SEMI-LINEAR DIFFERENCE SCHEMES FOR SINGULAR PERTURBATION PROBLEMS IN ONE DIMENSION

Jiachang Sun[1]

Technical Report #216/82
May 17, 1982

This research was supported in part by ONR Grant N00014-82-K-0184, and Yale University.

[1] Computing Center, Academia Sinica, Beijing, China

SEMI-LINEAR DIFFERENCE SCHEMES
FOR SINGULAR PERTURBATION PROBLEMS IN ONE DIMENSION.

ABSTRACT

This paper presents a class of semi-linear numerical differentiation formulas which is especially designed for functions with steep gradients. Basing on these, we construct a semi-linear second order difference scheme for solving the two-point singular perturbation problem

$$-\varepsilon u'' + p(x)\, u' + q(x)\, u = f(x), \qquad u(0) = u(1) = 0$$

The resulting discrete semi-linear system, $AU = d + Q(U)$, is solved by a simple iteration: $A\, U^{(k)} = d + Q(U^{(k-1)})$, $U^{(0)} = A^{-1}d$ which is convergent provided that the uniform mesh size h satisfies $h \leq \frac{2}{\|p\|_\infty} \varepsilon$. Moreover, with the same mesh size constraint it is shown that a second order semi-linear scheme has one more order of precision than its corresponding central difference scheme for small ε and a first order error estimate $\|u_s^h - u\|_\infty < C_\infty h$ is derived where the constant C_∞ is uniformly bounded for all $\varepsilon>0$. For the same accuracy, the present nonlinear scheme is much more efficient than the usual linear centered difference scheme for singular problems. For a linear and a semi-linear singular perturbation problems, numerical results agreeing with the above analysis are included.

ACKNOWLEDGEMENT

The author would like to thank Professor Martin H. Schultz for his helpful comments and Doctors Ding Lee and Tony Chan for their help in English to improve this manuscript.

1. Introduction

Numerical differential formulas play a very important role in constructing difference schemes of differential equations. Usual numerical differentiation formulas based on polynomial approximations are derived for smooth functions without large derivatives, it is possible for these formulas to lead to very poor results when the functions are not smooth. There are usually two ways to avoid this trouble: refine the mesh, or use higher order polynomial interpolation. Sometimes they are called h-version and p-version, respectively.

The approach presented in this paper is quite different. The main reason why the usual linear schemes lead to worse results for problems with large derivatives, especially those with singularity, is that the usual numerical differentiation formulas based on polynomial approximation is not accurate enough in this case, e.g. an asymptotic behavior near singularity is exponential . It seems hard to get high precision numerical differentiation formula near singularity if we restrict ourselves to use only piecewise polynomial approximations or other linear functional space. Hence, in this paper, we try to look for some new numerical differentiation formulas beyond linear functional space.

Thus, two questions arise. The first is how to find such kind of numerical differentiation formulas. The second is how to use them to construct difference scheme and how to solve the resulting discrete non-linear system efficiently. Besides, we expect that the resulting non-linear system should be 'not too far', in some sense, from being linear in order to simplify both the theoretical analysis and practical computation. Therefore, the numerical differentiation formulas and the resulting schemes, which we introduce below, are called semi-linear, or in some sense they are required to be 'weakly' non-linear.

The above main idea is based on the author's previous work ([8]-[10]). The purpose of this paper is to derive some of these semi-linear numerical differentiation formulas (section 2) and to construct the corresponding semi-linear difference schemes for a general elliptic singular perturbation problem (section 4): $-\varepsilon u''+pu'+qu=f$, $u(0)=u(1)=0$. The same problem is also used as a test problem in my another paper [11] which extended the Ritz-Galerkin method from linear subspaces to subsets. A similar study in two dimensional case will be discussed in [12]. The main idea of our scheme is that we use different difference schemes in different subdomains according to the size of the discrete first divided difference; while it is large the scheme will choose a semi-linear scheme automatically, otherwise the scheme is the same as the usual linear scheme, for instance, the centered difference scheme. It is shown that the semi-linear scheme has one more order of precision than the conventional centered difference scheme for the singular perturbation problem if $h \leq 2\varepsilon/\|p\|_{\infty}$, where h is a uniform mesh size in the 'singular' subdomain. In the larger 'regular' subdomain the mesh size can be used as larger as we desire. Moreover, for the same accuracy the semi-linear scheme costs less CPU time than the linear scheme, being a simple way to reduce the resulting semi-linear system to an iteration with the corresponding linear system. The numerical tests presented in section 5 match the above analysis very well.

