CNRS
Centre National de la Recherche Scientifique

INFN
Istituto Nazionale di Fisica Nucleare

SIESTA implementation of an accelerated convergence scheme for FFT simulations

Massimo Galimberti

Laboratoire des Matériaux Avancés
7, avenue Pierre de Coubertin, 69622 Villeurbanne (FRANCE)

m.galimberti@lma.in2p3.fr

VIR-0658A-10
Issue: 1
November 29, 2010

VIRGO * A joint CNRS-INFN Project
Project Office: Traversa H di Via Macerata, I-56021 S.Stefano a Macerata (Pisa)
Secretariat: Telephone (+39) 050 752 511 * FAX (+39) 050 752 550 * e-mail W3@virgo.infn.it

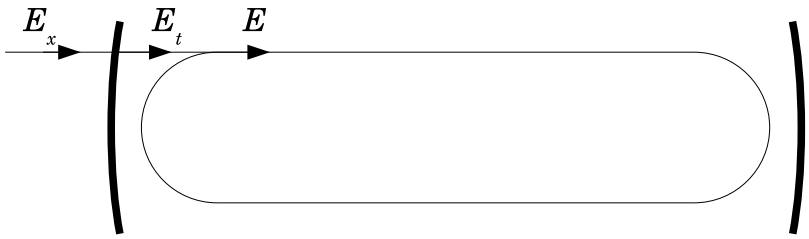

Figure 1: Notation for the fields used throughoute the note.

1 Scope of the document

This note describes how the accelerated convergence scheme for FFT simulations presented in [3] has been implemented in SIESTA, version v5r03. The implementation is described for the case of a simple Fabry-Perot cavity.

Sections 3–4 explain the scheme proposed in Saha's paper [3], from the point of view of a generic linear system. The scheme is then rewritten for the case of a Fabry-Perot cavity. Sections 5–6 deal with implementation in SIESTA and practical issues.

2 The problem

To avoid confusion, throughout this note we adopt for the fields a notation close to that employed in [3].

Consider a Fabry-Perot cavity, as depicted in figure 1. We are interested in finding the steady-state fields. The intracavity field E obeys the implicit equation (see e.g. [1, page 161]):

$$E = \mathcal{R}_1 \mathcal{P} \mathcal{R}_2 \mathcal{P} E + \mathcal{T}_1 E_x \tag{1}$$

where E_x is the input (external) beam, $\mathcal{T}_1$ is the transmittance operator of the input mirror, $\mathcal{R}_1$ and $\mathcal{R}_2$ are the refection operators of respectively the input and end mirrors, and $\mathcal{P}$ is the propagation operator. We will call $\mathcal{C} = \mathcal{R}_1 \mathcal{P} \mathcal{R}_2 \mathcal{P}$ the operator representing a cavity round-trip, and $E_t = \mathcal{T}_1 E_x$ the field entering the cavity.

The solution of equation 1 can be written in the form:

$$E = (\mathrm{I} - \mathcal{C})^{-1} E_t \tag{2}$$

where I is the identity operator. In numerical simulations, the problem becomes that of inverting the matrix $\mathrm{I} - \mathcal{C}$. In modal simulations, where the field E is represented by a vector in a convenient mode basis, this inversion can usually be done by standard numerical methods. On the other hand, in FFT simulations, E is sampled on a xy grid; if the grid has $n \times n$ gridpoints, then the matrix to be inverted has $n^2 \times n^2$ elements, and it can easily be seen that the memory and computation time needed for such an inversion are prohibitive even for "small" grids of 128×128 points. Iterative method are therefore the preferred choice for the solution of equation 1.

3 Review of some iterative methods to solve linear systems

For the content of this section, see e.g. [2], page 108.

Let us suppose we seek a solution, via an iterative method, for the linear system:

$$\mathrm{A}\mathbf{x} = \mathbf{b} \tag{3}$$

(we will go back to the field notation later). A general strategy to solve the system is the *splitting method*, which consists in decomposing A in the form $\mathrm{A} = \mathrm{P} - \mathrm{N}$, where P (called the *preconditioner*) is non-singular. The iterative solution can then be written in the form:

$$\mathrm{P}\mathbf{x}^{(k+1)} = \mathrm{N}\mathbf{x}^{(k)} + \mathbf{b} \tag{4}$$

Let us limit the discussion to the simple case where $\mathrm{P} = \mathrm{I}$. If we call $\mathbf{r}^{(k)} = \mathbf{b} - \mathrm{A}\mathbf{x}^{(k)}$ the *residual* vector at iteration k, equation 4 can be rewritten in the form:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{r}^{(k)} \tag{5}$$

where it is explicit that the solution at iteration $k+1$ is the solution at iteration k plus the residual.

