PFC/RR-84-18 DOE/ET/51013-138
UC20G

AGGRESSIVE TIME STEP SELECTION FOR THE TIME ASYMPTOTIC VELOCITY DIFFUSION PROBLEM

D. W. Hewett,* V. B. Krapchev, K. Hizanidis, A. Bers

MIT, Plasma Fusion Center
Cambridge, MA 02139

December 1984

*Now at LLNL, Livermore, CA

This work was supported by the U.S. Department of Energy Contract No. DE-AC02-78ET51013.

AGGRESSIVE TIME STEP SELECTION FOR THE TIME ASYMPTOTIC VELOCITY DIFFUSION PROBLEM

by

D.W. Hewett,* V.B. Krapchev, K. Hizanindis, and A. Bers
MIT Plasma Fusion Center
Cambridge, Massachusetts 02139

Abstract

An aggressive time step selector for an ADI algorithm is presented that is applied to the linearized 2-D Fokker-Planck equation including an externally imposed quasilinear diffusion term. This method provides a reduction in CPU requirements by factors of two or three compared to standard ADI. More important, the robustness of the procedure greatly reduces the work load of the user. The procedure selects a nearly optimal Δt with a minimum of intervention by the user thus relieving the need to supervise the algorithm. In effect, the algorithm does its own supervision by discarding time steps made with Δt too large.

* Now at LLNL, Livermore, CA.

Introduction

The success of current drive experiments in several toroidal magnetic fusion experiments has caused renewed interest in the behavior of the electron momentum distributions subject to externally imposed RF fields. One aspect of this problem is to determine the time asymptotic electron momentum distribution function subject to a momentum space diffusion operator. The equation to be solved

$$\frac{\partial f}{\partial t} = -\frac{\partial S_{\|}}{\partial p_{\|}} - \frac{1}{p_{\perp}}\frac{\partial}{\partial p_{\perp}}(p_{\perp} S_{\perp}) \tag{1}$$

where

$$S_{\|} = -D_{\|}\frac{\partial f}{\partial p_{\|}} - D_X\frac{\partial f}{\partial p_{\perp}} - F_{\|} f$$

and

$$S_{\perp} = -D_X\frac{\partial f}{\partial p_{\|}} - D_{\perp}\frac{\partial f}{\partial p_{\perp}} - F_{\perp} f$$

with

$$D_{\|} = \frac{\gamma}{p^3}\left[\left(\varsigma - \frac{\gamma^2}{2p^2}\right)p_{\perp}^2 + \gamma^2\frac{p_{\|}^2}{p^2}\right] + D_q$$

$$D_{\perp} = \frac{\gamma}{p^3}\left[\left(\varsigma - \frac{\gamma^2}{2p^2}\right)p_{\|}^2 + \gamma^2\frac{p_{\perp}^2}{p^2}\right]$$

$$D_X = -\frac{\gamma}{p^3}\left[\varsigma - \frac{3\gamma^2}{2p^2}\right]p_{\|}p_{\perp}$$

$$F_{\|} = \frac{\gamma^2}{p^3}p_{\|}$$

$$F_{\perp} = \frac{\gamma^2}{p^3}p_{\perp}$$

and

$$\varsigma = (Z+1)/2$$

$$p^2 = p_{\perp}^2 + p_{\|}^2$$

has been derived and motivated in a recent publication [1] and will not be discussed in detail here. This equation describes an idealized situation in which no spatial gradients are allowed and no consideration is given to the mechanisms which establish the (quasilinear) momentum diffusion $D_q(p_{\|})$. Even with these simplifications, eq. (1) requires substantial computation before yielding the time asymptotic state.

The most obvious and frequently used approach is to begin with a Maxwellian electron distribution $f_M(p^2)$ at time $t = 0$ which is subjected to the momentum

diffusion $D_q(p_\parallel)$. The distribution, discretized on a mesh in $p_\parallel$, $p_\perp$, is then advanced in time using an explicit finite time difference integration version of eq. (1). This approach has been pursued at Texas [2], Princeton [3], and IREQ [4]. Their experience, and our own, suggests that a considerable computational effort can be required to reach the asymptotic state using the largest fixed stable time step.

