Asynchronous distributed solution of large scale nonlinear inversion problems

V. Pereyra
Weidlinger Associates
4410 El Camino Real, #110
Los Altos, CA 94022, USA

Abstract

In this paper we consider large scale nonlinear least squares problems whose objective functions are very expensive to evaluate, and whose Jacobian matrices are generally ill-conditioned and have an almost block diagonal structure. We prove the convergence of an asynchronous iteration under standard assumptions. The method is naturally parallelizable and thus is applicable to practical inversion problems, as those arising in seismic exploration for hydrocarbons.

1 Introduction

We consider large scale nonlinear least squares problems whose objective functions are very expensive to evaluate, and whose Jacobian matrices are generally ill-conditioned.

For this class of problems, conventional iterative methods that require a large number of iterations to converge are impractical. Because of the ill-conditioning, we would like to use Singular Value Decomposition techniques as a useful tool for dynamic regularization. However, due to the large problem size, direct SVD based methods are also impractical.

On the positive side, the almost block diagonal structure facilitates the application of asynchronous block iterative methods that parallelize naturally. If the block sizes are chosen sufficiently small, then global optimization techniques and direct SVD methods are also feasible at the block level, and if the partitioning is done wisely, it will provide an *a priori* regularization mechanism, as we shall show later on. Regularization can also be introduced at the block level, if necessary, producing a stable, robust, and efficient algorithm.

In [15] we show how some three-dimensional seismic travel time inversion problems of significant practical importance belong to the class of problems that we are considering, and demonstrate the feasibility of our approach by implementing a complete algorithm which uses PVM (Parallel Virtual Machine, a language for distributed computing on networks), to distribute the computation on various multi-processor configurations. The performance on several medium to large scale synthetic problems is also exemplified there.

We have not found in the literature much previous work specifically for this type of problems, with the exception of the excellent research of Diniz-Ehrhardt and J.M. Martinez [5], which discusses a nonlinear version of a Cimmino type algorithm [4]. This leads to a fairly different method from ours; note also that that research is concurrent with ours, first reported in [13] (see also [9, 14]). More recently, Renaut and Mittelmann [16] have proposed a general class of multispliting methods that are related to our approach.

In the seismic inversion literature, large scale problems of this kind arise even in two-dimensions, because of the type of model parametrizations used. Typically, the methods used to solve the linearized problems are of the conjugate gradient type, and simplifications are made to avoid the large cost of recomputing full ray traced data. This often leads to artifacts and has resulted on generalized mistrust for the technique.

In the general minimization arena, the recent work of R. B. Schnabel and his collaborators [17, 7], in the area of molecular dynamic problems shares also some of the flavour of our approach. For a survey on parallel and distributed iterative algorithms see [2].

We would like to consider first some basic results in the theory of iterative methods for nonlinear operators as it pertains to the minimization of nonlinear functionals in R^n. Our aim is to present some precise statements on the convergence of asynchronous iterative methods. Then we go on to show how the methods that we have been developing can make use of this theory.

2 Iterative methods for nonlinear equations

We start with a number of elementary results on fix points and the convergence of standard iterative methods. Upon the introduction of asynchronous block iterations we establish a local convergence theory for such methods. We refer to [6, 1] for more details and additional information.

Definition 1 *The nonlinear operator $T : R^n \longrightarrow R^n$ is a contracting mapping in $X \in R^n$ if there exists $0 < q < 1$, such that for any $\mathbf{x}, \mathbf{y} \in X$:*

$$\| T(\mathbf{x}) - T(\mathbf{y}) \| \leq q \| \mathbf{x} - \mathbf{y} \| .$$

Definition 2 *A vector* $\mathbf{x}_*$ *such that* $\mathbf{x}_* = T(\mathbf{x}_*)$ *is called a* fix point *of* T.

