WaveletExtractor: A Bayesian well–tie and wavelet extraction program ⋆

James Gunning [a,*], Michael E. Glinsky [b]

[a] *CSIRO Division of Petroleum Resources, Bayview Ave, Clayton, Victoria, Australia, 3150, ph. +61 3 95458396, fax +61 3 95458380*

[b] *BHP Billiton Petroleum, 1360 Post Oak Boulevard Suite 150, Houston, Texas 77056, USA*

Abstract

We introduce a new open–source toolkit for the well–tie or wavelet–extraction problem of estimating seismic wavelets from seismic data, time-to-depth information, and well–log suites. The wavelet extraction model is formulated as a Bayesian inverse problem, and the software will simultaneously estimate wavelet coefficients, other parameters associated with uncertainty in the time–to–depth mapping, positioning errors in the seismic imaging, and useful amplitude–variation–with–offset (AVO) related parameters in multistack extractions. It is capable of multi–well, multi–stack extractions, and uses continuous seismic data-cube interpolation to cope with the problem of arbitrary well paths. Velocity constraints in the form of checkshot data, interpreted markers, and sonic logs are integrated in a natural way.

The Bayesian formulation allows computation of full posterior uncertainties of the model parameters, and the important problem of the uncertain wavelet span is addressed uses a multi-model posterior developed from Bayesian model selection theory.

The wavelet extraction tool is distributed as part of the *Delivery* seismic inversion toolkit. A simple log and seismic viewing tool is included in the distribution. The code is written in java, and thus platform independent, but the Seismic Unix (SU) data model makes the inversion particularly suited to Unix/Linux environments. It is a natural companion piece of software to *Delivery*, having the capacity to produce maximum likelihood wavelet and noise estimates, but will also be of significant utility to practitioners wanting to produce wavelet estimates for other inversion codes or purposes. The generation of full parameter uncertainties is a crucial function for workers wishing to investigate questions of wavelet stability before proceeding to more advanced inversion studies.

Key words: wavelet extraction, well tie, Bayesian, seismic, signature, inversion, open–source,

Article originally published in Computers & Geosciences 32 (2006)

1 Introduction

The common procedure of modeling post–stack, post-migrated seismic data as a simple convolution of subsurface 'reflectivity' with a band–limited wavelet, and the use of this model as the basis of various probabilistic inversion algorithms, has been the subject of some very vigorous – and occasionally heated – debates in the last 20 years or so. It is not our intention to rehearse the many and diverse arguments at length in this article: trenchant views both in support and opposition to this model have been expressed at length in the literature (see, e.g. the series of vigorous exchanges following Ziolkowski (1991)). Many detractors believe that this approach places excessive importance on statistical machinery at the expense of physical principles; a perfectly reasonable objection in cases where important experimental or modeling issues are oversimplified and the statistical analysis is disproportionately ornate. While these criticisms can have considerable weight, our view is that inverse modeling in geophysics must always deal with uncertainty and noise, and the community seems to have come to the practical conclusion that good modeling can dispense with neither solid physics nor sensible statistics.

There is a perfectly adequate theoretical justification for the convolutional model, as long as absorption and reflections are sufficiently weak, and the seismic processing preserves amplitudes. Closely related assumptions must also hold for the imaging/migration process to be meaningful; these are usually based on ray–tracing and the Born approximation. The migrated images will then be pictures of the true in–situ subsurface reflectivity – albeit bandlimited by an implied filter which embodies the wavelet we seek to extract. Since modern inversion/migration methods are closely related to Bayesian updates for the subsurface velocity model based on common image gathers (Tarantola, 1984; Gouveia and Scales, 1998; Jin et al., 1992; Lambare et al., 1992, 2003), it would be desirable if well log data were directly incorporated into the migration formula. This is not commonly done, for reasons that may relate primarily to the segregation of workflows, but also technical difficulties. There remains a practical need for wavelet extraction tools operating on data produced by the migration workflow.

Much of the skepticism about convolutional models has arisen from the observation that wavelets extracted from well ties often seen to vary considerably over the scale of a survey. This in itself does not seem to us a sufficient argument to dismiss the model as unserviceable. Firstly, a host of reasons associated with inadequacies in the seismic processing, may create such effects.

⋆ Updated manuscript with additional Appendices from citation (Gunning and Glinsky, 2006). Please cite as both (Gunning and Glinsky, 2006) and the website.

* Corresponding author.
Email address: `James.Gunning@csiro.au` (James Gunning).

For example, sometimes there is heavy attenuation from patchy shallow reefs which is imperfectly compensated for. Secondly, it is rarely – if ever – demonstrated that the difference in wavelets extracted at different parts of the survey is statistically significant. As before, an ideal solution would involve unifying the imaging problem with the well–tie problem for each new well, so imaged amplitudes are automatically constrained to log information. But until this is commonplace, independent parties will be responsible for the seismic processing and well–tie workflows, so the well–tie work has to proceed with the 'best case' processed seismic data at hand. In this situation, errors in the imaging have to be absorbed in modeling noise, and the practitioner should at least attempt to discern if the wavelets extracted at different wells are statistically different.

It is the author's experience that commercial wavelet extraction codes do not proceed from an explicit probabilistic approach to wavelet extraction, and thus are not capable of producing statistical output. Most appear to implement a reasonable least-squares optimization of a model misfit function, but produce only maximum-likelihood estimates (with no associated uncertainty measures), and often only cosmetically filtered versions of these. In addition, there are a number of parameters associated with the time-to–depth mapping (henceforth called 'stretch–and squeeze'), multi–stack and incidence angle (AVO) effects, and imaging errors that ought in principle to be jointly optimized along with the wavelet parameters to improve the well tie. To the authors knowledge, control of such parameters are not available in commercial codes. Many of the commercial programs quote the fine work of Walden and White (Walden and White, 1998) in their pedigree. These algorithms are entirely spectrally–based, which make them very fast and well suited to (probably) impatient users. However, the formulation is not explicitly probabilistic, and the spectral methods will no longer hold once extra modeling parameters are introduced which will move the well log in space or time. More recently, a paper by Buland and Omre (2003) presents a model very much in the same spirit as that we advocate, but no code is supplied. Some notable differences in modeling priority exist between our work and that of Buland. We consider the problem of the unknown wavelet span as very important, and devote considerable effort to modeling this. Conversely, Buland puts some focus on correlated noise models and assumes the wavelet span to be known apriori. It is our experience that the wavelet span, treated as an inferable parameter, couples strongly with the noise level, and is unlikely to be well known in a practical situation. We also prefer a more agnostic approach to modeling the effective seismic noise, since this is a complex mixture of forward modeling errors, processing errors, and actual independent instrumental noise. It seems unlikely that such a process would be Gaussian, and still less susceptible of detailed modeling of the two–point Gaussian correlation function.

We have no objection in principle to the application of cosmetic post-extraction

filtering in order to improve the apparent symmetry or aesthetic appeal of extracted wavelets, but feel that these requirements would be better embedded in some kind of prior distribution in a Bayesian approach. Similarly, the relationship between the sonic log and any time–to–depth information derived from independent sources (like a checkshot) seems most naturally expressed with a Bayesian likelihood function.

In summary, we feel that the simple convolutional model is likely to linger on indefinitely as the standard model for performing probabilistic or stochastic inversion. It is then crucially important to ensure that inversions are run with wavelets and noise parameters optimally derived from well data, checkshot information and the same seismic information. There is then a considerable need for quality software for performing well–ties using a full probabilistic model for the wavelet coefficients, span, time to depth parameters, and other related parameters that may be required for inversion or other quantitative interpretations.

We present just such a model and its software implementation details in this paper. Section 2 presents the Bayesian model specification, and suitable choices for the priors. The forward model, likelihoods and selection of algorithms is discussed in section 3, with the final posterior distribution described in section 4. Questions of statistical significance are addressed in section 5. Details about the code and input/output data forms are discussed in section 6, but most detail is relegated to the electronic appendices 2 and 3. A variety of examples are presented in section 7, and some final conclusions are then offered in section 8.

2 Problem Definition

2.1 Preliminaries

The wavelet extraction problem is primarily one of estimating a wavelet from well log data, imaged seismic data, and a time–to–depth relation that approximately maps the well-log onto the seismic data in time. We will use the following forward model in the problem, with notation and motivations developed in the remainder of this section. The observed seismic $\mathbf{S}_{\text{obs}}$ is a convolution of the reflectivity $\mathbf{r}$ with a wavelet $\mathbf{w}$ plus noise $\mathbf{e_n}$

$$\mathbf{S}_{\text{obs}}(x + \Delta x, y + \Delta y, t + \Delta t_R) = \mathbf{r}(x, y, t|\boldsymbol{\tau}) * \mathbf{w} + \mathbf{e_n}, \tag{1}$$

taking into account any lateral positioning $(\Delta x, \Delta y)$ and registration error Δt_R of the seismic data with respect to the well coordinates. The well log data is

mapped onto the time axis using time–to–depth parameters $\boldsymbol{\tau}$.

A few remarks about the various terms of this equation are required. The imaged seismic data $\mathbf{S}_{\text{obs}}$ is likely to have been processed to a particular (often zero) phase, which involves estimation of the actual source signature (gleaned from seafloor or salt-top reflections, or perhaps even explicit modeling of the physics of the impulsive source), and a subsequent removal of this by deconvolution and bandpass filtering. The processed data is then characterized by an 'effective' wavelet, which is usually more compact and symmetrical than the actual source signature. This processing clearly depends on the early-time source signature, so subsequent dispersion and characteristics of the amplitude gain control (and possible inverse–Q filtering) may also make the 'effective' wavelet at longer times rather different than the fixed wavelet established by the processing. Since the wavelet appropriate for inversion work will be that applicable to a time window centered on the inversion target, it may be very different in frequency content, phase and amplitude from that applicable to earlier times. Thus, we begin with the assumption that the user has relatively weak prejudices about the wavelet shape, and any more definite knowledge can be integrated into the well–tie problem at a later stage via appropriate prior terms.

Deviated wells impose the problem of not knowing precisely the rock properties in the region above and below a current point in the well, since the log is oblique. Given that a common procedure would be to krige the log values out into the near–well region, and assuming the transverse correlation range used in such kriging would be fairly long, a first approximation is to assume no lateral variations in rock properties away from the well, so the effective properties producing the seismic signal are those read from the log at the appropriate $z(t)$. The wavelet extraction could, in principle, models errors in the reflectivity calculation, which could then absorb errors associated which this long–correlation kriging assumption. Another way to think about this approximation is the leading term in a 'near–vertical' expansion. The seismic amplitudes to use will be those interpolated from the seismic cube at the appropriate $(x, y, t(z))$. In this approximation a change in the time–to–depth mapping will result in a new $t'(z)$, but not a new (x, y). Extraction of these amplitudes for each possible realization of the time–to–depth parameters must be computationally efficient.