Some of this previous work has emphasized a more accurate (higher order) representation in momentum for f. Our experience suggests that, for the majority of our parameter runs, the initial rapid time evolution subsides after a relatively small number of time steps. Subsequent evolution is much slower. A frequent error is that the evolution is assumed to be near the asymptotic limit when in fact it is still slowly evolving. Our approach is to use the simplest second-order finite-difference representation in momentum of eq. (1) so that the operations per time step are minimized. Errors made by this approach can be readily monitored and reduced as required by using a finer mesh. This simple form is solved easily by a noniterative ADI procedure – further relaxing some explicit stability contraints on the time step. Large reduction in CPU requirements have been achieved through the use of variable time step size. We have found that the number of time steps required to reach an asymptotic state can be reduced by a factor of three and the less sophisticated finite-difference can be two to four times cheaper per time step for comparable resolution.

In what follows we present our simple finite-difference version of eq. (1), the noniterative ADI procedure approach to the solution, the philosophy and details of the adaptive time step selection, and some representative results.

ADI Integration of the Finite-Difference Equation

The conservative finite-difference representation of eq. (1) used in this work is

$$\frac{f_{i,j}^{N+1} - f_{i,j}^{N}}{\Delta t} = \frac{S_{\parallel i+1/2,j}^{N+\omega_\parallel} - S_{\parallel i-1/2,j}^{N+\omega_\parallel}}{\Delta p_\parallel} + \frac{p_{\perp j+1/2} S_{\perp i,j+1/2}^{N+\omega_\perp} - p_{\perp j-1/2} S_{\perp j-1/2}^{N+\omega_\perp}}{p_{\perp j} \Delta p_\perp} \qquad (2)$$

where the superscript represents time level ($t = N\Delta T$) and $\omega_\parallel$, $\omega_\perp$ are parameters which specify intermediate time levels. We use a rectangular spatial mesh with the discretization

$$p_{\parallel i} = p_{\parallel \min} + (i - 1.5)(p_{\parallel \max} - p_{\parallel \min})/N_\parallel$$

and

$$p_{\perp j} = (j - 1.5) p_{\perp \max}/N_\perp$$

where $p_{\| \min} < p_{\| j} < p_{\| \max}$ and $0 < p_{\perp j} < p_{\perp \max}$ define the computation region which is the uniform orthoganal $N_{\|} \times N_{\perp}$ mesh.

An ADI time step is accomplished by the two half time steps

$$\begin{aligned} \frac{2}{\Delta t} f^{N+1/2} + H(f^{N+\omega_{\|}}) &= \frac{2}{\Delta t} f^{N} - V(f^{N+\omega_{\perp}}) \quad ; \quad \omega_{\|} = 1/2 \quad \omega_{\perp} = 0 \\ \frac{2}{\Delta t} f^{N+1} + V(f^{N+\omega_{\perp}}) &= \frac{2}{\Delta t} f^{N+1/2} - H(f^{N+\omega_{\|}}) \; ; \; \omega_{\|} = 1/2 \quad \omega_{\perp} = 1 \end{aligned} \tag{3}$$

where H (V) represents all the terms involving derivatives with respect to $p_{\|}$ ($p_{\perp}$). These two half steps can be shown [5,6] to be equivalent to a single second order time integration step from $N\Delta t$ to $(N+1)\Delta t$. (To see this eliminate $Hf^{N+\omega_{\|}}$ and assume H and V operators commute.) As the time asymptotic limit is approached the terms with Δt^{-1} subtract out leaving

$$(H+V)f = 0 \tag{4}$$

which is the desired result.

Aggressive Time Step Selection

Simply running the ADI procedure just outlined with a fixed time step provides a significant gain in the allowable explicit time step size. The explicit equation [eq. (2) with $\omega_{\|} = \omega_{\perp} = 0$] has a stability constraint on Δt given by

$$\Delta t < \frac{\Delta p}{4 D_0} \tag{5}$$

where D_0 is the maximum D_q and $\Delta p = \Delta p_{\|} = \Delta p_{\perp}$ [7]. We find that Δt can safely be several times larger than those required for the stable integration of the fully explicit form of eq. (2).