Theorem 3 (Banach Fix Point Principle) *Let* T *be contracting in* X *and let* $T(X) \subseteq X$. *Consider the iteration*

$$\mathbf{x}_{k+1} = T(\mathbf{x}_k), \quad \mathbf{x}_0 \in X.$$

Then, T *has a unique fix point in* X *and the iteration converges to it. Besides, we have the error bound:*

$$\| \mathbf{x}_k - \mathbf{x}_* \| \leq \frac{q^k}{1-q} \| \mathbf{x}_1 - \mathbf{x}_0 \| .$$

Definition 4

- *The operator* $V : X \longrightarrow X$ *is* bounded *if the image of each bounded set* $M \subset X$ *is bounded.*
- V *is* potential *if* $V(\mathbf{x}) = \nabla f(\mathbf{x})$, *for some scalar function* $f(\mathbf{x})$.
- V *is* monotone *if* $0 \leq < V(\mathbf{x}) - V(\mathbf{y}), \mathbf{x} - \mathbf{y} >$, *for each* $\mathbf{x}, \mathbf{y} \in X$.
- V *is* uniformly monotone *if there exists* $0 < m$ *such that:*

$$m \| \mathbf{x} - \mathbf{y} \|^2 \leq < V(\mathbf{x}) - V(\mathbf{y}), \mathbf{x} - \mathbf{y} >,$$

- V *is* Lipschitz *if there exists* $l > 0$ *such that:*

$$\| V(\mathbf{x}) - V(\mathbf{y}) \| \leq l \| \mathbf{x} - \mathbf{y} \| .$$

If V is simultaneously uniformly monotone and Lipschitz then $m \leq l$.

Theorem 5 *Let* $V : R^n \longrightarrow R^n$ *be Lipschitz and uniformly monotone. Then,* $T : R^n \longrightarrow R^n$ *defined by:*

$$T(\mathbf{x}) = \mathbf{x} - \tau V(\mathbf{x}),$$

is contracting for $\tau \in (0, \frac{2m}{l^2})$ *and*

$$\| T(\mathbf{x}) - T(\mathbf{y}) \| \leq q \| \mathbf{x} - \mathbf{y} \|,$$

where $q = (1 - 2m\tau + l^2\tau^2)^{1/2} < 1$.

The minimum value of q is obtained by choosing $\tau = m/l^2$:

$$q_{min} = | \ 1 - (m/l)^2 \ |^{1/2} .$$

3 Convergence of asynchronous iterations

We consider now the basic theory of convergence of asynchronous iterations for the solution of systems of nonlinear equations. Asynchronous iterations were introduced by Chazan and Miranker in [3] for linear systems of equations. We follow Baudet [1] to establish the results for the nonlinear problem that concerns us.

Definition 6 *Let $T : R^n \longrightarrow R^n$. An asynchronous iteration corresponding to the operator T and starting with a vector $\mathbf{x}(0)$ is a sequence $\mathbf{x}(j),\ j = 0, 1, ...,$ of vectors of R^n defined by:*

$$x_i(j) = \begin{cases} x_i(j-1) & if \quad i \notin J_j, \\ t_i(x_1(s_1(j)), ..., x_n(s_n(j))) & if \quad i \in J_j, \end{cases} \tag{1}$$

where $\mathcal{J} = \{J_j \mid j = 1, 2, ...\}$ is a sequence of nonempty subsets of $N = \{1, 2, ..., n\}$ and $\mathcal{S} = \{(s_1(j), ..., s_n(j)) \mid j = 1, 2, ...\}$ is a sequence of elements in N^n.

In addition, $\mathcal{J}$ and $\mathcal{S}$ are subject to the following conditions, for each $i \in N$:

- $s_i(j) \leq j - 1,\ j = 1, 2, ...;$
- $s_i(j) \longrightarrow_{j \to \infty} \infty;$
- *i occurs infinitely often in the sets $J_j,\ j = 1, 2, ...$*

Thus, this type of iteration allows for the asynchronous updating of groups of variables, with some minimal requirements that guarantee that every variable gets updated infinitely often.

Theorem 7 *If T is Lipschitz then it is also vector Lipschitz:*

$$\mid T(\mathbf{x}) - T(\mathbf{y}) \mid \leq A \mid \mathbf{x} - \mathbf{y} \mid,$$

where A is a nonnegative $n \times n$ matrix , $\mid \mathbf{z} \mid$ is the vector $\{\mid z_i \mid\}$, and the inequality above is meant componentwise.

