Reflectivity Synthetic, IASP-Quartz, Qs=100
Time (s)
Range (km)
d)
Amplitude Along P Event, IASP-Quartz Model, Qs=100
Amplitude
Offset (km)
e)
Amplitude Along P Event, IASP-Quartz Model, Qs=1000
Amplitude
Offset (km)

3) Energy and amplitude measurements of complex crust model reflectivity synthetics.
a) Energy of entire trace plotted as a function of the trace's offset, Q_{crust}=100. Energy decays with offset due to geometrical spreading.
b) Decrease in amplitude between 100 km and 200 km is less evident than in the other models.
c) No significant gap in high-energy arrivals is seen at small offsets.

d-g) Amplitude measured along a thin, 10s strip of single seismic events. Amplitudes are corrected for spreading. Expected slope, corrected by $exp(-f_{dom}t/Q_{crust})$ for effects of crustal attenuation, is shown as a line.
d) *P* event, corrected for spreading by *offset*, Q_{crust}=100.
e) *P* event, Q_{crust}=1000.
f) L_g, corrected for cylindrical spreading by $\sqrt{offset}$, Q_{crust}=100.
g) L_g, Q_{crust}=1000.
Decay occurs as expected for *P* events and L_g at Q_{crust}=100; for L_g at Q_{crust}=1000, decays more quickly than expected, especially for large offsets.




e)
Amplitude Along P Event, Complex Crust, Qs=1000
Amplitude
Offset (km)
f)
Amplitude Along Lg Event, Complex Crust, Qs=100
Amplitude
Offset (km)
g)
Amplitude Along Lg Event, Complex Crust, Qs=1000
Amplitude
Offset (km)

Appendix C: Scattering Model's Numerical Integration Code

Scattering model's numerical integration C++ code, written by Joel Duenow and Dr. Igor Morozov. Code is written within the framework of Dr. Morozov's SIA system. This class performs numerical integration as in Eq. 4. The output seismic trace at the receiver is *trc*.

```
/************************************************************************
      SIA - system for advanced seismic data analysis
      Copyright (c) 1995-2003, I. B. Morozov
************************************************************************/

/*--- RELEVANT CODE FROM syncoda_source.h ---*/

#ifndef MOD_SYNCODA_SOURCE
#define MOD_SYNCODA_SOURCE

#include "../interp/interp.h"
#include "../_lib/_lib.h"


struct SYNCODA_SOURCE : public INTERP_SECTION_TAUP, public SIA_List   // Source function
{
  SYNCODA_SOURCE(GATHER *src,SYNCODA_SOURCE *pred)

      : SIA_List(pred), INTERP_SECTION_TAUP(src,&SIA.root_gather())  {read();}

  TRACE *trace( double r ) { return INTERP_SECTION_TAUP::trace(r); }

  virtual TRACE *trace_xy( double x, double y ) { return NULL; }
      // returns a new trace (or NULL) corresponding to an estimated source signature
      // at (x,y) rotated by appropriate azimuth

  void read_out_hdrs();
      // read repeat group of parameters passed into the output trace
  void map_out_hdrs(TRACE *src, TRACE *tgt);
      //copy output headers from trace src to tgt

protected:

  ARRAY<HEADER_PAIR> out_hdr;     // output headers passed from the source traces
};



struct SYNCODA_POINT_SOURCE : public SYNCODA_SOURCE  // Synthetic Coda Point Source
{
  SYNCODA_POINT_SOURCE(GATHER *src,SYNCODA_SOURCE *pred);

  HEADER_PARAM      x_src, y_src;
      // position of the source

  TRACE *trace_xy( double x, double y );
      // returns a new trace (or NULL) corresponding to
      // an estimated source signature at (x,y)
      // rotated by appropriate azimuth
};
```

```
/*--- RELEVANT CODE FROM syncoda_source.C ---*/


SYNCODA_POINT_SOURCE::SYNCODA_POINT_SOURCE(GATHER *src,SYNCODA_SOURCE *pred)
       : SYNCODA_SOURCE(src,pred)
{
  x_src.read("X",REAL,0);
  y_src.read("Y",REAL,0);

  read_out_hdrs();  // read repeat groups
       // read repeat group of parameters passed into the output trace
}



TRACE* SYNCODA_POINT_SOURCE::trace_xy (double x, double y)
{
  x -= x_src.value();
  y -= y_src.value();
       // relative coordinates of scatter and source

  TRACE* trc = trace(sqrt( x*x + y*y ));
       //determine source signature at scatter point x,y from source point x_src, y_src

  if ( trc )
    trc->rotate(-MATH_atan2(y,x));
       //rotate by appropriate azimuth

  return trc;
}
```

Appendix D: Scattering Model SIA Job Files

This script for Dr. Igor Morozov's SIA system calls the scattering routine to produce output scattered seismic traces.

File runnumA.inc, giving the realization number for the model output.

```
###################################
# param file for Group A
###################################
*define
runnum 1
```

File full100a.job, specifying Q_{crust} value and input and output files, and creating directories, this example for the Quartz-4 model.

```
#################################################################
#   S  Y  N  C  O  D  A
#
#  Synthetic coda generation based on input synthetic section
#
#################################################################
*incl  runnumA.inc
*define

#########################################################
#   I  N  P  U  T     P  A  R  A  M  E  T  E  R  S
#########################################################
qsval  100    #Qcrust value


############################
# input synthetics used:
############################
modelS Morozova-qs{qsval}-Full-LowQSeds-t1572864-s200.sia
model  Morozova-qs{qsval}-Full-LowQSeds-t1572864-s200.sia
job    MorozovaP-qs{qsval}-{runnum}
outdir /transient6/jnduenow/scatter/MorozovaP{runnum}/{job}
dir1   /transient6/jnduenow/scatter/MorozovaP{runnum}

#######################################
# copy job files to output directory
#######################################
*unix  edit
mkdir {dir1}
*unix  edit
mkdir {outdir}

#######################################
#  include the main processing file
#######################################
*incl  scatter.inc
```