Wavelet extraction for multiple wells will be treated as though the modeling parameters at each well are independent (e.g. time–to–depth parameters). The likelihood function will be a product of the likelihood over all wells. The wavelet will be modeled as transversely invariant.

The wavelet is parametrized by a suitable set of coefficients over an uncertain span and will be naturally tapered. Determination of the wavelet span is intrin-

sic to the extraction problem. The time to depth mapping from, e.g. checkshot data, will be in general not exact, since measurement errors in first–break times in this kind of data may not be negligible. We allow subtle deviations from the initial mapping ('stretch–and–squeeze' effects) so as to improve the quality of the well–tie, and the extent of these deviations will be controlled by a suitable prior.

Other parameters that may be desirable to model are (1) *time registration errors* in the seismic: these are small time errors that may systematically translate the seismic data in time, and (2) *positioning errors*: errors that may systematically translate the seismic data transversely in space. The latter in particular are useful in modeling the effects of migration errors when using post–migration seismic data. Since the wavelet extraction may be performed with data from several wells, the positioning and registration errors may be modeled as either (a) independent at each location, which assumes that the migration error is different in widely separated parts of the survey, or (b) synchronized between wells, which may be an appropriate assumption if the wells are close.

The extraction is also expected to cope with multi-stack data. Here the synthetic seismic differs in each stack because of the different angle used in the linearized Zoeppritz equations. Because of anisotropy effects, this angle is not perfectly known, and a compensation error for this angle is another parameter which is desirable to estimate in the wavelet extraction. Users may also believe that the wavelets associated with different stacks may be different, but related, for various reasons associated with the dispersion and variable traveltimes of different stacks. It is desirable to build in the capacity to permit stretch and scale relations between wavelets associated with different stacks

Finally, the extraction must produce useful estimates of the size of the seismic noise, which is defined as the error signal at the well–tie for each stack.

2.2 Wavelet parameterization

2.2.1 Basic parameters

Let the wavelet $\mathbf{w}(\mathbf{a}_w)$ be parametrized by a set of coefficients $\mathbf{a}_w$, with suitable prior $p(\mathbf{a}_w)$. Like Buland and Omre (2003), we use a widely dispersed Gaussian of mean zero for $p(\mathbf{a}_w)$. The wavelet is parameterized in terms of a set of equispaced samples $i = -M \ldots N$ (k_W in total), spaced at the Nyquist rate associated with the seismic band edge (typically about $\delta t = 1/(4f_{\text{peak}})$). The first and last samples must be zero, and the samples for the wavelet at the seismic data rate (e.g. 2,4 ms) are generated by cubic splines with zero-derivative endpoint conditions. See Fig. 1. Note there are fewer fundamental parame-

ters than seismic samples. This parameterization enforces sensible bandwidth constraints and the necessary tapering. Cubic splines are a linear mapping,

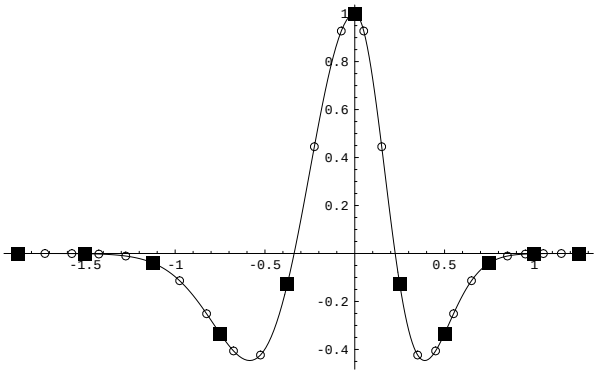

Fig. 1. Parameterization of wavelet in term of coefficients at Nyquist spacing associated with band edge (black boxes), and resulting coefficients generated at seismic time-sampling rate (circles) by cubic interpolation. Cubic splines enforce zero derivatives at the edges for smooth tapering.

so the coefficients at the seismic scale $\mathbf{a}_S$ are related to the coarse underlying coefficients $\mathbf{a}_W$ linearly. Given a maximum precursor and coda length, the two indices M, N then define a (usually small) set of wavelet models with variable spans and centering.

For two–stack ties, the default is to assume the same wavelet is used in the forward model for all stacks. However, users may believe the stacks might legitimately differ in amplitude and frequency content (far-offset stacks usually have about 10% less resolution than the near stack). We allow the additional possibility of 'stretching and scaling' the far wavelet (two additional parameters added on to $\mathbf{a}_w$) from the near wavelet to model this situation. The priors for the additional 'stretch and scale' factors are taken to be Gaussian, most commonly with means close to 1 and narrow standard deviations.

2.2.2 Wavelet phase constraints

Sensible bandwidth and tapering constraints are built into the wavelet parameterization. Additionally, users often believe that the wavelet ought to exhibit some particular phase characteristic, e.g. zero or constant phase. Since the wavelet phase ϕ is obtained directly from a Fourier transform of the wavelet $\tilde{\mathbf{w}} = F(\mathbf{w}(\mathbf{a}_w))$ as $\phi_i = \tan^{-1}(\Im(\tilde{w}_i)/\Re(\tilde{w}_i)$, for frequencies indexed i, a suitable phase–constraint contribution to the prior may be written

$$p(\mathbf{a_w}) \sim \exp(-\sum_i (\phi_i(a_w) - \bar{\phi})^2/2\sigma^2_{\text{phase}})$$

where the sum is over the power band of the seismic spectrum (the central frequencies containing about 90% of the seismic energy). Here $\bar{\phi}$ is a target phase, either user-specified, or computed as $\langle \phi_i(a_w) \rangle$ if a constant (floating) phase constraint is desired. A problem with this formulation is that the branch cuts in the arctan function cause discontinuities in the prior. To avoid this, we use the form

$$p_{\text{phase}}(\mathbf{a_w}) \sim \exp(-\sum_i (D(\tilde{w}_i, \bar{\phi})^2 / 2\sigma^2_{\text{phase}})$$

where $D(\tilde{w}_i, \bar{\phi})$ is the shortest distance from the point $\tilde{w}_i$ to the (one-sided) ray at angle $\bar{\phi}$ heading out from the origin of the complex $\tilde{w}$ plane. This formulation has no discontinuities in the error measure at branch cuts.

2.2.3 Wavelet timing

In many instances, users also believe that the peak of the wavelet response should occur at zero time, so timing errors will appear explicitly in the time–registration parameters, rather than being absorbed into a displaced wavelet. Very often, a zero–phase constraint is too strong a condition to impose on the wavelet to achieve the relatively simple effect of aligning the peak arrival, since it imposes requirements of strong symmetry as well. This peak–arrival requirement can be built into the prior with the additional term

$$p_{\text{peak–arrival}}(\mathbf{a_w}) \sim \exp(-(t_{\text{peak}}(\mathbf{a_w}) - \hat{t}_{\text{peak}})^2 / 2\sigma^2_{\text{peak}}),$$

where $\hat{t}_{\text{peak}}$ and σ_{peak} are user–specified numbers, and $t_{\text{peak}}(\mathbf{a_w})$ is the peak time of the wavelet inferred from the cubic spline and analytical minimization/maximization. Clearly the peak time is only a piece–wise continuous function of the wavelet coefficients, so we advise the use of this constraint only where an obvious major peak appears in the unconstrained extraction. If not, the optimizer's Newton schemes are likely to fail.

2.3 Time to depth constraints

2.3.1 Checkshots and markers

The primary constraint on time to depth is a series of *checkshots*, which produce data pairs $\{z_i^{(c)}, t_i^{(c)}\}$ with associated time uncertainty $\sigma_{t,i}^{(c)}$, stemming primarily from the detection uncertainty in the first arrival time. The depths are measured (well–path) lengths, but convertible to true depths from the well survey, and we will assume no error in this conversion. Such pairs can often

be sparse, e.g. 500ft spacings, and will not necessarily coincide with natural formation boundaries. See also Appendix 5.

Markers are major points picked from the seismic trace and identified with events in the logs. They form data triples $\{z_i^{(m)}, \Delta t_i^{(m)}, \sigma_{\Delta t,i}^{(m)}\}$ which are depths $z^{(m)}$ and relative timing errors $\Delta t_i^{(m)}$ with respect to the time–depth curve associated with the linearly interpolated checkshot. The picking error is $\sigma_{\Delta t,i}^{(m)}$. These can obviously be converted to triples $\{z_i^{(m)}, t_i^{(m)}, \sigma_{t,i}^{(m)}\}$, which are the same fundamental type of data as the checkshot. Here

$$t_i^{(m)} = t_{\text{interpolated–from–checkshot}}(z_i^{(m)}) + \Delta t_i^{(m)},$$

and $\sigma_{t,i}^{(m)} = \sigma_{\Delta t,i}^{(m)}$. This formula is used to convert marker data to checkshot–like data, and is not intended to imply or induce any correlation between the two sources of information in the prior.

We use the vector $\boldsymbol{\tau} = \{t_i^{(c)}, t_i^{(m)}\}$ (combining checkshots and marker deviations) as a suitable parameterization of the required freedom in the time–to–depth mapping. We assume that the errors at each marker or checkshot position are independent and Normal, so the prior for the time–to–depth components becomes

$$p(\tau_i) \sim \exp(-\frac{(t - t_i^{(c,m)})^2}{2\sigma_{t,i}^2})$$

The complete time prior distribution $p(\boldsymbol{\tau}) = \prod_i p(\tau_i)$ is truncated to preserve time ordering by subtracting a large penalty from the log–prior for any state $\boldsymbol{\tau}$ that is non–monotonic.

2.3.2 Registration and positioning errors

A further possibility for error in the time to depth mapping is that the true seismic is systematically shifted in time by a small offset. Such a registration error Δt_R is modeled with a Gaussian prior, and independently for each stack. Similarly, a simple lateral positioning error $\Delta r_p = \{\Delta x, \Delta y\}$ is used to model any migration/imaging error that may have mis–located the seismic data with respect to the well position. This too is modeled by a Gaussian prior $p(\Delta r_p) \sim \exp(-(\Delta x - \overline{\Delta x}_i)^2/2\sigma_{\Delta x,i}^2)\exp(-(\Delta y - \overline{\Delta y}_i)^2/2\sigma_{\Delta y,i}^2)$ for well i. Each seismic minicube (corresponding to each well) will have an independent error, but this can (optionally) be taken as the same for each stack at the well. Positioning errors for closely spaced wells can be mapped onto the same parameter. We write $\boldsymbol{\Delta}\mathbf{r}_R = \{\Delta t_{R,i}, \Delta x_i, \Delta y_i\}$ as the full model vector required for these registration and positioning errors. See also Appendix t2d.