Proof: Consider the l_1 norm $\| \mathbf{x} \| = \sum_{i=1}^n \mid x_i \mid$. Then, if $\| T(\mathbf{x}) - T(\mathbf{y}) \| \leq q \| \mathbf{x} - \mathbf{y} \|$, we can write:

$$\mid T_i(\mathbf{x}) - T_i(\mathbf{y}) \mid \leq \sum_{i=1}^{n} \mid T_i(\mathbf{x}) - T_i(\mathbf{y}) \mid = \| T(\mathbf{x}) - T(\mathbf{y}) \|,$$

and therefore T is vector Lipschitz with matrix $A = \{a_{ij}\} \equiv q$. Obviously, $\rho(A) = nq$, and therefore T is vector contracting iff T is contracting in l_1, with $q < 1/n$.□

Lemma 8 *Let A be nonnegative. Then $\rho(A) < 1$ iff it there exists $\omega > 0$, and a positive vector $\boldsymbol{\nu}$, such that:*

$$A\boldsymbol{\nu} \leq \omega\boldsymbol{\nu},\ \omega < 1.$$

Theorem 9 (Convergence of asynchronous iterations.) *If T is a vector contracting operator on a closed subset $D \subset R^n$, and $T(D) \subset D$, then the asynchronous iteration (1) with $\mathbf{x}(0) \in D$ converges to the unique fix point of T in D.*

Proof: Let $\boldsymbol{\psi}$ be the unique fix point of T in D. By considering the operator $T(\mathbf{x} + \boldsymbol{\psi}) - \boldsymbol{\psi}$ we can assume that $\boldsymbol{\psi} = T(\boldsymbol{\psi}) = 0$. Letting $\mathbf{y} = \boldsymbol{\psi} = \mathbf{0}$ in the Lipschitz condition, it yields: $\mid T(\mathbf{x}) \mid \leq A \mid \mathbf{x} \mid,\ \mathbf{x} \in D$.

Let $\omega, \boldsymbol{\nu}$ be the quantities associated with the contracting matrix A, as defined in Lemma 8. For every starting vector $\mathbf{x}(0)$, we can find $0 < \boldsymbol{\alpha}$, such that

$$\mid \mathbf{x}(0) \mid \leq \alpha\boldsymbol{\nu}. \tag{2}$$

We will show that there is a sequence of indices $j_p,\ p = 0, 1, ...,$ such that

$$\mid \mathbf{x}(j) \mid \leq \alpha\omega^p\boldsymbol{\nu},\ if\ j \geq j_p, \tag{3}$$

which, since $\omega < 1$, shows that $\mathbf{x}(j) \longrightarrow_{j\to\infty} \mathbf{0}$. Let us assume that inequality (3) is valid for $0 \leq j < k$, and consider $\mathbf{x}(k)$. Let $\mathbf{z}$ be the vector with components $z_i = x_i(s_i(k)),\ i \in N \equiv \{1, ..., n\}$. Clearly, for $i \notin J_k$,

$$| x_i(k) | = | x_i(k-1) | \leq \alpha\nu_i.$$

On the other hand, for $i \in J_k$, we have

$$x_i(k) = t_i(\mathbf{z}).$$

Since $s_i(k) < k$, then

$$| T(\mathbf{z}) | \leq A \mid \mathbf{z} | \leq \alpha A\boldsymbol{\nu} \leq \alpha\omega\boldsymbol{\nu},$$

and therefore

$$| x_i(k) | = | t_i(\mathbf{z}) | \leq \alpha\omega\nu_i.$$

Since, $0 < \omega < 1$, this also implies that

$$| x_i(k) | \leq \alpha\nu_i,$$

and we have proven by induction that for all k

$$| \mathbf{x}(k) | \leq \alpha\boldsymbol{\nu}.$$

This proves (3) for $p = 0$. Now, assume that j_p has been found and that (3) holds for $0 \leq p < q$. We want to find j_q such that (3) holds for q.

Let us first define: $r \equiv min\{k$, such that for each $j \geq k,\ s_i(j) \geq j_{q-1},\ i \in N\}$.