The ADI procedure still exhibits a stability requirement on Δt, but fortunately, insight can be gained from the empirical tests to determine the stability bound on Δt. The deviation of f in a run with Δt slightly above the stability limit compared with a run with conservative Δt is minimal for a few tens of time steps. Eventually the result will deviate exponentially. The distribution in the unstable case is eventually dominated by high-frequency grid oscillations – a standard indication associated with exceeding Δt limits [5,6]. If Δt is large enough, the error exponentiates and the run terminates. As Δt is reduced, the high frequency behavior becomes progressively smaller and takes more time steps to grow. Often no clear distinction exists for a

finite number of iterations between "stable" and "unstable" values of Δt. In practice a "conservative" Δt is selected by some ill-defined method and the integration proceeds until the finite-differenced time derivative of eq. (2) is suitably small across the mesh. Obviously a more systematic mode of operation needs to be found.

Because the high frequency errors require a finite amount of time to reach destructive levels, important gains can be made by running with larger Δt's and occasionally reducing Δt to damp out the spurious high frequency components. This tactic is especially effective in circumstances in which intermediate time solutions are unimportant; we are interested only in the time asymptotic state here.

In the procedure we now outline, the goal is to achieve the time asymptotic state as quickly as possible. This goal is achieved by using as few time steps as possible consistent with stability of the solution. The details of the evolution are unimportant. Eq. (1) has a unique time asymptotic solution if boundary conditions consistent with the elliptic character of the equation are imposed. We require some measure ϵ of the relative stability of the solution at each step in the integration. The ϵ is defined so that stability is indicated by it's monotonic decrease. As long as ϵ decreases, the solution is considered acceptable up to that "time" level and ϵ is saved for future comparison. The Δt is gradually increased until ϵ no longer decreases. An increasing ϵ indicates the onset of numerical instability. We follow this last time step, which is on the verge of instability, by a step with a smaller Δt which is intended to allow the solution to stabilize itself. If a reevaluation of ϵ confirms that the solution is again stable (ϵ is again decreasing), the new solution is accepted, and another time step series is initiated with this same Δt that is near the stability limit (at this time in the integration). If the small step does not stabilize the solution, a second small step is taken in a final attempt to stabilize it. Should the second attempt succeed, we would be making optimal progress toward the asymptotic limit; should it fail, the last big step and these two small steps are discarded, the solution is restored at the last acceptable configuration, and Δt is reduced substantially before an attempt is made to continue the solution.

This procedure provides for aggressive increases in Δt to the stability limit during periods of inactivity in the time dependence of f. Should the activity increase, a rapid retrenchment is triggered which is very protective of the solution. If a situation is encountered in which the only solution requires an increase in ϵ, this algorithm will fail. The signature of such a mode is that no further time steps are "acceptable" and Δt is reduced to a very small number. This mode is never observed in the present application to eq. (1).

These concepts are in some ways analogous with – and motivated by – those that provide an intuitive understanding of the highly successful MULTIGRID [7] approach to the solution of partial differential equations. The MULTIGRID approach is an

iterative procedure which can be considered a method that selectively reduces the residual in the part of the wavelength spectrum best represented by the mesh. Those modes in the residual with a wavelength on the order of the mesh spacing are the most rapidly decaying modes in relaxation schemes. When these residuals become suitably small, the algorithm shifts to a coarser or finer mesh to work on another part of the residual spectrum. The algorithm presented here does the analogous thing in time. Our algorithm is continuously attempting to reduce low frequency modes (by attempting larger Δt's) while high frequency residuals are held to low levels by occasional retreats to small Δt's.

An Aggressive Δt Selector

The strategy that our algorithm must accomplish is to take a three-time-step series consisting of a large step Δt followed by up to two small steps f/STEP to damp the high frequency modes stimulated by the large step. The goal is to increase the large step to the onset of instability and then restabilize the solution using the small steps. With appropriate measures of our progress, and optimization of Δt adjustment parameters, we may increase or decrease Δt in order to make optimal progress towards the asymptotic state. The details of the resulting algorithm are given in the flow chart in Fig. 1.