File scatter.inc, containing main scattering model processing parameters, this example for the *P* coda.

```
#################################################################
#   S  Y  N  C  O  D  A
#
#  Synthetic coda generation based on input synthetic section
#
#  Geometry:
#     One receiver at 2900 km offset
#
#         Sc  Sc  Sc  Sc     S  = source
#         Sc  Sc  Sc  Sc     R  = receiver
#         Sc  Sc  Sc  Sc     Sc = scatterers
#    S    Sc  Sc RSc  Sc
#         Sc  Sc  Sc  Sc
#         Sc  Sc  Sc  Sc
#         Sc  Sc  Sc  Sc
#
#################################################################
*define

#########################################################
#   I  N  P  U  T     P  A  R  A  M  E  T  E  R  S
#########################################################

####################
#mute parameters
####################
srcT    .11114
srcB    .28571
greenT  .11114
greenB  .28571

###################################
#scattering region parameters
###################################
scxmin  2.3e6           #Minimum X scatterpoint coordinate (m)
scymin  9.0e5           #Minimum Y scatterpoint coordinate (m)
scadd   1.2e6           #Square sizes have this length (m)
scnbr   5000            #total nbr of scatterpoints
amp     1               #scattering potential

###################################
#input synthetic section model
###################################
tstrt   0               #trace start time (ms)
trlenO  1300e3          #output trace length (ms)
trlen   1572864         #input trace length (ms)

#########################################
# Synthetic section names
#########################################
#input synthetic section filename
gffile  /transient6/jnduenow/synthetics/{model}
srcfile /transient6/jnduenow/synthetics/{modelS}

###################################
#source and receiver locations
###################################
#source location
X_SRC   0       #X coordinate of source (m)
Y_SRC   1500000 #Y coordinate of source (m)
```

```
#near reciever parameters, used to generate line of recievers
R_X     2900000 #near receiver x-coordinate (m)
R_Y     1500000 #near receiver y-coordinate (m)
R_SAMP  200             #receiver trace sample interval (ms)
R_TRLEN {trlenO}        #receiver trace length (ms)
R_START {R_X}           #starting receiver x position (m)
R_INC   10000           #receiver spacing x increment (m)
R_NBR   1               #number of receiver increments

###############################################
#irange list for use in interpolation routine
###############################################
DX      50000           #width used in interpolation (m)
PMIN    0.08            #minimum slowness (s/km)
PMAX    0.3             #maximum slowness (s/km)
FMIN    0.1             #min freq. (Hz)
FMAX    -1              #max freq. (-1 is the Nyquist)

####################################
#output filename parameters
####################################
trcmtv  {outdir}/SC-{job}-trc.mtv
outfile {outdir}/SC-{job}.sia      #SIA output file

#####################################################################
#####################################################################

################################
# write over previous files
################################
*setup
noapp

############################################
# read and store input synthetic section
############################################

################################
# Read Green's function file
################################
*call   dskrd   gffile

*if
list    comp                            #component 1, vertical
1

#apply front and back mute
*call   hdrmath
set     one     float   0       #top mute starts at 0 ms, increasing with
greenT  abs             x       #slope greenT specified above
set     two     float   10000   #bottom mute starts at 10000 ms, increasing
greenB  abs             x       #with slope greenB
*call   trcmath
mute    x                                       one
*call   trcmath
mute    x       0       3.5e6   two     #mute applied to all traces

*call   store   greens-fun              #store the GF traces

*endif  #for comp number

*end    #traces disappear here since they are stored, thus no longer needed
```

```
################################
# Read source file
################################
*call    dskrd    srcfile

*if
list     comp
1

#apply front and back mute
*call    hdrmath
set      one      float    0        #top mute starts at 0 ms, increasing with
srcT     abs               x        #slope srcT specified above
set      two      float    10000    #top mute starts at 10000 ms, increasing
srcB     abs               x        #with slope srcB
*call    trcmath
mute     x                                      one
*call    trcmath
mute     x        0        3.5e6    two      #mute applied to all traces

*call    store    greens-fun-src

*endif   #for comp number

*end             #close read of source file

################################################################
# generate EMPTY trace at receiver
################################################################
*call    generate R_SAMP   R_TRLEN
hrange   x        float    R_START  R_INC    R_NBR


#################################
# scatter point definition
#################################
*call    readtab  map      scattermap
count    integer  1
xsc      float
ysc      float
AMPL     float
argrang  count    1        1        scnbr
const
xsc      0
ysc      0
AMPL     {amp}

*call    tabmath  edit              scattermap
set      xsc      float    scxmin           #Monte-Carlo sampling
         rand-u            scadd            #within specified region

set      ysc      float    scymin           #Monte-Carlo sampling
         rand-u            scadd            #within specified region
```

```
################################################################
#    S  I  M  U  L  A  T  I  O  N
################################################################

################################
# generate synthetic coda
################################
*call    syncoda x       R_Y                                     #position of receivers
xsc      ysc     AMPL    0       1       0       scattermap      #position of scatterers
psource X_SRC    Y_SRC   x       greens-fun-src  #pos. of src, dist from src to scatt
greenf           x       greens-fun                              #dist from scatterers
to receivers
irange   DX      PMIN    PMAX    FMIN    FMAX


*info    eof     time

################################################
# write trace to SIA file for stacking later
################################################
*call   write           outfile

################################################
# change trace to instantaneous amplitude
################################################
*call    trcmath
iampl    x

################################################
# call plotmtv to plot the trace envelope
################################################
*call    plotmtv trcmtv
title    Trc at {R_START}, {job}
label-x Time (s)
label-y Amplitude
range-x 0       trlen0
scale   1e3     1       1
xyratio 1
line    solid   1
data    0       trlen0

*end
*end
```

Appendix E: τ-p Interpolation Routine

This is a C++ class performing τ-p interpolation of seismic traces at the selected interval, written by Joel Duenow within the framework of Dr. Morozov's SIA system. This routine was first implemented as an independent tool and tested separately from the numerical integration routine. It is used as part of the numerical integration routine for interpolating traces at appropriate source-scatterer and scatterer-receiver distances. The routine takes a stored seismic section as input and resamples the offset coordinate over the selected offset range and trace offset interval using the chosen number of traces with which to perform the interpolation. Slowness range and frequency range are also parameters.

