Asymptotic complexity of inverse nonstationary phase shift

Marcus R. Wilson and Robert J. Ferguson

ABSTRACT

We perform a speed test on an optimized conjugate-gradient based inversion to correct for surface statics and irregular spatial sampling. An ideally preconditioned scheme is run repeatedly on synthetic trace gathers of random size, up to 2^{15} traces. The runtime and the number of conjugate gradient iterations required for each trial is recorded. Iteratively refined polynomial regression is performed on the resulting data points to estimate the asymptotic complexity of the algorithm, and the number of conjugate gradient iterations is compared with the overall runtime to check for consistency. We find that the ideal preconditioned scheme gives a near optimal runtime of $\mathcal{O}(n^{1.082})$, which indicates that the method could feasibly be preconditioned to run on large data sets.

INTRODUCTION

A wave equation inversion for seismic data described by Ferguson (2006) recursively computes the extrapolated wavefield at depth using nonstationary phase shift operators (Ferguson and Margrave, 2002), which generalize the phase shift method of Gazdag (1978) to media where velocity varies laterally. The operator matrix is computed using an assumed velocity model and the wavefield at depth is computed using weighted damped least squares (Tarantola, 2005). In one inversion, this method corrects common shot and common receiver gathers for topography and near-surface heterogeneity, and interpolates missing traces.

The algorithm successfully generalizes phase-shift wave extrapolation to cases of laterally varying medium velocity and irregular trace coverage, but it carries a larger computational cost. Basic implementation of the algorithm requires computation of matrix representations of the nonstationary phase shift operators. Normal equations must then be formed by matrix multiplication, and the result inverted to extrapolate the wavefield. These computations are $\mathcal{O}(n^3)$ for a 2D shot record of n traces. For 3D surveys, the cost is also $\mathcal{O}(n^3)$, but n refers to the total number of traces in the output wavefield, given by $n = n_{inline} \times n_{xline}$. For large 3D surveys, this is computationally infeasible, as traces can number in the millions.

Ferguson (2006), Ferguson (2010) and Smith et al. (2009) target stages of the algorithm that carry the largest computational cost, and seek approximations that reduce the total effort. Ferguson (2006) restricts computation of the forward and adjoint matrices to d diagonals and sets the remaining entries to zero. This reduces the cost of constructing the Hessian by matrix multiplication to $\mathcal{O}(dn^2)$. Ferguson (2010) derives a series expansion to compute the entire Hessian in $\mathcal{O}(n^2 \log n)$. In both cases, direct matrix methods are used to invert the resulting matrix, so the complexity of these methods is $\mathcal{O}(n^3)$. Smith et al. (2009) inverts the Hessian matrix using conjugate gradients, with complexity $\mathcal{O}(C(n)n^2)$, where $C(n)$ is the number of conjugate gradient iterations required for an acceptable solution, which can be expected to increase with n. Finally, Wilson and Ferguson (2010b)

implements a conjugate gradient scheme using a programmed function in place of a matrix to further decrease the runtime of the inversion to $\mathcal{O}(C(n)vn\log n)$, although convergence of the conjugate gradient iterations was slow in the lower frequencies. Each method results in significant runtime reductions, but due to the unknown function $C(n)$, theoretical asymptotic runtimes could not be derived.

In this paper, we discuss the process of inversion of the phase shift operator by conjugate gradients on normal equations, and conduct a feasibility study to determine if it is worth pursuing a fast and accurate preconditioned scheme. The goal is to minimize computational complexity and mathematical error so that the result provides a stable platform for subsequent imaging on large data sets.

THEORY

Ferguson (2006) presents a simultaneous data regularization, elevation statics and datuming method that uses the non-stationary phase shift operators of Gazdag (1978) and Margrave and Ferguson (1999). Here we summarize the development of these operators and their applications. We then discuss the runtime of various numerical methods to speed the application of these operators, which will show us how to evaluate the asymptotic runtime of the method.