The previously described three-step series can easily be recognized in the flow chart. The parameters of the procedure explicitly displayed in the flow chart are now defined. In the UNSTABLE section, the parameter STEP determines how much smaller the small steps are than the initial step of the series Δt_N. Typically STEP is set at 3.0 to 10.0. The parameter SLOW is used if a second small step is required to reduce the next initial or big Δt. Should the second small step fail to stabilize the integration, a "backspace" is triggered and BACK reduces the next initial Δt by a substantial factor – typically BACK=10. In the STABLE section, SPED is the amount the next big Δt is increased if the initial Δt has failed to make the solution unstable. Best results are obtained for this problem for the modest SPED $\sim$ 0.5. If one small step is enough to stablize the solution, no substantial changes are required. It is found that the algorithm frequently stagnates in this state. STAG is a small change introduced to encourage the algorithm to try other values of Δt. Usually we run with STAG=1.01.

The measure of progress which seems to work best in this case is the largest absolute value of $\partial f/\partial t$ on the entire mesh. The requirement that $\epsilon = |\partial f/\partial t|_{\max}$ decrease after each three step series assures stable progress towards the asymptotic state. That ϵ always decreases is reasonable for diffusion processes. It also seems reasonable that other measures could prove acceptable. The L2 norm of $\partial f/\partial t$ will also increase if a spurious high frequency instability is imminent. For our purposes it worked no better and required more work to implement. Weighting ϵ so that regions

where f is largest contribute more to the selection of Δt failed universally. Apparently unchecked instability in regions of small f will quickly destroy the solution in all regions.

Results

A considerable amount of experience has been gained with this algorithm for the diffusion function

$$D_q = D_0 \exp\{-[(p_\| - p_0)/\Delta p]^4\} \tag{6}$$

We have considered two cases to present the method:

CASE 1	CASE 2
$D_0 = .5$	$D_0 = 10.$
$p_0 = 5.$	$p_0 = 7.$
$\Delta p = 1.$	$\Delta p = 1.$

The results for both problems are displayed in Table 1 for a 101×51 mesh with $-12.0 < p_\| < 12.0$ and $0.0 < p_\perp < 12.0$. Standard ADI (fixed Δt) results are labeled ADI; Aggressive ADI (variable Δt) results are labeled AADI.

TABLE 1

	CASE 1	CASE 2
explicit Δt	.0288	.00144
STABLE ADI Δt (UNSTABLE)	0.3333(0.4)	0.1(0.11)
ADI STEPS	1318	2362
ADI CPU	37.10 sec	66.55 sec
AADI CPU	25.60 sec	23.63 sec
ADI TIME	437.3	234.8
AADI "TIME"	79.7	41.03
TOLERANCE	10^{-10}	10^{-12}
BEST AADI PARAMETERS		
SPED	0.3	0.6
SLOW	3.5	3.5
STAG	1.01	1.01
STEP	10.0	10.0
BACK	10.0	10.0

The best means of evaluating the effectiveness of the aggressive Δt selector is to run our code with Δt fixed. Standard ADI is recovered by setting

SLOW=SPED=STEP=STAG=BACK=1.0. Setting all the AADI parameters equal to 1.0 requires that Δt remain constant and instability is evident the first time the initial Δt step is not accepted. Empirically we can optimize the standard ADI scheme by increasing Δt until our modified AADI algorithm detects an increasing ϵ. The comparison is still subjective in that our conclusions about the improvement can be colored by the degree of conservativeness used in selecting the fixed Δt – so we have also indicated the smallest unstable Δt we found. As shown in the first entry of Table 1, the ADI procedure provides some impressive increases in time step size over the traditional fully explicit limits.

Also shown in Table 1 are the AADI results along with the "best" AADI parameters–again determined empirically. A useful guideline in determining AADI parameters is that the Δt selection should be aggressive enough to invoke a backspace approximately every 100 iterations. The relative merit of ADI versus AADI is reflected in the CPU time required to achieve a residue ϵ less than a given tolerance. As also can be seen in the table, the number of "time units" required for solution are quite different. The resulting distribution is the same in each case but the meaning of accumulated time (summation of the Δt's) has no physical interpretation in the AADI run. The AADI is an elliptic equation solving procedure in that the only meaningful "times" are the initial guess at time $t = 0$ and the time asymptotic time $t = \infty$ state. Trying to follow the solution as a function of time with AADI is likely to be misleading.