For this work, a single interpolated trace at the required source-scatterer or scatterer-receiver distance was the offset range. The interpolation range was specified at 50 km so that at least 8 traces were used to interpolate a trace. The frequency range chosen for the interpolation was 0.1 Hz to the Nyquist frequency of 2.5 Hz. The slowness range used was 0.08 to 0.3 s/km.

```
/*********************************************************************
       SIA - system for advanced seismic data analysis
       Copyright (c) 1995-2003, I. B. Morozov
*********************************************************************/

/*--- RELEVANT CODE FROM interp.h ---*/

#ifndef MOD_INTERP
#define MOD_INTERP

#include "sia_module.C.h"
#include "sia_fft.h"


struct INTERP_SECTION      // a 2D time-offset section with trace interpolation
{
  INTERP_SECTION(GATHER *src=NULL, GATHER *tgt=&SIA.root_gather())
       : source(src),target(tgt),_in_spectra(FALSE),_to_spectra(TRUE) {}
       // constructor

  boolean check_sample_interval( double &si );
       // returns OK if sample intervals in the source gather are the same
       // and equal to 'si'
       // if si<0. on entry, starts from the sample interval
       // in its first trace and returns that interval in 'si'

  boolean read() { return OK; }
       //reads the stored section
       // no params required

  TRACE *first_trace() { return source ? source->first_trace() : NULL; }

  virtual TRACE *trace( double r );
       // returns a new trace object (or NULL) corresponding to
       // an interpolated trace at distance 'r' from the source

protected:

  GATHER     *source,      // pointer to the input trace gather
             *target;      // pointer to the output trace gather
  AHEADER    offset;       // input trace header containing the offset value

  boolean    _to_spectra,  // if TRUE, the section will be transformed to
```

```
                        // spectra upon initialization
            _in_spectra; // TRUE if the section is already in spectral form
};


struct INTERP_SECTION_TAUP : public INTERP_SECTION
      // a 2D time-offset section with trace interpolation
{
  INTERP_SECTION_TAUP()
      : INTERP_SECTION(SIA_input(),SIA_output()),f_domain(FALSE) {}
      //constructor

  INTERP_SECTION_TAUP(GATHER *src, GATHER *tgt=SIA_output(), Boolean
     fdomain=FALSE)
      : INTERP_SECTION(src,tgt),f_domain(fdomain) {}
      //constructor

  HEADER_PARAM      min_p, max_p,       // ray parameter range
                    min_f, max_f,       // frequency range
                    max_dx;             // width of the interpolation range

  virtual TRACE *trace( double r );
      // returns a new trace object (or NULL) corresponding to
      // an interpolated trace at distance 'r' from the source

  boolean read();
      // read params from PRANGE and FRANGE lists
      // return OK if read correctly

  void to_spectra();
      // converts all the traces at input into complex spectra

private:

  boolean    f_domain;
      // if TRUE, output frequency-domain traces
};



/*************************************************************************
      SIA - system for advanced seismic data analysis
      Copyright (c) 1995-2002, I. B. Morozov
*************************************************************************/

/*--- RELEVANT PART OF interp_section.C ---*/

/*--- routine that performs the actual interpolation ---*/


TRACE *INTERP_SECTION_TAUP::trace( double x )
{
  TRACE      *t = first_trace();
  if ( t )
  {
    /*--- read in job input parameters  ---*/
    double   dx  = max_dx.value(t),
             pmin = min_p.value(t),
             pmax = max_p.value(t),
             fmin = min_f.value(t),
             fmax = max_f.value(t);

    /*--- convert frequencies to kHz from Hz ---*/
    fmin = 0.001 * fmin;
    fmax = 0.001 * fmax;
```

```
/*--- default of fmax is the Nyquist frequency ---*/
if (fmax < 0)
    fmax = t->sample_interval()*t->num_samples();

/*--- variables used to find offset value in current trace ---*/
double   xoff;
int      iter = 0;
int      numNearTrace = 0;
double   nearTraceDist = HUGE;

struct   indexTrace
{
   double dist;
   TRACE *nearT;
};

indexTrace *nearTrace = ALLOC_TEMP(source->num_traces(),indexTrace);

/*--- variables used in computation of kernel ---*/
double   dp     = 1/(fmax*dx*spw);               //slowness inc. delta p
int      N      = int(ceil((pmax-pmin)/dp));     //total number of dp inc.
double   da     = 0;                             //a increment

/*--- find trace t that is nearest to x ---*/
for ( ; t; t = t->next_trace() )
{
  nearTrace[iter].dist  = fabs(x-offset.value(t));
  nearTrace[iter].nearT = t;
  iter++;
}

for (int i=0; i<iter; i++)
{
   if (nearTrace[i].dist < nearTraceDist)
   {
      numNearTrace  = i;
      nearTraceDist = nearTrace[i].dist;
   }
}

/*--- move pointer to the near trace ---*/
t = nearTrace[numNearTrace].nearT;

/*-- add an empty trace to the output with parameters of near trace ---*/
TRACE    *t_out  = target->add_ensemble()->add_trace(t);

/*--- set parameters for the new trace ---*/
t_out->clear_data();         // set trace values to 0

t_out->copy_header(t);
t_out->set_time_start(t->time_start());
offset.set_value(t_out,x);

/*--- calculate for each component (e.g. x,y,z) ---*/
for (int recNbr=0; recNbr<t_out->num_records(); recNbr++)
{
   COMPLEX      *c_cout = (COMPLEX*)t_out->data_start(recNbr);

   /*--- stack traces near 'x', weighted by kernel, into t_out ---*/
   for ( t = source->first_trace(); t; t = t->next_trace() )
   {
     xoff= offset.value(t);

     /*--- if trace in range then include in interpolation ---*/
     if ( fabs(xoff - x) < dx )
     {
```

```
          // phase shift increment between the different frequency points
          double dphase = TWO_PI * t->sample_interval() *
                          ( t->time_start() - t_out->time_start() );

            /*--- set kernel values, weight trace samples ---*/
            da          = TWO_PI * (x-xoff) * t->sample_interval();
           COMPLEX *c = (COMPLEX*)t->data_start(recNbr);

            /*--- for kernel at 0 frequency ---*/
            c_cout[0] += c[0] * N;

            /*--- calculation of frequency-dependent kernel, weighting ---*/
           if (fabs(x-xoff) < 0.001)   //special case at x-xoff= 0; kernel=N
            {
              for (int i=1; i<t->num_samples(); i++)
                c_cout[i] += c[i] * N * cuexp(dphase*i);
            }
            else
            {
              for (int i=1; i<t->num_samples(); i++)
                   c_cout[i] += c[i] * cuexp((dphase+pmin*da)*i)*(C_1-
                   cuexp(N*dp*da*i))/(C_1-cuexp(dp*da*i));
            }//end else
          }//end if fabs()
         } //end for loop of stacking and weighting traces
      } //end for loop of the different components

      if ( f_domain )        // output frequency-domain traces
        return t_out;

      t_out->convert(DATA);  //convert interp. trace back into time domain

      return t_out;          //return the trace

    }   //end if(t)

    return NULL;             //no data section if there was no input t

}//close function
```