Non-stationary Phase Shift Operators

Given a monochromatic wavefield $\varphi_z = \varphi(x,z,\omega)$ measured at some depth z across all lateral coordinates x, the extrapolated wavefield at depth $\varphi_{z+\Delta z}$ can be computed using the Fourier integral operator given by (Ferguson, 2006),

$$\begin{aligned}\varphi_{z+\Delta z} &= [P_{\Delta z}]\,\varphi_z \\ &= \int e^{2\pi i \Delta z k_z}\hat{\varphi}(k_x,z,\omega)e^{-2\pi i x k_x}dk_x,\end{aligned} \tag{1}$$

where $\hat{\varphi}(k_x,z,\omega)$ is the spatial Fourier transform of $\varphi(x,z,\omega)$, given by

$$\hat{\varphi}(k_x,z,\omega) = \int \varphi(x,z,\omega)e^{2\pi i x k_z}dx. \tag{2}$$

The vertical wavenumber k_z must be chosen to satisfy the dispersion relation (Ferguson and Margrave, 2002)

$$k_x^2 + k_z^2 = \left(\frac{\omega}{v}\right)^2, \tag{3}$$

where v is the velocity of the medium. We can choose the sign of k_z so that the operator propagates the wavefield in the direction of Δz in the wavelike region, where $|\frac{\omega}{v}| \leq |k_x|$, and attenuates energy in the evanescent region, where $|\frac{\omega}{v}| > |k_x|$. These conditions are satisfied in Ferguson (2010), where k_z is given by,

$$k_z = \mathrm{Re}\left\{\sqrt{\left(\frac{\omega}{v}\right)^2 - k_x^2}\right\} + i\,\mathrm{sgn}(\Delta z)\,\mathrm{Im}\left\{\sqrt{\left(\frac{\omega}{v}\right)^2 - k_x^2}\right\}. \tag{4}$$

When φ_z is a discretely sampled wavefield, such as would be measured in a seismic experiment, we can think of Equation 1 as taking the Fourier transform of the data, followed by pointwise multiplication by an appropriate sampling of the function $\alpha(\Delta z, k_z) = e^{2\pi i \Delta z k_z}$, known as the *symbol*, and then the inverse Fourier transform of the result. Each of these operations can be encoded in a matrix.

$$\varphi_{z+\Delta z} = [IFT][\alpha_{\Delta z}][FT]\varphi_z, \tag{5}$$

where $[\alpha_{\Delta z}]$ is a diagonal matrix, with the sampling of the symbol on the diagonal.

To accommodate velocity variation in depth, the medium is divided into a series of horizontal slabs of constant velocity (Gazdag, 1978). The Fourier integral operator is used to propagate the wavefield through each slab, and continuity of displacement is assumed to move the wavefield between adjacent slabs. For lateral velocity variation, it suffices to allow v in Equation 4 to vary with x. This complicates the computation of Equation 1, however, as the symbol must then be recalculated for each output location, and the subsequent inverse Fourier transform cannot be calculated quickly. Margrave and Ferguson (1999) accommodate lateral velocity variation using a set of constant velocity windows. The window function is defined for a given reference velocity v by

$$\Omega_v(x) = \begin{cases} 1 & \text{if } v(x) = v \\ 0 & \text{if } v(x) \neq v \end{cases}, \tag{6}$$

and Equation 5 becomes

$$\varphi_{z+\Delta z} = \sum_{v \in V} [\Omega_v][IFT][\alpha_{\Delta z, v}][FT]\varphi_z. \tag{7}$$

Statics and Trace Regularization

Ferguson (2006) presents an application of this phase-shift operator to correct for topography and near-surface heterogeneity, and for irregular trace spacing. Seismic data are modelled as a recursive upward phase shift of the data at depth, followed by the addition of random noise and setting a selection of the traces to zero to model irregular surface coverage. Due to the trace decimation the data can not be reliably phase shifted back down, so the problem is solved using weighted damped least squares (Menke, 1989).