2.4 Log data

For computational efficiency during the extraction, the sonic and density log data are first segmented into chunks whose thickness ought not to exceed $\lambda_B \approx 1/6 f_{\mathrm{BE}}$, where f_{BE} is the upper band–edge frequency of the seismic. We use the aggregative methods of Hawkins and ten Krooden (1978) (also Hawkins (2001(3)) to perform the blocking, based on the p–wave impedance. The effective properties of the blocked log are then computed using Backus averaging (volume weighted arithmetic density average plus harmonic moduli averages (Mavko et al., 1998)) yielding the triples $D_{\mathrm{well}} = \{v_p, v_s, \rho\}$ for the sequence of upscaled layers. The shear velocity may be computed from approximate regressions if required (a near–stack wavelet extraction will be relatively insensitive to it anyway). This blocking procedure is justified by the fact that the convolutional response of very finely layered systems is exactly the same as the convolutional response of the Backus–averaged upscaled system, providing the Backus average is done to 'first order' in any deviations of the logs from the mean block values.

The raw log data and its upscaled values can be expected to have some intrinsic error, which ultimately appears as an error e_r of the reflectivity values computed at the interfaces using the linearized Zoeppritz equations. The nature of errors in well logging are complex in general. A small white noise component always exists, but the more serious errors are likely to be systematic or spatially clustered, like invasion effects or tool–contact problems. For this reason, we make no attempt to model the logging errors using a naive model, and expect the logs to be carefully edited before use.

3 Forward model, likelihood, and the Bayes posterior

The forward model for the seismic is the usual convolutional model of equation (1). We suppress notational baggage denoting a particular well and stack. The true reflectivity is that computed from the well log projected onto the seismic time axis $\mathbf{r}(x, y, t|\boldsymbol{\tau})$ (which is a function of the current parameters time to depth map parameters $\boldsymbol{\tau}$).

The reflection coefficients $\mathbf{r}$ are computed from the blocked log properties, using the linearized Zoeppritz form for the p–p reflection expanded to $O(\theta^2)$ (top of p.63, Mavko et al. (1998)):

$$R_{pp}(B) = \frac{1}{2}\left(\frac{\Delta\rho}{\rho} + \frac{\Delta v_p}{v_p}\right) + B\theta^2 \left(\frac{\Delta v_p}{2\, v_p} - \frac{2\, {v_s}^2 \left(\frac{\Delta\rho}{\rho} + \frac{2\,\Delta v_s}{v_s}\right)}{{v_p}^2}\right), \tag{2}$$

with notation $\rho = \frac{1}{2}(\rho_1 + \rho_2)$, $\Delta\rho = \rho_2 - \rho_1$, etc, for the wave entering layer 2 from layer 1 above. Here, the factor B is introduced to model a situation where the angle θ for a given stack obtained from the Dix equation is uncertain. The stack has user–specified average stack velocity $V_{\rm st}$, event–time $T_{\rm st}$ and stack range $[X_{\rm st,min}, X_{\rm st,max}]$. Assuming uniform weighting, the mean–square stack offset is

$$\langle X_{\rm st}^2 \rangle = \frac{(X_{\rm st,max}^3 - X_{\rm st,min}^3)}{3(X_{\rm st,max} - X_{\rm st,min})}, \tag{3}$$

from which the θ^2 at a given interface is computed as

$$\theta^2 = \frac{v_{\rm p,1}^2}{V_{\rm st}^4 T_{\rm st}^2 / \langle X_{\rm st}^2 \rangle}. \tag{4}$$

Due to anisotropy and other effects related to background AVO rotation (Castagna and Backus, 1993), the angle may not be perfectly known, so B is introduced as an additional parameter, typically Gaussian with means close to unity and small variance: $p(B) \sim \exp(-(B - \bar{B})^2/2\sigma_B^2)$. The B parameter can be independent for each stack if multiple stacks are used.

The reflection R_{pp} is computed at all depths z where the blocked log has segment boundaries, and the properties used in its calculation are the segment properties from Backus averaging etc. The reflection is then projected onto the time axis using the current time to depth map and 4–point Lagrange interpolation. The latter approximates the reflection by placing 4 suitably weighted spikes on the 4 nearest seismic sampling points to the true projected reflection time (this is an approximation to true sinc interpolation, since sinc interpolation is far too expensive to use in an inversion/optimization problem).

3.1 Contributions to the likelihood

The optimization problem aims at obtaining a synthetic seismic close to the observed data for all wells and stacks, whilst maintaining a credible velocity model and wavelet shape. The prior regulates the wavelet shape, so there are contributions to the likelihood from the seismic noise and interval velocities. We address these in turn.

3.1.1 Seismic noise pdf parametrization

Physical seismic noise is largely a result of small waves with the same spectral character as the main signal (multiple reflections etc). However, the noise

process we model, $\mathbf{e}_n$, is a mixture of this real noise and complex modeling errors, much of the latter originated in the seismic processing. We take this as a random signal with distribution $P_N(\mathbf{e}_n|\mathbf{a}_n)$, the meta–parameters $\mathbf{a}_n$ (e.g. noise level, covariance terms etc) having prior $p(\mathbf{a}_n)$.

Probably the simplest approach to the meta–parameters in the noise modeling is to avoid the issue of the noise correlations as much as possible by subsampling. It is trivial to show that if a random process has, eg. a Ricker-2 power spectrum (i.e. $\sim f^2 \exp(-(f/f_{\text{peak}})^2)$), then 95% of the spectral energy in the process can be captured by sampling at times

$$\Delta T_s = 0.253/f_{\text{peak}}, \tag{5}$$

where f_{peak} is the peak energy in the spectrum (and often about half of the bandwidth). Most practical seismic spectra will yield similar results. To keep the meta–parameter noise description as simple as is reasonable, we choose to model the prior distribution of the noise for stack j as Gaussian, with N_j independent samples computed at this sampling–rate, mean zero, and overall variance $\sigma^2_{n,j}$. Since the variance is unknown, it must be given a suitable prior. Gelman et al. (1995) suggest a non–informative Jeffrey's prior ($P_N(\sigma_{n,j}) \sim 1/\sigma_{n,j}$) for dispersion parameters of this type, so the overall noise likelihood + prior will look like

$$P_N(\mathbf{e_n}, \boldsymbol{\sigma}_n) \sim P_N(\mathbf{e_n}|\boldsymbol{\sigma}_n)p(\boldsymbol{\sigma}_n) \sim \prod_{\text{stacks } j} \frac{1}{\sigma_{n,j}^{N_j+1}} \exp(-\mathbf{e_n}^2/2\sigma^2_{n,j}). \tag{6}$$

Correlations between the noise level on closely spaced stacks may be significant, so we assume the user will perform multi–stack ties only on well-separated stacks, so the priors for each stack are sensibly independent.

3.1.2 Interval velocities

Any particular state of the time depth parameter vector $\boldsymbol{\tau}$ corresponds to a set of interval velocities $\mathbf{V}_{\text{int}}$ between checkshot points. It is desirable for these to not differ too seriously from an upscaled version of the sonic log velocities. If $\mathbf{V}_{\text{int,log}}$ are the corresponding (Backus-upscaled) interval velocities from the logs (which we treat as observables), we use the likelihood term

$$p(\mathbf{V}_{\text{int,log}}|\boldsymbol{\tau}) \sim \exp(-(\mathbf{V}_{\text{int}}(\boldsymbol{\tau}) - \mathbf{V}_{\text{int,log}})^2/2\sigma^2_{\text{Vint}})$$

where σ_{Vint} is a tolerable interval velocity mismatch specified by the user. Typically acceptable velocity mismatches may be of the order of 5% or so, allowing for anisotropy effects, dispersion, modeling errors etc.

Even in the case that the interval velocity constraints are weak or disabled, a large penalty term is introduced to force monotonicity in the checkshot points, since the Gaussian priors will allow an unphysical non-monotonic time-to depth map.

4 Forming the posterior

For a given wavelet span, the Bayesian posterior for all the unknowns will then be

$$
\begin{aligned}
&\Pi(\mathbf{a}_w, \boldsymbol{\tau}, \boldsymbol{\Delta}\mathbf{r}_R, B, \boldsymbol{\sigma}_n) \sim \\
&p(\boldsymbol{\sigma}_n)p(B)p(\mathbf{a}_w)p_{\text{phase}}(\mathbf{a_w})p_{\text{peak–arrival}}(\mathbf{a_w}) \prod_{\text{wells } i} p(\boldsymbol{\tau}_i)p(\boldsymbol{\Delta}\mathbf{r}_i) && \text{(prior)} \\
&\times \prod_{\text{wells } i} p(\mathbf{V}_{\text{int,log}}|\boldsymbol{\tau}_i) && \text{(likelihoods)} \\
&\times \prod_{\substack{\text{wells } i \\ \text{stacks } j}} P_N(\mathbf{S}_{\text{obs,ij}}(x + \Delta x_i, y + \Delta y_i, t + \Delta t_{R,i}) - (\mathbf{r}_{ij}(\boldsymbol{\tau}_i) * \mathbf{w}(\boldsymbol{a}_w))|\boldsymbol{\sigma}_n)
\end{aligned}
\tag{7}
$$

The maximum aposteriori (MAP) point of this model is then a suitable estimate of the full set of model parameters, and the wavelet can be extracted from this vector. This point is found by minimizing the negative log posterior using either Gauss–Newton or standard Broyden–Fletcher–Goldfarb–Shanno (BFGS) methods (Nocedal and Wright, 1999; Koontz and Weiss, 1982). The optimizer by default is started at the prior–mean values, with typical scales set to carefully chosen mixtures of the prior standard deviations or other suitable scales. The parameter uncertainties are approximated by the covariance matrix formed from the quadratic approximation to the posterior at the MAP point, as per Appendix 1.

In some cases, especially those where registration or positioning terms exist in the model, the posterior surface is multimodal. The code can then employ a global optimization algorithm formed by sorting local optimization solutions started from dispersed starting points, at the user's request. The starting points are distributed in a hypercube formed from the registration/positioning parameters. This method is naturally more expensive to run than the default. An illustration of this accompanies Example 7.2.

Where lateral positioning errors are suspected and modeled explicitly, the maximum aposteriori parameters obtained in the optimization may not be especially meaningful if the noise levels are high. Better 'fits' to the tie can be obtained through pure chance as easily as through correct diagnosis of a mis-

positioning, and users have to beware of this subtle statistical possibility. A more detailed discussion can be found in Appendix 4. We urge the use of this facility with caution.

