Acceleration of the Finite-Element Gaussian Belief Propagation Solver Using Minimum Residual Techniques

Yousef El-Kurdi[1], David Fernández[1,2], Warren J. Gross[1], and Dennis D. Giannacopoulos[1]

[1]Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada
[2]Escuela de Ingeniería Eléctrica, Universidad del Zulia, Maracaibo 4002, Venezuela

The finite-element Gaussian belief propagation (FGaBP) method, introduced recently, provides a powerful alternative to the conventional finite-element method solvers to efficiently utilize high-performance computing platforms. In this paper, we accelerate the FGaBP convergence by combining it with two methods based on residual minimization techniques, namely, the flexible generalized minimum residual and the iterant recombination method. The numerical results show considerable reductions in the total number of operations compared with the stand-alone FGaBP method, while maintaining the scalability features of FGaBP.

Index Terms—**Finite-element method (FEM), Gaussian belief propagation (GaBP), iterative methods, Krylov subspace methods.**

I. Introduction

PARALLEL methods, such as the recently introduced finite-element Gaussian belief propagation (FGaBP) method [1], address the challenging problem of attaining high computational scalability on manycore computing architectures used in high-performance computing platforms. The FGaBP algorithm, when adapted in a multigrid setting [2], demonstrated considerable scalability over the conventional finite-element method (FEM) software. This was a direct consequence of reformulating the FEM as a probabilistic method in FGaBP, where computations are carried out using distributed message updates on a matrix-free data structure.

Most importantly, both the FGaBP method and its multigrid-adapted (FMGaBP) solver [2] eliminate the need to perform global algebraic operations, such as sparse matrix vector multiplies (SMVMs). Nonetheless, to help extend the applicability of the FGaBP solver to a wider range of applications, we here explore combining it using residual minimization with variants of Krylov subspace methods. While this may reduce the distributed behavior of FGaBP by introducing global algebraic operations after a number of FGaBP message update sweeps, our results show that an important reduction in the number of operations can be realized, making this solution very beneficial.

A key challenge emerges when accelerating FGaBP in the context of Krylov methods. The FGaBP solver uses distributed message computation supporting arbitrary update schedules, thus generating an iteration matrix is not tractable. As a result, a conventional Krylov preconditioner approach cannot be used.

In this paper, a residual minimization technique and a flexible Krylov subspace method are used with the FGaBP solver to accelerate its convergence. Our results demonstrate considerable reductions in the overall computational load of FGaBP using these new techniques. The introduced techniques are basically an outer loop over the parallel FGaBP method, as shown in Sections III and IV. Section I presents a brief summary of FGaBP, then the detailed procedure of the two acceleration schemes are described, and finally, we close with results and concluding remarks.

II. Background

The FGaBP formulates the FEM as a variational inference problem by modifying its functional form as follows:

$$\mathcal{P}(U) = \frac{1}{Z}\prod_{s\in\mathcal{S}}\Psi_s(U_s) \tag{1}$$

where Z is a normalizing constant, $\Psi_s(U_s)$ is the local factor function corresponding to each finite element indexed by s in the S set of finite elements, U is the variables vector, and U_s is the subset variables connected to factor Ψ_s. For symmetric positive definite problems, Ψ_s takes a multivariate Gaussian form albeit unnormalized. Correspondingly, the nodal variables are each assumed to model a random Gaussian variable. It can be shown that the solution to the underlying FEM problem can alternatively be obtained by inferring the marginal mean and variance parameters of each of the Gaussian variables in U. This, in turn, motivates the use of the Belief Propagation (BP) algorithm [3] as a computational inference algorithm. The BP is a recursive message passing algorithm that exhibits highly distributed computations by using intermediate results, generally referred to as local beliefs. The resulting FGaBP algorithm communicates messages between variable nodes representing U and factor nodes representing each finite element in a localized matrix-free form. Since the FGaBP messages take Gaussian forms, each message is composed of two parameters: 1) a first-order parameter (β) and 2) a second-order parameter (α). In [2], it was shown that the FGaBP can be adapted into a completely distributed and stationary multigrid process resulting in high computational scalability.