$$\left[P^*_{-\Delta z} W_e P_{-\Delta z} + \varepsilon W_m\right] \varphi_{z+\Delta z} = P^*_{-\Delta z} W_e \varphi_z. \tag{8}$$

Here $P_{-\Delta z}$ is an upward application of the phase shift operator (Equation 7), $P^*_{-\Delta z}$ is the adjoint of $P_{-\Delta z}$, W_e models irregular trace coverage, W_m is a smoothing operator, and ε is a user parameter that controls the amount of smoothing (Menke, 1989). These equations can be solved for the unknown wavefield $\varphi_{z+\Delta z}$ given the known wavefield φ_z. Since W_e is not invertible, the smoothing operator W_m can be used to constrain the system to give preference to solutions that satisfy some user-defined notion of smoothness. Note that W_m acts as a stabilizing factor in Equation 8, which can force the linear system to have a unique

solution (Margrave, 2009). The matrix $H_{\Delta z} = \left[P^*_{-\Delta z} W_e P_{-\Delta z} + \varepsilon W_m\right]$ is often referred to as the Hessian matrix.

As presented, this algorithm is computationally expensive, so optimization strategies must be considered. Ferguson (2006), and Ferguson (2010) explore different approximations to speed computation of the Hessian matrix, and Smith et al. (2009) describes an implementation of this algorithm that uses Conjugate Gradients to speed inversion. These authors demonstrate the utility of this algorithm in reconstructing missing traces and accounting for near-surface velocity effects, and they report decreased runtime as a result of their specific design features. An asymptotic complexity function is not constructed in these works. Such a function might be helpful in determining the optimum design and runtime of the method.

Preconditioned Conjugate Gradients

Many techniques exist to solve linear systems, and the cost of solving the normal equations (Equation 8) will vary depending on the inversion method used. Matrix methods such as Gaussian elimination and LU factorization are widely used due to their versatility and ease of use, but require the matrix form of the operator to be inverted and generally carry the largest computation cost (Burden and Faires, 2001). When speed is desired, iterative methods such as conjugate gradients can be used to compute an approximate solution with fewer computations.

For ease of notation, denote by H the Hessian operator on the left-hand side of Equation 8, and b the transformed wavefield vector on the right-hand side. The problem can then be written as a linear system given by,

$$Hx = b, \tag{9}$$

where we wish to compute the unknown vector x. For a thorough discussion of the various computational options, see Wilson and Ferguson (2010a). For our purposes it suffices to note that, given an $n \times n$ matrix H which represents the Hessian operator, computing Hx for a given vector x requires $\mathcal{O}(n^2)$ operations, whereas computing the same output using a preprogrammed function that uses the fast Fourier transform can be accomplished in $\mathcal{O}(vn \log(n))$ operations, which is a significant speedup for large values of n.

Given the matrix form of H and the data vector b, we could solve for the unknown vector x by Gaussian elimination, at a cost of $\mathcal{O}(n^3)$. This is infeasible for large values of n, which are likely to be encountered in large 3D surveys. In this case, we can turn to conjugate gradients to reduce this cost, provided the system is positive definite (or semi-definite) and symmetric, as is the case when working with normal equations.

To solve a linear system by conjugate gradients we choose an initial value x_0, and compute the residual vector $r_0 = b - Hx_0$. The cost of this step is dominated by the cost of applying the operator H to x_0. This residual vector defines a search direction that we use to refine our guess. Subsequent iterations are similar, except the search directions are adjusted to take advantage of the positive definite structure of the operator H. Given perfect arithmetic, this method is guaranteed to produce an exact solution to the system after n iterations (Hestenes and Stiefel, 1952), and an acceptable approximation can be

attained using machine arithmetic in fewer iterations if the matrix is well conditioned. In this case, we should be able to solve the matrix form of the system in $\mathcal{O}(C(n)n^2)$, and the functional form in $\mathcal{O}(C(n)vn\log(n))$, where $C(n)$ is the number of conjugate gradient iterations required. This is a significant cost decrease when n is very large.