Such a number exists because of the second condition on the $s_i(j)$. Besides $r > j_{q-1}$, which shows in particular that

$$| \mathbf{z} | \leq \alpha\omega^{q-1}\boldsymbol{\nu}.$$

From the contracting property of operator T we get:

$$| T(\mathbf{z}) | \leq A \mid \mathbf{z} | \leq \alpha\omega^{q-1}A\boldsymbol{\nu} \leq \alpha\omega^{q}\boldsymbol{\nu},$$

which shows that if $i \in J_j$, then $x_i(j)$ satisfies:

$$| x_i(j) | = | t_i(\mathbf{z}) | \leq \alpha\omega^{q}\nu_i. \tag{4}$$

On the other hand, if $i \notin J_j$ the ith component is not modified. However, by the conditions on $s_i(j)$, we know that each component will eventually be updated, at which point (4) will hold. Thus, let us define: $j_q = min\{j$, such that for $j \geq r$ and $\bigcup_{s=r}^{j} J_s = N\}$. Then, for any $j \geq j_q$, every component will be updated at least once between the rth and jth iterations, and therefore inequality (4) holds for all $i \in N$. This shows that inequality (3) holds for $p = q$ and proves the theorem. □

4 Block Gauss-Seidel-Marquardt iteration

Now we consider the minimization of nonlinear functionals of special type by the Levenberg-Marquardt iterative method (see [?]). Let $\mathbf{F} : R^m \longrightarrow R^n$, and let $f(\mathbf{x}) = \frac{1}{2} \parallel \mathbf{F}(\mathbf{x}) \parallel_2^2$ be twice continuously differentiable in an open convex set $D \subset R^m$. We also assume that the problem will be ill-conditioned and large.

Let us consider the Singular Value Decomposition of $J(\mathbf{x})$:

$$\mathbf{J}(\mathbf{x}) = \mathbf{USV^T}.$$

We introduce the mollified pseudoinverse of $\mathbf{J}$ as:

$$\tilde{\mathbf{J}}(\mathbf{x}, \mu) = \mathbf{V}\mathbf{D}\mathbf{S}^{\dagger}\mathbf{U}^{\mathbf{T}},$$

Here the diagonal matrix $\mathbf{D}$ is defined as:

$$\mathbf{D} = diag\{d_i\} = diag\{\frac{k_i^2}{k_i^2 + \mu^2}\},$$

where $k_i = s_i/s_1$ are the normalized singular values of J. Thus, $\mu > 0$ provides a mollified cutoff point for the normalized singular values (see [8]). With this we can define the Levenberg-Marquardt iteration as:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \tilde{\mathbf{J}}(\mathbf{x}_k, \mu)^{\dagger}\mathbf{F}(\mathbf{x}_k). \tag{5}$$

If n is large (> 100 say), direct Singular Value Decomposition methods will not be practical, and we will loose the advantage that the information on singular values gives to assess and handle the ill-conditioning inherent to these problems. In order to preserve this advantage, and also to open the door to distributed computing methods, we consider a domain decomposition approach for problems of this type that have some additional structure.

The idea is to partition the data space so that for each block of observations only a small part of parameter space is relevant. We have exemplified in Pereyra (1995) how such models occur in geophysical applications. Schnabel (1995) has used similar ideas in molecular dynamics applications.

The tool we use to quantitatively determine which are the relevant parameters associated with a given subset of observations is the Singular Value Decomposition itself, applied to the Jacobian matrix of the residual vector corresponding to the selected observations.

Our hypothesis regarding the essentially local character of the model parametrization, together with the judicious choice of the subsets of observations, imply a priori that there will be many zero columns in the Jacobian matrix, corresponding to parameters that do not influence the value of the residual norm at all for that subset of observations. This has the effect of bringing the sub-problems down to a manageable size.

