SRJ source code:
https://www.dropbox.com/s/utkkm3tsz4pnckp/LaplaceSolver.zip?dl=0

To appear in The Journal of
Computational Physics
DOI: 10.1016/j.jcp.2014.06.010
Available online at
http://www.sciencedirect.com/science/article/pii/S0021999114004173

Acceleration of the Jacobi iterative method by factors exceeding 100 using scheduled relaxation

Xiang Yang, Rajat Mittal

Department of Mechanical Engineering, Johns Hopkins University, MD, 21218

Abstract

We present a methodology that accelerates the classical Jacobi iterative method by factors exceeding 100 when applied to the finite-difference approximation of elliptic equations on large grids. The method is based on a schedule of over- and under-relaxations that preserves the essential simplicity of the Jacobi method. Mathematical conditions that maximize the convergence rate are derived and optimal schemes are identified. The convergence rate predicted from the analysis is validated via numerical experiments. The substantial acceleration of the Jacobi method enabled by the current method has the potential to significantly accelerate large-scale simulations in computational mechanics, as well as other arenas where elliptic equations are prominent.

Keywords: Iterative method, Jacobi method, Elliptic equations

1. Introduction

Elliptic equations appear routinely in computational fluid and solid mechanics as well as heat transfer, electrostatics and wave propagation. Discretization of elliptic partial differential equations using finite-difference or other methods leads to a system of linear algebraic equations of the form $Au = b$, where u is the variable, b the source term, and A, a banded matrix that represents the coupling between the variables. In the context of computational fluid mechanics, which is of particular interest to us, the Poisson equation for pressure appears in the majority of incompressible Navier-Stokes solvers [1, 2], and is by far, the most computationally intensive component of such simulations. Thus, any effective methods that can accelerate the numerical solution of such equations would have a significant impact on computational mechanics and numerical methods.

The early history of iterative methods for matrix equations goes back to Jacobi [3] and Gauss [4], and the first application of such methods to a finite-difference approximation of an elliptic equation was by Richardson [5]. The method of Richardson, which can be expressed as $u^{\mathrm{n}+1} = u^{\mathrm{n}} - \omega_n \left(Au^{\mathrm{n}} - b\right)$, where 'n' and ω are the iteration index and the relaxation factor respectively, was a significant advance since it introduced the concept of

Email addresses: xyang44@jhu.edu (Xiang Yang), mittal@jhu.edu (Rajat Mittal)

convergence acceleration through successive relaxation. Richardson further noted that ω could be chosen to successively eliminate individual components of the residual. However, this required knowledge of the full eigenvalue spectrum of A, which was impractical. Given this, Richardson's recipe for choosing ω was to distribute the "nodes"(or zeros) of the amplification factor evenly within the range of eigenvalues of A. This was expected to drive down the overall amplification factor for the iterative scheme, and the advantage of this approach was that it required knowledge of only the smallest and largest eigenvalue of A.

Richardson's method subsequently appeared in the seminal doctoral dissertation of Young [6]. Noting, however, that "*... it appears doubtful that a gain of a factor of greater than 5 in the rate of convergence can in general be realized unless one is extremely fortunate in the choice of the values of ω*", Young discarded this method in favour of successive over-relaxation of the so-called Liebmann Method [7], which was essentially the same as the Gauss-Seidel method. This seems to signal the end of any attempts to accelerate Jacobi or Jacobi-like methods for matrix equations resulting from discrete approximations of elliptic equations.

In this article we describe a new approach for accelerating the convergence of the Jacobi iterative method as applied to the finite-difference approximation of elliptic equations. Using this approach, gains in convergence-rate well in excess of a factor of 100 are demonstrated for problems sizes of practical relevance. The increase of processor count in parallel computers into the tens of thousands that is becoming possible with multi-core and GPU architectures [8], is leading to an ever-increasing premium on parallelizability and scalability of numerical algorithms. While sophisticated iterative methods such as multigrid (MG) are highly efficient on a single processor [9], it is extremely difficult to maintain the convergence properties of these methods in large-scale parallel implementations. The domain decomposition approaches associated with parallel implementations negatively impact the smoothing properties of the iterative solvers used in MG, and also limit the depth of coarsening in such methods; both of these can significantly deteriorate the convergence properties of MG methods. In addition to this, the ratio of computation to communication also decreases for the coarse grid corrections, and this further limits the scalability of these methods. Within this context, the iterative method described here, preserves the insensitivity of the Jacobi method to domain decomposition, while providing significant convergence acceleration.

Another class of methods that is extensively used for solving elliptic equations is conjugate gradient (CG) [10]. CG methods, however, require effective preconditioners in order to produce high convergence rates; in this context, the method proposed here could eventually be adapted as a preconditioner for CG methods. Thus, the method described here could be used as an alternate to or in conjunction with these methods, and as such, could a have a significant impact in computational mechanics as well as other fields such as weather and climate modelling, astrophysics and electrostatics, where elliptic equations are prominent.

Finally, the slow convergence rate of the Jacobi iterative method and the inability to accelerate this method using relaxation techniques is, at this point, considered textbook material [10, 11, 12]. In most texts, a discussion of the Jacobi method and its slow convergence is followed immediately by a discussion of the Gauss-Seidel method as a faster and more practical method. In this context, the method described here demonstrates that it is in-fact, relatively easy to increase the convergence rate of the Jacobi method by factors exceeding those of the classical Gauss-Seidel method. It is therefore expected that the method presented here will have a fundamental impact on our view of these methods, and spur further analysis of the acceleration of these basic methods.

2. Jacobi with successive over-relaxation (SOR)

We employ a 2D Laplace equation in a rectangular domain of unit size as our model problem: $\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2 = 0$. A 2$^{\text{nd}}$-order central-difference discretization on a uniform grid followed by the application of the Jacobi iterative method with a relaxation parameter ω, leads to the following iterative scheme:

$$u_{i,j}^{n+1} = (1-\omega)\, u_{i,j}^{n} + \frac{\omega}{4}\left(u_{i,j-1}^{n} + u_{i,j+1}^{n} + u_{i-1,j}^{n} + u_{i+1,j}^{n}\right) \qquad (1)$$

where n is the index of iteration. von Neumann analysis [11] of the above scheme results in the following amplification factor:

$$G_{\omega}(\kappa) = (1-\omega\kappa) \text{ with } \kappa(k_x, k_y) = \sin^2(k_x \Delta x/2) + \sin^2(k_y \Delta y/2) \qquad (2)$$

where Δx and Δy are the grid spacings and k_x and k_y the wave-numbers in the corresponding directions. The largest value of κ is given by $\kappa_{\max} = 2$. The smallest non-zero value of κ depends on the largest independent sinusoidal wave that the system can admit. A Neumann(N) problem allows waves to be purely one-dimensional, i.e. elementary waves can have $k_x = 0$ or $k_y = 0$, whereas for Dirichlet(D) problems, one-dimensional waves are, in general, not admissible, and k_x, k_y must all be non-zero. Therefore the corresponding $\kappa_{\min}$ are given by:

$$\kappa_{\min}^{\text{N}} = \sin^2\left(\frac{\pi/2}{\max(N_x, N_y)}\right);\ \kappa_{\min}^{\text{D}} = \sin^2\left(\frac{\pi/2}{N_x}\right) + \sin^2\left(\frac{\pi/2}{N_y}\right) \qquad (3)$$

The above expressions are true for a uniform mesh and section 8 describes the extension of this approach to non-uniform meshes.

The convergence of the iterative scheme requires $|G| < 1$ for all wave numbers, and it is easy to see from Eqn.(2) that over-relaxation of the Jacobi method violates this requirement. Furthermore, for a given grid, $\kappa_{\min}^{\text{N}} < \kappa_{\min}^{\text{D}}$; thus Neumann problems have a wider spectrum and are therefore more challenging than the corresponding Dirichlet problem. We therefore focus most of our analysis on the Neumann problem. We also note that while the above analysis is for 2D problems, corresponding 1D and 3D problems lead to exactly the same expressions for the amplification factors, and similar expressions for $\kappa_{\min}$, with a pre-factor different from unity.