Note that we expect $C(n)$ to increase with n, although the way $C(n)$ changes with n is not obvious. We expect that $C(n) < n$, and Burden and Faires (2001) states that for a well conditioned system, $C(n) \approx \sqrt{n}$. Thus we are left to try to measure $C(n)$ for small values of n, and use polynomial regression to estimate its behaviour for larger values.

If the system is particularly sensitive to rounding errors, this method might not find a solution to the system quickly, and may fail to find an acceptable approximation at all. We call such a system "ill conditioned." In fact, Wilson and Ferguson (2009) and Wilson and Ferguson (2010a) implement this algorithm with no preconditioning, and note that the algorithm tends to converge quickly in the high frequencies, where no evanescent filter is applied, and very slowly or not at all in the lower frequencies. Such a system must first be preconditioned. For a complete discussion on preconditioning, see Wilson and Ferguson (2010a). For our purposes, it suffices to imagine the "ideally" conditioned operator, with all evanescent filter effects removed. This operator can be run through the inversion method, and the filter can be applied to the result to get the desired output. This operator is obtained by factoring the filter portion of the operator out of the symbol α,

$$\begin{aligned} \alpha_{\Delta z} &= \exp(2\pi i \Delta z k_z) \\ &= \exp(2\pi i \Delta z Re\{k_z\} - |\Delta z| Im\{k_z\}) \\ &= \exp(2\pi i \Delta z Re\{k_z\}) \exp(-|\Delta z| Im\{k_z\}). \\ &= \alpha^P_{\Delta z} \alpha^F_{\Delta z}. \end{aligned} \tag{10}$$

In other words, we can factor the symbol matrix into two diagonal matrices. One performs the phase shift, and one applies the filter. After an approximation and some algebra, we attain a modified set of normal equations given by,

$$\bar{P}^*_{-\Delta z} W_e \varphi_z = \left[\bar{P}^*_{-\Delta z} W_e \bar{P}_{-\Delta z} + \varepsilon (F^{-1})^* W_m F^{-1}\right] F\varphi_{z+\Delta z}, \tag{11}$$

where $\bar{P}_{-\Delta z}$ is just $P_{-\Delta z}$ but with $[\alpha_{\Delta z}]$ replaced by $[\alpha^P_{\Delta z}]$, and F is the filter portion defined by,

$$F = [\alpha^F_{\Delta z}][FT] \tag{12}$$

The inversion can be run on the new well conditioned system to recover $F\varphi_{z+\Delta z}$, and the result can be multiplied by F^{-1} to recover the extrapolated wavefield. Using this preconditioning scheme, the system can be feasibly solved for larger values of n, so we can run the inversion for several sizes of input and try to recover the complexity function. Once we have these functions, it will be simple to estimate the runtime of the algorithm for any more practical preconditioners we might encounter.

Regression

To estimate the complexity function, we require a model of the growth of the system. We expect that the number of iterations required is less than n, and Burden and Faires

(2001) asserts that a well conditioned scheme should converge in approximately $\sqrt{n}$ iterations, so it is reasonable to assume that the iterations function is of the form,

$$C(n) = kn^{\beta}, \tag{13}$$

where $k > 0$ and $0 \leq \beta \leq 1$. So, given a vector of problem sizes $\vec{n}$ and a corresponding vector of the iterations function $\vec{C}(\vec{n})$, we assume that $C_i \approx kn_i^{\beta}$. Taking the logarithm of both sides gives us

$$\log(C_i) \approx \log(k) + \beta \log n_i. \tag{14}$$

We can recover $\log(k)$ and β by computing the least squares solution of the resulting linear system,

$$\log \vec{C} = \begin{bmatrix} \vec{1} & \log \vec{n} \end{bmatrix} \begin{bmatrix} \log(k) \\ \beta \end{bmatrix}. \tag{15}$$

Likewise we assume the total runtime function is given by $t(n) = mn^{\gamma}$, where $m > 0$ and $0 \leq \gamma \leq 3$, and proceed in the same manner.