Although some care was taken with these examples to achieve the smallest CPU times in each case for a fair comparision, the typical savings in practice tended to be larger owing to user conservativeness with ADI. The ADI scheme with Δt just over the stability boundary requires a substantial fraction of the anticipated run time before revealing the instability. The ensuing frustration gives rise to excessive caution in selecting the next Δt. The most practical feature of the AADI Δt selection process is that even with less than optimal choices for the parameters the method will not fail due to the time step growing too large – a relief to those who must make a large number of parameter studies.

A typical result of this procedure is shown in Fig. (2) – several other examples can be found in references [8], [9], and [10]. We have found that for problems using a "smooth" D_q, ϵ can generally be reduced to a value less than 10^{-10} in 30 to 120 seconds. More iterations are generally needed as the position of the maximum D_q is moved to greater velocities or its profile is made sharper. The more pathological choices of D_q, such as discontinuous profiles chosen for ease in analytic comparisons [8] have required as many as a few thousand iterations even with AADI.

Summary and Conclusions

We have developed an aggressive time step selector for the standard ADI algorithm. The algorithm is easily tuned for our test problem – the linearized 2-D Fokker-Planck equation with quasilinear diffusion The procedure selects a nearly optimal Δt with a minimum of intervention by the user.

The time integration is accomplished by a standard ADI scheme run in a series of three time steps: a large Δt that is large enough or increased to be large enough to trigger high frequency numerical instability followed by as many as two small Δt's in an attempt to "restabilize" or smooth out the high frequencies. Standards are defined for successful control of the incipient instability. If unsuccessful, a retrenchment of the integration is triggered which discards the latest time step series and tries again with a substantially smaller first Δt.

Run times for our problem are reduced by factors of around three over standard ADI improvements. More important, however, is the fact that the algorithm supervises its own Δt selection–greatly reducing the user attentiveness required to accomplish parameter studies.

Acknowledgment

This work was supported by DOE Contract No. DE-AC02-78ET-51013.

References

1. K. Hizanidis and A. Bers, *Phys. Fluids*, November 1984 (in press).
2. J. C. Wiley, D-I. Choi, and W. Horton, *Phys. Fluids* 23, 2193 (1980).
3. C.F.F. Karney and N. J. Fisch, *Phys. Fluids* 22, 1817 (1974).
4. V. Krapchev, K. Hizanidis, A. Bers, and M. Shoucri, "Current Drive by LH Waves in the Presence of a DC Electric Field," *Bull. Amer. Phys. Soc.* 27, Oct. 1982, p. 1006; and M. Shoucri, V. Krapchev, and A. Bers, "A SADI Numerical Scheme for the Solution of the 2-D Fokker-Planck Equation," *IEEE Conference Record - Abstracts, 1983 IEEE Int. Conf. on Plasma Science*, p. 67.
5. R.D. Richtmeyer and K.W. Morton, *Difference Methods for Initial-Value Problems*, Interscience Publishers, John Wiley and Sons, Second Edition, 1967.
6. P.J. Roache, *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, 1976.
7. A. Brandt, *AIAA J.*, 18, 1165 (1980).
8. V. B. Krapchev, D. W. Hewett, and A. Bers, *Phys. Fluids* (to appear Feb. 1985).
9. D. Hewett, K. Hizanidis, V. Krapchev and A. Bers, "Two-Dimensional and Relativistic Effects in Lower-Hybrid Current Drive" (invited), Proceedings of the IAEA Technical Committee Meeting on Non-Inductive Current Drive in Tokamaks, Culham, U.K., April 18-21, 1983; Culham Report CLM-CD (1983).
10. K. Hizanidis, D. Hewett and A. Bers, "Solution of the Relativistic 2-D Fokker-Planck Equation for LH Current Drive," *Proc. of the 4th Int. Symp. on Heating in Toroidal Plasmas*, Int. School of Plasma Physics and ENEA, Rome, Italy, 1984, pp. 668-673.

FIGURE CAPTIONS

Figure 1 Flow chart of the Aggressive ADI Algorithm.

Figure 2 Contours of the steady-state distribution function in momentum space: time-asymptotic numerical solution of Eq. (1) for a given D_q (see Ref. 10).