3. Jacobi with over-relaxation: an example

A simple example is presented here in order to motivate the current method. We start by noting that for relaxation factors ≥ 0.5, there is one wave number with an amplification factor equal to zero, and this implies that for a discretized system with N unknowns, it should be possible to obtain the solution in N iterations by employing a successively relaxed iteration with relaxation factors at each iteration chosen to successively eliminate each wave-number component of the error. We explore the above idea further via the following simple one-dimensional example:

$$\frac{\partial^2 u}{\partial x^2} = 0, x \in [0,1], u(0) = 0, u(1) = 0$$

The above equation is discretized on a five-point equispaced grid. Since the boundary values (u_1 and u_5) are given, the resulting 3×3 system that is to be solved is as follows:

$$\begin{bmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Application of the Jacobi method to this system gives:

$$\begin{bmatrix} u_2^{n+1} \\ u_3^{n+1} \\ u_4^{n+1} \end{bmatrix} = \begin{bmatrix} 0 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \end{bmatrix} \begin{bmatrix} u_2^n \\ u_3^n \\ u_4^n \end{bmatrix}$$

Incorporating successive over-relaxation in to this iteration gives:

$$\begin{bmatrix} u_2^{n+1} \\ u_3^{n+1} \\ u_4^{n+1} \end{bmatrix} = \left((1-\omega)I + \omega \begin{bmatrix} 0 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \end{bmatrix} \right) \begin{bmatrix} u_2^n \\ u_3^n \\ u_4^n \end{bmatrix}$$

We now need to choose a relaxation factor for each iteration such that we can reach the exact solution in three iterations. This condition leads to a cubic equation for ω_k

$$\prod_{k=1}^{3} \left((1-\omega_k)I + \omega_k \begin{bmatrix} 0 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \end{bmatrix} \right) = 0$$

The roots of this equation can be uniquely determined and they are as follows (in descending order of magnitude):

$$\omega_1 = 2 + \sqrt{2}, \; \omega_2 = 1 \text{ and } \omega_3 = 2 - \sqrt{2}$$

Thus, with the above unique choice of ωs, the exact solution (barring round-off error) can be obtained in 3 iterations starting from an arbitrary initial guess. This idea of eliminating one particular wave-number component of the error in each iteration is however not practical, since it is difficult to know *a-priori*, the correct relaxation factor for each iteration. Besides, even if appropriate ω_k could be determined, the error-component for any wave number that is eliminated in a given iteration could reappear due to roundoff error and possibly be amplified in subsequent iterations since over-relaxation does generate amplification factors that exceed unity for some wave numbers. The above analysis is nevertheless insightful: considering the fact that the ω_k above are uniquely determined, it necessarily implies that constrained to under-relaxation (note that $\omega_1 > 1$), it is not possible to obtain the solution in $\leq N$ steps. In fact, it is interesting to note that without any over- or under-relaxation, the error would be reduced by a factor of only $(1/2)^3 = 1/8$ (approximately one order-of-magnitude) in three iterations. Thus, over-relaxation does not necessarily produce divergence in the iteration; on the contrary, when combined with under-relaxation, it can help accelerate the convergence of JM.

4. Scheduled Relaxation Jacobi (SRJ) Schemes

The method described here (termed "SRJ" for Scheduled Relaxation Jacobi) consists of an iteration cycle that further consists of a fixed number (denoted by M) of SOR Jacobi iterations with a prescribed relaxation factor scheduled for each iteration in the cycle. The M-iteration cycle is then repeated until convergence. This approach is inspired by the observation that over relaxation of Jacobi damps the low wavenumber residual more effectively, but amplifies high wavenumber error. Conversely, under-relaxation with the Jacobi method damps the high wave number error efficiently, but is quite ineffective for reducing the low wavenumber error. The method we present here, attempts to combine under- and over-relaxations to achieve better overall convergence. Our goal is to provide a set of schemes that practitioners can simply select from and use, without going through any of the analysis performed in this paper.

4.1. Nomenclature

Each SRJ scheme is characterized by the number of 'levels', which corresponds to the number of distinct values of ω used in the iteration cycle. We denote this parameter by P and the P distinct values of ω used in the cycle are represented by the vector $\vec{\Omega} = \{\omega_1, \omega_2, \ldots, \omega_P\}$. For the uniqueness of $\vec{\Omega}$, we sequence those P distinct values of relaxation factors in descending order, i.e. $\omega_1 > \omega_2 > \ldots > \omega_P$. Second, we define $\vec{Q} = \{q_1, q_2, \ldots, q_P\}$, where q_i is the number of times the value ω_i is repeated in an SRJ cycle. We also define $\vec{\beta} = \{\beta_1, \beta_2, ..., \beta_P\}$ where $\beta_i = q_i/M$, is the fraction of the iterations counts of ω_i in one SRJ iteration cycle. Due to the linearity of Eq.(1), the sequencing of the ωs within a cycle does not affect the analysis of convergence. In practice, however, due to the roundoff and arithmetic overflow associated with digital computers, appropriate sequencing of the ω 's might be require; this issue that will be discussed later in the paper. P, $\vec{\Omega}$ and $\vec{Q}$ (or $\vec{\beta}$) therefore uniquely define an SRJ scheme. It is noted that Richardson proposed to use a different relaxation factor for each iteration in a cycle; that idea may be viewed as a special case of the SRJ scheme with $P = M$ and $q_i \equiv 1$.

4.2. Amplification Factor

If one iteration of SOR-Jacobi attenuates the wave with a vector wavenumber (k_x, k_y) by the factor of $G(\kappa(k_x, k_y))$, then the amplification factor for the one full SRJ cycle (defined as $G(\kappa; \vec{\Omega}, \vec{Q})$) consisting of M iterations is given by

$$G(\kappa; \vec{\Omega}, \vec{Q}) = \prod_{i=1}^{P} \left(G_{\omega_i}\left(\kappa\right)\right)^{q_i} \tag{4}$$

where G_{ω_i} raised to the power of q_i is simply because the relaxation factor ω_i is repeated q_i times in one cycle. Following this, we define the per-iteration amplification factor for this cycle as the geometric mean of the modulus of the cycle amplification factor $G(\kappa; \vec{\Omega}, \vec{Q})$, i.e.

$$\Gamma\left(\kappa\left(k_x.k_y\right)\right) = \left|G(\kappa; \vec{\Omega}, \vec{Q})\right|^{\frac{1}{M}} = \prod_{i=1}^{P} \left|1 - \omega_i \kappa\right|^{\beta_i} \tag{5}$$

5. Two-Level (P=2) Schemes

We begin our analysis with the two-level ($P = 2$) SRJ scheme, for which $\vec{\Omega} = \{\omega_1, \omega_2\}$ and $\vec{\beta} = \{\beta_1, \beta_2\}$. The per-iteration amplification factor for this 2-level scheme can be expressed as

$$\Gamma\left(\kappa\right) = \left[\left|1 - \omega_1 \kappa\right|^{q_1} \quad \left|1 - \omega_2 \kappa\right|^{q_2}\right]^{\frac{1}{M}} = \left|1 - \omega_2 \kappa/\alpha\right|^{\beta} \left|1 - \omega_2 \kappa\right|^{1-\beta} \tag{6}$$

where $\alpha = \omega_2/\omega_1$, and β_1, which is equal to $(1 - \beta_2)$, is replaced by β for notational simplicity. The three variables α, ω_2, β control the convergence of any $P = 2$ SRJ. Since $\Gamma(\kappa) < 1$ must be satisfied, we cannot have both ω_1, ω_2 larger than 1, and because we order relaxation factors such that $\omega_1 > \omega_2$, ω_2 is necessarily ≤ 1. Moreover, by definition, α and β also lie within $[0, 1]$.

The maximum magnitude of Γ in the closed interval $[\kappa_{\min}, \kappa_{\max}]$, denoted as $\Gamma_{\max}$, determines the asymptotic convergence of an iterative scheme[11]. Following convention, the convergence rate of a given iterative scheme can be measured by the number of iterations required to reduce the residual by an order of magnitude. This number, denoted here by $\mathcal{N}_{0.1}$, is given by the following formula: $\mathcal{N}_{0.1} = \ln(0.1)/\ln\left(\Gamma_{\max}\right)$. Finally, for each iterative method, we also define a convergence performance index ρ as the ratio of $\mathcal{N}_{0.1}$ for the classic Jacobi method to $\mathcal{N}_{0.1}$ for the iterative method under consideration.