Appendix F: Matlab Code to Measure Coda Envelope Slopes

This is Matlab code to measure *P* and L_g coda envelope slopes of scattered traces output by the scattering model. Code computes, plots, and writes to file Q_{crust} values and their corresponding measured L1 fit Q_{coda}/f values. The L1 routine is from Aster *et. al.* (2002). Output files are read by code in appendix H.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Best-fit lines and plotting of synthetic traces
%     - Creates plots with line and equation
%     - Fits P events from all traces as one; same for Lg
%
%  1)  Clear output files
%  2)  Read all traces (amp. vs. time) into arrays of P and Lg data
%  3)  Fit P or Lg event with LSQR line
%  4)  Fit P or Lg event with L1 line
%  5)  Write out resulting slopes for both LSQR and L1
%  6)  Plot entire trace and P or Lg coda with best-fit lines
%         for each input Qcrust value (50:50:1000)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% initializations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
model   = 'Combined';           %velocity model
wvtp    = 'Full';           %waveform
fnum    = '2';              %frequency nbr (1,2,3)
fold    = [model,wvtp];
spcChar = 'P';               %to give diff. output name

runnum=[1:1:16];

qvals = [1000:-50:50];   %the input Qcrust value files

if (strcmp(fold,'CombinedFull'))
    %time window
    Pstart=390; Pend=470; LgStart=550; LgEnd=551;%LgStart=550; LgEnd=650;
    %plot label locations
    PLabHt=2e-6; LgLabHt=2e-6; RgLabHt=1e-7;
    pLoc=30; lgLoc=600; rgLoc=1400;
    plot2loc=430; plot2LabHt=5e-7; plot3loc=590; plot3LabHt=5e-7;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'LenaFull'))
    %time window
    Pstart=435; Pend=600; LgStart=880; LgEnd=950; %LgStart=720; LgEnd=721;
    %plot label locations
    PLabHt=1e-5; LgLabHt=1e-5; RgLabHt=1e-7;
    pLoc=340; lgLoc=710; rgLoc=1300;
    plot2loc=540; plot2LabHt=4e-6; plot3loc=760; plot3LabHt=4e-6;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'IASPFull'))
    %time window
    Pstart=400; Pend=480; LgStart=700; LgEnd=701;
    %plot label locations
    PLabHt=1e-5; LgLabHt=2e-6; RgLabHt=1e-7;
    pLoc=420; lgLoc=780; rgLoc=1400;
    plot2loc=540; plot2LabHt=1e-9; plot3loc=760; plot3LabHt=1e-9;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'ScottFull'))
```

```
    %time window
    Pstart=510; Pend=575; LgStart=1040; LgEnd=1085;%LgStart=660; LgEnd=800;
    %plot label locations
    PLabHt=5e-6; LgLabHt=1e-8; RgLabHt=1e-7;
    pLoc=420; lgLoc=710; rgLoc=1400;
    plot2loc=535; plot2LabHt=6e-6; plot3loc=710; plot3LabHt=2e-8;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
else
    fprintf('Error in folder name ',fold,' \n');
end


%specify sample range for P and Lg measurement
PstartS  = Pstart * 5;      %in terms of samples
PendS    = Pend * 5;
LgStartS = LgStart * 5;
LgEndS   = LgEnd * 5;
PRange  = PstartS:1:PendS;
LgRange = LgStartS:1:LgEndS;

%establish output file names
outdir='/transient6/jnduenow/dataAnalysis/combi/';
locfilename=[outdir,fold,fnum,spcChar,'/LSQRqf.txt'];
locfilenameL1=[outdir,fold,fnum,spcChar,'/L1qf.txt'];

%clear the output files
locfile=fopen(locfilename,'w');
fclose(locfile);
locfileL1=fopen(locfilenameL1,'w');
fclose(locfileL1);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Perform plotting/fitting for each Q value
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for (iter=qvals)

    fprintf('Q value:  %d\n',iter);
    clear datavec datavecSub new newSub dataPvec dataPvecSub dataLgvec dataLgvecSub pVec
    LgVec;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % loop over runnum to read data
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for(iteration=runnum)
        fprintf('Run number:  %d\n',iteration);

        %set text names for input files
        if (iteration>=10)
            runnumS=num2str(iteration,'%2d');
        else
            runnumS=num2str(iteration,'%1d');
        end
        folderS =[fold,runnumS];   %folder name
        qnbrS=num2str(iter,'%4d');
        dirname=['stack-',fold,'-qs',qnbrS,'-',spcChar];
        mtvname=['stack-',fold,'-qs',qnbrS,'-',spcChar,'-',runnumS,'.mtv'];
        currentdir=[srcdir,dirname,'/'];
        datafile=[currentdir,mtvname];

