

Note

Accelerated Convergence of the Steepest Descent Method for Magnetohydrodynamic Equilibria

1. INTRODUCTION

Recently, a steepest-descent moment method [1] has been implemented for determining magnetohydrodynamic (MHD) equilibria. In Ref. [1] the Fourier amplitudes (R_{mn}, Z_{mn}) of the cylindrical coordinates (R, ϕ, Z) were discretized on a radial mesh. The resulting system of moment equations was solved using a descent algorithm to obtain the magnetic flux coordinate mapping.

In this paper, the ϵ -algorithm [2] is considered as a technique to accelerate the convergence rate of the steepest-descent method. The improved convergence rate of geometric sequences given by the ϵ -algorithm is well known. However, the relevance of this method specifically for the MHD equilibrium problem, and more generally in conjunction with the descent equations, has not been previously discussed.

The steepest-descent equations may be represented by the recursion relation:

$$O_n^{(i)} \mathbf{X}(n) = -\Delta W\{\mathbf{X}(n)\}. \tag{1}$$

Here, \mathbf{X} denotes the vector of discrete field amplitudes. For the MHD problem, $\mathbf{X} = \{R_{mn}, Z_{mn}\}$. The Δ notation symbolizes the gradient-difference operator for the positive definite energy functional W , and $O_n^{(i)}$ is a first ($i=1$)- or second ($i=2$)-order linear operator. Here, n assumes the role of a discrete time variable. The vector sequence $\mathbf{X}(n), n = 1, \dots$, generated by the iteration of Eq.(1) may be decomposed as follows:

$$\mathbf{X}(n) = \mathbf{X}_g(n) + \mathbf{X}_{nl}(n) + \mathbf{X}_\delta(n). \tag{2}$$

Here, the dominant asymptotic limit of $\mathbf{X}(n)$ corresponds to the geometric vector sequence,

$$\mathbf{X}_g(n) = \mathbf{X}_\infty + \sum_{j=1}^{m_x} A_j r_j^n \mathbf{E}_j, \tag{3a}$$

$$|r_j| < 1; \quad |r_1| \geq \dots \geq |r_{m_x}|, \tag{3b}$$

where \mathbf{X}_∞ is the stable equilibrium limit. The asymptotically vanishing nonlinear correction \mathbf{X}_{nl} is at least of second order, $\mathbf{X}_{nl}(n \rightarrow \infty) \simeq O(|\mathbf{X}_g(n) - \mathbf{X}_\infty|^2)$. The error term $\mathbf{X}_\delta(n)$ contains the effects of round-off and truncation error. If this term

is random, it should not produce numerical stability problems in the application of the ε -algorithm.

In Eq. (3), r_j are the complex characteristic roots of the linearized version of Eq. (1). They are related through a dispersion relation to the eigenvalues λ_j of the Jacobian matrix $\mathbf{J} \equiv -\nabla^2 W(\mathbf{X}_\infty)$ evaluated at the equilibrium. E_j are the eigenvectors associated with λ_j . The eigenvalues of \mathbf{J} are negative for a stable equilibrium, $\lambda_j \in [-|\lambda|_{\max}, -|\lambda|_{\min}]$, and for the MHD energy functional W they satisfy [1] the stiffness property $\delta_\lambda \equiv \lambda_{\max}/\lambda_{\min} \gg 1$.

The goal of any acceleration scheme is to use a finite subset of the sequential data $\mathbf{X}(n)$, $n = 1, \dots$, generated by Eq. (1), to predict the desired equilibrium value \mathbf{X}_∞ . Consider the application of the scalar ε -algorithm to each component of the vector geometric sequence, Eq. (3a). A typical component of Eq. (3a) has the form ($m \leq m_x$)

$$(\mathbf{X}_g)_s(n) \equiv S_n = S_\infty + \sum_{j=1}^m \alpha_j r_j^n. \tag{4}$$