5.1. *Optimal P=2 schemes*

The asymptotic convergence of the iterative scheme can be maximized by minimizing the maximum value of $\Gamma_{\max}$ for any given grid size N. A simple analysis of Eq.(6) shows that $\Gamma_{\max}$ for $P = 2$ schemes could be located at one of the following locations:

$$\kappa_1 = \kappa_{\min}; \;\; \kappa_2 = \left(\alpha + \beta - \alpha\beta\right)/\omega_2 \text{ and } \kappa_3 = \kappa_{\max} \equiv 2 \tag{7}$$

where κ_2 is the coordinate of the internal extremum for (6). This is determined by first noting that the local extremum of $\log\Gamma\left(\kappa\right)$ coincides with that of $\Gamma\left(\kappa\right)$, and then solving the equation $\frac{d\log\Gamma(\kappa)}{d\kappa} = 0$ for the form of Γ given in Eqn.(6)

Among $\Gamma\left(\kappa_1\right)$, $\Gamma\left(\kappa_2\right)$, $\Gamma\left(\kappa_3\right)$, the value of $\Gamma(\kappa_1)$ is strongly constrained by the condition $\Gamma(0) \equiv 1$ and therefore, the minimization of $\Gamma(\kappa_1)$ has to be the driving factor in the convergence maximization. Thus, $\Gamma(\kappa_1)$ should be the *global* maximum and we therefore require:

$$\Gamma(\kappa_2) \leq \Gamma(\kappa_{\min}),\ \Gamma(\kappa_3) \leq \Gamma(\kappa_{\min}) \tag{8}$$

However, Γ, is a function of three independent variables: ω_2, α and β, and minimization of $\Gamma(\kappa_{\min})$ under the constraints of (8) is not straightforward. The mathematical problem can, however, be simplified by first we noting that $\Gamma(\kappa_{\min})$, is a monotonically decreasing function of $(\omega_2\kappa_{\min})$ near $\omega_2\kappa_{\min} = 0$. Since we have

$$\frac{d\log\left(\Gamma\right)}{d(\omega_2\kappa)} = \frac{1}{\Gamma}\frac{d\Gamma}{d\omega_2\kappa} \tag{9}$$

the sign of $d\log\left(\Gamma\right)/d(\omega_2\kappa)$ is the same as that of $d\Gamma/d\omega_2\kappa$. Hence to check the monotonicity near 0, we only need to check the sign of $d\log\left(\Gamma\right)/d(\omega_2\kappa)$ at $\omega_2\kappa_{\min} = 0$. This is given by

$$\left.\frac{d\log\left(\Gamma\right)}{d(\omega_2\kappa)}\right|_{\omega_2\kappa_{\min}=0} = -1 + \beta - \frac{\beta}{\alpha} < 0 \tag{10}$$

Since $\kappa_{\min}\omega_2$ is very close to 0, it is safe to argue that with other variable unchanged, a smaller $\omega_2\kappa_{\min}$ could result in a smaller $\Gamma\left(\kappa_{\min}\right)$. In addition, we also have

$$\Gamma(\kappa_2) = \left|\frac{\beta}{\alpha}(1-\alpha)\right|^{\beta}\left|(1-\alpha)(1-\beta)\right|^{1-\beta} \tag{11}$$

which indicates that $\Gamma(\kappa_2)$ is independent of ω_2. Therefore, if the first condition in (8) is satisfied strictly with an inequality, it is possible to increase ω_2 to reduce $\Gamma(\kappa_{\min})$, the maximum of Γ, without violating the first condition in (8). Hence to minimize $\Gamma(\kappa_{\min})$, the first condition in (8) should be satisfied with an equality. Employing the same argument, it is easy to see that the second condition in (8) should also be satisfied with an equality. Condition (8) is therefore reduced to:

$$\Gamma(\kappa_2) = \Gamma(\kappa_{\max}) = \Gamma(\kappa_{\min}) \tag{12}$$

Maximization of convergence rate of the P=2 SRJ scheme is therefore equivalent to minimizing $\Gamma_{\max}$ under the two conditions in (12). Given that $\kappa_{\min}$ is essentially a surrogate for the grid size (see Eq.(3)) the above provides sufficient conditions to uniquely determine a set of α, β, ω_2 values that maximize the asymptotic convergence rate for a given grid.

The equations that result from the above procedure are shown in Appendix A where the three primary unknowns are augmented by two ancillary variables $\partial\alpha/\partial\beta$ and $\partial\omega_2/\partial\beta$. The addition of these two variables facilitates the implementation of the above procedure. We note that these equations are highly non-linear and coupled in a complex way.

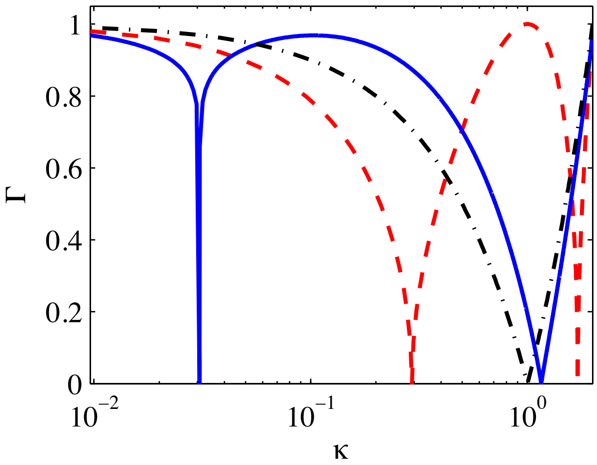

Figure 1: Amplification factor curves for selected optimal P=2 SRJ schemes. Comparison between Jacobi method ($\cdot - \cdot$), optimal $M = 2$ ($-$) and optimal $P = 2$ ($M > 2$) ($--$) scheme for N=16.

5.2. M=2 schemes

A closed-form solution of the above equations is obtained only for schemes with $M = 2$ ($\beta = 0.5$) and is as follows:

$$\alpha = \left(-b - \sqrt{b^2 - 4}\right)/2 \text{ and } \omega_2 = (\alpha + 1) / (2 + \kappa_{\min}) \\ \text{where } b = \frac{2\kappa_{\min} - 6\left(2 + \kappa_{\min}\right)^2}{\kappa_{\min} + \left(2 + \kappa_{\min}\right)^2} \tag{13}$$

The above expressions asymptote for $\kappa_{\min} \ll 2$ which, based on (3), occurs at fairly small values of N. The asymptotic values of ω_1 and ω_2 are 3.414213 and 0.585786 respectively, and the asymptotic value of ρ for this scheme is precisely equal to 2.0. Numerical tests of this for a 2D Neumann problem and a random initial guess indicate a ρ value of 1.99 which confirms the predicted convergence. Thus, even the two-step SRJ scheme, which is the simplest possible extension of the classic Jacobi method, doubles the rate of convergence, a rate that matches that of the classical point-iterative Gauss-Seidel method.

5.3. $M > 2$ schemes

For $P = 2; M > 2$ schemes, we have employed a MATLAB based numerical procedure to solve the equation-set shown in Appendix A, and Table 1 shows the optimal parameters obtained from this process for a 2D Neumann problem on a uniform $N \times N$ grid. Fig.1 shows plots of the amplification factor for optimal $M = 2$ and $M > 2$ scheme for $N = 16$ along with the amplification factor for the classic Jacobi method. It is noted that while the Jacobi method has one node, the $P = 2$ SRJ schemes have two nodes. Furthermore, the plots clearly show the effect of relaxing the $M = 2$ constraint for two-level schemes: the first node in Γ is pushed closer to $\kappa = 0$ thereby reducing $\Gamma(\kappa_{\min})$.

Table 1: Parameters and performance of optimized two-level (P=2) SRJ schemes for a range of grid sizes N. ρ_{test} is the value obtained from numerical tests for the Neumann problem and N excludes the boundary points.

N	SRJ Scheme	$\mathcal{N}_{0.1}$	ρ	(q_1, q_2)	ρ_{test}
16	$\vec{\Omega} = \{32.60, 0.8630\}$ $\vec{\beta} = \{0.064291, 0.93570\}$	72	3.31	(1,15)	3.41
32	$\vec{\Omega} = \{81.22, 0.9178\}$ $\vec{\beta} = \{0.032335, 0.96766\}$	251	3.81	(1,30)	4.00
64	$\vec{\Omega} = \{190.2, 0.9532\}$ $\vec{\beta} = \{0.015846, 0.98415\}$	923	4.14	(1,63)	4.51
128	$\vec{\Omega} = \{425.8, 0.9742\}$ $\vec{\beta} = \{0.0076647, 0.99233\}$	3521	4.34	(1,130)	4.34
256	$\vec{\Omega} = \{877.8, 0.98555\}$ $\vec{\beta} = \{0.0036873, 0.996312\}$	13743	4.45	(1,257)	4.50
512	$\vec{\Omega} = \{1972, 0.99267\}$ $\vec{\beta} = \{0.0017730, 0.998227\}$	54119	4.52	(1,564)	4.69
1024	$\vec{\Omega} = \{4153, 0.99615\}$ $\vec{\beta} = \{0.00085251, 0.9991474\}$	214873	4.55	(1,1172)	5.35

Table 1 shows that ω_1 for these optimal schemes is significantly larger than unity for all N, and it increases rapidly with this parameter. This is expected since $\kappa_{\min}$ decreases with increasing N and the scheme attempts to move the first node closer to $\kappa = 0$ in order to reduce the amplification factor at $\kappa_{\min}$. On the other hand, ω_2 is very close to unity and becomes increasingly so with N. It is also noted that β_1 is significantly smaller than β_2 (which, is equal to $(1 - \beta_1)$) for all cases and becomes smaller with increasing N. Thus, the scheme compensates for an increasing over-relaxation parameter ω_1 primarily by increasing β_2 relative to β_1. Furthermore, $\mathcal{N}_{0.1}$ is found to increase with N and this rate of increase with N will be addressed later in the paper.