For convenience, an analysis is given in section 3 for a simplified model problem: $-\varepsilon u''+u'=0$, $u(0)=0, u(1)=1$, which was discussed by many authors, e.g. [1]-[3].

2. 'Semi-linear' numerical differentiation formulas

Let u(x) be a function defined in (a,b) with a large first derivative. Without loss of generality, let us consider u(x) is monotonic in the interval, and suppose that u = Fx is one-to-one mapping such that $x = F^{-1}u$.

Denote $x_{-1} = a < x_0 < x_1 = b$, and let G be defined as an indefinite integral of F, i.e.

$$G(x) = \int F dx, \tag{2.1}$$

By the Mean value theorem, there exist two points z_{-1} and z_1: $x_{-1} < z_{-1} < x_0 < z_1 < x_1$, such that

$$[F^{-1}u_0, F^{-1}u_{-1}]\ G = u(z_{-1}),\ [F^{-1}u_1, F^{-1}u_0]\ G = u(z_1), \tag{2.2}$$

where [x1, x2]Y is a notation of divided difference. Now we look for an approximate formula for the first derivative at the node $x=x_0$, based on the formulas (2.2), as follows:

$$u'(x_0) \sim \frac{2}{x_1-x_{-1}} ([F^{-1}u_1, F^{-1}u_0]\ G - [F^{-1}u_0, F^{-1}u_{-1}]\ G). \tag{2.3}$$

For the simplest case, take F as the identity mapping, and $G(x) = x^2/2$, then

$$u'(x_0) = \frac{u(x_1)-u(x_{-1})}{x_1 - x_{-1}} \tag{2.4}$$

Formula (2.3) is the same as the usual central difference formula based on the quadratic interpolation. In this case, it is obvious that the above numerical differential formula (2.3) is an extension of the usual central difference formula.

Now we assume F to be an admissible one-to-one mapping such that G can be obtained from (2.1) directly. For any such F, (2.3) defines a numerical formula for the first derivative at the node $x = x_0$.

As an example, let $F^{-1} f = f^r$, where r is a real parameter. For $u(x) > 0$ in (x_{-1}, x_1), from (2.3) and (2.1), the following approximation formula is obtained

$$u'(x_0) = \frac{2r}{(1+r)(x_1-x_{-1})}\left\{\frac{u_1^{1+r} - u_0^{1+r}}{u_1^{r} - u_0^{r}} - \frac{u_0^{1+r} - u_{-1}^{1+r}}{u_0^{r} - u_{-1}^{r}}\right\} \qquad (2.5)$$

where $u_0 = u(x_0)$, $u_{-1} = u(x_{-1})$, $u_1 = u(x_1)$.

When $r = \frac{1}{2}$, $x_0 = \frac{1}{2}(x_1+x_{-1})$, the corresponding formula to (2.5) is

$$u'(x_0) = \frac{1}{3h}(u_1^{1/2} - u_{-1}^{1/2})(u_1^{1/2} + u_0^{1/2} + u_{-1}^{1/2}). \qquad (2.6)$$

In general, we have the following error estimate:

<u>Theorem 2.1</u>: Let u, F ε $C^{k+1}(x_{-1}, x_1)$, where k=3 or 4, $F^{-1}u$ is a one-to-one mapping, $h = x_0-x_{-1} = x_1-x_0$, then, the remainder of the numerical differentiation formula (2.3) equals to

$$\frac{1}{h}\left([F^{-1}u_1, F^{-1}u_0]\, G - [F^{-1}u_0, F^{-1}u_{-1}]\, G\right) = u'(x_0) +$$

$$\frac{h^2}{12}\frac{d}{dx}\left\{2u'' + u'^2 \frac{d^2F^{-1}u}{du^2}\left(\frac{dF^{-1}u}{du}\right)^{-1}\right\}\Big|_{x=x_0} + O(h^k). \qquad (2.7)$$

<u>Proof</u>:

Applying the Taylor expansion for G(y) upon z, one obtains

$$G(y) - G(z) = (y-z)G'(z) + (y-z)^2 G''(z)/2 + (y-z)^3 G^{(3)}(z)/3!$$

$$+ (y-z)^4 G^{(4)}(z)/4! + O((y-z)^5)$$

Hence

$$W(y_1, y_0, y_{-1}) = \frac{G(y_1) - G(y_0)}{y_1 - y_0} - \frac{G(y_0) - G(y_{-1})}{y_0 - y_{-1}}$$

$$= \frac{y_1 - y_{-1}}{2} G''(y_0) + \frac{(y_1-y_0)^3 - (y_0-y_{-1})^3}{4!} G^{(4)}(y_0) + O((y_1-y_{-1})^5)$$

By means of rules for finding derivative function in implicit case, it is easyly seen that

$$W(F^{-1}u_1,\ F^{-1}u_0,\ F^{-1}u_{-1})$$

$$= hu'(x_0) + \frac{h^3}{12}\frac{d}{dx}\{2u'' + u'^2 \frac{d^2F^{-1}u}{du^2}(\frac{dF^{-1}u}{du})^{-1}\}|_{x=x_0}$$

$$+ O(h^{k+1}), \quad \text{for } k=3,4. \tag{2.8}$$

Q.E.D.

Corollary 1. If the mapping F satisfies the relation

$$\frac{d}{dx}\{2u'' + u'^2 \frac{d^2F^{-1}u}{du^2}(\frac{dF^{-1}u}{du})^{-1}\}|_{x=x_0} = 0 \tag{2.9}$$

then, the formula (2.3) has an error of third or fourth order for $k = 3$ or 4, respectively.

Corollary 2. If $k = 2$, the error estimate is of the second order, and the second term in the right side of (2.7) is evaluated at a point ξ ε (x_{-1}, x_1).

The dominant term of the truncation error of formula (2.5) is

$$\frac{h^2}{12}\frac{d}{dx}\{2u'' + (r-1)\frac{u'^2}{u}\}|_{x=x_0}$$

This leads to a fourth order numerical differentiation formula if r is so chosen that the above term vanishes. For instance, r can be chosen as

$$r = 1 - 2\frac{uu''}{u'^2}|_{x=x_0}, \quad \text{or} \tag{2.10}$$

$$r = -1 + 2\left(\frac{u_1}{u'_1} - \frac{u_{-1}}{u'_{-1}} \right) \tag{2.11}$$

Both (2.10) and (2.11) are implicit and they can not be used directly for numerical differentiation. However, they are useful in constructing a high order difference scheme for differential equations , even for partial differential equations.

The limiting case of (2.5) in which r tends zero is very interesting for solving singular perturbation problems. The corresponding formula with remainder term becomes

$$u'(x_0) = \frac{1}{h} \left\{ \frac{u(x_1)-u(x_0)}{\mathrm{Log}(u(x_1)/u(x_0))} - \frac{u(x_0)-u(x_{-1})}{\mathrm{Log}(u(x_0)/u(x_{-1}))} \right\}$$
$$- \frac{h^2}{12} \frac{d}{dx}\{2u'' - \frac{u'^2}{u}\}\Big|_{x=x_0} + O(h^4) \tag{2.12}$$

It follows from Theorem 2.1 that the formula (2.5) is fourth order if the function u(x) satisfies the non-linear second order differential equation

$$2uu'' - (1-r)u'^2 = bu, \quad \text{where} \quad b = \text{constant}. \tag{2.13}$$

One particular solution of (2.13) is the 'parabolic' function

$$u(x) = (cx + d)^{2/(1-r)} \tag{2.14}$$

where c, d are constants. This means that the 'logarithmic type' numerical differentiation formula (2.12), specifically designed for solutions with large derivative, is also good for smooth functions, such as parabolic function.

Similarly, we can derive numerical differentiation formulas for higher derivatives. For instance, there is an analogue of Theorem 2.1 for second derivatives.