5 Selecting subspaces through the SVD decomposition

Jupp and Vozoff (1975) introduced the idea of relevant and irrelevant parameters, based on Singular Value Decomposition linearized analysis. We write the Taylor expansion around a given point $\mathbf{x}$:

$$\mathbf{F}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{F}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta\mathbf{x} + \mathbf{R}(\mathbf{x}, \delta\mathbf{x}), \tag{6}$$

where $\mathbf{J}$ is the Jacobian of $\mathbf{F}(\mathbf{x})$ and $\mathbf{R}$ represents high order terms in the perturbation $\delta\mathbf{x}$. From

$$\mathbf{J} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathbf{T}} = \sum_{i=1}^{n} s_i \mathbf{u}_i \mathbf{v}_i,$$

we have that the pseudoinverse of $\mathbf{J}$ can be written as:

$$\mathbf{J}^{+} = \mathbf{V}\mathbf{S}^{\dagger}\mathbf{U}^{\mathbf{T}} = \sum_{i=1}^{r} \frac{1}{s_i}\mathbf{v_i}\mathbf{u_i^T},$$

where $r \leq n$ is the rank of $\mathbf{J}$.

Introducing the *rotated parameters* (in tangent space):

$$\delta \mathbf{p} = s_1 \mathbf{V^T} \delta \mathbf{x},$$

and neglecting the higher order terms, we can write (6) as:

$$\delta \mathbf{F} \approx \mathbf{J} \delta \mathbf{x} = \sum_{i=1}^{r} k_i \delta p_i \mathbf{u_i},$$

which shows the direct relationship between the singular values and the rotated parameters, and also their influence on the variation of the misfit functional.
Thus,

$$\| \delta \mathbf{F} \|_2^2 = \sum_{i=1}^{r} k_i^2 \delta {p_i}^2,$$

since $\| \mathbf{u}_i \|_2 = 1$. This equation shows that the parameters δp_i associated with singular values that are small relative to s_1, will not contribute much to variations in the misfit functional.

That is the key to our algorithm for partitioning the parameter set in blocks, given a partition of the data set in K blocks d^k.

1. Given a block of data, $\mathbf{d^k}$ with m^k elements, and a base point $\mathbf{x}$, calculate $\mathbf{J(x)}$ for this data set (i.e., $\mathbf{J}$ has only m^k rows).
2. Delete all zero columns of $\mathbf{J}$ and calculate its Singular Value Decomposition.
3. Inspect the columns of $\mathbf{V^T}$, and select the largest components in absolute value. Choose the indices of the variables in parameter space corresponding to these entries to form the subset IC_k of parameters that are most influenced by the data set $\mathbf{d}^k$.

The different subsets of parameters may in general overlap. Observe also that it is possible for the union of all the subsets to be smaller than the entire set of parameters. This will indicate that there are parameters that cannot be resolved by the given data, at least in a neighborhood of the base point $\mathbf{x}$. Since this analysis is local, it should be periodically revised as the optimization process advances.

Once this partition has been effected, we use an outer block nonlinear Gauss-Seidel iteration (Ortega and Rheinboldt, 1970), in order to obtain the solution to the full problem.

To make this more precise, let us partition the index sets $\bar{M} = [1, m], \bar{N} = [1, n]$, into the subsets $\{IR_k\}, \{IC_k\}$, i.e., the index sets that describe the partition of our problem into blocks. The sub-problems can now be written as:

$$\min_{\mathbf{x}^k \in \Re^k} \| \mathbf{F^k}(\mathbf{x^k}) \| = \min_{\mathbf{x}^k \in \Re^k} f^k(x^k), \tag{7}$$

where, $\mathbf{x}^k = \{x_j^k\}_{j \in IC_k}$, $\mathbf{d}^k = \{d_i^k\}_{i \in IR_k}$,

$$f_i^k(\mathbf{x^k}) = f_i(\mathbf{x}), i \in IC_k,$$

and $x_i = x_i^k$, $if\ i \in \bar{N} - IC_k$. In other words, we fix the values of the parameters that are not in the vector $\mathbf{x^k}$. Observe that the dimension of the subproblems is then $m^k \times n^k$, where $n^k = |(IC_k)|$, $m^k = |(IR_k)|$ ($|\ .\ |$ stands for the number of elements in a set). By considering enough partitions $k = 1, ..., K$, we can make these dimensions small, especially n^k, and therefore make the sub-problems (7) amenable to global optimization techniques.