Now suppose that we want to modify the splitting method in order, for a specific linear system, to speed up convergence. One idea is to apply *successive overrelaxations* by introducing a parameter ω_k in equation 5:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \omega_k \mathbf{r}^{(k)} \tag{6}$$

Of course, ω_k must be conveniently chosen in order to accelerate convergence. Equation 6 can be rewritten as:

$$\mathbf{x}^{(k+1)} = (1 - \omega_k)\mathbf{x}^{(k)} + \omega_k(\mathrm{N}\mathbf{x}^{(k)} + \mathbf{b}) \tag{7}$$

where it appears that the solution at iteration $k+1$ is a linear combination of the solution at iteration k and of the solution that would be obtained with the simple relaxation method of equation 4.

Let us call this latter $\mathbf{x}_{\mathrm{SR}}^{(k)}$, and let us generalize the preceding equation to:

$$\mathbf{x}^{(k+1)} = \alpha_k \, \mathbf{x}^{(k)} + \beta_k \, \mathbf{x}_{\mathrm{SR}}^{(k+1)} \tag{8}$$

For a convenient choice of the coefficients α_k and β_k, the overrelaxation method can converge faster than the simple relaxation one.

4 Application to the steady-state solution of a Fabry-Perot cavity

Let us apply what exposed in section 3 to the problem of finding the steady-state field of a Fabry-Perot cavity:

$$(\mathrm{I} - \mathcal{C})E = E_t \tag{9}$$

With respect to the notation in the previous section, $\mathbf{x} = E$, $\mathbf{b} = E_t$. The system matrix $\mathrm{A} = \mathrm{I} - \mathcal{C}$ is already written in its split form, with $\mathrm{P} = \mathrm{I}$, and $\mathrm{N} = \mathcal{C}$. The simple relaxation solution is therefore naturally written as:

$$E_{\mathrm{SR}}^{(k+1)} = \mathcal{C}E^{(k)} + E_t \tag{10}$$

A convergence criterion must be chosen. A natural choice is to ask for the normalized residual to be smaller than a fixed tolerance. Calling $\Delta^{(k)} = E^{(k+1)} - E^{(k)}$ the residual, iterations stop when:

$$\frac{\|\Delta^{(k)}\|}{\|E^{(k)}\|} < \text{tolerance} \tag{11}$$

Now, the first field $E^{(0)}$ of the iterative procedure is of course a "guess" field. The convergence speed of the simple relaxation scheme depends on how close the initial guess field is to the steady-state solution. For an initail guess field far from the steady-state, the iterative procedure is not (much) faster than computing the cavity dynamics, i.e., starting with a null field inside the cavity and letting the circulating field build-up to steady state with time steps equal the round-trip time of the cavity. This is why an accelerated convergence scheme is of interest.

If we apply the overrelaxation method (equation 8) to the cavity equation 9, we can write the overrelaxed iterative solution as:

$$E^{(k+1)} = \alpha_k E^{(k)} + \beta_k E_{\text{SR}}^{(k+1)} \tag{12}$$

The problem is now that of finding a convenient choice of α_k and β_k. Following [3], we choose the values that minimize the norm of the residual at the next iteration $\|\Delta^{(k+1)}\|$. This is a least-square minimization which translates in the solution of the system:

$$\begin{bmatrix} \langle D^{(k)}, D^{(k)} \rangle & \langle D^{(k)}, D_{\text{SR}}^{(k+1)} \rangle \\ \\ \langle D_{\text{SR}}^{(k+1)}, D^{(k)} \rangle & \langle D_{\text{SR}}^{(k+1)}, D_{\text{SR}}^{(k+1)} \rangle \end{bmatrix} \begin{bmatrix} \alpha_k \\ \\ \beta_k \end{bmatrix} = \begin{bmatrix} \langle D^{(k)}, E_t \rangle \\ \\ \langle D_{\text{SR}}^{(k+1)}, E_t \rangle \end{bmatrix} \tag{13}$$

where: $\langle \cdot, \cdot \rangle$ indicates the scalar product; $D^{(k)} = E^{(k)} - \mathcal{C}E^{(k)}$; and $D_{\text{SR}}^{(k+1)} = E_{\text{SR}}^{(k+1)} - \mathcal{C}E_{\text{SR}}^{(k+1)}$.