In essence, the FGaBP exploits the inherent structure of the FEM problem resulting in localized computational operations on small matrices of size n or less, where n represents the number of densely connected variables in the resulting computational structure. Most importantly, n is fixed and

Manuscript received July 5, 2015; revised September 3, 2015; accepted September 30, 2015. Date of publication October 6, 2015; date of current version February 17, 2016. Corresponding author: D. Fernández (e-mail: davidmoises@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMAG.2015.2487683

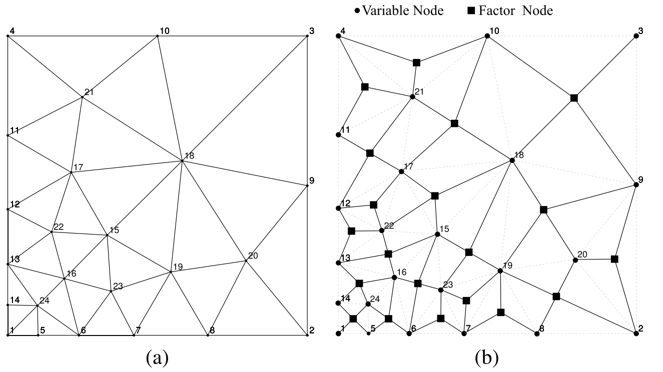

Fig. 1. Sample FEM mesh and a corresponding FEM-FG instance. (a) Sample mesh. (b) FEM-FG instance.

small compared with the total number of variables in the FEM problem resulting in the high scalability of the FGaBP algorithm. This computational structure is reflected in a bipartite graphical modeling approach specific to the FEM problem referred to as the FEM factor graph (FEM-FG). In the FEM-FG graphical model, distributed messages are communicated according to a particular schedule along the edges of the graph between two types of processing nodes: 1) variable nodes and 2) factor nodes. Distributed message schedules can flexibly be varied on the structure of the FEM-FG graph, such as element merging, allowing adaptable memory bandwidth utilization and enhanced overall parallel efficiency [4]. Fig. 1 shows a sample FEM mesh and a corresponding FEM-FG graph.

III. Acceleration Using Iterant Recombination

The FGaBP can be accelerated using message relaxation, as shown in [5], resulting in considerable iteration reductions. However, such an approach can be limited, since the iterative solution is approximated using information only from the previous iteration. In this paper, we aim to obtain better solution approximations using a longer history of previous approximations. The successive solution approximations are obtained using the criterion to minimize the residual. The framework of this method is highlighted in [6, pp. 280–282] and [7] and is referred to as acceleration by iterant recombination (IR). The successive solution estimates $\bar{u}^{(m)}$ at iteration m is obtained as a linear combination of $\tilde{m}$ previous solutions as follows:

$$\bar{u}^{(m)} = u^{(m)} + \sum_{i=1}^{\tilde{m}} a_i (u^{(m-i)} - u^{(m)}). \tag{2}$$

Here, the factors a_i are chosen such that the residual L_2-norm is minimized as follows:

$$a_0 = \underset{a}{\operatorname{argmin}} \left\| d^{(m)} + \sum_{i=1}^{\tilde{m}} a_i (d^{(m-i)} - d^{(m)}) \right\|_2 . \tag{3}$$

This equation shows how to choose the coefficients a_i's of the new solution from (2) such that the new solution minimizes the residual d. The IR method, while presented for multigrid in [6], maybe adapted to other methods as is the case in this paper, where it exhibits great flexibility for the FGaBP algorithm. The FGaBP algorithm, as shown in [2], can be restarted from an arbitrary approximate solution by correspondingly approximating its intermediate messages along the FEM-FG graph. At the IR step, the method needs to perform global operations, such as SMVM and dot products; however, in between the IR iterations, the FGaBP can perform a number of update sweeps maintaining its distributed nature. Here, the SMVM operation utilizes the FGaBP matrix-free data structure without a major impact on memory other than storing the truncated Krylov subspace since typically $\tilde{m} \leq 10$.

```
 1: Generate Factor Node matrices and source vectors
 2: Setup the FGaBP data-structure
 3: Incorporate boundary conditions
 4: Initialize: FGaBP messages $\alpha_{ij} = 1,\ \beta_{ij} = 0,\ \forall i, j$
 5: Define $m$ vectors $D_m$ and $U_m$ (solution)
 6: Define vectors right-hand-side $R$ and correction $C$
 7: Set $R$ equal to system right-hand-side
 8: Define H $(m-1 \times m-1)$, B and a $(m-1 \times 1)$
 9: $R =$ FGaBP.get-right-hand-side$()$
10: Initialize $D_m$ as follows:
11: loop $\{i = 1,$ to $m\}$
12:     FGaBP.compute$(I)$, where $I$ is a fixed inner iteration
        count
13:     $U_i =$ FGaBP.get-solution-vector$()$
14:     $D_i =$ FGaBP.multiply-with$(U_i)$
15:     $D_i = R - D_i$
16: end loop
17: repeat $\{$FGABP iteration: $t = 1, 2, \cdots\}$
18:     Compute $H$ and $B$ using
        Gram-Schmidt orthogonalization
19:     Solve $Ha = B$ (LU with complete pivoting)
20:     Compute solution $\bar{U}$ using (2)
21:     $\bar{D} =$ FGaBP.multiply-with$(\bar{U})$
22:     $\bar{D} = R - \bar{D}$
23:     FGaBP.set-right-hand-side$(\bar{D})$
24:     Initialize the FGaBP messages $\beta_{ij} = 0$
25:     FGaBP.compute$(I)$
26:     $C =$ FGaBP.get-solution-vector$()$
27:     $U_m = \bar{U} + C$, discard oldest $U_1$
28:     $D_m =$ FGaBP.multiply-with$(U_m)$, discard oldest $D_1$
29: until Convergence check
30: Output solution: $U_m$
```

