Available online at www.sciencedirect.com
ScienceDirect
Computational Materials Science 40 (2007) 345–353
COMPUTATIONAL MATERIALS SCIENCE
www.elsevier.com/locate/commatsci

Enhancing structure relaxations for first-principles codes: An approximate Hessian approach

James M. Rondinelli [1], Bin Deng, Laurence D. Marks *

Department of Materials Science and Engineering, Northwestern University, 2220 N Campus Drive, Evanston, IL 60208-3108, USA

Received 15 August 2006; received in revised form 11 January 2007; accepted 16 January 2007
Available online 13 March 2007

Abstract

We present a method for improving the speed of geometry relaxation by using a harmonic approximation for the interaction potential between nearest neighbor atoms to construct an initial Hessian estimate. The model is quite robust, and yields approximately a 30% or better reduction in the number of calculations compared to an optimized diagonal initialization. Convergence with this initializer approaches the speed of a converged BFGS Hessian, therefore it is close to the best that can be achieved. Hessian preconditioning is discussed, and it is found that a compromise between an average condition number and a narrow distribution in eigenvalues produces the best optimization.

PACS: 02.60.Pn; 31.15.−p; 31.15.Ar; 71.15.−m

Keywords: DFT; Structure relaxation; Hessian; BFGS optimization

1. Introduction

In many cases the slowest step in a density functional calculation (DFT) or other *ab initio* calculations is finding the optimal atomic positions which minimize the total energy. With older minimization approaches, such as the conjugate gradient method, the number of evaluations scales proportionally with the system size. More powerful are quasi- Newton methods, in particular the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method, which can show quadratic convergence provided that breakdowns of the curvature condition (discussed later) are protected against. Essential to the quasi-Newton methods are estimates for the gradient and curvature of the potential energy surface; the latter being stored in a matrix commonly referred to as the Hessian, which contains all second derivatives (or atomic force constants). The classic BFGS method uses a simple diagonal matrix as the initial Hessian estimate, perhaps with the initial diagonal term using the Shanno–Phua scaling [1]; see also the discussion by Nocedal and Wright [2]. In principle one could achieve far better convergence by some appropriate choice of the initial Hessian estimate, as suggested by some recent analysis [3–6].

In this paper, we detail an approach for improving on the estimate of the starting Hessian, using a harmonic potential describing the interactions between nearest neighbor atoms. We find that it is important to combine this with a diagonal component plus an appropriate scaling term. Slightly unexpectedly, what turns out to be important is a balance between making the initial Hessian estimate replicate that of the true problem and keeping the condition number of the estimate small.

The structure of this note is as follows. First, we briefly review conventional optimization methods (Section 2), with some comments about how they might be improved for density functional theory (DFT) calculations. Second,

* Corresponding author.
E-mail addresses: rondo@mrl.ucsb.edu (J.M. Rondinelli), b-deng@northwestern.edu (B. Deng), L-marks@northwestern.edu (L.D. Marks).
[1] Present address: Materials Department, University of California, Santa Barbara, CA 93106-5050, USA.

0927-0256/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.commatsci.2007.01.004

we outline the algorithm for generating the Hessian estimate and implementing it into the all-electron (linearized) augmented-plane wave + local orbitals (L/APW + lo) package WIEN2k [7] (Section 3). The robustness of the program is tested by performing geometry relaxations for various classes of materials (Section 4). Finally, we conclude with a discussion on the importance of Hessian preconditioning, and we propose a general scheme for resolving these problems.

2. Optimization methods

At the heart of quasi-Newton methods is an expansion of the energy in the form

$$E^{\dagger} = E + \mathbf{g}^{\mathrm{T}}\mathbf{s} + \frac{1}{2}\mathbf{s}^{\mathrm{T}}\mathbf{H}\mathbf{s} \tag{1}$$

where $E^{\dagger}$ is the predicted energy, E and $\mathbf{g}$ are the energy and gradient for a step $\mathbf{s}$ from the current state, and $\mathbf{H}$ is the Hessian matrix. The optimum step can be obtained directly in principle as