6 Sequential block nonlinear Gauss-Seidel iteration

Having partitioned the problem in blocks we can define the sequential nonlinear Gauss-Seidel iteration for solving the least squares problem (7); given $\hat{\mathbf{x}}$, the current value of the parameter vector:

i) For $\ell = 1, ..., ITMAX$ do
For $k = 1, ..., K$ do
$x_i = \hat{x}_i, i \notin IC_k$
$\min_{x_i, i \in IC_k} \| \epsilon^{(k)}(\mathbf{x}) \|_2^2$

ii) Update $\hat{\mathbf{x}}$ by replacing the components $\hat{x}_i,\ i \in IC_k$, with the optimum value from above.

end do k

iii) if $\| \epsilon \|_2^2 = \sum \| \epsilon^{(k)} \|_2^2 \leq eps \quad stop$
end do ℓ

7 Asynchronous block nonlinear Gauss-Seidel iteration

Chazan and Miranker introduced in 1969 the idea of chaotic relaxation for the iterative solution of systems of linear equations. This was extended to the nonlinear case by (Miellou, 1974, 1975).

The purpose of chaotic relaxation was to facilitate the parallel implementation of iterative methods in a multi-processor system or in a network of computers, by reducing the amount of communication and synchronization between cooperating processes, and also by allowing that assigned sub-tasks go unfulfilled.

This was achieved by not requiring that the relaxations follow a pre-determined sequence of computations, but rather the different processes start their evaluations from a current, centrally managed value of the unknowns, and return updated values to this central location as they complete their alloted tasks.

In (Baudet, 1978) the author defines the class of asynchronous iterative methods, and shows that it includes chaotic iterations. We paraphrase a somewhat more restricted definition for the case of block nonlinear optimization that concerns us. Although Baudet's method will apply directly to calculating the zeroes of the gradient of the goal functional, we prefer to describe the method in the optimization context in which it will be used.

Definition 10 *Let* $\mathbf{f}(\mathbf{x}) = \sum_{k=1}^{K} \| \mathbf{F}^k(\mathbf{x}) \|_2^2 = \sum_{k=1}^{L} \mathbf{f}^k(\mathbf{x})$ *be a nonlinear functional from* $\mathbf{R}^n$ *to* $\mathbf{R}$. *An asynchronous iteration for calculating* $\min_{\mathbf{x}} \mathbf{f}(\mathbf{x})$ *starting from the vector* $\mathbf{x}^{(0)}$, *is a sequence* $\mathbf{x}^{(k)}$ *defined recursively by:*

$$\mathbf{x}_j^{(k)} = arg_j \min_{\mathbf{x}} \mathbf{f}^k(\mathbf{x}) \quad if \quad j \in IC_k,$$

$$subject\ to\ x_j^{(k)} = x_j^{(k-1)},\ j \notin IC_k, k = 1, ..., n_k, \tag{8}$$

where IC_k *are subsets of integers in* $[1, n]$.

The initial vector for the kth minimization is $\mathbf{x^{(k-1)}} = (x_1^{s_1(k)}, ..., x_n^{s_n(k)})$, where $S = \{s_1(k), ..., s_n(k)\}$ is a sequence of elements in $\mathcal{N}^n$ that indicates at which iteration a particular component was last updated. In addition, the following conditions should be satisfied:

- $s_i(k) \leq k - 1$,
- $s_i(k) \to \infty,\ as\ k \to \infty$.

These conditions guarantee that all the variables are updated often enough, while the formulation allows for the use of updated subsets of variables "as they become available".

Combining the theoretical results of the previous sections we obtain a convergence theorem for this process.

Theorem 11 Convergence of the asynchronous block Gauss-Seidel- Marquardt iteration.
Let $\mathbf{V}(\mathbf{x}) = \tilde{\mathbf{J}}(\mathbf{x}, \boldsymbol{\mu})^{\dagger}\mathbf{F}(\mathbf{x})$ *be Lipschitz and uniformly monotone in a neighborhood X of a stationary point of* $\mathbf{f}(\mathbf{x})$*, with* $2m/l^2 > 1$*. Then,* $\mathbf{T} = \mathbf{x} - \mathbf{V}(\mathbf{x})$ *is vector contracting and Marquardt's method for minimizing* $\mathbf{f^k}(\mathbf{x})$ *is convergent if* $\mathbf{x^{(0)}} \in \mathbf{X}$*, and the asynchronous iteration (8) is also well defined and convergent.*