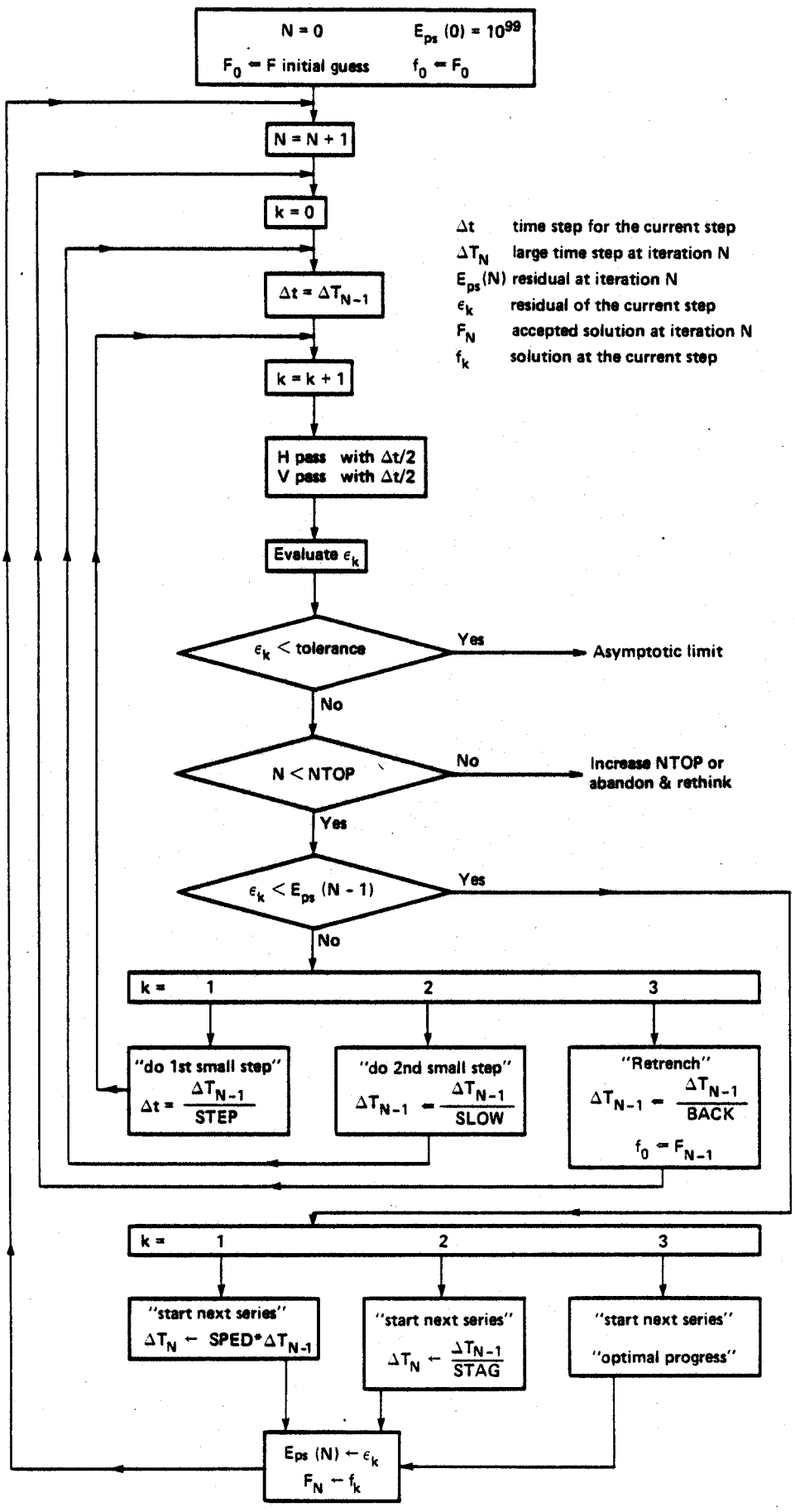
N = 0
$E_{ps}(0) = 10^{99}$
$F_0 \leftarrow$ F initial guess
$f_0 \leftarrow F_0$
N = N + 1
k = 0
$\Delta t = \Delta T_{N-1}$
k = k + 1
H pass with Δt/2
V pass with Δt/2
Evaluate ϵ_k
$\epsilon_k <$ tolerance
Yes
Asymptotic limit
No
N < NTOP
No
Increase NTOP or abandon & rethink
Yes
$\epsilon_k < E_{ps}(N-1)$
Yes
No
k = 1 2 3
"do 1st small step"
$\Delta t = \frac{\Delta T_{N-1}}{STEP}$
"do 2nd small step"
$\Delta T_{N-1} \leftarrow \frac{\Delta T_{N-1}}{SLOW}$
"Retrench"
$\Delta T_{N-1} \leftarrow \frac{\Delta T_{N-1}}{BACK}$
$f_0 \leftarrow F_{N-1}$
k = 1 2 3
"start next series"
$\Delta T_N \leftarrow SPED * \Delta T_{N-1}$
"start next series"
$\Delta T_N \leftarrow \frac{\Delta T_{N-1}}{STAG}$
"start next series"
"optimal progress"
$E_{ps}(N) \leftarrow \epsilon_k$
$F_N \leftarrow f_k$
Δt time step for the current step
ΔT_N large time step at iteration N
$E_{ps}(N)$ residual at iteration N
ϵ_k residual of the current step
F_N accepted solution at iteration N
f_k solution at the current step