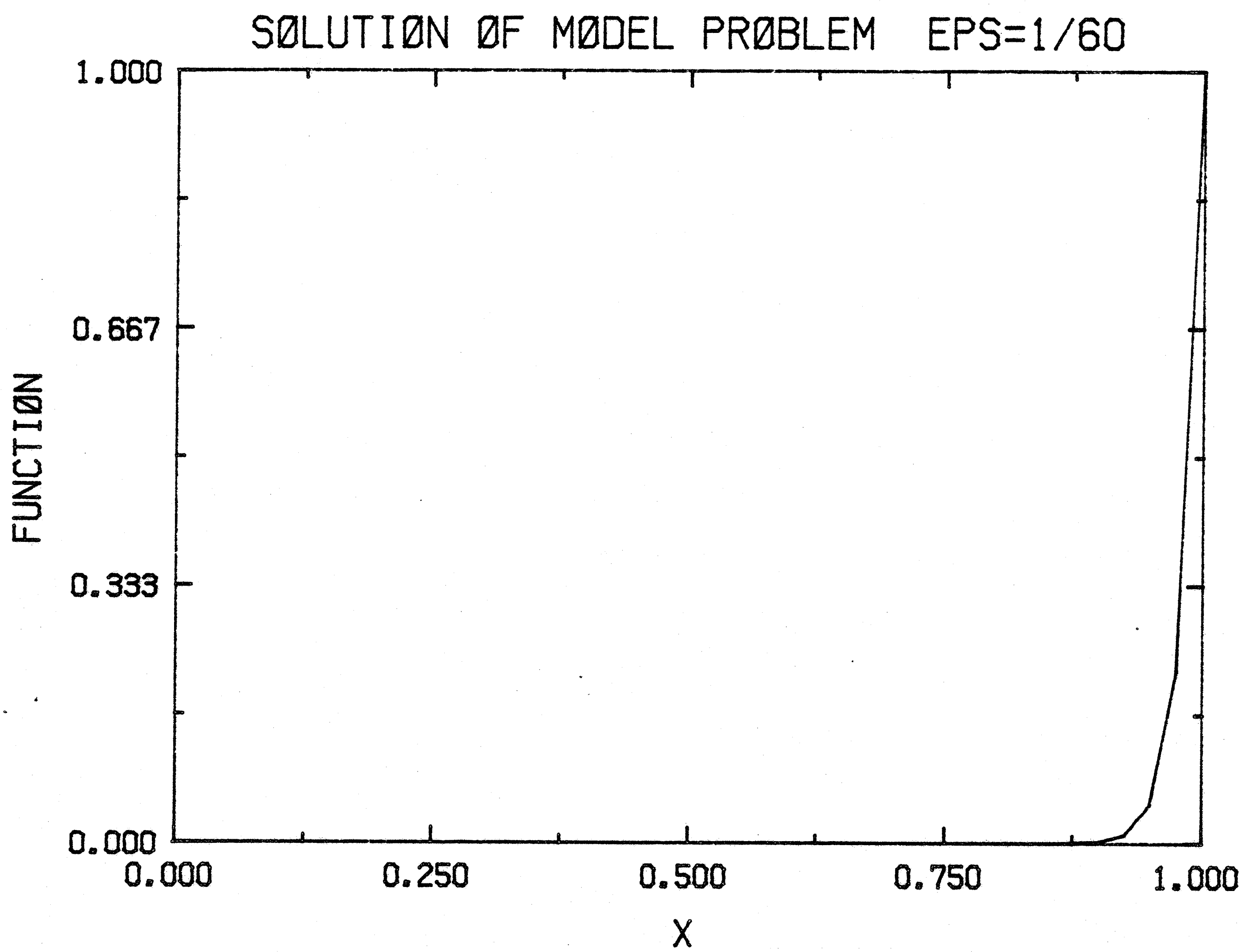
<u>Theorem 2.2</u>: If $u, F \varepsilon C^4(x_{-1}, x_1)$, $h = x_0 - x_{-1} = x_1 - x_0$, and $F^{-1}u$ is one-to-one mapping in the interval, then

$$u''(x_0) = \frac{2}{h^2}\{[F^{-1}u_1, F^{-1}u_0]\, G + [F^{-1}u_0, F^{-1}u_{-1}]\, G - 2u_0\}$$
$$- \frac{u'^{\,2}_0}{3} \frac{d^2F^{-1}u}{du^2}\left(\frac{dF^{-1}u}{du}\right)^{-1}\Bigg|_{x=x_0} + O(h^2) \qquad (2.15)$$

where G is defined by (2.1) and u'_0 is given by (2.3).

The proof of Theorem 2.4 follws the proof of Theorem 2.1, using the Taylor expansion and differentiation of implicit functions. (2.15) is an extension of the second order central difference scheme, and the latter is only a particular case of F = I.