This regression solves for the line of best fit between $\log(n)$ and $\log(C)$ (or $\log(t)$), so we can expect the error to be roughly uniformly distributed with respect to the logarithm of the data if the relationship is indeed linear. However, when we return the linear system to its original polynomial state, the resulting error will be exponentially greater for larger values of n. We can accommodate this by multiplying the system on the right by a weight matrix that assigns higher weight to the larger values of n, resulting in values of β and k (or γ and m) will more closely model those larger data values than the unweighted system, but this may also worsen the damaging effects of large outliers.

Data Analysis

For this experiment, we choose a random n between 1 and 2^{15}. A synthetic gather of n traces is forward modelled through a random blocky velocity model with $\log_2(n)$ reference velocities. Random noise at a level of -40db is added and 30% of the traces are set to zero to model irregular trace spacing. The inversion is run on the "ideally preconditioned" operator (Equation 11) running on the resulting noisy decimated data. After running, the algorithm returns the number of traces used, the number of conjugate gradient iterations required, the total runtime of the inversion, and the residual error after the inversion. This algorithm is run on a random sample of trace sizes, and the results analysed for trends.

A polynomial regression is performed on the data to generate an initial best fit curve. Weights are declared as the difference between the best fit curve values and the true values, and the weighted regression is then performed on the original data. The results are given in Figure 1.

The average number of CG iterations per frequency is given in Figure 1(a). For this ideal preconditioned scheme, the number of iterations does not seem to grow appreciably with the size of the problem. The blue line indicates the best fit curve, which has constant $k = 11.96$ and exponent $\beta = 0.0158$. The total runtime is given in Figure 1(b), with the best fit line in blue. The best fit curve has constant $m = 0.00931$ and exponent $\gamma = 1.082$.

This data would indicate that the inversion is $\mathcal{O}(n^{1.082})$, and that the CG iterations function $C(n)$ is $\mathcal{O}(n^{0.0158})$, which far exceeds our expectation of $\mathcal{O}(\sqrt{n})$. This is roughly consistent with the theory that the complexity of the inversion is $\mathcal{O}(C(n)vnlog(n))$, although we would expect this estimate to change if we were to compute the runtime function for larger values of n, as our regression does not take into account the $\log(n)$ term.

It is promising to note that $C(n)$ increases so slowly, as it results in a huge decrease in runtime for the inversion. In fact, if a non-ideal preconditioner were to be found that increases the convergence to the target level of $\sqrt{n}$ without sacrificing accuracy, we could expect a total runtime in the neighborhood of $\mathcal{O}(vn^{1.5}log(n))$, which is a huge speedup over previously derived methods.

Extrapolating this function to larger values of n, we could estimate that running this inversion on $1,000,000$ traces would take approximately eight hours. That is eight hours per depth step per trace gather. This sounds grim until we note that the constant m can be viewed as a measure of the user's computer architecture and program design. This algorithm runs independently on each frequency, so it parallelizes trivially, and we could split the work of this experiment between up to 122 nodes - one node per frequency imaged - effectively dividing m by 122. Using a pre-compiled programming language and enterprise level computing power could shrink m even more, until a call of this function runs in minutes or seconds.

CONCLUSION

Regression analysis on our observed data indicates that the asymptotic runtime of our ideal preconditioned scheme is approximately $\mathcal{O}(n^{1.082})$. Ideal preconditioning results in an iterations function $C(n)$ that is almost constant time, so complexity is dominated by cost of applying the forward operator, which is itself very fast. This suggests that if a preconditioner were to be found that improves convergence of the conjugate gradient scheme to the target of $\mathcal{O}(\sqrt{n})$ without sacrificing accuracy, the resulting algorithm could quickly and accurately interpolate seismic traces for subsequent imaging in an everywhere varying velocity medium.