It is known [3, 4] that for geometric sequences of the form given by Eq. (4) any successive $2m + 1$ elements S_n, \dots, S_{n+2m} can be used to determine S_∞ (the generalized Shank transform). However, because the theory makes use of Hankel determinants, it is impractical for computing with stiff systems. An alternative recursive nonlinear scheme [5], the ε -algorithm, also yields S_∞ after a finite number of operations involving $2m + 1$ successive S_n members. Because of its recursive structure, the ε -algorithm is appropriate for stiff systems. The basic recursion relation is

$$\varepsilon_{k+1}^{(j)} = \varepsilon_{k-1}^{(j+1)} + [\varepsilon_k^{(j+1)} - \varepsilon_k^{(j)}]^{-1}, \quad j = 0, \dots; k = 0, \dots, \tag{5a}$$

$$\varepsilon_{-1}^{(j)} = 0, \quad \varepsilon_0^{(j)} = S_j, \quad \text{for all } j. \tag{5b}$$

It can be proven [6] that if the parent geometric sequence satisfies certain minimal conditions then the $k = 2m$ column, $\varepsilon_{2m}^{(j)}$, will have the limiting value S_∞ .

For stiff systems, a large number of iterations, N , of Eq. (1) is required to attain a single e -folding of the smallest characteristic root in Eq. (4), regardless of the order of $O_n^{(i)}$. In particular [1], $N^{(i)} \simeq 0.5(\delta_\lambda)^{1/i}$ for $i = 1, 2$. Alternatively, it is possible to use $2m + 1 \lesssim N^{(i)}$ successive sequence members $\{S_n\}$ in the ε -algorithm [Eq. (5)] to obtain S_∞ directly. Since many e -foldings are generally required to obtain S_∞ by direct iteration alone, the use of the epsilon algorithm in this context can be very economical.

Depending on the distribution of characteristic roots [for $O^{(1)}$] or amplitudes [for $O^{(2)}$] contributing to the parent sequence, considerable sequence acceleration may be manifested by columns of the ε -algorithm of even order less than $2m$. This is apparent from the explicit form of the scalar algorithm of even order [7]

$$\varepsilon_{2k}^{(j)} = S_\infty + \frac{\sum_{l=1}^m \prod_{l=1}^{k+1} (\alpha_{\rho_l} r_{\rho_l}^l) \prod_{1 \leq p < q \leq k+1} (r_{\rho_p} - r_{\rho_q})^2}{\sum_{l=1}^m \prod_{l=1}^k [\alpha_{\rho_l} r_{\rho_l}^j (r_{\rho_l} - 1)^2] \prod_{1 \leq p < q \leq k} (r_{\rho_p} - r_{\rho_q})^2}. \tag{6}$$

Here, Ω_k^m denotes the set of all distinct permutations of length k taken from a set of length m ; and ρ_l is the subscript of the l th term in the particular permutation comprising Ω_k^m . Note that Ω_{m+1}^m is the null set, so that $\varepsilon_{2m}^{(i)} = S_\infty$ as required.

For first-order systems, the fact that all of the characteristic roots in Eq. (3) have distinct moduli leads to the following asymptotic form for $\varepsilon_{2k}^{(j)}$:

$$\lim_{j \rightarrow \infty} \varepsilon_{2k}^{(j)} \sim S_\infty + O(r_{k+1}^j). \quad (7)$$

This estimate was obtained by noting that the dominant contributions to the numerator and denominator of Eq. (6) arise from terms containing the product of roots with the largest modulus. Recalling the root ordering in Eq. (3), it is apparent that each successive even ε -column converges faster than the preceding one and, in particular, faster than the parent sequence (the $k=0$ column of the ε -algorithm).

For second-order central difference schemes [1] the characteristic roots of Eq. (3) have the same modulus, $r_j = \beta^{1/2} \exp(i\theta_j)$, where $\beta \sim 1 - \varepsilon$ and $\varepsilon \sim \delta_\lambda^{-1/2} \ll 1$. Although the theorem $\varepsilon_{2m}^{(j)} = S_\infty$ still pertains, the asymptotic behavior of the even ε -columns will differ from that of first-order systems as given by Eq. (7). Nevertheless, some convergence improvement may be anticipated even for the $2k < 2m$ ε -column, based on a smooth asymptotic approach (for increasing k) to the limiting value S_∞ . The numerical results in Section 3 also seem to confirm this expectation.