        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % read the data from plotmtv file
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        data=fopen(datafile);
        if (data == -1)
```

```
        continue
    else
        junk=fgets(data);

        while junk(1)=='%' | junk(1)=='$'
            junk=fgets(data);
        end
        datavecSub = fscanf(data,'%e',[2 10000])';
        fclose(data);
        newSub=[1e-3.*datavecSub(:,1) log(datavecSub(:,2))];

        dtest=exist('datavec');
        if (dtest>0)
            datavec=[datavec; datavecSub];
            new=[new; newSub];
        else
            datavec=datavecSub;
            new=newSub;
        end

        newPSub=newSub(PRange,:);
        dataPvecSub=datavecSub(PRange,:);
        newLgSub=newSub(LgRange,:);
        dataLgvecSub=datavecSub(LgRange,:);
        atest=exist('pVec');
        if (atest>0)
            dataPvec=[dataPvec; dataPvecSub];
            pVec=[pVec; newPSub];
            dataLgvec=[dataLgvec; dataLgvecSub];
            LgVec=[LgVec; newLgSub];
        else
            pVec=newPSub;
            dataPvec=dataPvecSub;
            LgVec=newLgSub;
            dataLgvec=dataLgvecSub;
        end
    end

end %end for loop over runnum values

pVec=sortrows(pVec,[1 2]);
LgVec=sortrows(LgVec,[1 2]);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% set plot vars
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
plot1fname=[outdir,fold,fnum,spcChar,'/fulltrace',qnbrS,'.fig'];
plot2fname=[outdir,fold,fnum,spcChar,'/bestfitPL1',qnbrS,'.fig'];
plot3fname=[outdir,fold,fnum,spcChar,'/bestfitLgL1',qnbrS,'.fig'];
plot1title=['Crustal Qs=',qnbrS,', ',model,' Model, ',wvtp,' Waveform, Full Trace'];
plot2title=['Crustal Qs=',qnbrS,', ',model,' Model, ',wvtp,' Waveform, P, L1 fit'];
plot3title=['Crustal Qs=',qnbrS,', ',model,' Model, ',wvtp,' Waveform, Lg, L1 fit'];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make LSQR best fit lines
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%P best-fit line, all runnum runs
ppoly=polyfit(pVec(:,1),pVec(:,2),1);
Pline=(1e-3.*dataPvecSub(:,1)).*ppoly(1) + ppoly(2);
QfP = -pi/ppoly(1);
textlineP=['Measured Q/f = ',num2str(QfP,'%4.0f')];

%Lg best-fit line
lgpoly=polyfit(LgVec(:,1),LgVec(:,2),1);
Lgline=(1e-3.*dataLgvecSub(:,1)).*lgpoly(1) + lgpoly(2);
```

```
QfLg = -pi/lgpoly(1);
textlineLg=['Measured Q/f = ',num2str(QfLg,'%4.0f')];


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make L1 best fit lines
% algorithm is from inverse theory text, Ch 2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rdelta=.002;     %determines how long to iterate
counterP=0;      %count while-loop iterations
%%%% L1 line for P slopes %%%%
%make matrix with ones in 1st col., x-values in 2nd col.
GwP = [ones(length(pVec),1),pVec(:,1)];
%1 and 2 are switched compared to polyfit; use as initial values
mP(1)=ppoly(2);
mP(2)=ppoly(1);
%residual vector rprevP
rprevP = abs ( pVec(:,2) - GwP*mP' );
%make into a matrix, values on diagonal
RP = spdiags((1./rprevP),0,length(pVec),length(pVec));  %sparse
%calculate new slope and intercept
mP = (inv(GwP'*RP*GwP)*GwP'*RP*pVec(:,2))';
%and the new residuals that result
rcurrP = abs ( pVec(:,2) - GwP*mP' );
while ( ( norm((rcurrP - rprevP),1) /norm (rprevP,1) )  > rdelta)
    rprevP = rcurrP;    %set rprevP to the old rcurrP
    RP = spdiags((1./rcurrP),0,length(pVec),length(pVec));  %sparse
    mP = (inv(GwP'*RP*GwP)*GwP'*RP*pVec(:,2))';
    rcurrP = abs ( pVec(:,2) - GwP*mP' );
    counterP=counterP+1;
end
fprintf('Total iterations for L1 P: %d\n',counterP);
%calculate the L1 best fit line
PlineL1=mP(2).*(1e-3.*dataPvecSub(:,1)) + mP(1);
QfPL1 = -pi/mP(2);
textlinePL1=['Measured Q/f = ',num2str(QfPL1,'%4.0f')];

%%%% L1 line for Lg slopes %%%%
counterLg=0;
GwLg = [ones(length(LgVec),1),LgVec(:,1)];
mLg(1)=lgpoly(2);
mLg(2)=lgpoly(1);
rprevLg = abs ( LgVec(:,2) - GwLg*mLg' );
RLg = spdiags((1./rprevLg),0,length(LgVec),length(LgVec));  %sparse
mLg = (inv(GwLg'*RLg*GwLg)*GwLg'*RLg*LgVec(:,2))';
rcurrLg = abs ( LgVec(:,2) - GwLg*mLg' );
while ( ( norm((rcurrLg - rprevLg),1) /norm (rprevLg,1) )  > rdelta)
    rprevLg = rcurrLg;    %set rprevP to the old rcurrP
    RLg = spdiags((1./rcurrLg),0,length(LgVec),length(LgVec));  %sparse
    mLg = (inv(GwLg'*RLg*GwLg)*GwLg'*RLg*LgVec(:,2))';
    rcurrLg = abs ( LgVec(:,2) - GwLg*mLg' );
    counterLg=counterLg+1;
end
fprintf('Total iterations for L1 Lg: %d\n',counterLg);
%calculate the L1 best fit line
LglineL1=mLg(2).*(1e-3.*dataLgvecSub(:,1)) + mLg(1);
QfLgL1 = -pi/mLg(2);
textlineLgL1=['Measured Q/f = ',num2str(QfLgL1,'%4.0f')];