Figure 1


SØLUTIØN ØF MØDEL PRØBLEM EPS=1/60
1.000
0.667
0.333
0.000
FUNCTIØN
0.000
0.250
0.500
0.750
1.000
X

3. Model problem analysis

In order to give a detailed analysis, in this section our attention is devoted to the following model problem which is similar to those discussed by various authors cf. Christie, Mitchell (1978), Zienkiewicz and others (1978), Barrett, Morton (1980). Consider

$$Lu = -\varepsilon u'' + u' = 0 \quad \text{in } (0,1), \quad u(0) = 0,\ u(1) = 1. \tag{3.1}$$

with solution (Figure 1)

$$u(x) = \frac{e^{x/\varepsilon} - 1}{e^{1/\varepsilon} - 1}. \tag{3.2}$$

Let $x_j = jh$ ($j = 0,1,\dots,N$; $h = 1/N$), using the conventional central difference for the first and second derivatives in (3.1) leads to the following difference equation

$$L_h U^h_j = -(a+\tfrac{1}{2})U^h_{j-1} + 2aU^h_j + (\tfrac{1}{2} - a)U^h_{j+1} = 0, \quad 1 \le j \le N-1. \tag{3.3}$$

with $U^h_0 = 0$ and $U^h_N = 1$, where $a \equiv \frac{\varepsilon}{h}$.

The exact solution of the difference equation (3.3) is given by

$$U^h_j = \frac{p(h)^j - 1}{p(h)^N - 1}, \text{ where } p \equiv \frac{a+1/2}{a-1/2}. \tag{3.4}$$

In order to preserve the increasing monotonic property of the solution (3.2), it is reasonable to demand

$$a > \frac{1}{2}, \qquad \text{i.e.} \quad h < 2\varepsilon. \tag{3.5}$$

Let $E^h_j = u(x_j) - U^h_j$, $E^h = \operatorname{Max} E^h_j$. From (3.2), (3.4) and (3.5), for small ε, we have an asymptotic estimate for small h,

$$E^h_j = \frac{xe^{x/\varepsilon} - e^{1/\varepsilon} + e^{(1+x)/\varepsilon}(1-x)}{12\varepsilon\, e^{2/\varepsilon}\, a^2} \Big|_{x=x_j} \tag{3.6}$$

with $E^h_0 = E^h_N = 0$, and

$$0 < E^h_j < E^h_{N-1} = E^h \sim \frac{1}{12}a^{-2} = \frac{h^2}{12\varepsilon^2}. \tag{3.7}$$

Thus, the error will be of zero order if h is the same order as ε. Hence, in this case the difference scheme can't converge for small ε, unless the mesh size h is excessively small. Meanwhile, (3.7) shows that the maximum error always occurs at the last interior mesh point. The difficulty comes from the fact that the scheme (3.3) yields a bad approximate solution near the layer boundary at the righthand end point. As a matter of fact, substituting the exact solution (3.2) of (3.1) into (3.3) yields

$$\begin{aligned} Tr\, u_j &\equiv -\frac{\varepsilon}{h}(u_{j+1}-2u_j+u_{j-1}) + \frac{1}{2}(u_{j+1}-u_{j-1}) \\ &= h\{ -\varepsilon[u''_j + \frac{h^2}{12}u^{(4)}_j + O(h^4)] + u'_j + \frac{h^2}{6}u^{(3)} + O(h^4)\} \\ &= \frac{h^3}{12}\{-\varepsilon u^{(4)}_j + 2u^{(3)}_j\} + O(h^5) = \frac{h^3}{12}u^{(3)}_j + O(h^5), \end{aligned} \tag{3.8}$$

where the equation (3.1) is used and

$$h^3u^{(3)}(x) = \frac{e^{x/\varepsilon}}{(e^{1/\varepsilon}-1)a^3}. \tag{3.9}$$

Thus, the local truncation error near x=1 is only of zero order if ε is the same order of h.

Therefore, it is why we need to look for a scheme with better approximation near the layer boundary. To construct such a scheme we try to use the results described in last section.