2. NUMERICAL STABILITY

Since the ε -algorithm is applied in practice to a sequence [Eq. (2)] that involves numerical perturbations of the exact asymptotic geometric sequence S_n , it is relevant to assess the numerical stability of the algorithm. One approach is to perturb an individual $\varepsilon_k^{(j)}$ element and to generate the corresponding perturbations of $\varepsilon_{k+1}^{(j)}$ and $\varepsilon_{k+2}^{(j)}$ as given by the recursive algorithm, Eq. (5). In this manner [8] the following stability criterion for the relative error, $\delta_k^{(j)} \equiv |\delta \varepsilon_k^{(j)}| / |\varepsilon_k^{(j)}|$, is obtained (for $S_\infty \neq 0$):

$$\lim_{j \rightarrow \infty} \delta_{2k+2}^{(j)} = R_{k+1} \delta_{2k}^{(j)}, \quad (8)$$

$$R_{k+1} = |r_{k+1}|^2 / |1 - r_{k+1}|^2 \leq 1.$$

Clearly $R_{k+1} \leq 1$ is desirable for stability and corresponds to either $\text{Re}(r_k) \leq 0$ or $|r_k| < \frac{1}{2}$.¹ From Ref. [1], it is clear that $|r_k|_{\max} \simeq 1$, regardless of a first- or second-order formulation. Because of stiffness, it is also generally not possible to find explicit finite difference schemes for $O_n^{(i)}$ in Eq. (1) with all characteristic roots satisfying $-1 < \text{Re}(r_k) < 0$. Nevertheless, by preconditioning the iterative scheme,

¹ Also note that in addition to Eq. (8) one must also have the usual condition $|r_k| < 1$.

it may be possible [9] to obtain effective characteristic roots that do satisfy $\text{Re}(r_k) < 0$ for all k .

If there were a preponderance of roots with negative real parts, then the majority of steepest-descent-generated sequences should be consistent with the sequence stability conditions. Some preliminary indications of this favorable root distribution are provided by the estimate $|\lambda|_{\max}^{1/2} \sim \{\text{No. of radial mesh points}\}$ [1]. Thus, the eigenvalues of $\nabla^2 W$ in Eq. (1) are expected to be packed closer to $-|\lambda|_{\max}$ rather than to $-|\lambda|_{\min}$. The additional observation that for both first- and second-order systems the optimum $O_n^{(i)}$ corresponds to roots satisfying² $\text{Re}[r(-|\lambda|_{\max})] = -\text{Re}[r(-|\lambda|_{\min})] < 0$ would support the expectation that the majority of steepest descent sequences are stable with respect to an ε -algorithm analysis.

An alternative interpretation of the above is that the ε -algorithm is more stable for nonmonotonic sequences, $\text{Re}(r_k) < 0$, than for monotonic ones. For the latter, $R_{k+1} \leq 1$ can be satisfied by taking appropriate subsequences of the parent sequence. Consider $S_n^* = S_{(In)}$, $n = 1, 2, \dots$. By selecting I so that

$$|r_j|_{\max}^I = \frac{1}{2}, \quad (9)$$

or $I \simeq 0.7N^{(i)}$, it is clear that the corresponding ε -algorithm will be numerically stable. The example presented in the next section corresponds to this situation.

3. NUMERICAL EXAMPLE

In Ref. [1] the following second-order operator was used in Eq. (1):

$$O^{(2)} \mathbf{X}_{(n)} \equiv \frac{\mathbf{X}(n+1) + \mathbf{X}(n-1) - 2\mathbf{X}(n)}{\Delta t^2} + \frac{\mathbf{X}(n+1) - \mathbf{X}(n-1)}{2\tau \Delta t}. \quad (10)$$

The magnetic axis was treated accurately by adopting a Galerkin expansion for the $m=0$ Fourier component of the radial inverse coordinate $R(\rho, \theta, \phi)$,

$$R^{O_n}(\rho) = R_0^{O_n} + \sum_{k=1} u_k L_k(\rho^2), \quad (11)$$

where L_k are Legendre polynomials. Now consider the application of an ε -algorithm analysis to the steepest-descent sequences for u_1 and u_2 . Two possible second-order theories have been considered here. One of these corresponds to keeping τ and Δt fixed (nonoptimized situation). The other involves prescribing optimal τ values [1]. It will be seen that application of the ε -algorithm to the first set of data, which proceeds to 3000 iterations, yields better results than the optimized τ -

² For first-order operators, $r_k = 1 - |\lambda_k| \Delta t$, where Δt is chosen optimally so that $\max(r_k)$ is minimized with respect to k , i.e., $\Delta t^{\text{opt}} = 2/(|\lambda|_{\max} + |\lambda|_{\min}) \simeq 2/|\lambda|_{\max}$. Hence, $r_k(-|\lambda|_{\max}) \simeq -1$. For second-order operators, a similar argument applies; see Handy [9].