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% write Q/f value obtained to file
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%write this line with Q/f info to files
qline=[qnbrS,' %4.0f ',qnbrS,' %4.0f\n'];
```

```
    %write LSQR Q/f values to a local file
    locfile=fopen(locfilename,'a');
    fprintf(locfile,qline,QfP,QfLg);
    fclose(locfile);

    %write L1 Q/f values to file
    locfileL1=fopen(locfilenameL1,'a');
    fprintf(locfileL1,qline,QfPL1,QfLgL1);
    fclose(locfileL1);

    %%%%%%%%%%%%%%%%%%%%%%%%%
    % make plots
    %%%%%%%%%%%%%%%%%%%%%%%%%
    %plot entire trace with events labeled
    figure(1)
    plot1 = semilogy(1e-3.*datavec(:,1),datavec(:,2),'.k');
    title(plot1title)
    xlabel('Time (s)')
    ylabel('Log Amplitude')
    text(pLoc,PLabHt,'P Coda','FontSize',16)
    text(lgLoc,LgLabHt,'Lg Coda','FontSize',16)
    text(rgLoc,RgLabHt,'Rg Coda','FontSize',16)
    set(plot1,'LineWidth',2)
    %save the first and last plots
    if (qvals==100 | qvals==1000)
        saveas(gcf,plot1fname);
    end

    %plot P with L1 best-fit line
    figure(2)
    plot2 = semilogy(1e-3.*dataPvec(:,1),dataPvec(:,2),'.k',
      1e-3.*dataPvecSub(:,1),exp(PlineL1),'-r');
    title(plot2title)
    xlabel('Time (s)')
    ylabel('Amplitude')
    text(plot2loc,plot2LabHt,textlinePL1,'FontSize',18)
    set(plot2,'LineWidth',2)
    %save first and last plots
    %if (qvals==100 | qvals==1000)
        saveas(gcf,plot2fname);
        %end

    %plot Lg with best-fit line
    figure(3)
    plot3 = semilogy(1e-3.*dataLgvec(:,1),dataLgvec(:,2),'.k',
      1e-3.*dataLgvecSub(:,1),exp(LglineL1),'-r');
    title(plot3title)
    xlabel('Time (s)')
    ylabel('Amplitude')
    text(plot3loc,plot3LabHt,textlineLgL1,'FontSize',18)
    set(plot3,'LineWidth',2)
    if (qvals==100 | qvals==1000)
        saveas(gcf,plot3fname);
    end