The ρ for the optimal $P = 2$ schemes is found to range from 3.31 to 5.35 for the values of N studied here. Thus, even within the constraint of two-levels, loosening of the $M = 2$ constraint results in a significant additional increase in the convergence rate. Note also that these optimal two-level SRJ schemes are now roughly twice faster than GSM. Furthermore, this speed-up is similar to the maximum gain possible with the method of Richardson [6].

For numerical validation of the above schemes, we choose $q_1 \equiv 1$ and $q_2 = \lceil \frac{\beta_2}{\beta_1} \rceil$. The use of the ceiling function increases the under-relaxation very slightly over the optimal value and should enhance numerical stability. This value of $\vec{Q}$ is also shown in Table 1 and we note that q_2 increases from 15 for $N = 16$ to 1172 for $N = 1024$. Thus, in all of these schemes, one over-relaxation step is followed by a fairly large number of under-relaxations. We have implemented the schemes with (q_1, q_2) given in the table for a 2D Laplace equation on a uniform, isotropic grid with a cell-centered arrangement. Results for the Neumann problem only are reported here since the Dirichlet problems exhibit similar trends.

The initial guess in these tests corresponds to a random value in the range $[0, 1]$ at each grid point and Table 1 shows that the numerical tests exhibit a convergence performance index (ρ_{test}) that is equal to or slightly higher than the predicted value. It is noted that the condition of optimality equalizes the multiple extrema in Γ. However, due to the finite-precision mathematics employed in the optimization process as well as the round-off errors inherent in the numerical validation, there is a small but finite imbalance between the various values of $\Gamma_{\max}$ and this can make the convergence behaviour somewhat sensitive to the initial guess. Numerical tests with other initial guesses such as a Dirac δ-function confirm this sensitivity. However, for every initial guess that has been tried, the asymptotic convergence rates for the optimal SRJ schemes either equal or slightly exceed the corresponding prediction from the mathematical analysis have been observed.

5.4. Sensitivity of optimal SRJ schemes

For practical implementation of the SRJ schemes, it is desired that $\mathcal{N}_{0.1}$ not increase significantly if parameters that are close, but not exactly equal to the optimal set, are employed. This issues is addressed in the current section.

We first prove that if a set of parameters that is optimized for a given system size (i.e. given N) is employed to solve a larger system, the acceleration index ρ will *not* decrease below that of the smaller system. For this, we note that for sufficiently large N, $\Gamma(\kappa_{\min})$ can be approximated via a Taylor series as $\Gamma(\kappa_{\min}) \approx \Gamma(0) + \kappa_{\min}\partial\Gamma/\partial\kappa|_{\kappa=0} = 1 + \kappa_{\min}\partial\Gamma/\partial\kappa|_{\kappa=0}$. Assuming further that the scheme under consideration is optimized using the process described in the previous sections, $\Gamma_{\max} = \Gamma\left(\kappa_{\min}\right)$, and the convergence performance index can then be estimated as:

$$\rho \approx \frac{\ln\left(1+\kappa_{\min}\frac{\partial\Gamma_{\text{SRJ}}}{\partial\kappa}\Big|_{\kappa=0}\right)}{\ln\left(1+\kappa_{\min}\frac{\partial\Gamma_{\text{JM}}}{\partial\kappa}\Big|_{\kappa=0}\right)} \approx \frac{\frac{\partial\Gamma_{\text{SRJ}}}{\partial\kappa}\Big|_{\kappa=0}}{\frac{\partial\Gamma_{\text{JM}}}{\partial\kappa}\Big|_{\kappa=0}} = \sum_{i=1}^{P}\omega_i\beta_i \qquad (14)$$

It is noted that the above expression is independent of $\kappa_{\min}$ and therefore, for a fixed set of parameters (i.e. $\vec{\Omega}$ and $\vec{\beta}$), ρ is independent of system size N, as long as $\Gamma_{\max} = \Gamma(\kappa_{\min})$. Since $\Gamma\left(\kappa_{\min}\right)$ has a negative slope at $\kappa = 0$, and $\kappa_{\min}$ is smaller for a larger N, $\Gamma_{\max} = \Gamma(\kappa_{\min})$ will be true if a set of parameters that is optimized for a smaller system is applied to a larger system. The converse, i.e. using a set of parameters that is optimized for a large system to a smaller system, is however not recommended; in that case $\Gamma(\kappa_{\min})$ decreases and because all the interior extrema have to be balanced with $\Gamma\left(\kappa_{\min}\right)$, $\Gamma(\kappa_{\min})$ is not guaranteed to be $\Gamma_{\max}$. We also note that in this analysis, the dimension of the problem does not enter in to consideration and therefore, parameters optimized for a 2D problem can be safely used for corresponding 3D problem. This is demonstrated later in the paper.

We further explore the sensitivity of the convergence rate to scheme parameters via Fig. 2, which shows the effect of cycle size M on ρ. Each point on the curve corresponds to a scheme optimized for a given M for the 2D Neumann problem. The maximum in ρ observed in the various curves (and marked by circles) correspond to the global optima that are identified in Table 1. The plot indicates that for small values of N, the maximum in ρ is sharp, indicating a relatively high sensitivity to the scheme parameters. However, for larger values of N, while the performance deteriorates rapidly as M is reduced from the optimal

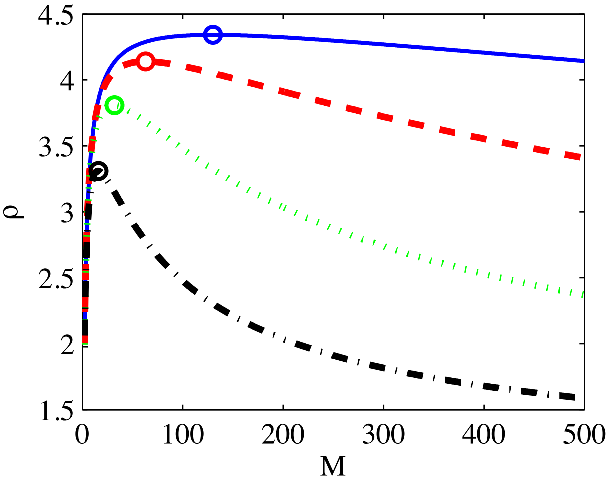

Figure 2: Convergence performance index ρ versus M for $P = 2$ schemes that are optimized for a given value of M. $N = 16$ $(\cdot - \cdot)$; $N = 32(\cdots)$; $N = 64(--)$; and $N = 128(\ -)$. Circles identify the globally optimal scheme for each N.

value, the effect on performance is relatively weak as M is increased beyond the optimal value. This insensitivity is particularly pronounced for the highest values of N in the plot, which exhibits a very broad maxima. This is particularly useful since problems of practical relevance in computational mechanics are expected to have large grid sizes.

While small changes that are introduced to enhance the stability of the given scheme are not expected to deteriorate the convergence significantly, caution should be exercised in making *ad-hoc* modifications in some parameters. For instance, based on Table 1, there might be a temptation to simply put ω_2 equal to 1 for large N. However, this would eliminate the under-relaxation that is necessary to balance out the over-relaxation, and would lead to divergence. Similarly, starting from an optimal set of parameters, the over-relaxation factor ω_1 should only be decreased since increasing it would lead to instability. Thus, it is advisable to round *down* the value of the over-relaxation parameters determined from the optimization process.

6. Multilevel ($P > 2$) Schemes

In this section, we examine the convergence gains possible with multilevel ($P > 2$) schemes. The general procedure for optimizing multilevel schemes is summarised in Appendix B and the ideas behind the equations are briefly described here. The constraint that all local extrema are equalized with $\Gamma_{\max}$ is still enforced. The parameters $\beta_i, i = 1, 2, ..., P-1$ are considered to be truly free parameters, with ω_i depending on them. The optimization is then with respect to $\beta_i, i = 1, 2, ..., P-1$, and this procedure results in intermediate unknowns like $\partial\omega_i/\partial\beta_j$. We note that β_P is not a free parameter because of the constraint $\sum_{i=1}^{P} \beta_i = 1$. The size of this complex non-linear system grows proportional to P^2 and this system of equations can therefore only be solved numerically. Furthermore, large values of

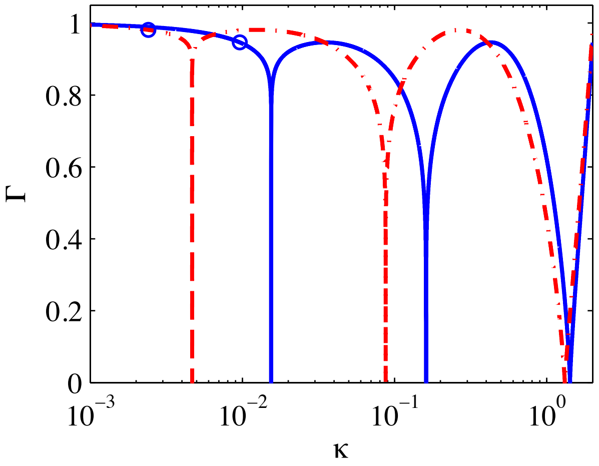

Figure 3: Comparison between optimal $P = 3$ scheme for $N = 16$ (· − ·)and $N = 32$ (−). $\Gamma(\kappa_{\min})$ for each case is denoted by ∘. For the curve of $N = 16$, $\kappa_{\min} = 0.0096$, $\kappa_1 = 0.0351$, $\kappa_2 = 0.4282$ and $\Gamma_{\max} = 0.946$, while for the $N = 32$ curve, $\kappa_{\min} = 0.0024$, $\kappa_1 = 0.0125$, $\kappa_2 = 0.2689$ and $\Gamma_{\max} = 0.981$.