4.1 The uncertain span problem

From the user–specified maximum precursor and coda times, a set of candidate wavelet models with indices M, N can be constructed which all lie within the acceptable bracket. These can be simply enumerated in a loop. The posterior space is then the joint space of models and continuous parameters, and each model is of different dimensions. We treat the wavelet span problem as a model–selection problem where we seek the most likely wavelet model k (from among these $k = 1 \ldots N_m$ models) given the data D ($D = \{S_{\text{obs}}, D_{\text{well}}\}$). These models are assumed to have equal prior weight. The maximum–aposteriori wavelet model measured by the *marginal likelihood of the model k*

$$P(k|D) \sim \int \Pi(\mathbf{a}_w, \boldsymbol{\tau}, \boldsymbol{\Delta}\mathbf{r}_R, B, \boldsymbol{\sigma}_n) d\mathbf{a}_w d\boldsymbol{\tau} d\boldsymbol{\Delta}\mathbf{r}_R dB d\boldsymbol{\sigma}_n \tag{8}$$

is an appropriate measure to determine the maximum likelihood wavelet. For linear models, it is well known that the overall model probability computed from this relation (and the associated Bayes factors formed by quotients of these probabilities when comparing models) includes a strong tendency to penalize models that fit only marginally better than simpler models. A simple approximation to the integral is the standard Bayesian information criterion (BIC) penalty (Raftery, 1996), which adds a term $\frac{1}{2} n_p \log(n_d)$ to the negative log–posterior, n_d being the number of independent data points (which will be the number of near–Nyquist samples of the noise $\mathbf{S}_{\text{obs}} - \mathbf{S}_{\text{synth}}$ when the mismatch trace is digitized over the time–interval of interest), and n_p is the number of parameters in the wavelet. We do not use the BIC directly, but evaluate the integral (8) using the Laplace approximation (Raftery, 1996), based on the numerical posterior covariance matrix $\tilde{C}$ obtained in Appendix 1.

Users are sometimes confused as to why long wavelets yield better ties than short ones, and how to choose the length. Use of the formal marginal model likelihood immediately solves this problem: the better ties are invariably not statistically significant. The Bayes factor concentrates the marginal likelihood on the wavelet span of most statistical significance.

Readers unfamiliar with model–selection problems should be aware that the choice of the prior standard deviations for the wavelet coefficients is no longer benign when there are different size models to compare. If the prior standard deviations are chosen too large, the model–selection probabilities suffer from the Lindley Paradox (Denison et al., 2002) and the posterior probability always

falls on the shortest wavelet. To prevent this, the characteristic scale in the prior is chosen about 3 times larger than the RMS seismic amplitude divided by a typical reflection coefficient.

The code computes a normalized list of all posterior wavelet–mode probabilities, so if the user requests realizations of the wavelet from the full posterior, these are sampled from the joint model and continuous–coefficient space. Very often, the posterior probability is concentrated strongly on a particular span.

5 Questions of statistical significance

Even with a sophisticated well–tie code such as that described in this paper, wavelet extraction is not a trivial workflow. In many cases, the main reflections in the data do not seem to correspond very well to major reflections in the well logs, and well-matched synthetics can be produced only by switching on many degrees of freedom, such as positioning, registration errors, and highly flexible checkshot points, with narrow time gates for the extraction. In such situations, the tie may well be totally spurious. A useful flag for such a situation is when the posterior probability lands almost entirely on the shortest wavelet, and also when the quality of the tie degrades rapidly with increasing time gate width.

There are ways to address the problem more rigorously. The usual statistical apparatus required to detect such situations requires a forward model that can generate *all* the seismic data, and parameter uncertainties are then estimated using a Monte Carlo scheme, using an ensemble of extractions on a set of synthetic data sets. A closely related 'null' experiment is to generate synthetic plausible seismic data sets, *totally* uncorrelated from the well data, perform an ensemble of extractions, and see where the true extraction falls on the Monte–Carlo distribution based on the synthetic ensemble. Strictly speaking, when we model registration or positioning errors, this scheme is possible only if we construct a full geostatistical earth model from which we can generate realizations of the seismic minicube. This is a detailed, fussy piece of modeling, requiring both categorical and continuous rock property modeling for an appreciable volume of space.

Instead, we make a 'poor–man's' approximation to this model, generating seismic minicubes using a FFT method based on the power spectrum

$$S(\boldsymbol{\omega}) = w_z^2 \exp(-w_z^2) \exp(-(w_x^2 + w_y^2))$$

where w_z is scaled from a Ricker–2 fit to the data's average vertical power spectrum, and $w_{x,y}$ are scaled so as to give a lateral correlation length specified

by the user. The CDF of the final minicube is then mapped to that of the data, so the univariate statistics of the data minicubes are preserved. The latter step is important, as processed seismic is not univariate Gaussian, and the wavelet extraction is sensitive to amplitudes.

This Monte–Carlo test can be run in cases where the user is suspicious of the meaningfulness of the fit, and generates a CDF of the noise parameters obtained over the Monte–Carlo ensemble. This should have no appreciable overlap with the true–extraction noise best estimates in cases of meaningful ties. An example is shown in section 7.3.

6 The code

The wavelet extraction code is written in java, and based largely on efficient public domain linear algebra [1] and optimization (Koontz and Weiss, 1982) libraries, along with the seismic handling libraries of its companion software *Delivery* [2]. It comprises about 50k lines of source code. Wavelet extraction is a relatively light numerical application: simple extractions take seconds and more complex problems may take minutes.

Users are expected to be able to provide seismic data in big–endian 'minicubes' centered at each well (the seismic resides entirely in RAM), plus log data in ASCII LAS or simple geoEAS format. Checkshots and well surveys in a simple geoEAS are also required. Details of these formats are given in Appendix 2, but see also Appendix 5.

Outputs are a mixture of seismic SU [3] files for wavelets, synthetic–and–true SU seismic pairs, and simple ASCII formats for the parameter estimations and uncertainties. A small graphical visualization tool (extractionViewer) is provided as part of the distribution which produces the typical cross-registered log and seismic displays shown in the examples. Details of the output files are in Appendix 3.

The code is available for download (Gunning and Glinsky, 2004),[3], under a generic open–source agreement. Improvements to the code are welcome to be

[1] Hoschek, W., The CERN **colt** java library. `http://dsd.lbl.gov/~hoschek/colt/`.

[2] Gunning, J., 2003. **Delivery** website: follow links from `http://www.petroleum.csiro.au`.

[3] Cohen, J. K., Stockwell, Jr., J., 1998. CWP/SU: Seismic Unix Release 35: a free package for seismic research and processing,. Center for Wave Phenomena, Colorado School of Mines, `http://timna.mines.edu/cwpcodes`.

submitted to the author. A set of simple examples is available in the distribution, and are briefly illustrated here.

7 Examples

7.1 Simple Single Reflection

The simplest example possible is a set of logs that produce a single reflection, so the synthetic trace reflects the wavelet directly. Two synthetic data minicubes were generated from the logs with zero and a modest noise level. With the noise–free minicube, the wavelet extracts perfectly, and the posterior distribution converges strongly on the wavelet model just long enough to capture the true wavelet energy.

More interesting is the extraction on the noisy minicube, shown in Fig. 2. Here the central checkshot point was deliberately shifted 10ms, and it shares a 10ms uncertainty with all the checkshot points spaced at 500ft. The naive extracted wavelet is clearly offset. A symmetrical wavelet can be forced by switching on a zero–phase constraint, and the checkshot point then moves about 10ms backwards, to allow the zero–phase recovered wavelet to still generate a 'good' synthetic.

This extraction is run with the command line
```
% waveletExtractor WaveletExtraction.xml --dump-ML-parameters
--dump-ML-synthetics --dump-ML-wavelets --fake-Vs -v 4 -c -NLR
```
where the main details for the extraction are specified in the XML file `WaveletExtraction.xml`. The XML shows how to set up an extraction on a single trace, where the 3D interpolation is degenerate. The runtime options correspond to frequently changed user preferences. Their meanings are documented by the executable `waveletExtractor` in SU style self–documentation [2]. Amongst the most important options are a) those related to the set of wavelet spans (c.f. section 4.1): `-c,-l` denote respectively a centered set of wavelets, or the longest only. b) Log-data options: `-NLR`; assume no reflections outside log (so entire log can be used), `--fake-Vs`; concoct approximate fake shear log from p–wave log and density.

Standard graphical views of the MAP extraction are obtained with
```
% extractionViewer MLgraphing-file.MOST_LIKELY.well_log.txt
```
and the most likely wavelet with (on a little–endian machine)
```
% cat MLwavelet.MOST_LIKELY.su | suswapbytes | suxgraph
```

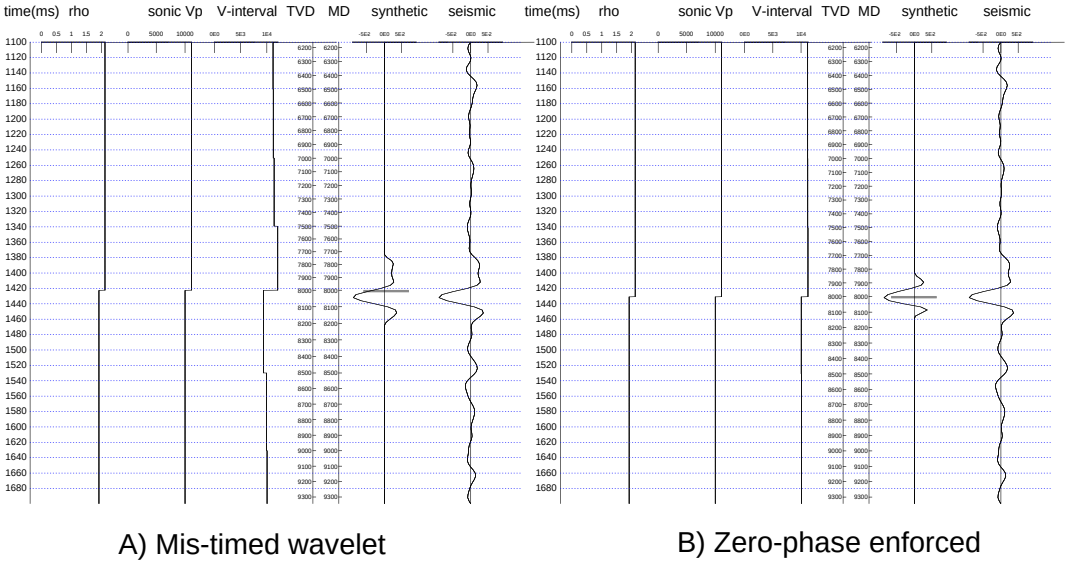

Fig. 2. Example of recovery of wavelet from single reflection with noisy data. A) Case where time-to-depth error forces mis–timing of wavelet. B) Case where phase–constrained prior forces timing error to be absorbed in uncertain time-to-depth map. Viewing program displays co–registered time, total vertical depth (TVD), measured depth (MD) scales for MAP (best–estimate) time-to-depth model. Also shown are (in order) density logs ρ, sonic v_p, $V_{\text{int,log}}$ (interval velocity from upscaled blocked sonic), synthetic seismic $(\mathbf{r}(x, y, t|\boldsymbol{\tau}) * \mathbf{w})$, observed seismic $\mathbf{S}_{\text{obs}}(x + \Delta x, y + \Delta y, t + \Delta t_R)$. Viewing display can toggle on acoustic impedance, a blocked v_p log, slownesses etc.