varying situation at 4000 iterations. (Application of the ε -algorithm to the latter data set is invalid because of the variation of τ during the iteration.) Table I corresponds to the nonoptimized ($\tau = \text{const.}$) case. The parent sequence, up to 3000 iterations, is given in intervals of 100 iterations. At 3000 iterations, $u_1 = -5.147 \times 10^{-2}$ and $u_2 = -9.332 \times 10^{-3}$. Because of the monotonic nature of the sequence, a subsequence analysis of the type discussed in the preceding section was implemented. Accordingly, using the expression $I \approx 0.7N^{(2)}$ and the estimate [1] $N^{(2)} = 2\tau/\Delta t$ yields $I = 90$ for $\Delta t = 0.04$ and $\tau^{-1} = 0.4$. The corresponding ε -algorithm analysis is represented in Table 3. Note the predicted values $u_1 = -5.128 \times 10^{-2}$ and $u_2 = -9.50e \times 10^{-3}$.

These latter values for u_1 and u_2 compare well with those generated from an optimized (τ -varying) second-order formulation as presented in Table II. At 3000 iterations, $u_1^{\text{opt}} = -5.132 \times 10^{-2}$ and $u_2^{\text{opt}} = -9.457 \times 10^{-3}$, which are consistent with the ε -algorithm estimates. Indeed, the results contained in Tables II and III at 3000 iterations are far better converged than the corresponding entries of Table I. However, one can see that the results of the ε -algorithm actually surpass those of the optimized second-order code (Table II), because already on the basis of 3000 iterations of the Table I code the ε -algorithm predicts a reasonably stable limit value for both u_1 and u_2 . In contrast, the data in Table II show that u_2 does not

TABLE I
Nonoptimized, Second-Order Steepest-Descent Sequence for u_1 and u_2

Number of iterations	u_1	u_2
1000	-5.475 ($\times 10^{-2}$)	-6.135 ($\times 10^{-3}$)
1100	-5.425	-6.716
1200	-5.387	-7.163
1300	-5.353	-7.480
1400	-5.323	-7.748
1500	-5.294	-7.997
1600	-5.271	-8.208
1700	-5.253	-8.386
1800	-5.236	-8.542
1900	-5.221	-8.677
2000	-5.208	-8.791
2100	-5.197	-8.889
2200	-5.186	-8.974
2300	-5.180	-9.047
2400	-5.173	-9.109
2500	-5.166	-9.162
2600	-5.161	-9.208
2700	-5.157	-9.246
2800	-5.153	-9.279
2900	-5.150	-9.308
3000	-5.147	-9.332

begin to asymptote until at least 4000 iterations. There is some indication that the ultimate limit for u_2 will be less than the -9.469×10^{-3} entry and perhaps will even approach the ϵ -algorithm estimate of -9.50×10^{-3} .

It will be noted from the data in Table III that the ϵ -algorithm data for u_1 are slightly more susceptible to resonance effects than that of u_2 . This is because the original parent sequence for u_1 , as given in Table I, already converges much faster than that of u_2 . Thus, considering Eq. (4) one can see that if two successive elements satisfy $\epsilon_k^{(j+1)} = \epsilon_k^{(j)}$ then the ensuing recursively computed expression $\epsilon_{k+1}^{(j)}$ will be singular. Clearly, round-off error makes such infinities anomalously large. Theoretically, one can regulate away [10] such potential infinities. In practice, one

TABLE II
Optimized Second-Order, Steepest-Descent Parent Sequences for u_1 and u_2