N (i.e. small values of $\kappa_{\min}$) combine with large values of P to increase the stiffness of the system of equations. Thus, optimal schemes presented in the paper are currently limited to $P = 5$.

6.1. Optimal multilevel schemes

Table 2, 3 and 4 shows the optimal values that have been obtained from the numerical optimization process for $P = 3, 4$ and 5 SRJ schemes, respectively. The trends in the table are similar to those observed for the $P = 2$ schemes. First, except for one stage of under-relaxation, all other stages in the sequence are over-relaxations, and the over-relaxation factors increase monotonically with N. Fig.3 shows plots of the amplification factor for optimal $P = 3$ schemes for $N = 16$ and 32. The plots clearly show that the optimization process moves the first node in Γ closer to the origin as N increases.

For one case ($N = 64$) in each table, we have also included results for a three-dimensional case. As predicted from the analysis in Sec. 5.4, the increase in the dimensionality of the problem from one to three does not affect the performance of the scheme. Thus, schemes identified as optimal from the 1D analysis can be used safely in the corresponding 2D and 3D problems.

The tables therefore clearly demonstrate significant increases in convergence acceleration with increasing P. For instance, comparing the case of $N = 512$ for which we have derived optimized $P = 2$, 3, 4 and 5 schemes, we note that ρ increases from about 5 for $P = 2$ to about 68 for $P = 5$. Higher gains in convergence will likely be obtained for larger P although this remains to be formally demonstrated. Nevertheless, it is clear that multilevel SRJ schemes offer significantly higher convergence rates.

Table 2: Parameters for optimized $P = 3$ SRJ schemes for various values of N. For $N = 64$ the ρ_{test} for the corresponding 3D problem is also included.

N	Optimal Scheme Parameters	ρ	ρ_{test}
16	$\vec{\Omega} = \{64.66, 6.215, 0.7042\}$ $\vec{\beta} = \{0.039715, 0.18358, 0.77669\}$ M=27; $\vec{Q} = \{1, 5, 21\}$	5.71	6.5
32	$\vec{\Omega} = \{213.8, 11.45, 0.7616\}$ $\vec{\beta} = \{0.019004, 0.13416, 0.84683\}$ M=53; $\vec{Q} = \{1, 7, 45\}$	7.90	8.1
64	$\vec{\Omega} = \{684.3, 20.73, 0.8149\}$ $\vec{\beta} = \{0.0085938, 0.093707, 0.89769\}$ M=118; $\vec{Q} = \{1, 11, 106\}$	10.2	12.9 (3D:10.1)
128	$\vec{\Omega} = \{2114, 36.78, 0.8611\}$ $\vec{\beta} = \{0.0037113, 0.063178, 0.93311\}$ M=270; $\vec{Q} = \{1, 17, 252\}$	12.5	13.4
256	$\vec{\Omega} = \{6319, 63.99, 0.8989\}$ $\vec{\beta} = \{0.0015454, 0.041468, 0.95698\}$ M=653; $\vec{Q} = \{1, 27, 625\}$	14.6	15.6
512	$\vec{\Omega} = \{18278, 109.2, 0.9282\}$ $\vec{\beta} = \{0.000626, 0.0266, 0.972\}$ M=1615; $\vec{Q} = \{1, 68, 3955\}$	16.4	17.7
1024	$\vec{\Omega} = \{51769.1, 184.31, 0.95025\}$ $\vec{\beta} = \{0.00024857, 0.016941, 0.98281\}$ M=1615; $\vec{Q} = \{1, 43, 1571\}$	17.8	17.9

Table 3: Parameters for optimized $P = 4$ SRJ schemes for various values of N. For $N = 64$ the ρ_{test} for the corresponding 3D problem is also included.

N	Optimal Scheme Parameters	ρ	ρ_{test}
16	$\vec{\Omega} = \{80.154, 17.217, 2.6201, 0.62230\}$ $\vec{\beta} = \{0.031495, 0.082068, 0.25554, 0.63089\}$ M=31; $\vec{Q} = \{1, 2, 8, 20\}$	7.40	7.7
32	$\vec{\Omega} = \{289.46, 40.791, 4.0877, 0.66277\}$ $\vec{\beta} = \{0.015521, 0.053883, 0.22041, 0.71018\}$ M=64; $\vec{Q} = \{1, 3, 14, 46\}$	11.3	11.8
64	$\vec{\Omega} = \{1029.4, 95.007, 6.3913, 0.70513\}$ $\vec{\beta} = \{0.0072290, 0.033832, 0.18222, 0.77671\}$ M=146; $\vec{Q} = \{1, 5, 26, 114\}$	16.6	15.2 (3D:16.3)
128	$\vec{\Omega} = \{3596.4, 217.80, 9.9666, 0.74755\}$ $\vec{\beta} = \{0.0032024, 0.020392, 0.145608, 0.83079\}$ M=343; $\vec{Q} = \{1, 7, 50, 285\}$	23.1	24.7
256	$\vec{\Omega} = \{12329, 492.05, 15.444, 0.78831\}$ $\vec{\beta} = \{0.0013564, 0.011845, 0.11316, 0.87362\}$ M=760; $\vec{Q} = \{1, 9, 86, 664\}$	30.8	34.2
512	$\vec{\Omega} = \{41459, 1096.3, 23.730, 0.82597\}$ $\vec{\beta} = \{0.00055213, 0.0066578, 0.085990, 0.90680\}$ M=1818; $\vec{Q} = \{1, 12, 155, 1650\}$	39.0	44.2

Table 4: Parameters for optimized $P = 5$ SRJ schemes for various values of N. For $N = 64$ the $\rho_{\rm test}$ for the corresponding 3D problem is also included.

N	Optimal Scheme Parameters	ρ	$\rho_{\rm test}$
16	$\vec{\Omega} = \{88.190, 30.122, 6.8843, 1.6008, 0.58003\}$ $\vec{\beta} = \{0.026563, 0.050779, 0.12002, 0.28137, 0.52126\}$ M=43; $\vec{Q} = \{1, 2, 5, 12, 23\}$	8.5	8.8
32	$\vec{\Omega} = \{330.57, 82.172, 13.441, 2.2402, 0.60810\}$ $\vec{\beta} = \{0.013467, 0.031695, 0.092173, 0.26580, 0.59686\}$ M=76; $\vec{Q} = \{1, 2, 7, 20, 46\}$	14.0	13.2
64	$\vec{\Omega} = \{1228.8, 220.14, 26.168, 3.1668, 0.63890\}$ $\vec{\beta} = \{0.0064862, 0.019035, 0.068043, 0.24139, 0.66504\}$ M=158; $\vec{Q} = \{1, 3, 10, 38, 106\}$	22.0	20.4 (3D:20.2)
128	$\vec{\Omega} = \{4522.0, 580.86, 50.729, 4.5018, 0.67161\}$ $\vec{\beta} = \{0.0029825, 0.011020, 0.048530, 0.21238, 0.72508\}$ M=343; $\vec{Q} = \{1, 3, 16, 73, 250\}$	33.2	31.0
256	$\vec{\Omega} = \{16459, 1513.4, 97.832, 6.4111, 0.70531\}$ $\vec{\beta} = \{0.0013142, 0.0061593, 0.033568, 0.18206, 0.77689\}$ M=778; $\vec{Q} = \{1, 4, 26, 142, 605\}$	48.3	43.1
512	$\vec{\Omega} = \{59226, 3900.56, 187.53, 9.1194, 0.73905\}$ $\vec{\beta} = \{0.00055665, 0.0033286, 0.022588, 0.15273, 0.82079\}$ M=1824; $\vec{Q} = \{1, 6, 40, 277, 1500\}$	67.7	59.9

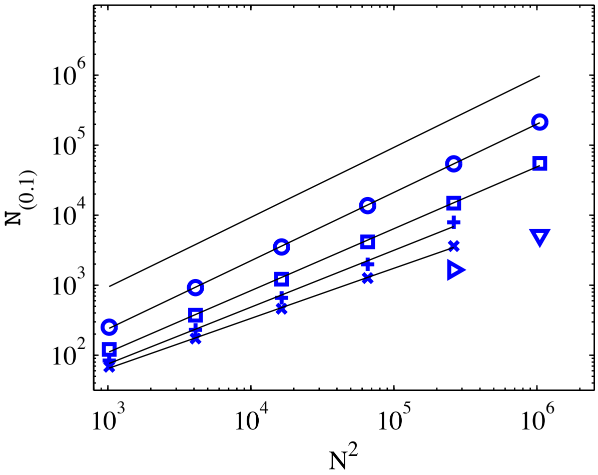

Figure 4: Effect of grid size N on asymptotic convergence rate of various optimized SRJ schemes ($P = 2 : (\circ)$; $P = 3 : (\square)$, $P = 4 : (+)$ and $P = 5 : (\times)$). The slopes for the best-fit lines for P =2, 3 and 4 schemes are 0.97, 0.88, 0.81 and 0.71 respectively. The topmost line corresponds to the classical Jacobi method. The isolated symbols in the plot identify the SRJ schemes described in Table 5 (a: $(\triangleright)$; d:$(\triangledown)$).