From (3.2), we observe that

$$u^{(3)}(x) = \frac{e^{x/\varepsilon}}{(e^{1/\varepsilon}-1)\varepsilon^3}, \qquad \frac{d}{dx}\frac{u'^2}{u} = u^{(3)}(x)\{ 1 - \frac{e^{x/\varepsilon}-1}{(e^{x/\varepsilon}-1)^2} \},$$

clearly both terms of the above formulas are very large with the same sign near the layer boundary, but their difference is very small for small ε if x is close to the right end, in fact

$$u^{(3)}(x) - \frac{d}{dx}\frac{u'^2}{u} \longrightarrow \varepsilon^{-3}e^{-1/\varepsilon} \qquad \text{as } x \longrightarrow 1.$$

Hence, if we could find a scheme of (3.1) such that the dominant coefficient $u^{(3)}$ in the truncation error formula (3.8) could be instead by $u^{(3)}(x) - \frac{d}{dx}\frac{u'^2}{u}$, it would greatly reduce the approximation error near the layer boundary. Therefore, it reminds us to adopt the 'logarithmic type' numerical differential formula (2.12) near layer boundary.

Thus, the difference scheme (3.3) is changed now as follows:

$$L_hU^h{}_j = -(a+\tfrac{1}{2})U^h{}_{j-1} + 2aU^h{}_j + (\tfrac{1}{2}-a)U^h{}_{j-1}, \text{ for } 2j < N+1$$

$$L_hU^h{}_j = -(a+\tfrac{1}{2})U^h{}_{j-1} + 2aU^h{}_j + (\tfrac{1}{2}-a)U^h{}_{j+1} - Q(U^h{}_{j-1}, U^h{}_j, U^h{}_{j+1}),$$

$$\text{for } N+1 \leq 2j \leq 2N \tag{3.10}$$

where

$$Q(U_{j-1},U_j,U_{j+1}) = \frac{U_{j+1}-U_{j-1}}{2} - \left\{ \frac{U(x_{j+1})-U(x_j)}{\mathrm{Log}(U(x_{j+1})/U(x_j))} - \frac{U(x_j)-U(x_{j-1})}{\mathrm{Log}(U(x_j)/U(x_{j-1}))} \right\} \tag{3.11}$$

with $U^h{}_0 = 0$ and $U^h{}_N = 1$, $a = \frac{\varepsilon}{h}$.

The local truncation error of the scheme becomes

$$L_h u = \begin{cases} \dfrac{h^3}{12} u^{(3)}{}_j & \text{if } \quad 2j < N+1 \end{cases} \tag{3.12}$$

$$\frac{h^3}{12}\{u^{(3)}-\frac{d}{dx}\frac{u'^2}{u}\}|_j \qquad \text{if} \quad N+1\le 2j\le 2N$$

or

$$12h^{-3}Tr^h_j = \begin{cases} \dfrac{e^{x/\varepsilon}}{(e^{1/\varepsilon}-1)\varepsilon^3} & \text{if} \quad 2j < N+1 \\[2ex] \dfrac{e^{x/\varepsilon}}{(e^{x/\varepsilon}-1)^2\varepsilon^3} & \text{if} \quad N+1\le 2j\le 2N \end{cases} \qquad (3.13)$$

where x = jh.

Denote the two right hand sides of (3.13) by R1(x) and R2(x), respectively. Let

$$D(x) = \frac{R1(x)}{R2(x)} = \frac{(e^{x/\varepsilon}-1)^2}{e^{1/\varepsilon}-1}.$$

Since

$$D'(x)>0,\ D(0)=0,\ D(1)=e^{1/\varepsilon}-1,\ D(\tfrac{1}{2}) = \frac{e^{1/(2\varepsilon)}-1}{e^{1/(2\varepsilon)}+1} < 1,$$

the solution of the equation D(x) = 1 is equal to

$$x^* = \frac{1}{2} + \varepsilon(e^{-1/(2\varepsilon)}-\frac{1}{2}e^{-1/\varepsilon}) + O(e^{-2/\varepsilon})$$

Then the ratio function R1(x)/R2(x) increases from zero to a very large number in the interval [0, 1], and the asymptotic solution of R1(x) = R2(x) is equal to $\frac{1}{2} + o(\varepsilon)$, for small ε. This is reasonable to choose $\frac{1}{2}$ as the separate point of the two different scheme in (3.10). and now the maximum local truncated error of (3.10) is given by

$$Tr^h \equiv \text{Max}\ Tr^h_j \sim \frac{h^3}{12}R2(1/2) = \frac{h^2}{12}\frac{e^{1/(2\varepsilon)}}{\varepsilon^3(e^{1/(2\varepsilon)}-1)^2} \sim \frac{h^3}{12}e^{-1/(2\varepsilon)}\varepsilon^{-3}$$

It is worth to note that there is a striking contrast between the truncation errors for the revised scheme (3.10) and the original linear scheme (3.3). In the linear scheme (3.3) the maximum truncation error has no power of the mesh size h. But the truncation error of the revised scheme (3.10) has an h^2 term with a coefficient which tends to 0 as ε does. Naturally, we expect the revised scheme (3.10), which we will call a semi-linear scheme based on the numerical differentiation formula in section 2, to give much better numerical results for the Model equation (3.1) than (3.3) does.