ACKNOWLEDGEMENTS

The authors wish to thank the sponsors, faculty, staff and students of the Consortium for Research in Elastic Wave Exploration Seismology (CREWES), and the Natural Sciences and Engineering Research Council of Canada (NSERC, CRDPJ 379744-08) for their support of this work.

REFERENCES

Burden, R., and Faires, J. D., 2001, Numerical Analysis: Brooks/Cole, 511 Forest Lodge Road, Pacific Grove, CA 93950 USA.

Ferguson, R. J., 2006, Regularization and datuming of seismic data by weighted, damped least squares: Geophysics, **71**, No. 5, U67–U76.

Ferguson, R. J., 2010, Efficient simultaneous wave-equation statics and trace regularization by series approximation: Geophysics, **75**, No. 3, U19–U28.

Ferguson, R. J., and Margrave, G. F., 2002, Prestack depth migration by symmetric nonstationary phase shift: Geophysics, **67**, No. 2, 594–603.

Gazdag, J., 1978, Wave equation migration with the phase-shift method: Geophysics, **43**, No. 7, 1342–1351.

Hestenes, M. R., and Stiefel, E., 1952, Methods of conjugate gradients for solving linear systems: Journal of Research of the National Bureau of Standards, **49**, No. 6.

Margrave, G. F., 2009, Methods of Seismic Data Processing: University of Calgary Bookstore, 2500 University Drive NW, Calgary Alberta, T2N 1N4.

Margrave, G. F., and Ferguson, R. J., 1999, Wavefield extrapolation by nonstationary phase shift: Geophysics, **64**, No. 4, 1067–1078.

Menke, W., 1989, Geophysical Data Analysis: Academic Press, 525B Street, Suite 1900, San Diego, California 92101-4495.

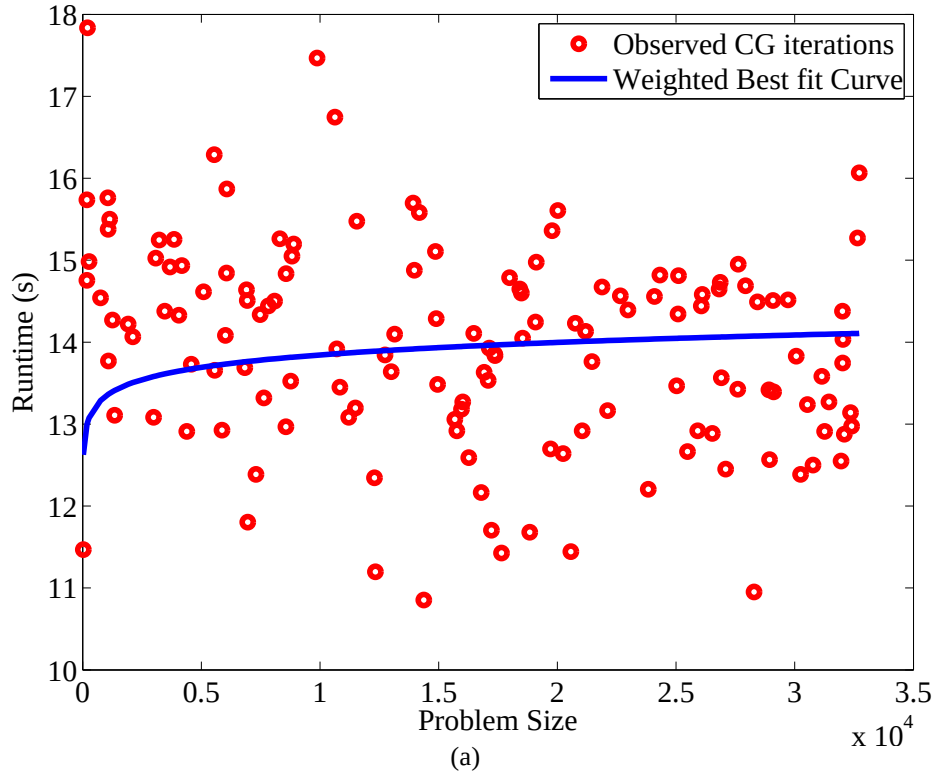
Smith, D. R., Sen, M. K., and Ferguson, R. J., 2009, Optimized regularization and redatum by conjugate gradients: 71st EAGE Conference & Exhibition, **71**.