5 SIESTA implementation

From a practical point of view, this is the scheme of the algorithm implemented in SIESTA:

1. start iteration k, knowing from iteration $k-1$:
   - the coefficients α_{k-1}, β_{k-1}
   - the fields $E^{(k-1)}$, $E_{\text{SR}}^{(k)}$
   - the fields $\mathcal{C}E^{(k-1)}$, $\mathcal{C}E_{\text{SR}}^{(k)}$
2. compute field $E^{(k)} = \alpha_{k-1}\, E^{(k-1)} + \beta_{k-1}\, E_{\text{SR}}^{(k)}$
3. compute field $\mathcal{C}E^{(k)} = \alpha_{k-1}\, \mathcal{C}E^{(k-1)} + \beta_{k-1}\, \mathcal{C}E_{\text{SR}}^{(k)}$
4. compute field $E_{\text{SR}}^{(k+1)} = \mathcal{C}E^{(k)} + E_t$
5. compute residual $\Delta^{(k)} = E_{\text{SR}}^{(k+1)} - E^{(k)}$

6. if $\frac{\|\Delta^{(k)}\|}{\|E^{(k)}\|} <$ tolerance, stop iterations ($E^{(k)}$ is then retained as the approximation to the steady-state solution); otherwise go on
7. compute field $\mathcal{C}E_{\mathrm{SR}}^{(k+1)}$; this is the *only* step where FFT propagation is employed
8. compute $D^{(k)} = E^{(k)} - \mathcal{C}E^{(k)}$, and $D_{\mathrm{SR}}^{(k+1)} = E_{\mathrm{SR}}^{(k+1)} - \mathcal{C}E_{\mathrm{SR}}^{(k+1)}$
9. compute the matrix elements of equation 13 and solve the system to find α_k, β_k (see below for details)
10. restart from point 1 with a new iteration.

Note that, although both simple- and over-relaxation fields are in play, only one round-trip FFT propagation is performed for every iteration. The only exception is iteration 0, where we start with a guess field $E^{(0)}$ equal to the field predicted by theory for perfect infinite mirrors, and let it do a first round-trip to obtain $E_{\mathrm{SR}}^{(1)}$.

One important difference between SIESTA implementation and the theory described in section 4 is the computation of the matrix elements of equation 13. In general, all the elements in equation 13 are complex, and α_k and β_k are complex as well. In SIESTA, it has empirically been observed that the use of complex α_k and β_k makes the algorithm diverge. This is perhaps due to the fact that, since the phase of the circulating field has already been set by adjusting the cavity to resonance, letting α_k and β_k touch the phase of $E^{(k)}$ can only push the circulating field farther from resonance. Therefore, the algorithm has been implemented in order to force α_k and β_k to be real numbers. This has been obtained by solving only the real part of equation 13.

6 Evaluation of convergence speed

As a benchmark to evaluate the effectiveness of the fast convergence scheme for the steady-state of a Fabry-Perot cavity, 1000 simulations have been run with the following configuration (Advanced Virgo arm cavity, as specified in [4]):

- grid size = 400 mm, number of grid points = 128×128
- cavity length = 2998.8 m
- input mirror: diameter = 340 mm; RoC = 1420 m; transmission = 1.4%
- end mirror: diameter = 340 mm; RoC = 1683 m; transmission = 5 ppm
- flatness defects extracted randomly from a 1-D power spectral density of the kind f^{-2}, f being the radial spatial frequency of the defects; the (unweighted) rms value of the defects is 10 nm over a diameter of 340 mm
- input beam = TEM00 matched with a perfect cavity with the same length and RoC's; power = 1 W
- cavity length adjusted to resonance with a tolerance of 10^{-7} rad
- tolerance for the iterative process set to 10^{-6} (i.e., iterations stop when the relative residual is less than 10^{-6}, see equation 11)

	default algorithm	fast algorithm
CPU time (s)	11.2 ± 0.9	4.2 ± 1.0
no. of iterations to converge	1370 ± 62	390 ± 109
circulating power (W)	201.28 ± 15.35	
average difference in circulating power (W)	$8 \cdot 10^{-4}$	
round-trip losses (ppm)	2446 ± 639	
average difference in round-trip losses (ppm)	0.04	

Table 1: Results of the benchmark test, expressed as mean $\pm$ standard deviation over 1000 simulations.

The test has been performed on a PC equipped with a 2.13 GHz Intel Core2 CPU (using only one of the two processors), running Scientific Linux 5. The results of the benchmark are reported in table 1.

We checked that both methods converge to the same solution by looking at the circulating power and the round-trip losses. The difference in these two figures between the default and the fast method is negligible. What is interesting in the results is the number of iterations needed to converge (the CPU time, in fact, takes into account also the time needed to adjust the cavity microscopic length, which does not of course depend on the convergence scheme, and in some cases is a non-negligible fraction of the total CPU time). From the number of iterations, we see that the fast convergence scheme gains on average a factor of about 3.8.

References

[1] The VIRGO physics book, vol. 2, april 2006.

[2] A. Quarteroni, R. Sacco, and F. Saleri. *Matematica numerica*. Springer, 2000.

[3] P. Saha. Fast estimation of transverse fields in high-finesse optical cavities. *J. Opt. Soc. Am. A*, 14(9):2195–2202, 1997.

[4] R. L. Ward. Advanced Virgo optical design parameters summary. VIR-0541A-10, issue 1, october 2010.