However, the first important question arisen is how to solve the semi-linear system (3.10). Denote the resulting system by

$$A\,U = d + Q(\,U\,) \tag{3.15}$$

where A and d coincide with the corresponding linear scheme, and the non-linear term Q(U) arises from using the semi-linear scheme for $x > \frac{1}{2}$.

Let

$$A_n = \begin{vmatrix} 2a & -(a-1/2) & & & \\ -(a+1/2) & 2a & -(a-1/2) & & \\ & \cdots & \cdots & \cdots & \\ & & \cdots & \cdots & \cdots \\ & & -(a+1/2) & 2a & -(a-1/2) \\ & & & -(a+1/2) & 2a \end{vmatrix}$$

$D_n = \mathrm{Det}(A_n)$, $\beta_n = \dfrac{D_{n-1}}{D_n}$, then

$$D_n = 2aD_{n-1} - (a^2-\tfrac{1}{4})D_{n-2},\quad D_0=1,\ D_1 = 2a,$$

$$\beta_n = \frac{1}{2a-(a^2-1/4)\beta_{n-1}},\quad \beta_{-1} \equiv 0.$$

When $a \geq \frac{1}{2}$, $\beta \equiv \lim \beta_n = \dfrac{1}{a+1/2}$, and

$$\frac{1}{2a} \leq \beta_{n+1} - \beta_n \leq \frac{a-1/2}{a+1/2}\,(\beta_n - \beta_{n-1}),$$

$$\frac{1}{2a} \leq \beta_n \leq \beta = \lim \beta_n = \frac{1}{a+1/2}. \tag{3.17}$$

The inverse matrix A^{-1} can be easily found:

$$A^{-1} = (\alpha^{-1}_{i,j}),$$

$$\text{where } \alpha^{-1}_{i,j} = \begin{cases} (a+1/2)^{i-j} D_{j-1} D_{n-i}/D_n & \text{if } i>j \\ (a-1/2)^{j-i} D_{i-1} D_{n-j}/D_n & \text{if } i<j \end{cases} \tag{3.18}$$

We need to introduce the following Lemmas:

<u>Lemma 3.1</u>: For $a \geq \frac{1}{2}$, A^{-1} is a positive matrix given by (3.18). Moverover

$$\begin{aligned} \alpha^{-1}_{i,j} &\geq \alpha^{-1}_{i,j-1} && \text{if } i \geq j \\ &\leq \alpha^{-1}_{i,j-1} && \text{if } i < j \end{aligned} \tag{3.19}$$

$$\|A^{-1}\|_\infty \leq (N-1)\beta_{N-1} < \frac{1}{h(a+1/2)}.$$

Suppose that a,b, and c are positive, let

$$r(a,b) = \frac{a+b}{2} - \frac{b-a}{\mathrm{Log}(b/a)},$$

$$Q(a,b,c) = r(b,c) - r(a,b) \tag{3.20}$$

then

$$\begin{aligned} Q_a &\equiv \frac{\partial Q}{\partial a} = -\frac{1}{2} - \frac{1}{\mathrm{Log}(b/a)} + \frac{(b-a)/b}{\{\mathrm{Log}(b/a)\}^2}, \\ Q_c &\equiv \frac{\partial Q}{\partial c} = \frac{1}{2} - \frac{1}{\mathrm{Log}(c/b)} + \frac{(c-b)/c}{\{\mathrm{Log}(c/b)\}^2}, \\ Q_b &\equiv \frac{\partial Q}{\partial b} = \frac{1}{\mathrm{Log}(b/a)} - \frac{(b-a)/a}{\{\mathrm{Log}(b/a)\}^2} + \frac{1}{\mathrm{Log}(c/b)} - \frac{(c-b)/b}{\{\mathrm{Log}(c/b)\}^2}. \end{aligned} \tag{3.21}$$

The term $\frac{b-a}{\mathrm{Log}(b/a)}$ is a Generalized Mean defined by Jiachang Sun [9] between the arithmetic Means and Geometric Means of a and b:

<u>Lemma 3.2</u>: If $a,b > 0$, then

$$(ab)^{1/2} \leq \frac{a-b}{\mathrm{Log}(a/b)} < \frac{a+b}{2} \tag{3.22}$$

where the equality holds if and only if a=b.