7.2 Standard dual–well extraction

Here we illustrate a typical, realistic dual well extraction for two widely separated wells 'Bonnie–House' and 'Bonnie–Hut'. The checkshots are of high quality (bar a few points), so time—to–depth errors are confined to registration effects only. We illustrate here the peak–arrival constraint prior (c.f. section 2.2.3), which forces the wavelet peak amplitude to arrive at $t = 0 \pm 1ms$. Multi-start global optimization was used. A Monte–Carlo study of the extraction here shows the extraction to be significant with very high probability. See Fig. 3.

The vertical wells 'Bonnie–House' and 'Bonnie–Hut' have relatively good quality checkshots. For this reason, time–registration effects were the only unknown parameters included in the time-to-depth mapping. This kind of problem creates an obvious non–uniqueness in the inverse problem if the wavelet is long enough to offset itself in time by the same offset as the registration error. The prior in the registration error in principle should remove the non-uniqueness of the global minimum in this case, but a variety of poorer local minima do appear in the problem, in which the local optimizer is easily trapped.

This problem illustrates a typical interactive workflow. 1) The extraction is per-

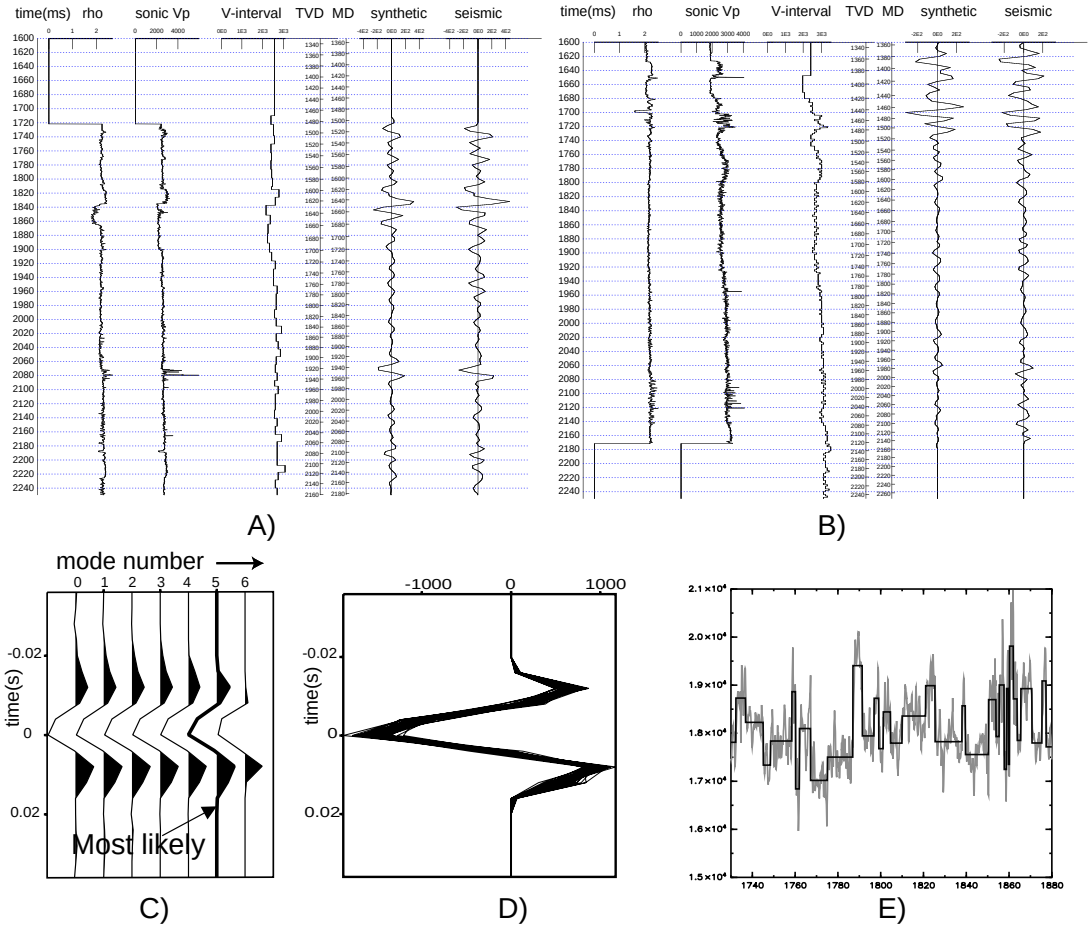

Fig. 3. Example of typical dual–well extraction for 'Bonnie' wells: A) Tie at 'Bonnie–House', B) 'Bonnie–Hut'. Note close agreement between interval velocities and sonic logs. C) Maximum likelihood (ML) wavelet for each model–span, with ML wavelet the solid curve. D) Wavelet realizations from full posterior (section 4.1) showing small posterior uncertainty. E) Example of the acoustic impedance blocking on a section of the Bonnie–House log.

formed with no phase or timing constraints, often for a single long wavelet (the `-l` option), on each well at a time. The resulting wavelets are then examined to see if the peak response (assuming the expectation of a symmetric–looking wavelet) is systematically shifted in time. Very bad ties may be noted, and the guilty wells excluded. 2) Large systematic offsets can then be directly entered into the XML file for each well (in consultation with the imaging specialist!), possibly allowing registration timing errors around the shift. 3) The individual extractions can then be rerun, confirming that major timing problems have been removed 4) A variety of joint–well extractions may then be attempted, perhaps turning on phase or peak–arrival constraints.

In principle, the manual setting of offsets is something that can be automated as part of a (expensive) global optimization routine (available using the `--multi-start` heuristic), but users usually want to examine the well–ties

on an individual basis before proceeding to joint extractions. It is often the case that imaging around a well is suspicious for various reasons – which translates into poor ties, so the best decision may be to omit the well altogether, rather than lump it into what may otherwise be a good joint extraction.

The actual extraction here is run with

```
% waveletExtractor BonnieHousePlusBonnieHut.xml --fake-Vs
--dump-ML-parameters --dump-ML-synthetics --dump-ML-wavelets -v 4
[--constrain-peak-arrival 0 1] -c --multi-start -NLR
```

An illustration of the complexity of the basins of attraction is shown in Fig. 4.

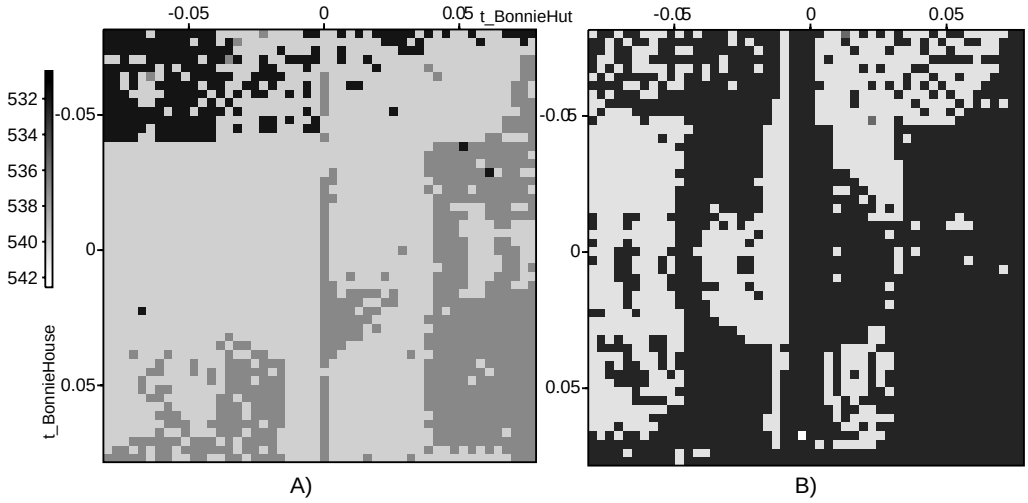

Fig. 4. Example of basins of attractions for two registration parameters for Bonnie–House and Bonnie–Hut. Grayscale represents MAP negative log–posterior value obtained starting from the values on the axes. All other parameters are started at their prior means. A) Basins for the problem without peak–arrival constraints B) Case with peak–arrival constraint ±1ms (`--constrain-peak-arrival 0 1`). The darkest shade is the global minimum. Note how peak–arrival term in prior greatly enlarges basin of attraction of desired global optimum, without affecting overall MAP value.

7.3 AVO extraction for 'Bitters' well

Here we illustrate a typical single–well extraction on log data that has a strong AVO character. The two stacks are at about 5 and 30 degrees. The seismic here was generated synthetically, with a large level of white noise added to the reflection sequence before convolution. The extraction is then of poorish quality, but still statistically significant at about the 10% level, even though the most likely wavelet corresponds to the shortest model. The same wavelet was used for both stacks, so no wavelet stretch–and–scale parameters were used.