Tarantola, A., 2005, Inverse problem theory and methods for model parameter estimation: Society for Industrial and Applied Mathematics, 3600 University City Science Center, Philadelphia, PA 19104-2688.

Wilson, M. R., and Ferguson, R. J., 2009, Trace interpolation and elevation statics by conjugate gradients: CREWES Research Report, **21**.

Wilson, M. R., and Ferguson, R. J., 2010a, Implicit and explicit preconditioners for fourier integral operators: CREWES Research Report, **22**.

Wilson, M. R., and Ferguson, R. J., 2010b, Trace interpolation and elevation statics by conjugate gradients: GeoCanada 2010 Conference Abstracts.



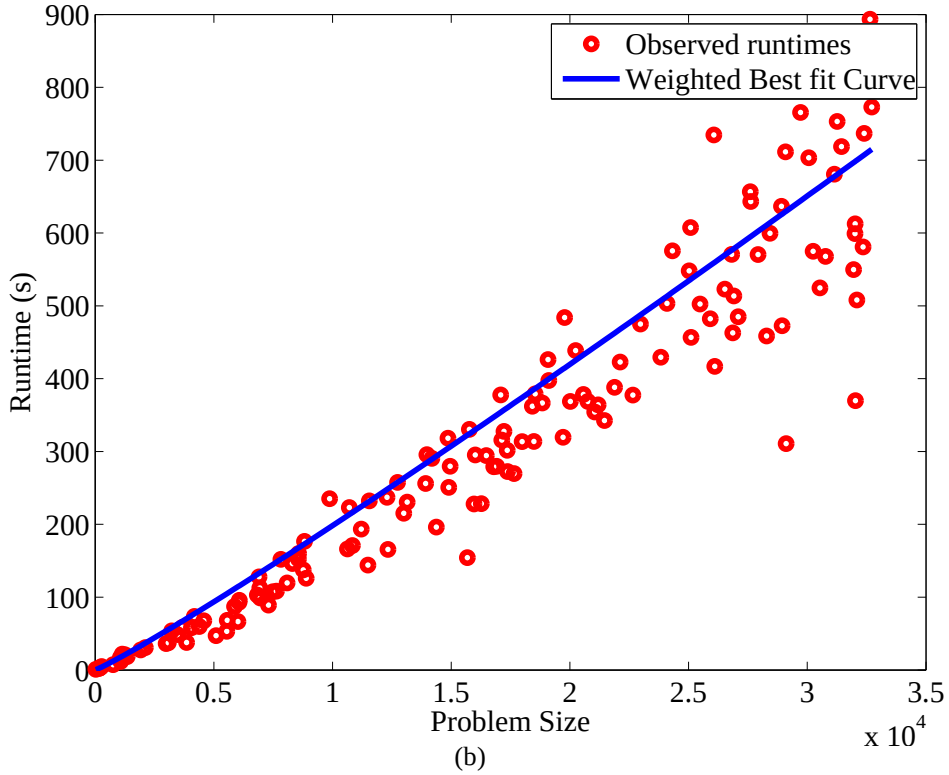


FIG. 1. (a) The number of CG iterations vs the number of traces. (b) The total runtime of the inversion vs the number of traces. Regression estimates that the iterations function is given by $C(n) = 11.96n^{0.0158}$, and the runtime function is given by $t(n) = 0.00931n^{1.082}$