Fig. 2. IR-accelerated FGaBP algorithm.

The IR accelerated FGaBP algorithm is shown in Fig. 2. All designated vectors are of length equal to the total unknowns (N) in the problem. Most of the algorithms computational load is in Steps 12 and 25 where the parallel FGaBP algorithm executes a fixed number of inner iterations (I). Steps 14 and 21 perform SMVM operations using the FGaBP data structure. In Step 24, only the β edge messages are reinitialized to zero, since we are working in the correction space; the α edge messages are left unaltered. Finally, setting up the right-hand side, as in Step 23, involves taking the nodal elements of the vector and evenly distributing

```
1: repeat $\{t = 1, 2, \cdots\}$
2:    Compute $r_0 = b - Au_0$, $\beta = \|r_0\|_2$, and $v_0 = r_0/\beta$
3:    loop $\{j = 1,$ to $m\}$
4:       Compute $z_j = M_j^{-1} v_j$, as FGaBP solve!
5:       Compute $w = Az_j$
6:       loop $\{i = 1,$ to $j\}$
7:          $h_{i,j} = (w, v_j)$
8:          $w = w - h_{i,j} v_j$
9:       end loop
10:      Compute $h_{j+1,j} = \|w\|_2$ and $v_{j+1} = w/h_{j+1,j}$
11:      Define $Z_m := [z_1, \ldots, z_m]$
12:      Define $\overline{H}_m = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le m}$
13:   end loop
14:   Compute $y_m = \arg\min_y \|\beta e_1 - \overline{H}_m y\|$
15:   Compute $u_m = u_0 + Z_m y_m$
16: until Convergence check. If not converged $u_0 = u_m$ and
    goto 1
17: Output solution: $u_m$
```

Fig. 3. FGMRES algorithm with FGaBP preconditioner.

it to the corresponding elements in the source vectors of the factor nodes. It is noteworthy to mention that using the edge α messages as weighting factors in setting the right-hand side can sometimes improve the numerical properties of the algorithm.

IV. Acceleration Using GMRES Preconditioning

The next approach we consider is to use the FGaBP to the effect of a preconditioner for a Krylov subspace method. Preconditioning an iterative Krylov method is somewhat of an art. This is even more so when implementing iterative methods on parallel computing systems. Moreover, by applying FGaBP as a preconditioner to generalized minimum residual (GMRES), we are effectively applying a variable operator as a preconditioner, further complicating this task.

A modified Krylov method of the classical GMRES referred to as the flexible GMRES (FGMRES) method [8, pp. 287–290] can accommodate a dynamically varying preconditioner, such as the FGaBP. The FGMRES computes the solution vector using a linear combination of the preconditioned orthonormalized subspace $z_j = M^{-1} v_j$, as shown in Fig. 3 (line 4). The FGaBP preconditioner is configured to stop in a few iterations ranging from 2 to 5, which becomes a tunable parameter.