6.2. Scaling of convergence acceleration with grid size

Interesting trends emerge with respect to the effect of N on convergence acceleration. In Fig. 4 we compare $\mathcal{N}_{0.1}$ for various grids for the entire set of optimal SRJ schemes that have been identified so far and in the later sections. For the classical Jacobi method, $\mathcal{N}_{0.1}$ scales linearly with the total number of grid points N^2 and this is indeed borne out in our tests. On the other hand, optimal SRJ schemes seem to provide a slower than linear increase in $\mathcal{N}_{0.1}$ with total grid size. The slopes estimated for the lines in Fig. 4 are 0.97, 0.88, 0.81 and 0.71 for P=2, 3, 4 and 5 respectively. Thus, the advantage of SRJ schemes over the classical Jacobi is not just through a fixed multiplicative factor in the convergence rate (as is the case for Gauss-Seidel method that has a constant factor of two) but through a factor that increases with grid size. This is highly desirable, since large grid simulations on massively parallel computers, are precisely where this method would be most appropriate.

6.3. Trends in cycle size (M)

The cycle size M of optimal SRJ schemes is also of interest, since large cycle sizes would be unattractive from a practical point-of-view. The cycle size M is plotted against N for all P in Fig.5(a) and we find that cycle size grows nearly linearly with N for all values of P. This is not unexpected since increasing N increases the range of wavenumbers and additional iterations are needed in a given cycle to damp the error at each of these wavenumbers.

Figure 5(b) shows the cycle size M plotted against P for all N and we note that while for each N, the cycle-size increases with P, it rapidly approaches an asymptotic value. This combined with the trend shown in Fig. 4 implies that higher convergence rates obtained by increasing P do not require substantial additional increase in cycle-size.

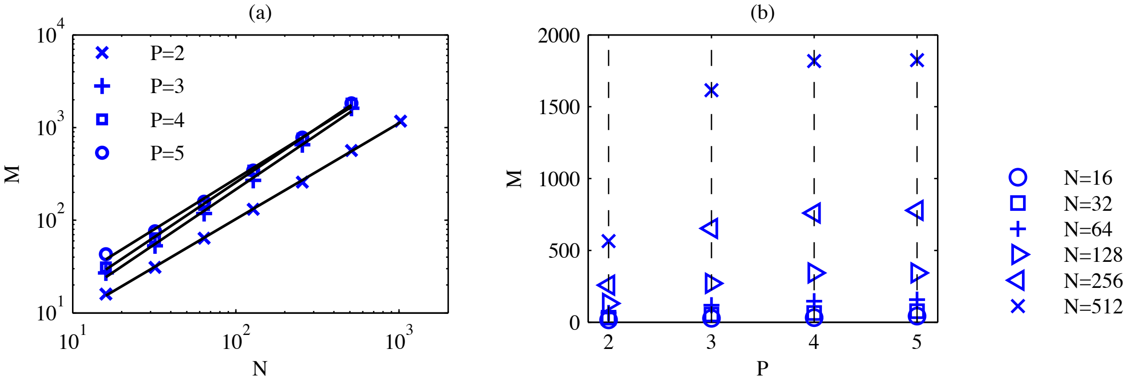

Figure 5: Variation of cycle size M for various multilevel schemes: (a) M versus N for various P; (b) M versus P for various N.

6.4. Relaxation Schedule

The optimal multilevel schemes identified above have been subjected to numerical validation and in this process, the importance of appropriate scheduling of the iterations during the SRJ cycles becomes apparent. We have previously mentioned that the schedule of relaxation factors does not effect the *mathematical analysis* of any SRJ scheme. However, in practice, due to roundoff errors and the arithmetic overflow associated with digital computers, appropriate scheduling of the ω 's during a SRJ cycle might be required to ensure numerical convergence of the SRJ scheme.

To understand this issue, we note first that for any single iteration that employs an over-relaxation, $|G|_{\max} = |1 - \omega\kappa_{\max}| \equiv 2\omega - 1$. Now consider, for example, the optimal scheme for $P = 4$ and $N = 256$ in Table 3 that consists of 760 iterations, of which, 86 iterations have a relaxation factor of 15.444. If all of these 86 iterations are carried out in succession, any initial error at $\kappa = 2$ would get amplified by a factor of $(29.888)^{86}$, which would lead to overflow even if the initial error at this wavenumber was machine-zero. Therefore, the appropriate approach to avoid overflow in these schemes is to appropriately distribute the relatively few over-relaxation in between multiple under-relaxation. In this way, any intermediate growth in the high wavenumber residual due to over-relaxation can be damped appropriately by the under-relaxations.

Our numerical experiments indicate that in most cases an even distribution of the over-relaxations over the entire cycle is sufficient to avoid overflow. Thus for instance, in the particular case of the optimal scheme for $P = 4$, $N = 256$ in Table 3 for which $M = 760$, $\vec{\Omega} = \{12329, 492.05, 15.444, 0.78831\}$ and $\vec{Q} = \{1, 9, 86, 664\}$, the first iteration would be the over-relaxation with $\omega_1 = 12329$. The 9 iterations with $\omega_2 = 492.05$ would be spaced $760/9 \approx 84$ iterations apart and the 86 iterations with $\omega_3 = 15.444$ would be spaced $760/86 \approx 8$ iterations apart. All of the intervening iterations would be with the 664 under-relaxations with $\omega_4 = 0.78831$.

In addition to the simple rule for scheduling the iterations provided above, a more formal procedure is also available to generate a robust schedule that guarantees avoidance of overflow. This robust schedule is determined as follows: we assume a constant unit spectrum

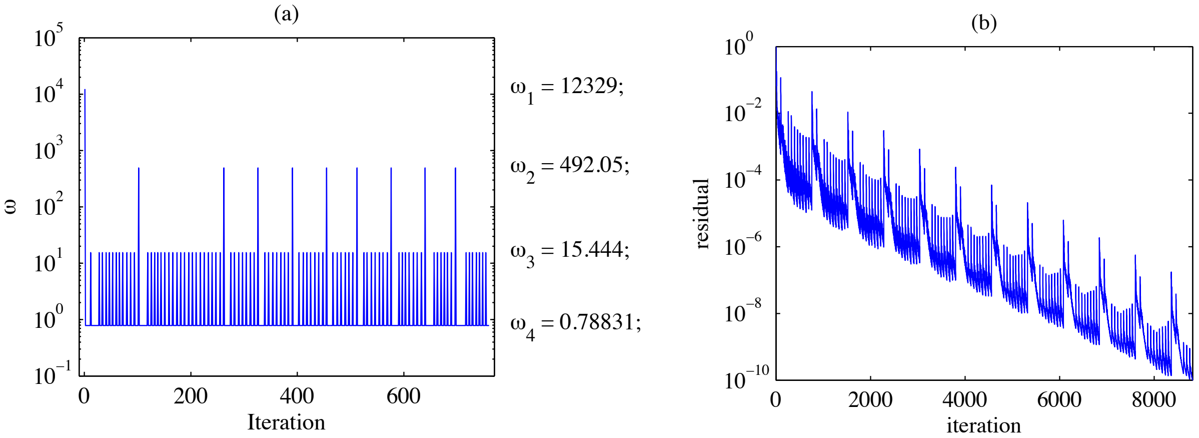

Figure 6: (a) Robust relaxation for optimal $P = 4$ scheme with $N = 256$. (b) Reduction in residual associated with the robust relaxation.

for the initial error $E_0(\kappa) = 1$ and use ω_1 (the largest over-relaxation) for the first iteration. According to Eq. (2), the spectrum after the first iteration would be $E_1(\kappa) = |1 - \omega_1\kappa|$. A recipe is now needed to choose the relaxation factor for the next iteration from those available, i.e. $(\omega_2, \omega_3, \ldots, \omega_P)$. This is done by identifying the maximum of $E_1(\kappa)$ and then choosing an ω from those available that produces the largest reduction in this maximum error. This second iteration generated a new error spectrum and the process of identifying the maximum in this spectrum and choosing an ω to maximize the reduction of this maximum is repeated until all the iterations in the cycle are exhausted. In Fig.6(a) we plot the relaxation schedule obtained from this procedure for the optimal $P = 4$ for $N = 256$. We can see that this particular sequence is not significantly different from an even spacing of the over-relaxations across the whole iteration cycle.