Number of iterations	u_1	u_2	$1/\tau$
1000	$-5.488 (\times 10^{-2})$	$-6.006 (\times 10^{-3})$	$3.940 (\times 10^{-1})$
1100	-5.430	-6.624	3.060
1200	-5.368	-7.289	1.779
1300	-5.306	-7.885	2.358
1400	-5.259	-8.269	5.453
1500	-5.235	-8.477	1.351
1600	-5.212	-8.728	2.084
1700	-5.191	-8.984	1.141
1800	-5.161	-9.207	5.803
1900	-5.152	-9.260	6.503
2000	-5.152	-9.279	3.673
2100	-5.150	-9.315	3.970
2200	-5.144	-9.352	2.951
2300	-5.141	-9.376	3.165
2400	-5.139	-9.395	3.665
2500	-5.137	-9.412	2.077
2600	-5.135	-9.428	1.518
2700	-5.134	-9.441	4.211
2800	-5.133	-9.450	4.007
2900	-5.132	-9.454	1.312
3000	-5.132	-9.457	1.586
3100	-5.132	-9.461	2.228
3200	-5.132	-9.464	5.863
3300	-5.131	-9.465	3.277
3400	-5.131	-9.466	5.838
3500	-5.131	-9.467	3.896
3600	-5.131	-9.468	3.311
3700	-5.131	-9.468	8.631
3800	-5.131	-9.469	6.143
3900	-5.131	-9.469	2.589
4000	-5.131	-9.469	2.798

may ignore them, provided they have no deleterious effects on successively higher-order columns of the ε -algorithm ansatz. This is clearly the behavior of both u_1 and u_2 , as given in Table III.

TABLE III
 ε -Algorithm as Applied to Data of Table I, for u_1 and u_2 , Respectively

ε_2	ε_6	ε_{10}	ε_{14}	ε_{18}
-5.243 ($\times 10^{-2}$)				
-5.064				
-5.098	-5.006			
-4.453	-5.055			
-5.183	-5.139	-5.128		
-5.188	-5.139	-5.128		
-4.947	-5.112	-5.128	-5.128	
-5.109	-5.165	-5.129	-5.128	
-5.124	-3.002 ($\times 10^5$)	-5.130	-5.128	-5.128
-5.137	3.603	-5.122	-5.128	-5.128
-5.148	-5.136	-5.127	-5.128	
-5.116	-5.153	-5.128	-5.128	
-5.124	-5.087	-5.129		
9.999 ($\times 10^5$)	-5.112	-5.127		
-5.149	-5.134			
-5.141	-5.135			
7.206 ($\times 10^6$)				
-5.141				
-8.654 ($\times 10^{-3}$)				
-8.253				
-9.214	-9.190			
-1.126 ($\times 10^{-2}$)	-9.502			
-9.380	-9.523	-9.526		
-9.346	-9.521	-9.509		
-9.648	-9.494	-9.485	-9.491	
-9.545	-9.491	-9.489	-9.518	
-9.410	-9.487	-9.489	-9.485	-9.499
-9.500	-9.494	-9.489	-9.491	-9.510
-9.530	-9.479	-9.486	-9.492	
-9.491	-9.485	-9.484	-9.490	
-9.458	-9.488	-9.481		
-9.474	-9.474	-9.470		
-9.510	-9.473			
-9.427	-9.470			
-9.497				
-9.518				

ACKNOWLEDGMENTS

The authors thank J. C. Whitson and Professor R. Mickens for useful discussions. Research was sponsored by the Office of Fusion Energy, U.S. Department of Energy, under Contract W-7405-eng-26 and UN. CAR./OR 19X43371C(S/C) with the Union Carbide Corporation.

REFERENCES

1. S. P. HIRSHMAN AND J. C. WHITSON, *Phys. Fluids* **26** (1983), 3553.
2. A. GENZ, in "Pade Approximants" (P. R. Graves-Morris, Ed.), Institute of Physics, London, 1972.
3. D. SHANKS, *J. Math. Phys.* **34** (1955), 1.
4. C. M. BENDER AND S. A. ORSZAG, "Advanced Mathematical Methods for Scientists and Engineers," McGraw-Hill, New York, 1978.
5. P. WYNN, *Math. Tables Aids Comput.* **10** (1956), 91.
6. J. B. MCLEOD, *Computing (Arch. Electron. Rechnen)* **7** (1971), 17.
7. P. WYNN, *SIAM J. Numer. Anal.* **3** (1966), 91.
8. J. WIMP, "Sequence Transformations and Their Applications," Academic Press, New York, 1981.
9. C. R. HANDY, "The Preconditioning of ϵ -Algorithm Steepest Descent Methods for Linear Stiff Systems," Atlanta University preprint, 1984.
10. P. WYNN, *Nord. Tidskr. Informat.-Behandl.* **3** (1963), 175.

RECEIVED: April 17, 1984

C. R. HANDY

*Department of Physics
Atlanta University
Atlanta, Georgia 30314*

S. P. HIRSHMAN

*Oak Ridge National Laboratory
Oak Ridge, Tennessee 37830*