The FGaBP solver is used as a right preconditioner solving $AM^{-1}x = b$ system with $u = M^{-1}x$. In this implementation, it is not necessary to produce or store the auxiliary vector x. On the other hand, due to the variability of the FGaBP preconditioner, both the $\{v_j\}$ and $\{z_j\}$ m-vectors, as designated by the GMRES algorithm, need to be generated and stored in order to obtain the solution using a linear combination of these alternate correction spaces as follows:

$$u_m = u_0 + Z_m y_m \qquad (4)$$

where y_m is the vector defining the weights to use in the linear combination of the Z_m correction vectors and $\{v_{m+1}\}$ are the original vectors generated in the orthonormalization process (i.e., Gram–Schmidt). The weights y_m are computed by a minimum residual approach as done for the IR method in equation (3). It is relevant to emphasize that the main difference with the right preconditioned GMRES is the generation and storage of the $\{z_j\}$ m-vectors, not required with a fixed preconditioners. In this second approach, we are again confronted with several SMVM operations that are carried out using the FGaBP framework, thus reducing the impact due to global linear algebra operations.

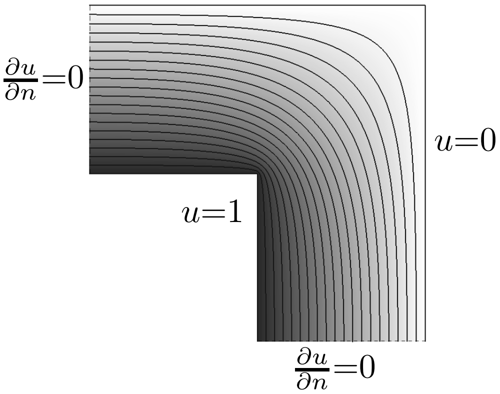
$\frac{\partial u}{\partial n}=0$
$u=0$
$u=1$
$\frac{\partial u}{\partial n}=0$

Fig. 4. Equipotential lines of the top-right symmetrical corner of the square conductor.

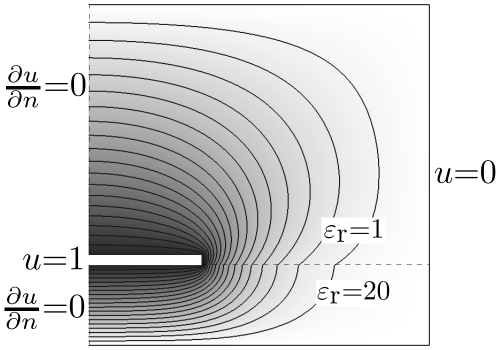
$\frac{\partial u}{\partial n}=0$
$u=0$
$\varepsilon_r=1$
$u=1$
$\varepsilon_r=20$
$\frac{\partial u}{\partial n}=0$

Fig. 5. Equipotential lines of the shielded microstrip conductor between two dielectric media.

V. Results

The behavior of the new algorithms is tested using two experiments. The first experiment is the well-known 2-D square conductor Laplace potential problem that is shown in Fig. 4. The problem uses Dirichlet and Neumann boundary conditions and has a dimension of 1 cm. A quadrilateral mesh is used to discretize the domain containing one of the corners of the square conductor along the two lines of symmetry. The second experiment is the shielded microstrip conductor placed between two dielectric media, as shown in Fig. 5. The top dielectric media is air, while the relative permittivity of the bottom media is varied between 10^6 and 10^{15}. All the experiments were terminated when the normalized residual's L_2-norm dropped below 10^{-6}.

The plots in Fig. 6 show the iteration reduction ratios of the first experiment using the FGaBP accelerated by IR (IR-FGaBP) and FGMRES (FGMRES-FGaBP). Runs are performed for three sets of degrees of freedom (DoFs), each with six variations on FGaBP inner iterations and