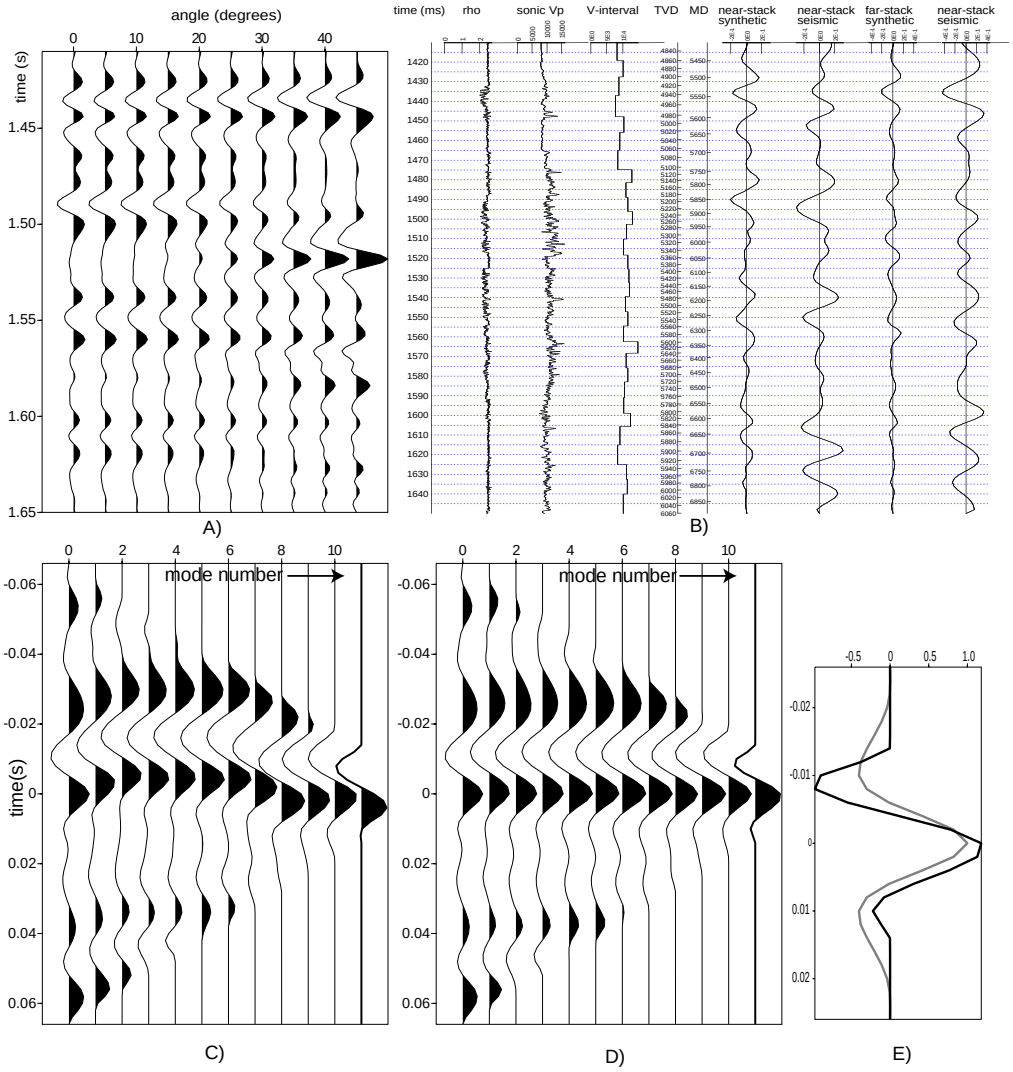

Fig. 5. A) AVO character of synthetic seismic using Ricker–2 wavelet. Note strong variation around 1.5s. B) Two–stack extraction screen capture: far–stack synthetic and observed traces are appended at the right. C) ML wavelets vs. wavelet length (mode) with no peak–constraints. ML wavelet (boldest trace) is shortest mode. D) Same, with peak–constraints in prior. E) Recovered wavelet vs underlying wavelet used to generate synthetics.

Here the XML shows how to set up an extraction on a single seismic line, where the 3D interpolation is approximated by perpendicular 'snapping' onto the 2D line. The extraction module can be used to generate AVO gathers of the kind seen in Fig. 5a). The command

```
% waveletExtractor Bitters.xml --dump-ML-wavelets -v 4 -NLR
--make-synthetic-AVO-gather 80 2
```

was used in this example, which generates 80Hz synthetics sampled at 2ms. Angles are stored in the SU header word f2. Displays can then be obtained via, e.g.

```
% cat synth_seis.Bitters.las.AVO-gather.su | [suswapbytes]
```

```
|suxwigb x2beg=0
```

The synthetic seismic minicubes are generated using the `--make-synthetic freq dt noise` option, so, e.g.

```
% waveletExtractor Bitters.xml -v 4 -NLR --dump-ML-wavelets
--make-synthetic 80 2 2.3
```

makes a pair of synthetic minicubes using a Ricker 2 wavelet of 80Hz band edge, sampled at 2ms, with a noise reflectivity 2.3 times the RMS reflection power measured in the logs added on to the log reflectivity. The time to depth curve is the prior mean read from the checkshot.

The second stack is specified naturally in the XML file, and the extraction run with

```
% waveletExtractor Bitters.xml --dump-ML-parameters
--dump-ML-synthetics --dump-ML-wavelets -v 4 -c -NLR
```

Significance testing is obtained by the command

```
% waveletExtractor Bitters.xml --dump-ML-parameters
--dump-ML-synthetics --dump-ML-wavelets -v 4 -c -NLR
--monte-carlo 10
```

which specifies a lateral correlation length of 10 traces in the Monte–Carlo synthetic minicubes. The position of the ML noise parameters with respect to the Monte–Carlo noise parameter distribution is found by examining the file `MC_ParameterSummary.txt`. In this case, they sit about 10% of the way into the Monte–Carlo distribution.

With very noisy data sets like this one, use of discontinuous or strongly non-linear terms in the likelihood such as the phase–constraint or wavelet–peak terms is somewhat dangerous. If the extraction is barely significant without these constraints (a suggested first test), it will be even less so when they are added.

8 Conclusions

We have introduced a new open–source software toolkit for performing well–ties and wavelet extraction. It can perform multi–well, deviated well, and multi–stack ties based on imaged seismic data, standard log files, and check-shot information. Uncertainties in the time–to-depth conversion and the imaging and wavelet–aesthetic constraints are automatically included by the use of Bayesian techniques. The module produces maximum–apriori ('best') estimates of all the relevant parameters, plus measures of their uncertainty.

Stochastic samples of the extracted wavelet can be produced to illustrate the uncertainty in the tie, and Monte Carlo tests of the well–tie significance can be performed for highly suspicious ties. Our experience is that the wavelet and noise estimates produced are of critical importance for inversion packages like *Delivery* and many other applications.

Inputs for the module are common seismic data minicubes, well log files, simple ascii representations of checkshot information, and a simply–written XML file to specify all the needed parameters. Users will be readily able to tackle their own problems by starting from the provided examples.

The authors hope that this tool will prove useful to the geophysical and reservoir modeling community, and encourage users to help improve the software or submit suggestions for improvements.

Acknowledgments

The first author gratefully acknowledges generous funding from the BHP Billiton technology program. Helpful comments from the two reviewers are also appreciated.

References

Buland, A., Omre, H., 2003. Bayesian wavelet estimation from seismic and well data. Geophysics 68 (6), 2000–2009.

Castagna, J. P., Backus, M. M. (Eds.), 1993. Offset-dependent reflectivity – Theory and practice of AVO Analysis. Society of Exploration Geophysicists.

Denison, D. G. T., et al., 2002. Bayesian Methods for Nonlinear Classification and Regression. Wiley.

Deutsch, C. V., Journal, A., 1998. GSLIB Geostatistical Software Library and User's Guide, 2nd edition. Oxford University Press.

Gelman, A., Carlin, J. B., Stern, H. S., Rubin, D. B., 1995. Bayesian Data Analysis. Chapman and Hall.

Gouveia, W., Scales, J., 1998. Bayesian seismic waveform inversion: Parameter estimation and uncertainty analysis. Journal of Geophysical Research 103 (B2), 2759–2780.

Gunning, J., Glinsky, M., 2004. Delivery: an open-source model-based Bayesian seismic inversion program. Computers and Geosciences 30 (6), 619–636.

Gunning, J., Glinsky, M., 2006. WaveletExtractor: A Bayesian well–tie and wavelet extraction program. Computers and Geosciences 32, 681–695.

Hawkins, D. M., 2001(3). Fitting multiple change-point models to data. Computational Statistics and Data Analysis, 323–341.

Hawkins, D. M., ten Krooden, J. A., 1978. A review of several methods of segmentation. In: Gill, D., Merriam, D. (Eds.), Geomathematical and Petrophysical Studies in Sedimentology. Pergamon, pp. 117–126.

Jin, S., Madariaga, R., Virieux, J., Lambare, G., 1992. Two–Dimensional Asymptotic Iterative Elastic Inversion. Geophysical Journal International 108, 575–588.

Koontz, R. S. J., Weiss, B., 1982. A modular system of algorithms for unconstrained minimization. Tech. rep., Comp. Sci. Dept., University of Colorado at Boulder, Steve Verrill's Java translation at `http://www1.fpl.fs.fed.us/optimization.html`.

Lambare, G., Operto, S., Podvin, P., Thierry, P., 2003. 3d ray+Born migration/inversion – Part 1:Theory. Geophysics 68, 1348–1356.

Lambare, G., Virieux, J., Madariaga, R., Jin, S., 1992. Iterative asymptotic inversion in the acoustic approximation. Geophysics 57, 1138–1154.

Mavko, G., Mukerji, T., Dvorkin, J., 1998. The Rock Physics Handbook. Cambridge University Press.

Nocedal, J., Wright, S. J., 1999. Numerical Optimization. Springer.

Raftery, A. E., 1996. Hypothesis testing and model selection. In: Markov Chain Monte Carlo in Practice. Chapman and Hall.

Tarantola, A., 1984. Linearized inversion of reflection seismic data. Geophysical Prospecting 32, 998–1015.

Walden, A. T., White, R., 1998. Seismic wavelet estimation: a frequency domain solution to a geophysical noisy input output problem. IEEE Transactions on Geoscience and Remote Sensing.

Zellner, A., 1986. On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In: Bayesian Inference and Decision Techniques: Essays in honour of Bruno de Finetti. North–Holland, pp. 233–243.

Ziolkowski, A., 1991. Why don't we measure seismic signatures? Geophysics 56, 190–201.

Appendices: (supplementary material on journal archive)

Appendix 1: Bayesian update formulas for approximate multi-Gaussian posteriors including the noise parameters

The standard linear model with a Jeffreys prior on a single noise parameter has a simple analytical posterior featuring a (right skewed) Inv–χ^2 marginal distribution for the noise (Gelman et al., 1995, section 8.9). Although this posterior is not strictly integrable to obtain a marginal model likelihood, a Gaussian approximation at the MAP point will be. When there are multiple noise parameters and some parts of the likelihood are not scaled by the unknown noise, the posterior is somewhat messier. The amount of data commonly available in well–tie problems is usually sufficient to make the posterior distribution of the noise components fairly compact, so will develop the relevant formulae on the assumption that a full multi–Gaussian form for the joint model and noise parameters is adequate.