FIGURE 1

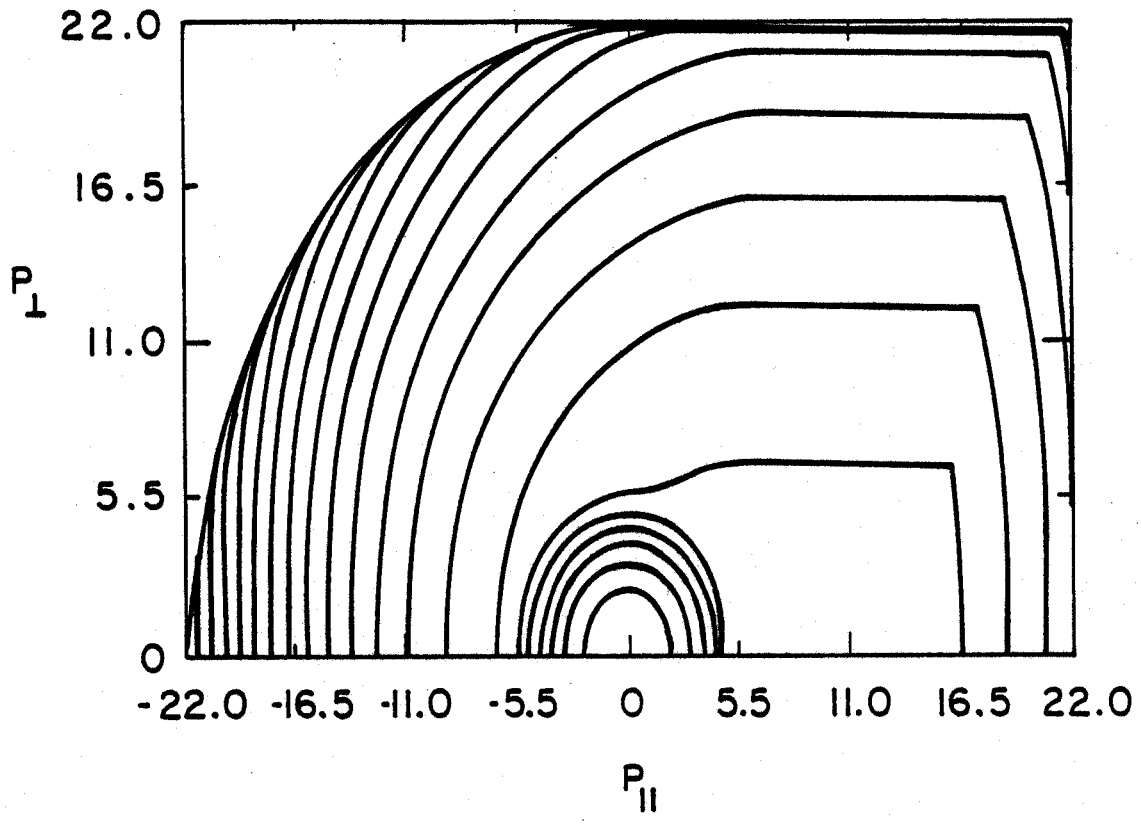

Figure 2