end %end for loop over Q values
```

Appendix G: Matlab Code to Calculate Error Bounds

This is Matlab code to calculate error bounds for slope measurements. Code reads in 50% of total trace pool, selecting traces at random. This is performed 10 times; minimum and maximum values of slope measured are written to file and used as error bounds.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Best-fit lines and plotting of synthetic traces
%     - Creates plots with line and equation
%     - Fits P events from all traces as one; same for Lg
%
%  1)  Clear output files
%  2)  Read all traces (amp. vs. time) into arrays of P and Lg data
%  3)  Fit P or Lg event with LSQR line
%  4)  Fit P or Lg event with L1 line
%  5)  Write out resulting slopes for L1 fit at each Qcrust value
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% initializations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
model   = 'Combined';   %velocity model
wvtp    = 'Full';       %waveform
fnum    = '1';          %freq nbr
fold    = [model,wvtp];
spcChar = 'P';      %use to give a different output name



trcPct  = 50;       %use 50% of traces in error bar sums
nbrRep  = 10;       %use this many repetitions of
                    %randomly-chosen traces to determine error bars
runRng  = 1:1:16;   %range of input file numbers

qvals = [1000:-50:50];   %the input Qcrust value files

if (strcmp(fold,'CombinedFull'))
    Pstart=390; Pend=470; LgStart=550; LgEnd=551;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'LenaFull'))
    Pstart=435; Pend=600; LgStart=880; LgEnd=950;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'IASPFull'))
    Pstart=400; Pend=480; LgStart=700; LgEnd=701;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
elseif (strcmp(fold,'ScottFull'))
    Pstart=510; Pend=575; LgStart=1040; LgEnd=1085;
    srcdir=['/transient6/jnduenow/stack/',model,wvtp,fnum,'/'];
else
    fprintf('Error in folder name\n');
end

%specify sample range for P and Lg measurement
PstartS  = Pstart * 5;      %in terms of samples
PendS    = Pend * 5;
LgStartS = LgStart * 5;
LgEndS   = LgEnd * 5;
PRange  = PstartS:1:PendS;
LgRange = LgStartS:1:LgEndS;

%directories
```

```
outdir='/transient6/jnduenow/dataAnalysis/combi/';

%establish output file names
locfilenameL1=[outdir,fold,fnum,spcChar,'/L1qfErrBar.txt'];

%clear the output files
locfileL1=fopen(locfilenameL1,'w');
fclose(locfileL1);

%compute number of traces to use in determining error bars
range = (max(runRng)-min(runRng) + 1);
trcTot = floor( trcPct/100 * range );
runnum=zeros(nbrRep,trcTot);
%determine which ones to use
for (x=1:nbrRep)
    trcNbrs = floor ( runRng(1) + range .* rand(trcTot,1) )';
    runnum(x,:)=trcNbrs;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Perform plotting/fitting for each Q value
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for (iter=qvals)

    clear datavec datavecSub new newSub dataPvec dataPvecSub dataLgvec dataLgvecSub pVec
LgVec;

    fprintf('Q value:  %d\n',iter);

    for(erbIter=1:length(runnum(:,1)));
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % loop over runnum to read data
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        for(iteration=runnum(erbIter,:))
            fprintf('Run number:  %d\n',iteration);

            %set text names for input files
            if (iteration>=10)
                runnumS=num2str(iteration,'%2d');
            else
                runnumS=num2str(iteration,'%1d');
            end
            folderS =[fold,runnumS];   %folder name
            qnbrS=num2str(iter,'%4d');
            dirname=['stack-',fold,'-qs',qnbrS,'-',spcChar];
            mtvname=['stack-',fold,'-qs',qnbrS,'-'spcChar,'-',runnumS,'.mtv'];
            currentdir=[srcdir,dirname,'/'];
            datafile=[currentdir,mtvname];



            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            % read the data from plotmtv file
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            data=fopen(datafile);
            if (data == -1)
                continue
            else
                junk=fgets(data);
                while junk(1)=='%' | junk(1)=='$'
                    junk=fgets(data);
                end
                datavecSub = fscanf(data,'%e',[2 10000])';
                fclose(data);
                newSub=[1e-3.*datavecSub(:,1) log(datavecSub(:,2))];
```

```
        dtest=exist('datavec');
        if (dtest>0)
            datavec=[datavec; datavecSub];
            new=[new; newSub];
        else
            datavec=datavecSub;
            new=newSub;
        end

        newPSub=newSub(PRange,:);
        dataPvecSub=datavecSub(PRange,:);
        newLgSub=newSub(LgRange,:);
        dataLgvecSub=datavecSub(LgRange,:);
        atest=exist('pVec');
        if (atest>0)
            dataPvec=[dataPvec; dataPvecSub];
            pVec=[pVec; newPSub];
            dataLgvec=[dataLgvec; dataLgvecSub];
            LgVec=[LgVec; newLgSub];
        else
            pVec=newPSub;
            dataPvec=dataPvecSub;
            LgVec=newLgSub;
            dataLgvec=dataLgvecSub;
        end
    end
end %end for loop over runnum values


pVec=sortrows(pVec,[1 2]);
LgVec=sortrows(LgVec,[1 2]);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% determine error bars
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make LSQR best fit lines
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%P best-fit line, all runnum runs
ppoly=polyfit(pVec(:,1),pVec(:,2),1);
Pline=(1e-3.*dataPvecSub(:,1)).*ppoly(1) + ppoly(2);

%Lg best-fit line
lgpoly=polyfit(LgVec(:,1),LgVec(:,2),1);
Lgline=(1e-3.*dataLgvecSub(:,1)).*lgpoly(1) + lgpoly(2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make L1 best fit lines
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rdelta=.002;     %determines how long to iterate
counterP=0;      %count while-loop iterations
GwP = [ones(length(pVec),1),pVec(:,1)];
mP(1)=ppoly(2);
mP(2)=ppoly(1);
rprevP = abs ( pVec(:,2) - GwP*mP' );
%make into a matrix, values on diagonal
RP = spdiags((1./rprevP),0,length(pVec),length(pVec));  %sparse
%calculate new slope and intercept
mP = (inv(GwP'*RP*GwP)*GwP'*RP*pVec(:,2))';
%and the new residuals that result
rcurrP = abs ( pVec(:,2) - GwP*mP' );

while ( ( norm((rcurrP - rprevP),1) /norm (rprevP,1) )  > rdelta)
```

```
            rprevP = rcurrP;    %set rprevP to the old rcurrP
            RP = spdiags((1./rcurrP),0,length(pVec),length(pVec));  %sparse
            mP = (inv(GwP'*RP*GwP)*GwP'*RP*pVec(:,2))';
            rcurrP = abs ( pVec(:,2) - GwP*mP' );
            counterP=counterP+1;
        end
        fprintf('Total iterations for L1 P, Error Bar: %d\n',counterP);
        %calculate the L1 best fit line
        PlineL1=mP(2).*(1e-3.*dataPvecSub(:,1)) + mP(1);
        QfPL1err(erbIter) = -pi/mP(2);


        %%%% L1 line for Lg slopes %%%%
        counterLg=0;
        GwLg = [ones(length(LgVec),1),LgVec(:,1)];
        mLg(1)=lgpoly(2);
        mLg(2)=lgpoly(1);
        rprevLg = abs ( LgVec(:,2) - GwLg*mLg' );
        RLg = spdiags((1./rprevLg),0,length(LgVec),length(LgVec));  %sparse
        mLg = (inv(GwLg'*RLg*GwLg)*GwLg'*RLg*LgVec(:,2))';
        rcurrLg = abs ( LgVec(:,2) - GwLg*mLg' );
        while ( ( norm((rcurrLg - rprevLg),1) /norm (rprevLg,1) )  > rdelta)
            rprevLg = rcurrLg;    %set rprevP to the old rcurrP
            RLg = spdiags((1./rcurrLg),0,length(LgVec),length(LgVec));  %sparse
            mLg = (inv(GwLg'*RLg*GwLg)*GwLg'*RLg*LgVec(:,2))';
            rcurrLg = abs ( LgVec(:,2) - GwLg*mLg' );
            counterLg=counterLg+1;
        end
        fprintf('Total iterations for L1 Lg, Error Bar: %d\n',counterLg);
        %calculate the L1 best fit line
        LglineL1=mLg(2).*(1e-3.*dataLgvecSub(:,1)) + mLg(1);
        QfLgL1err(erbIter) = -pi/mLg(2);

    end  %end of for loop over length of runnum

    QfPL1min = min(QfPL1err);
    QfPL1max = max(QfPL1err);
    QfLgL1min = min(QfLgL1err);
    QfLgL1max = max(QfLgL1err);


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % write Q/f value obtained to file
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %write this line with Q/f info to files
    qline=[qnbrS,' %4.0f ',' %4.0f ',qnbrS,' %4.0f ',' %4.0f\n'];

    %write L1 Q/f values to file
    locfileL1=fopen(locfilenameL1,'a');
    fprintf(locfileL1,qline,QfPL1min,QfPL1max,QfLgL1min,QfLgL1max);
    fclose(locfileL1);

end %end for loop over Q values
```