Since the noise posteriors are usually right skewed, it is sensible to work in terms of the joint model vector $\mathbf{M} = \{\mathbf{m}, \mathbf{m}^{(\mathrm{n})}\}$ where $\mathbf{m}^{(\mathrm{n})}$ is defined as $\mathbf{m}^{(\mathrm{n})}_j = \log(\sigma_{\mathrm{stack}\ j})$, the noise level for stack j. The non–noise components $\mathbf{m}$ are formed from the wavelet components $\mathbf{a}_w$ (and coupling parameters for the coupled near/far wavelet mode of operation), time to depth map knots $\boldsymbol{\tau}$, positioning and registration errors $\boldsymbol{\Delta}\mathbf{r}_R$ and AVO scale term B respectively:

$$\mathbf{m} \equiv \{\mathbf{a}_w, \boldsymbol{\tau}, \boldsymbol{\Delta}\mathbf{r}_R, B\}.$$

We assert a Gaussian prior for $\mathbf{M}$ of form $P(\mathbf{M}) = N(\bar{\mathbf{M}}, C_M)$, where $C_M = \mathrm{diag}\{C_p, C_p^{(n)}\}$ and the $C_p^{(n)}$ entries for noise components $\mathbf{m}^{(\mathrm{n})}$ are very weak (we choose the mean to be about the log of the signal RMS, and the standard deviation large on the log scale), and absorb (for computational convenience) the Jeffrey's noise prior and wavelet peak/phase prior terms into the collected likelihood

$$L(\mathbf{y}|M) = \frac{P_{\mathrm{Jeffreys}}(\mathbf{m}^{(\mathrm{n})})}{|C_D(\mathbf{m}^{(\mathrm{n})})|^{1/2}} \exp(-(\mathbf{y} - f(\mathbf{m}))^T C_D(\mathbf{m}^{(\mathrm{n})})^{-1}(\mathbf{y} - f(\mathbf{m}))/2).$$

This likelihood collects all non–Gaussian terms in (7), so $\mathbf{y}$ is an 'equivalent data' vector, formed by the concatenation of the seismic data for all wells and stacks, sonic–log derived interval velocities, and (from the prior) desired phase/timing characteristics of the wavelet. Similarly $f(\mathbf{m})$ is the concatenation of the synthetic seismics, model–derived interval velocities, and the wavelet–phase/timing measures. Only that part of $C_D(\mathbf{m}^{(\mathrm{n})})$ pertaining to the

seismic mismatch depends on $\mathbf{m}^{(\mathrm{n})}$. Note that C_D is diagonal, with $\exp(2\mathbf{m}_j^{(\mathrm{n})})$ terms for the seismic mismatch on stack j, σ^2_{Vint} terms for the interval velocities, and $\sigma^2_{\mathrm{phase}}, \sigma^2_{\mathrm{peak-arrival}}$ terms for the wavelet phase/timing.

Under the logarithmic change of variables, the Jacobean and coefficient of the above equation conspire to yield the log-likelihood

$$-\log(L(\mathbf{y}|M)) = \mathbf{e}^T.\mathbf{m}^{(\mathrm{n})} + \frac{1}{2}(\mathbf{y} - f(\mathbf{m}))^T C_D(\mathbf{m}^{(n)})^{-1}(\mathbf{y} - f(m)).$$

where $e_j = N_{\mathrm{stack}\ j}$, the number of seismic mismatch data points for stack j. If all terms in the log–likelihood and prior are expanded to quadratic order in $\{\mathbf{m}, \mathbf{m}^{(\mathrm{n})}\}$ around some current guess $\mathbf{M}_0 = \{\mathbf{m}_0, \mathbf{m}_0^{(\mathrm{n})}\}$ and collected, the resulting quadratic form corresponds to a posterior Gaussian distribution for $\mathbf{M}$ which may be solved using the following expressions:

$$\Delta\tilde{y}_i = \frac{\mathbf{y}_i - f_i(\mathbf{m}_0)}{C_{D,ii}^{1/2}} \tag{9}$$

$$\tilde{X}_{ij} = \frac{f_i(\mathbf{m}_0 + \delta m_j) - f_i(\mathbf{m}_0)}{\delta m_j C_{D,ii}^{1/2}} \quad \text{(sensitivity matrix from finite diffs.)} \tag{10}$$

$$\hat{Q}_{i,j} = -2\delta_{I(i),j}\Delta\tilde{\mathbf{y}}_j \qquad (N_{\mathrm{stacks}} \times \dim(\mathbf{y}) \text{ matrix}) \tag{11}$$

$$\tilde{P}_{ij} = 2\sum_k (\Delta\tilde{\mathbf{y}}_k)^2 \delta_{I(k),i}\delta_{I(k),j} \qquad (N_{\mathrm{stacks}} \times N_{\mathrm{stacks}} \text{ matrix}) \tag{12}$$

where $I(i)$ is an indicator variable that is the index of the stack–noise parameter in $\mathbf{m}^{(\mathrm{n})}$ that applies to component i in the 'data' vector $\mathbf{y}$. It is zero for non seismic–mismatch components.

With these definitions, the linear system for the model update $\Delta\mathbf{M}$ is

$$\begin{pmatrix} C_p^{-1} + \tilde{X}^T\tilde{X} & -\tilde{X}^T\hat{Q}^T \\ -\hat{Q}\tilde{X} & C_p^{(\mathrm{n})\text{-}1} + \tilde{P} \end{pmatrix} \cdot \begin{pmatrix} \Delta\mathbf{m} \\ \Delta\mathbf{m}^{(n)} \end{pmatrix} = \begin{pmatrix} C_P^{-1}(\bar{\mathbf{m}} - \mathbf{m}_0) + \tilde{X}^T\Delta\tilde{y} \\ -\mathbf{e} - C_p^{(n)-1}(\mathbf{m}_0^{(\mathrm{n})} - \bar{\mathbf{m}}^{(\mathrm{n})}) - \frac{1}{2}\hat{Q}\Delta\tilde{y} \end{pmatrix} \tag{13}$$

The update then has mean $\bar{\mathbf{M}}' = \mathbf{M}_0 + \Delta\mathbf{M}$ and a covariance $\tilde{C}$ which is the inverse of the coefficient matrix in (13):

$$\tilde{C} \equiv \begin{pmatrix} C_p^{-1} + \tilde{X}^T\tilde{X} & -\tilde{X}^T\hat{Q}^T \\ -\hat{Q}\tilde{X} & C_p^{(\mathrm{n})} + \tilde{P} \end{pmatrix}^{-1}. \tag{14}$$

These update formula form the basis of a Newton scheme for finding the maxi-

mum aposteriori point for a particular model with wavelet vector $\mathbf{a}_w$. The RHS of (13) is effectively a gradient of the posterior density, so the updates vanish when the mode is reached. The actual implementation is a typical Gauss–Newton scheme with line searches in the Newton direction and additional step-length limits applied for big Newton steps. Laplace estimates based on the determinate of the final $\tilde{C}$ are used in forming the marginal model likelihood.

Appendix 2: Data input forms

The data needed to perform the wavelet extraction should be in the following forms

- Configuration file. For a particular extraction, the configuration is defined by an XML file which sets up all necessary parameter choices and file names. It is controlled by a schema file (XSD), so configuration is simple with the BHP XML schema-driven editor supplied with the distribution. The schema is located at `au/csiro/JamesGunning/waveletextraction/xsd/` relative to the root of the distribution, and the editor at `scripts/xmledit.sh`. Certain self–documenting (SU style) runtime flags are able to modify the behavior at execution for frequently chosen options. We recommend users read the self docs carefully and follow the examples.
- Seismic data as a 'mini–cube', for each well, in big–endian SU format. Inline, crossline and gx,gy (group coordinate) header fields need to be correctly set. The cube should be big enough to safely enclose each well on the desired extraction time interval, but not so large as to swamp memory (10x10 cubes are typically fine for straight holes). If x,y headers are not available, a triplet of non–collinear points can be specified in the XML file which enables the code to map inline,crossline to x,y. Single line or trace seismic files are also acceptable, provided suitable flags are set in the XML.
- Log data is read in ascii, either standard ascii LAS files or the geoEAS format used by GSLIB (Deutsch and Journal, 1998). The critical depth to be specified in these files is measured depth (MD) along the well trajectory. There must be density and DT (slowness) fields for the sonic p–wave and shear logs. A rudimentary commandline switch (`--fake-Vs`) will concoct a placeholder shear log from the p–wave log if shear is not available, which is useful for near–normal incidence extractions.
- UTM files with the well trajectories. These are geoEAS files, which must have x, y, MD and TVD columns. Kelly bushing data will be required to convert to true depth TVD.
- Checkshot data. Again, geoEAS format. We require MD, T, sigma_T columns.
- Markers (optional): geoEAS format also, with MD, T, sigma_T columns.

A note on units: the code performs automatic time units detection for time data, which is expected to be input in either s or ms. All density (mass) units cancel internally. Equation (2) involves a velocity ratio from log data to stacking–velocity V_{st} (see eqn (3)), and the code will again attempt to auto–detect a feet vs meters difference here (logs are commonly in μS/ft, V_{st} may be in m/s). Another velocity ratio occurs in (3.1.2); again, auto–detection is performed, since $\mathbf{V}_{\mathrm{int}}(\boldsymbol{\tau})$ is derived from a checkshot that may be imperial or metric. The auto–detection code is based on the assumption fact that velocities will be specified in m/s or ft/s, and that sonic logs will be in μS/ft or (very unusually) metric.

Users are recommended to look at the examples in the distribution closely when setting up their own problems.

Appendix 3: Outputs

A variety of outputs are generated by the code. Many of these are selectable by commandline switches (the code self–documents like SU codes). All SU outputs are big–endian. In roughly diminishing order of importance, these are

- ASCII dumps of the cross-registered logs, observed and synthetic seismic, for each well and wavelet span (typically `MLgraphing-file.*`). Used as direct input for the ExtractionViewer module.
- Maximum likelihood (ML) wavelet, SU format, for each wavelet span, and overall (`MLwavelet.*.su`).
- ASCII files of the most likely parameter values and uncertainties (diagonal entries of the posterior covariance matrix), for each wavelet span, and overall (`MLparameters.*.txt`).
- Multiple realizations of the wavelets from the posterior, also SU format (`wavelet_realisations.su`).
- Maximum likelihood 5–trace seismic sets along the well trajectory, for the five quantities {observed–seismic, synthetic seismic, reflectivity, x, y}, again, for each wavelet span, and overall (`MLsynth_plus_true_seis.*.su`) . These are very useful for third party software applications. Two extra traces for P and S reflectivity can be obtained with `--dump-PS-reflectivity`.
- ASCII files for the most likely time-to–depth map, for each well, for each wavelet span, and overall (`MLtime-to-depth-map.*.txt`)
- ASCII files of the most likely wavelet phase spectrum, for each wavelet span, and overall (`MLwavelet-phase-spectrum*`).
- ASCII dumps of the well log blocking (`*_blocked.txt`).
- ASCII dumps of negative log–posterior cross-sections of the posterior surface, shown as slices for each model parameter (`posteriorXsection*.txt`). These are useful for checking for possible multimodality, or serious non–

linearity.

Interpretation of the time scales in these files requires some diligence; read especially Appendix 5.

Appendix 4: Issues in positional uncertainty modeling

When using the extraction model with general positioning/registration parameters, this amounts to using a generalized non–linear regression of form

$$\mathbf{S}_{\text{obs}}(m) = \mathbf{r}(m) * \mathbf{w}(m) + \mathbf{e_n},$$

where some parameters in m determine (by continuous interpolation) the local trace data $\mathbf{S}_{\text{obs}}(m)$ to be used in the wavelet extraction.