$$\mathbf{s} = -\mathbf{H}^{-1}\mathbf{g} \tag{2}$$

assuming that the Hessian is known. The concept of a quasi-Newton method is to calculate an approximation to the Hessian (or its inverse depending upon the exact method used) from previous gradient information. The most successful approaches use what are called secant methods [8], in particular the Broyden–Flecher–Goldfarb–Shanno (BFGS) method [9–12]. The most important contribution from these minimization algorithms is the use of Hessian updating techniques, which allow for the collection of more information about the potential energy surface (PES). In general, after each cycle the Hessian is updated during the minimum search until the convergence criterion is satisfied. It is important to recognize that convergence can be achieved without ever reaching the true Hessian, which suggests that the efficiency of the structure relaxation depends on both the starting geometry and the initial conditioning of the Hessian estimate (discussed later). In fact, the true Hessian of the problem is not always the optimal one, and a compromise between conditioning and accuracy is much more desirable for optimization problems; as Baali has shown much of the success of quasi-Newton methods relies on self-scaling algorithms [13,14]. The first estimate for the Hessian is usually a unitary matrix, although this is not required if physical knowledge of the system is available. For instance, in an earlier version of the WIEN2k code [7] an estimate of the bonding force constants and atom multiplicities was used for the initial diagonal elements—this worked much better than a simple constant. As we will see, one can do better than this.

The mathematics behind the secant method is that a typical iteration for the minimization of $f(x)$ is given by the form

$$x_{k+1} = x_k + \alpha_k d_k \tag{3}$$

where $d_k = -\mathbf{B}_k^{-1}\nabla f(x_k)$ and $\mathbf{B}_k$ is the approximation for the true Hessian that is updated and the step size α_k is chosen by a line search or a trust-region method (as here) [15–17].

For any two consecutive iterations, x_k and x_{k+1}, with their gradients, $\nabla f(x_k)$ and $\nabla f(x_{k+1})$, information about the curvature of the surface (the Hessian) is known since

$$[\nabla f(x_{k+1}) - \nabla f(x_k)] \approx \mathbf{B}_{k+1}[x_{k+1} - x_k] \tag{4}$$

writing $\mathbf{s}_k = x_{k+1} - x_k$ and $\mathbf{q}_k = \nabla f(x_{k+1}) - \nabla f(x_k)$, this can be rewritten as

$$\mathbf{q}_k = \mathbf{B}_{k+1}\mathbf{s}_k \tag{5}$$

The expression given in Eq. (5) is known as the secant equation. An important constraint is that $\mathbf{B}_{k+1}$ needs to be positive definite for the step to be downhill. Multiplying Eq. (5) on the left by $\mathbf{s}_k$ yields what is called the curvature condition $\mathbf{s}_k \cdot \mathbf{q}_k > 0$. This is equivalent to the geometric interpretation that over the step length the object function has positive curvature (i.e. the step is taken in a lower energy direction). When this condition is satisfied, Eq. (5) will always have a solution and the BFGS update

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \Delta\mathbf{B}_k, \quad \Delta\mathbf{B}_k = \frac{\mathbf{q}_k\mathbf{q}_k^{\mathrm{T}}}{\mathbf{q}_k^{\mathrm{T}}\mathbf{s}_k} - \frac{\mathbf{B}_k\mathbf{s}_k\mathbf{s}_k^{\mathrm{T}}\mathbf{B}_k}{\mathbf{s}_k^{\mathrm{T}}\mathbf{B}_k\mathbf{s}_k} \tag{6}$$

will maintain a positive definite approximation to the Hessian.

It is worth mentioning that the curvature condition does not always hold, so it must be explicitly enforced otherwise the BFGS method can fail completely; this is one of the weaknesses of these updating methods. This often occurs when the character of the Hessian changes substantially during the course of the minimization, which is more likely to occur if one starts far from the minimum. Fortunately, the BFGS update is rather well behaved, in that the Hessian estimate will tend to correct itself in a few steps, as compared to other approaches [2]. Three conventional techniques exist for handling the case when the curvature condition fails:

1. The calculations are restarted from the current position with a diagonal initial estimate.
2. A skipping strategy is employed on the BFGS update ($\mathbf{B}_{k+1} = \mathbf{B}_k$).
3. The use of a revised (damped) BFGS update [2] which modifies the definition of $\mathbf{q}_k$.

For the first case, any important curvature information is lost and previous steps are essentially wasted. The second technique allows one to incorporate the curvature information at previous iterations. However, it requires careful control, and too many updates may be skipped resulting in further loss of curvature information. (The limited memory method [18,2] can do this better because it skips steps far from the current location.) The particular code we employed used the third method where the scalar $\mathbf{t}_k$ is