Appendix H: Matlab Code to Determine Q_{coda}-Q_{crust} Relation

This is Matlab code to plot $\log\left(\frac{f}{Q_{coda}}\right)$ vs. $\log Q_{crust}$ and determine values for parameters κ and γ. The code fits an L1 line to the data; γ is the slope, $\log\kappa$ is the intercept.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Best-fit line to determine parameters
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear;
close all;

model   = 'Combined';
wvf     = 'Full';
nbr     = '1';  %freq range nbr
wavetyp = 'P';   %or 'Lg'

if (strcmp(nbr,'1'))
    titleAdd=', Peak Freq. 0.3 Hz';
elseif (strcmp(nbr,'2'))
    titleAdd=', Peak Freq. 0.7 Hz';
elseif (strcmp(nbr,'3'))
    titleAdd=', Peak Freq. 1.3 Hz';
else
    fprintf('Error in freq. range specification');
end


folder = [model,wvf,nbr,wavetyp];
filenm  = [folder,'/L1qf.txt'];
fileErr = [folder,'/L1qfErrBar.txt'];
srcdir = '/transient6/jnduenow/dataAnalysis/combi/';

%%%%%%%%%%%%%%%%%%%%%%%%%
% open the input file
%%%%%%%%%%%%%%%%%%%%%%%%%
file=[srcdir,filenm];
data=fopen(file);
datavec = fscanf(data,'%e',[4 300])';
%datavec = fscanf(data,'%e',[2 300])';       %for 1 set only
fclose(data);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% open the error bar input file
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
file=[srcdir,fileErr];
dataErr=fopen(file);
errvec = fscanf(dataErr,'%e',[6 300])';
%errvec = fscanf(dataErr,'%e',[3 300])';    %for 1 set only
fclose(dataErr);


%input files
if (strcmp(wavetyp,'P'))
    pdata(:,1) = log (datavec(:,1));        % Qcrust
    pdata(:,2) = log (1./datavec(:,2));     %1 / measured Qcoda/f
    txtPosA=-6.75;
    txtPosB=-7.05;
    perrrng(:,1) = log(1./errvec(:,2));     % 1 / low Qcoda/f
    perrrng(:,2) = log(1./errvec(:,3));     % 1 / high Qcoda/f
```

```
    perrbarL = pdata(:,2)-perrrng(:,2);     % the lower bar, but high Q since inverted
    perrbarU = perrrng(:,1)-pdata(:,2);     % the upper bar, but low Q
end

if (strcmp(wavetyp,'Lg'))
    pdata(:,1) = log (datavec(:,3));        %input Qcrust
    pdata(:,2) = log (1./datavec(:,4));     %1 / measured Qcoda/f
    txtPosA=-4.5;
    txtPosB=-4.7;
    perrrng(:,1) = log(1./errvec(:,5));     % 1 / low Qcoda/f
    perrrng(:,2) = log(1./errvec(:,6));     % 1 / high Qcoda/f
    perrbarL = pdata(:,2)-perrrng(:,2);     % the lower bar, but high Q since inverted
    perrbarU = perrrng(:,1)-pdata(:,2);     % the upper bar, but low Qcoda
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make best fit line, LSQR
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ppoly=polyfit(pdata(:,1),pdata(:,2),1);
pline=pdata(:,1).*ppoly(1) + ppoly(2);
fp = 1/ppoly(1);


%figure(2)
%hold on
%h=plot(pdata(:,1),pdata(:,2),'.k',pdata(:,1),pline,'b');
%title('lsqr fit')
%xlabel('Log Qcrust')
%ylabel('Log Measured f/Qcoda')
%text(4.7,txtPosA,textlinePL1,'FontSize',18)
%text(4.7,txtPosB,textlinePL1b,'FontSize',18)
%set(h,'MarkerSize',23,'LineWidth',1)
%j=plot(pdata(:,1),PlineL1,'k');
%set(j,'LineWidth',2)
%hold off


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make best fit line, L1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rdelta=.0002;     %determines how long to iterate
counterP=0;      %count while-loop iterations
GwP = [ones(length(pdata),1),pdata(:,1)];
mP(1)=ppoly(2);     %intercept
mP(2)=ppoly(1);     %slope
rprevP = abs ( pdata(:,2) - GwP*mP' );
RP = diag(1./rprevP);
mP = (inv(GwP'*RP*GwP)*GwP'*RP*pdata(:,2))';
rcurrP = abs ( pdata(:,2) - GwP*mP' );
while ( ( norm((rcurrP - rprevP),1) /norm (rprevP,1) )  > rdelta)
    rprevP = rcurrP;    %set rprevP to the old rcurrP
    RP = diag(1./rcurrP);
    mP = (inv(GwP'*RP*GwP)*GwP'*RP*pdata(:,2))';
    rcurrP = abs ( pdata(:,2) - GwP*mP' );
    counterP=counterP+1;
end
fprintf('Total iterations for L1 P: %d\n',counterP);
%calculate the L1 best fit line
PlineL1=mP(2).*pdata(:,1) + mP(1);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% plots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
titleLine2=['Log Coda f/Q vs. Log Crustal Q, ',model,' Model, ',wavetyp,titleAdd];
textlinePL1 =['Measured Y value = ',num2str(mP(2),'%1.2f')];
textlinePL1b=['Measured K value = ',num2str(exp(mP(1)),'%1.3f')];
%output figure file
plot3fname =
['/transient6/jnduenow/dataAnalysis/combi/',folder,'/',folder,wavetyp,'.fig'];


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error bar plot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(3)
hold on
h=errorbar(pdata(:,1),pdata(:,2),perrbarL,perrbarU,'.k');
title(titleLine2)
xlabel('Log Qcrust')
ylabel('Log Measured f/Qcoda')
text(3.75,txtPosA,textlinePL1,'FontSize',18)
text(3.75,txtPosB,textlinePL1b,'FontSize',18)
set(h,'MarkerSize',23,'LineWidth',1)
j=plot(pdata(:,1),PlineL1,'k');
set(j,'LineWidth',2)
saveas(gcf,plot3fname);
hold off
```