Figure 6(b) shows the convergence history for the optimal $P = 4$, $N = 256$ scheme with the robust relaxation schedule. The plot shows a reduction in the residual of nearly eight orders of magnitude over 11 SRJ cycles and we note that while the residual does not reduce monotonically, the overall trend shows robust convergence despite the extremely large over-relaxations employed. Results of the numerical tests carried out using the robust schedule are shown in the tables and these are very much inline with the predicted values.

A MATLAB script has been provided in the supplementary material to help the user determine the robust relaxation sequence for any given optimal SRJ scheme. The user is required to input N, $\vec{\Omega}$ and $\vec{Q}$, and a vector named W of length M is generated with its i^{th} value being the relaxation factor for i^{th} iteration.

7. $\rho \sim$ O(100) and Beyond

As mentioned before, the increasing stiffness of the equations in Appendix B for large P makes it difficult to extract higher-level optimal SRJ schemes. This stiffness can however be alleviated if $\vec{\beta}$ (or $\vec{Q}$) is prescribed and only $\vec{\Omega}$ determined from the optimization procedure. In that case, the free parameters are specified and we merely need to solve for the constraints

Table 5: SRJ schemes for some selected N that can accelerate JM by two orders of magnitude. The results are for 2D Laplace(ρ_L) and Poisson(ρ_P) equation with Neumann boundary conditions.

index	N	SRJ Scheme	ρ	ρ_L	ρ_P
a	512	$\vec{\Omega} = \{91299, 25979, 3862.1, 549.90, 80.217, 11.992, 1.9595, 0.59145\}$ $\vec{Q} = \{1, 3, 9, 27, 81, 243, 729, 1337\}$	148	147	151
b	512	$\vec{\Omega} = \{83242, 14099, 1334.1, 126.45, 12.193, 0.79246\}$ $\vec{Q} = \{1, 4, 16, 64, 256, 2504\}$	90	92	91
b	1024	$\vec{\Omega} = \{178919, 8024.1, 349.03, 15.9047, 0.799909\}$ $\vec{Q} = \{1, 7, 49, 343, 3087\}$	90	95	97
d	1024	$\vec{\Omega} = \{300015, 47617, 4738.4, 428.51, 39.410, 3.9103, 0.65823\}$ $\vec{Q} = \{1, 3, 13, 55, 227, 913, 2852\}$	190	199	197

that equalize all the interior extremums to $\Gamma_{\max}$, and with no intermediate unknowns like $\partial\omega_i/\partial\beta_j$. The parameter set obtained from this procedure will not correspond to the global optimum for the given N and P, but could still provide significant convergence acceleration. The goal here is to demonstrate convergence rates for grid sizes of practical interest that are two orders of magnitude better than the Jacobi method.

Our preceding analysis suggests that $\rho \sim O(100)$ would likely necessitate going to higher than 4-level schemes, and we have therefore focused our search on $P \geq 5$ schemes. Taking a cue from the observed trends for $P \leq 4$ SRJ schemes, we select a rapidly increasing sequence for β_i. In the absence of any other available rule, we choose $q_i = R^i$ for $i = 1$ to $P-1$, where R is an integer, and q_P was chosen to be significantly larger than $(q_1 + q_2 + \ldots + q_{P-1})$. Table 5 shows four selected schemes for values of N being 512 or 1024. We note that P ranges from 5 to 8 for these schemes and ρ ranges from 90 to 190. Following this, we execute our MATLAB algorithm to determine $\vec{\Omega}$ that maximizes the convergence for the given choice of $\vec{Q}$. The fourth scheme in the table was obtained serendipitously during our analysis and is noted here since it provides the highest (190-fold) convergence acceleration found so far.

The convergence performance predicted from the analysis is tested numerically by employing a robust relaxation schedule as described in section 6.4. The numerical validation does indeed bear out the prediction from the analysis and the results presented in this section therefore indicate that it is relatively easy to construct SRJ schemes that provide a two orders-of-magnitude increase in convergence rate over the Jacobi method. Also included in the table are convergence performance results for a corresponding Poisson equation with homogeneous Neumann boundary conditions where the source term consists of a dipole (equal and opposite delta functions located at $(x, y) = (0.25, 0.25)$ and $(0.75, 0.75)$). As expected, the presence of the source term has no significant effect on the convergence of the scheme. The performance is also insensitive to the exact placement of the sources.

8. Practical Considerations

8.1. Convergence on non-uniform grids

The vast majority of simulations in computational mechanics employ non-uniform meshes and in this section, we briefly describe the effect of grid non-uniformity on the performance of SRJ schemes. The ideas discussed here may be considered as being based notionally on a localized von Neumann analysis. Furthermore, for simplicity, we focus on a 1D problem although the ideas are equally applicable to multi-dimensional problems. For any non-uniform grid with N_0 grid points over a domain L, the minimum grid spacing ($\Delta_{\min}$) and an associated grid number $N_{\max} = L/\Delta_{\min}$ can be identified. It is trivial to note that $N_{\max} > N_0$. For such a grid, based on Eq. (3), a lower bound on $\kappa_{\min}$ is $\sin^2(\pi/2N_{\max})$, whereas a $\kappa_{\min}$ based on an average grid spacing would be $\sin^2(\pi/2N_0)$, which is the same as that for a uniform grid with N_0 grid points.

It has been noted before that any optimal SRJ scheme identified for a given grid can be applied to a larger grid with no loss in convergence rate. Based on this and on the fact that $N_{\max} > N_0$ for a nonuniform grid, we postulate that application of a SRJ scheme optimized for uniform grid with N_0 points can be applied to a non-uniform grid with the same number of points without reduction in ρ. We have tested this numerically for a variety of canonical non-uniform grids (with grid distributions based on exponential, tan-hyperbolic and cosine distributions) and found the postulate to be valid. Finally, we note that it is possible that application of SRJ scheme optimized for a grid number that is larger than N_0 but smaller than $N_{\max}$ might yield faster convergence for a non-uniform grid with N_0 points. However, depending on the grid distribution, such an approach could also diminish the convergence rate and even lead to divergence.

8.2. User implementation of SRJ schemes

In this section we briefly discuss how practitioners can utilize the SRJ schemes in their simulations. Appendix A and B provide the system of equations that are used to obtain an optimal SRJ scheme for a given grid size N. However, practitioners are not asked to solve this system repeatedly or concurrently with their simulations. The parameters of these optimal schemes can be computed once for a given grid and simply encoded into a given simulation. Thus, the use of the SRJ scheme does not entail an additional recurring computational cost of solving for the SRJ scheme.

Also, as noted, optimal schemes identified for a given grid size (say N_1) can be used for larger grids ($N > N_1$) while retaining the convergence acceleration (ρ) associated with the smaller (N_1) grid. In the current paper, we have identified multiple optimal SRJ schemes up to $P = 5$ for grids ranging from $N = 16$ to 512. Practitioners can therefore simply pick the scheme most appropriate for their grid from the ones provided. For example, if a user is solving a problem on a 600×600 grid, they could choose any of the optimal scheme identified in the table above for $N = 512$, and they would be guaranteed a convergence acceleration (ρ) equal to that associated with $N = 512$ for the chosen scheme. Once a scheme has been selected by the user, the relaxation sequence could be scheduled based on either of the two procedures described in section 6.4.

9. Comparison with Richardson's Method

Finally, we compare the current method to that proposed by Richardson [5]. The Jacobi method with SOR can be expressed as $u^{n+1} = u^n - \omega_n D^{-1}(Au^n - b)$ where D is the diagonal of A. Thus, for a uniform grid, the above expression is virtually identical to that of the Richardson method. However the current approach to maximizing convergence is fundamentally different from that of Richardson. In particular, in the context of the current analysis, Richardson's approach was to reduce Γ uniformly over the range $[\kappa_{\min}, \kappa_{\max}]$ by generating equispaced nodes of Γ in this range. In contrast, our strategy is to minimize $|\Gamma|_{\max}$ and this results in two key differences: first the nodes in the SRJ method are not equispaced in the interval $[\kappa_{\min}, \kappa_{\max}]$; second, optimal SRJ schemes naturally have many repetitions of the same relaxation factor whereas Richardson method generated distinct values of ω in each iteration of a cycle. A major consequence of the above differences is that while optimal SRJ schemes actually gain in convergence rate over Jacobi method as grids get larger, the convergence rate gain for the Richardson's procedure never exceeds $\rho = 5$.