Proof: This is a direct consequence of the results of Sections 2 and 3.□

8 A Block nonlinear Jacobi approach

An alternative to the above procedure is to consider a nonlinear block Jacobi iteration. The main difference here is that all the processes will use the same initial vector $\mathbf{x}^{(k-1)}$ at the begining of a sweep, and they are expected to terminate successfully. When all the processes terminate, $\mathbf{x}^{(k-1)}$ is updated. Since there may be multiple contradictory updates for some of the components that appear in more than one block, some kind of weighted average needs to be used.

This implies that the end of a sweep is a synchronization point, making load balancing and the reliability of the processors an important issue. Potentially, the asynchronous theory is also applicable here, so that one could use different strategies to declare a sweep complete, even if some processes have not terminated.

References

[1] Baudet, G.M. (1978), *Asynchronous Iterative Methods for Multiprocessors,* Journal A.C.M., 25, 226-244.

[2] Bertsekas, D.P, and Tsitsiklis, J.N. (1991), *Some Aspects of Parallel and Distributed Iterative Algoritms-A Survey,* Automatica, 27, 3-21.

[3] Chazan, D., and Miranker, W. (1969), *Chaotic Relaxation,* Linear Algebra and Appl., 2, 199-222.

[4] Cimmino, G., 1938, *Calcolo Approssimato per le Soluzioni dei Sistemi di Equazioni Lineari,* La Ricerca Scientifica Ser II, IV, 326-333.

Dennis, J.E., and Schnabel, R.B., *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice Hall, New Jersey, 1983.

[5] Diniz-Ehrhardt, M.A., and Martinez, J.M. (1993), *A Parallel Projection Method for Overdetermined Nonlinear Systems of Equations,* Numerical Algorithms 4, 241-262.

[6] Evtushenko, Y.G., *Numerical Optimization Techniques.* Optimization Software, Inc., New York, 1985.

[7] Feng, D., and Schnabel, R.B. (1993), *Globally Convergent Parallel Algorithms for Solving Block Bordered Systems of Nonlinear Equations,* Optimization Methods and Software, 2, 269-295.

[8] Jupp, D.L.B., and Vozoff, K. (1975), *Stable Iterative Methods for the Inversion of Geophysical Data,* Geophys. J. R. astr. Soc. 42, 957-976.

[9] Koshy, M., Meza J., and Pereyra, V. (1995), *Asynchronous Global Optimization Techniques for Medium and Large Inversion Problems,* SEG 65 Annual Meeting, Houston. Extended Abstracts, 1091-1094.

[10] Miellou, J.-C., (1974), *Iterations Chaotiques a Retards,* Comptes Rendus de l'Acad. Sci. Paris, Ser. A, 278, 957-960.

[11] Miellou, J.-C., (1975), *Algorithmes de Relaxation a Retards,* Revue d' Automatique, Informatique et Recherche Operationelle, 9, 55-82.

[12] Ortega, J.M., and Rheinboldt, W.C., *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, 1970.

[13] Pereyra, V. (1993), *Parallel Block Inversion of Geophysical Data*, Proceedings of Mathematical Methods in Geophysical Imaging, SPIE Intern. Symp. on Optical Applied Science and Eng., San Diego, CA, 2033, 192-198.

[14] Pereyra, V. (1995), *Modeling, Ray Tracing, and Block Nonlinear Travel Time Inversion in 3D.* To appear in Pure and Applied Geophysics.

[15] Pereyra, V., Carcione, L., Vassiliou, A., and Williams, M. (1998), *Improving a Bad Static Problem in the Arapahoe Field by 3D Nonlinear Travel Time Inversion,* submitted to the Leading Edge.

[16] Renaut, R.A., and Mittelmann, H.D. (1995), *Parallel Multisplittings for Optimization.* Parallel Algorithms and Appl., 7, 17-27.

[17] Schnabel, R.B. (1995), *View of the Limitations, Opportunities, and Challenges in Parallel Nonlinear Optimization,* Parallel Computing, 21, 875-905.