In this kind of model, one has to be wary of the optimization 'latching' on to a particularly favorable location parameter which is controlled merely by chance rather than a genuinely better fit to the regression model. A better insight into this problem may be obtained by examining a simplified discrete 'choose–the–data–set' problem, in which spatial fluctuations in y are mimicked by a choice of one of two data sets y_1 or y_2, and the 'choice' parameter is controlled by the Bayes factors at the maximum likelihood point. We can then examine how the expected fluctuations in the Bayes factor affect the uncertainty in the 'choice' parameter by Monte–Carlo methods.

A canonical linear regression model for a specific data set y would be

$$y = X.m + \mathbf{e_n}$$

where X is the design matrix and $e_n \sim N(0, \sigma^2)$ is the noise. The standard likelihood is

$$-\log(L(d|m)) \sim (y - X.m)^T (y - X.m)/2\sigma^2 + n \log(\sigma)$$

A Jeffrey's prior for the noise variance σ^2, coupled with an independent Gaussian $N(0, \Sigma_m)$ for the coefficients leads to an improper posterior marginal probability for the model (Gelman et al., 1995), so we use a class of conjugate priors – Zellners' g-priors (Zellner, 1986) – which leads to simple analytic forms for the posterior. These priors take the form

$$p(\sigma^2) \sim 1/\sigma^2 \tag{15}$$

$$p(m|\sigma^2) = N(0, g\sigma^2 (X^T X)^{-1}) \tag{16}$$

where g is a single hyper-parameter. For the linear model, the posterior marginal probability has the simple form

$$\Pi = \int (L(d|m)p(m|\sigma^2)p(\sigma^2)dmd\sigma^2 \sim (1+g)^{(n-k-1)/2}(1+g(1-R^2))^{-(n-1)/2}$$

where there are n data points, k parameters in m, and R^2 is the usual coefficient of determination. The Bayes factor for two identical models fitted to different data sets (traces) y_1 and y_2 is then

$$B_{12} = \left(\frac{(1+g(1-R_1^2)}{(1+g(1-R_2^2))} \right)^{-(n-1)/2} .$$

It is instructive to evaluate this quantity for realistic models of the relative fluctuation in R_1 and R_2 as we imagine extracting the data sets y_1, y_2 from a trace in a typical near–well region. To illustrate, we have fitted the 'noisy straight line' data sets(s)

$$y_{(1,2),i} = 1 + i/n + q_i + \epsilon_{(1,2),i} \qquad i = 1, 2 \ldots n$$

for $n = 100$ data points $i = 1, 2 \ldots n$, with X describing a standard linear model, $q \sim N(0, 0.25^2)$ a *fixed* (shared) error realization of relative RMS amplitude typical for a well–tie, and the two samples y_1, y_2 distinguished by the $\epsilon_{(1,2),i}$ term, with the setting $\epsilon_{(1,2),i} \sim N(0, 0.05^2)$ emulating small spatial fluctuations.

Different samples of y are generated by different realisations of ϵ. We have used the recommended setting $g = n$ to set the prior covariance $g(X^T X)^{-1}$ to be $O(1)$ (see Zellner (1986) and Clyde [4] for a more extensive discussion of the role of g). Under this model, the Monte–Carlo distribution of B_{12} formed when y_1 and y_2 are drawn from different realisations of ϵ (for a fixed, random q), is illustrated in Fig. 6. Large fluctuations in the Bayes factor associated with 'position' (of which ϵ is the surrogate) are induced by the relatively weak (5%) additional noise. It is easy to imagine that the deep local minima associated with these large Bayes factors will also occur in the full nonlinear wavelet extraction problem, where the 'choice' parameter is a continuous spatial variables, rather than a discrete label. If so, it is likely that strongly constrained positioning parameters (associated with deep local minima in the map of the Bayes factor as we move around in the near–well region) are not trustworthy.

In summary, we recommend using the lateral position modeling capabilities of this package with extreme caution. Particular skepticism ought to be exercised

[4] Clyde, M., George, E. I., 2003. Model uncertainty. Statistical and Applied Mathematical Sciences Institute Technical Report #2003–16, see `www.samsi.info`.

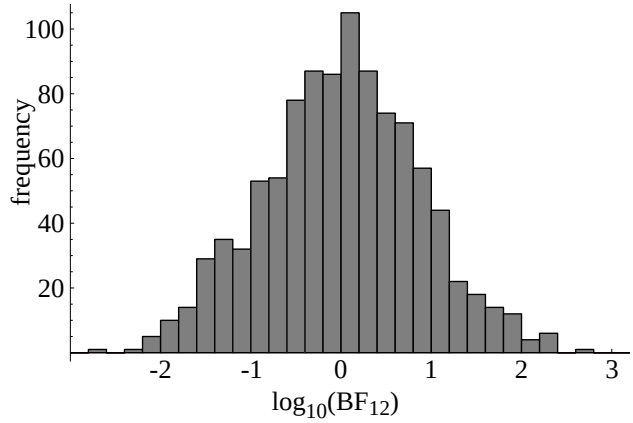

Fig. 6. Variation in Bayes factor due to trace variations of the order of 5% added to a linear model. Note the logarithmic scale. The wide variations in the Bayes factor would lead to unjustifiably strong conclusions about the optimal positioning of the seismic data if their (at least partial) origin in pure chance is not considered.

with respect to the MAP Hessian–derived standard deviations attached to the positioning parameters.

Appendix 5: Detail on the time and depth fields in the log, survey and checkshot data, plus output time scales

The extraction code has two notions of depth: i) one connected to the depth column of the log files ("MD"), which must be consistent between log files, the survey, and the checkshot, and ii) one tied to some notion of TVD (usually subsea (SS), for a marine survey), which is obtained from the survey files. The link between the two is the survey file, which contains both an MD and TVD column. It doesn't really matter where MD is measured from (e.g. the kelly bushing (KB), or perhaps the surface), so long as the "MD" used in the survey file is the same as the "MD" in the log file and the "MD" in the checkshot file. It also doesn't matter exactly what the datum is for TVD; the outputs are given in TVD as well as MD for user convenience.

For time scales, the time in the checkshot must have the same meaning as the time in the seismic minicubes, so this is usually TWT vertical (usually SS for marine data). Service companies that provide checkshot data virtually always provide a vertical TWT field and a MD field, but occasionally one is unlucky and the MD field has been lost in favour of the TVD, in which case the MD field will need to be recovered from the TVD entries, the survey file, and some interpolation tool such as Matlab/Mathematica. Most commonly, one ends up with MD from KB, and TVD from SS, and TWT vertical.

Users need to be careful in interpreting seismic trace outputs when the stack–

specific time-registration parameters are used. The parameters tagged <time_to_depth_local_systematic_time_shift_mean_ms> in the XML file translate the seismic minicubes in time, *not* the checkshot data. The general parametrisation allows for *different* registration errors for each stack, in which case the registration errors cannot be simply absorbed into a single new time–to–depth mapping. But in the case that there is only one stack, or the option `--synch-stacks` is supplied, there is then a unique registration error applicable for each well, and this can in principle be absorbed into the time–to–depth map (or checkshot). If the commandline and parameters permit such a possibility, the extractor outputs will include a *second* time scale corresponding to the seismic minicube time scale, which will facilitate interpretations on the origial seismic minicubes. Specifically

- The (`MLsynth_plus_true_seis.*.su`) files will be written *using the original seismic minicube times*. Resetting to checkshot time is always possible by resetting the f1 header word, i.e. subtracting the posterior estimate of the required *t_shift** parameter in the matching `MLparameters.*` file.
- The `MLgraphing-file.*` files have the extra seismic time scale, which the *extractionViewer* will display clearly.
- The `MLtime-to-depth-map.*.txt` files (a–posteriori time–to–depth maps) will contain an extra seismic time column.
- If requested (`-b`), the blocked log files (`*_blocked.txt`) will also contain the checkshot time scale and seismic time scale.

Appendix 6: Correlated noise estimates

In the case of multi–stack extractions, the error process $\Delta\mathbf{y} = \mathbf{y} - f(\mathbf{m})$ central to the discussion of Appendix 1, which contains two blocks of entries for "near" and "far" stacks, will be correlated between stacks. This is due to common terms in both the forward physics model and the imaging algorithms. Since the prior information on the noise parameters is very weak, the MAP estimate of the noise σ_n for each stack j can be shown to be

$$\sigma_{nj}^2 = \frac{1}{N_{\text{stack } j}} \sum_i (y_i - f(\mathbf{m})_i)^2,$$

coinciding with the vanishing of the RHS of equation (13). Defining the matrix $(\Delta\mathbf{y})_{ji} = y_{ji} - f(\mathbf{m})_{ji}$ for stack j and samples i running over the (subsampled) seismic–mismatch amplitudes, the full noise covariance matrix MAP estimate can be defined as

$$(\boldsymbol{\sigma}_n^2)_{jj'} = \frac{1}{N_{\text{stack}}} \sum_i (y_{ji} - f(\mathbf{m})_{ji})(y_{j'i} - f(\mathbf{m})_{j'i}),$$

taking (as is practically the case) $N_{\text{stack } j} = N_{\text{stack}}$ for all stacks. More succinctly, we can write

$$\boldsymbol{\sigma}_n^2 = \frac{1}{N_{\text{stack}}} \Delta\mathbf{y}\Delta\mathbf{y}^T,$$

and further define

$$\boldsymbol{\sigma}_n^2 = \text{diag}(\boldsymbol{\sigma}_n^2)^{1/2}.\boldsymbol{\rho}.\text{diag}(\boldsymbol{\sigma}_n^2)^{1/2},$$

so $\text{diag}(\boldsymbol{\sigma}_n^2)^{1/2}$ is a diagonal matrix of the "noise standard deviations" and $\boldsymbol{\rho}$ is a symmetric correlation–coefficient matrix (with unit diagonals) whose off-diagonals encapsulate the correlation between stacks. The $\text{diag}(\boldsymbol{\sigma}_n^2)^{1/2}$ estimates appear as *noise_stack_** entries in the `MLparameters.*` files, and the off-diagonal elements of $\boldsymbol{\rho}$ are named *noise_corr_coeff_*_** . Experience shows that typical correlation–coefficient entries might be around $\rho \sim 0.3 - 0.4$.

6.1 Applications

Correlated noise estimates can in principle be introduced to applications like *Delivery* (Gunning and Glinsky, 2004) or sparse spike inversion. In general, the log–posterior or objective function will look like

$$\chi^2 = \frac{1}{2}\Delta\mathbf{y}^T.(\boldsymbol{\sigma}_n^2)^{-1}.\Delta\mathbf{y} + \text{Bayesian prior terms or regularisation terms}$$

and we expect the effect of increasing correlation in the stack–error-processes to decrease the precision of estimated parameters, since the extra data's partial redundancy is being explicitly taken into account. In the limit of full correlation, the posterior estimates will be equivalent to only a single stack inversion. The advantage of this formalism is that it prevents the creation of artificially precise estimates of inversion parameters via "double–dipping" into the data.