10. Conclusions

A method for increasing the convergence rate of the Jacobi iterative method as applied to the solution of finite-difference approximations to elliptic partial differential equations, is presented. The method consists of a repeated sequence of iterations where a schedule of over- and under-relaxation is employed for the iterations within each sequence. These new schemes can be categorized naturally in terms of the number of 'levels', which are the distinct values of the relaxation parameter that are used in each sequence.

We determine the mathematical conditions that maximize the convergence rate for a scheme of any level, and for a given number of grid points. These mathematical conditions present as a set of implicit, non-linear coupled equations. The increasing stiffness of these equations for higher-level schemes currently precludes a solution for arbitrarily high level SRJ schemes, but systematic solution for up to level-five, generates SRJ schemes that provide a sixty fold increase in convergence over the classic Jacobi method. The performance of the schemes predicted by our analysis is validated via numerical experiments for a canonical two and three-dimensional elliptic problem. By prescribing some parameters of the scheme, we are able to derive higher (up to 8) level schemes, which provide more than a factor of hundred speedup over the Jacobi method. The analysis also indicates that additional gains in convergence rate may be possible by going to higher level schemes. These schemes therefore hold tremendous potential for accelerating large-scale parallel simulations in computational mechanics, as well as other fields where elliptic equations are a key component of the computational model.

A number of further extensions of this work are possible and worth exploring. First, SRJ schemes of levels higher than five and grid sizes larger than 512 could be derived by using more sophisticated methods to solve the equations in Appendix B. Such schemes would provide even higher convergence rates that what is noted in the current paper. Second, the schemes derived here are directly applicable to Laplace and Poisson equations on uniform or non-uniform grids, but could also be extended to the Helmholtz equation with relatively little effort. Third, as mentioned before, the methods described here could be modified into preconditioners for conjugate gradient methods. Finally, SRJ based "smoothers"[9] could be developed for use in multigrid methods. The advantage of such smoothers would be that their convergence and smoothing properties would be insensitive to domain-decomposition. Some of these extensions are currently being explored and will be presented in the future.

Acknowledgments

XY was supported by ONR grant N00014-12-1-0582. RM would like to acknowledge support from ONR as well as NSF (grants IOS1124804, IIS1344772 and OISE 1243482).

Appendix A. Equations for Optimal P=2 SRJ Schemes

The following is the system of five coupled equations that need to be solved for obtaining optimal parameters (ω_2, α and β) for the two-level ($P = 2$) SRJ scheme:

$$\begin{aligned}
-\ln\frac{\left(1-\frac{\omega_2\kappa_{\min}}{\alpha}\right)}{(1-\omega_2\kappa_{\min})}+\left(\frac{\beta}{\kappa_{\min}\omega_2-\alpha}+\frac{\beta}{\alpha}\right)\frac{\partial\alpha}{\partial\beta} &= \left(\frac{\kappa_{\min}\beta}{\omega_2\kappa_{\min}-\alpha}+\frac{(1-\beta)\kappa_{\min}}{\omega_2\kappa_{\min}-1}\right)\frac{\partial\omega_2}{\partial\beta} && \text{(A.1)}\\
\ln\left(\frac{2\omega_2}{\alpha}-1\right)-\ln(2\omega_2-1)-\left(\frac{\beta}{2\omega_2-\alpha}+\frac{\beta}{\alpha}\right)\frac{\partial\alpha}{\partial\beta} &= -\left(\frac{2\beta}{2\omega_2-\alpha}+\frac{2(1-\beta)}{2\omega_2-1}\right)\frac{\partial\omega_2}{\partial\alpha} && \text{(A.2)}\\
\ln\left(\beta\frac{1-\alpha}{\alpha}\right)-\ln(1-\alpha-\beta-\alpha\beta)-\left(\frac{\beta}{\alpha}+\frac{1}{1-\alpha}\right)\frac{\partial\alpha}{\partial\beta} &= 0 && \text{(A.3)}\\
\left(\frac{\beta+\alpha-\beta\alpha}{\alpha}-1\right)^{\beta}(1-(\alpha+\beta-\alpha\beta))^{(1-\beta)} &= \left(1-\frac{\omega_2\kappa_{\min}}{\alpha}\right)^{\beta}(1-\omega_2\kappa_{\min})^{1-\beta} && \text{(A.4)}\\
\left(\frac{\beta+\alpha-\beta\alpha}{\alpha}-1\right)^{\beta}(1-(\alpha+\beta-\alpha\beta))^{(1-\beta)} &= \left(\frac{\omega_2\kappa_{\max}}{\alpha}-1\right)^{\beta}(\omega_2\kappa_{\max}-1)^{1-\beta} && \text{(A.5)}
\end{aligned}$$

Appendix B. Equations for Optimal Multilevel (P $>$ 2) SRJ Schemes

Starting with the definition of the amplification factor for a P-level scheme shown in Eqn. 2 and following a process similar to that described above for the $P = 2$ scheme, we identify the following unknowns:

1. P distinct values of relaxation factor: $\omega_i,\ i = 1, 2, ..., P$;
2. their relative weight in a cycle: $\beta_i,\ i = 1, 2, ..., P$;
3. κ values of $P - 1$ local maxima: $\kappa_i^{\max},\ i = 1, 2, ..., P - 1$. These local extrema lie between two adjacent nodes and therefore $\kappa_i^{\max} \in \left(\frac{1}{\omega_i}, \frac{1}{\omega_{i+1}}\right)$
4. Finally, the partial derivatives: $\frac{\partial\omega_i}{\partial\beta_j},\ i = 1, 2, ..., P;\ j = 1, 2, ..., P - 1$, and
5. $\frac{\partial\kappa_i^{\max}}{\partial\beta_j},\ i = 1, 2, ..., P - 1;\ j = 1, 2, ..., P - 1$.

The above add up to a total of $2P^2$ unknowns. In order to solve for these unknowns we need an equal number of equations and these are:

1. the constraint $\sum\limits_{i=1}^{P} \beta_i = 1$;
2. a set of $P-1$ equations that determine $\kappa_i^{\max}$: $\frac{\partial}{\partial\kappa}\Gamma(\kappa_i^{\max}) = 0,\ i = 1, 2, 3, ...P-1$;
3. a second set of P constraints: $\Gamma(\kappa_{\min}) = \Gamma(\kappa_1^{\max}) = \Gamma(\kappa_2^{\max}) = ... = \Gamma(\kappa_{P-1}^{\max})... = \Gamma(\kappa_{\max})$;
4. a third set of $P-1$ equations to minimize $\Gamma(\kappa_{\min})$: $\frac{\partial}{\beta_j}\Gamma(\kappa_{\min}) = 0,\ j = 1, 2, 3, ...P-1$;
5. and finally, a set of $2P^2 - 3P + 1$ equations that relate $\frac{\partial\omega_i}{\partial\beta_j}, \frac{\partial\kappa_i^{\max}}{\partial\beta_j}$ to the other variables.

[1] Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations. Mathematics of computation, 22(104), 745-762.

[2] Patankar, S. (1980). Numerical heat transfer and fluid flow. CRC Press.

[3] Jacobi, C. G. J. Ueber eine neue Auflsungsart der bei der Methode der kleinsten Quadrate vorkommenden lineren Gleichungen. Astronomische Nachrichten, 22(20),(1845) 297-306.

[4] Gauss, C. F. Theoria motus corporum coelestium in sectionibus conicis solem ambientium, Perthes and Besser,(1809) Hamburg, Germany.

[5] Richardson, L. F. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, 210,(1911) 307-357.

[6] Young, D.M. Iterative methods for solving partial difference equations of elliptic type. PhD. Thesis,(1950) Department of Matematics, Harvard University.

[7] Liebmann, H. Die angen aherte Ermittlung harmonischer Funktionen und konformer Abbildungen. Sitzgsber. bayer. Akad. Wiss., Math.-phys. Kl,(1918) 385-416.

[8] Rosner, R. The opportunities and challenges of exascale computing. US Dept. of Energy Office of Science, Summary Report of the Advanced Scientific Computing Advisory Committee (ASCAC) Subcommittee. (2010)

[9] Briggs, W. L., McCormick, S. F. (2000). A multigrid tutorial (Vol. 72). Siam.

[10] Press, W. H. (2007). Numerical recipes 3rd edition: The art of scientific computing, chapter 10, Cambridge university press.

[11] Smith, G. D., Numerical Solution of Partial Differential Equations: Finite Difference Methods, Oxford University Press, 3rd ed. (1985)

[12] Isaacson, E. and Keller, H.B., Analysis of Numerical Methods, Wiley, 1966.