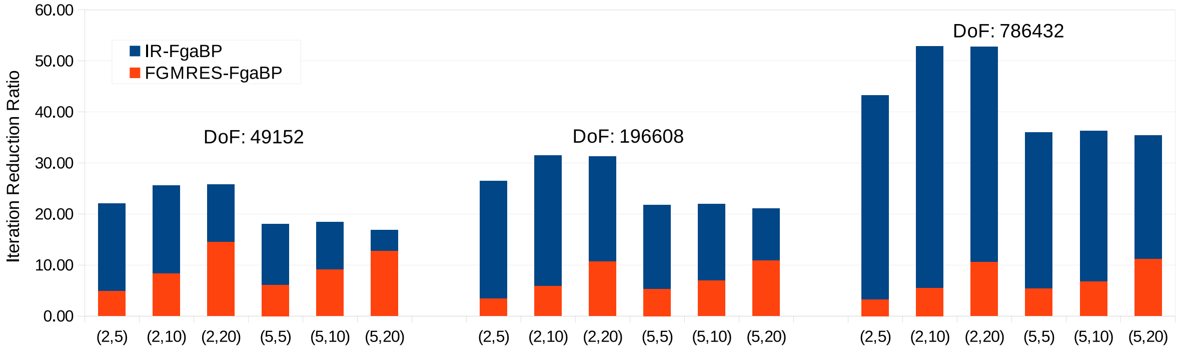

Fig. 6. Iteration reduction ratios of the FGaBP accelerated using the IR and the GMRES methods. The top numbers represent the DoF for each set, and the lower numbers in parentheses (· , ·) represent the inner FGaBP iterations and the size of the subspace, respectively.

TABLE I
SHIELDED MICROSTRIP EXPERIMENT ITERATION REDUCTIONS BY THE IR METHOD

Dielectric ratio	IR(2,10) reduction	IR(5,10) reduction
10^6	41.35	34.58
10^9	46.71	36.42
10^{12}	45.15	38.14
10^{15}	45.44	40.88

three subspace sizes. The numbers in parentheses (· , ·) represent the inner FGaBP iterations and the size of the subspace, respectively. The reduction ratios are obtained by dividing the total number of FGaBP iterations by itself with relaxation [5] over the total iterations of the accelerated method, which indicate the reductions on total floating point operations. The IR-FGaBP obtained the highest ratios on all experiments showing a growing trend of reduction ratios with increasing DoFs. As DoFs increase, the FGMRES-FGaBP benefits from more inner FGaBP iterations, but stagnates rapidly compared with the IR-FGaBP.

Table I shows the iteration reduction factors of the shielded microstrip experiment. Since the IR method showed consistently better results in the first experiment than the GMRES preconditioning, the IR method is primarily used in the second experiment. Here, we vary the relative permittivity of the bottom media while keeping the number of unknowns constant at 341 313. The IR method has produced considerable reductions in all cases; however, no noticeable effect was observed due to varying the relative permittivity.

VI. CONCLUSION

The highly parallel FGaBP algorithm was demonstrated to be amicable for acceleration using variants of Krylov methods resulting in considerable iteration reductions. The details of both acceleration methods, the IR and the FGMRES preconditioning, were presented. The new methods were tested on two different experiments showing considerable reductions in iterations without considerably impacting parallel scalability. For all executed experiments, the IR method showed considerably higher iteration reductions than the FGMRES preconditioning.

It is worth noting that the parallelism exhibited by FGaBP in the previous work is almost retained in the two combined methods proposed (namely, IR-FGaBP and FGMRES-FGaBP), considering that the most costly linear algebra operations are performed in a matrix-free form using special kernels developed within the FGaBP framework.

ACKNOWLEDGMENT

This work is sponsored by the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

[1] Y. El-Kurdi, D. Giannacopoulos, and W. J. Gross, "Parallel solution of the finite element method using Gaussian belief propagation," in *Proc. 15th Biennial IEEE Conf. Electromagn. Field Comp. (CEFC)*, Oita, Japan, Nov. 2012, Art. ID 7014304.

[2] Y. El-Kurdi, W. J. Gross, and D. Giannacopoulos, "Parallel multigrid acceleration for the finite-element Gaussian belief propagation algorithm," *IEEE Trans. Magn.*, vol. 50, no. 2, Feb. 2014, Art. ID 7014304.

[3] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[4] Y. El-Kurdi, M. M. Dehnavi, W. J. Gross, and D. Giannacopoulos, "Parallel finite element technique using Gaussian belief propagation," *Comput. Phys. Commun.*, vol. 193, pp. 38–48, Aug. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010465515001186

[5] Y. El-Kurdi, D. Giannacopoulos, and W. J. Gross, "Relaxed Gaussian belief propagation," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2012, pp. 2002–2006.

[6] U. Trottenberg, C. W. Oosterlee, and A. Schüller, *Multigrid*. San Diego, CA, USA: Academic, 2001.

[7] T. Washio and C. W. Oosterlee, "Krylov subspace acceleration for nonlinear multigrid schemes," *Electron. Trans. Numer. Anal.*, vol. 6, pp. 271–290, Dec. 1997.

[8] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA, USA: SIAM, 2003.