In fact, integrating t from 0 to 1 on the both sides of the following inequalities: $a^{1-t}b^{t} \leq (1+t)a + tb$, it leads to the righthand of (3.22), and the lefthand can be obtained by

$$\int_0^1 (\tfrac{a}{b})^{t-1/2}\, dt = \int_0^{1/2} \{(\tfrac{a}{b})^{t-1/2} + (\tfrac{b}{a})^{t-1/2}\}\, dt \geq 1.$$

<u>Corollary</u>. For a,b > 0

$$0 \leq r(a,b) \equiv \frac{a+b}{2} - \frac{a-b}{\mathrm{Log}(a/b)} \leq \frac{1}{2}(b^{1/2}-a^{1/2})^2, \qquad (3.23)$$

There exist a series of inequalities for the function Q and its derivatives, based on (3.22).

<u>Lemma</u> 3.3: If a,b, and c are positive, then

(i) $-\frac{1}{2}(b^{1/2}-a^{1/2})^2 \leq Q(a,b,c) \leq \frac{1}{2}(c^{1/2}-b^{1/2})^2.$

(ii) If $c\geq b\geq a>0$ and $b^2\leq ac$, then $Q(a,b,c) \geq 0$ with '=' iff c = a.

(iii) $0 \leq \frac{1}{2}\{(\frac{b}{a})^{1/4} - 1\} \leq \frac{\partial Q}{\partial a} \leq \frac{1}{2}(\frac{a}{b} - 1)$, if $b\geq a>0$

$-\frac{1}{2}\{1-(\frac{b}{a})^{1/2}\} \leq \frac{\partial Q}{\partial a} \leq -\frac{1}{4}(1-(\frac{b}{a})^{1/2}))\leq 0$, if $a\geq b>0$.

$0 \leq \frac{1}{4}\{1-(\frac{b}{c})^{1/2}\} \leq \frac{\partial Q}{\partial c} \leq \frac{1}{2}(1-\frac{c}{b})$, if $c\geq b>0$

$-\frac{1}{4}(\frac{b}{c}-1) \leq \frac{\partial Q}{\partial c} \leq -\frac{1}{2}((\frac{b}{a})^{1/4}-1)\leq 0$, if $b\geq c>0$.

Now we look for an estimate of $\|A^{-1}J(Q(U))\|$, where J(Q) is the Jacobi matrix of Q. First, we prove a very interesting relationship for the scheme (3.10)

$$\{J(Q(u))\, u\}_j = \{Q(u)\}_j, \quad \text{if } j < N-1. \qquad (3.24)$$

Usually, (3.24) is only true for Q(u) = Au, where A is a matrix, it shows again that the semi-linear term Q(u) is really not far from linear. In fact, from (3.11) and (3.21), $\{J(Q(u))\, u\}_j = 0$ if $j < n$, where $n = [N/2]$. And for $n < j < N-1$

$$\{J(Q(u))\, u\}_j = \frac{\partial Q}{\partial u_{j-1}} u_{j-1} + \frac{\partial Q}{\partial u_j} u_j + \frac{\partial Q}{\partial u_{j+1}} u_{j+1}$$
$$= \frac{u_{j+1}-u_{j-1}}{2} - \{ \frac{u(x_{j+1})-u(x_j)}{\mathrm{Log}(u(x_{j+1})/u(x_j))} - \frac{u(x_j)-u(x_{j-1})}{\mathrm{Log}(u(x_j)/u(x_{j-1}))} \} = Q_j(u)$$

Hence

$$J(Q(u))\, u = \{0,\ldots,0,Q_n,\ldots,Q_{N-2},Q^*_{N-1}\},$$

where

$$Q_j = r_{j+1/2} - r_{j-1/2},$$
$$r_{j-1/2} \equiv \frac{u_j+u_{j-1}}{2} - \frac{u_j - u_{j-1}}{\mathrm{Log}(u_j/u_{j-1})},$$
$$Q^*_{N-1} = Q_{N-1} - \frac{\partial Q_{N-1}}{\partial u_N} u_N. \tag{3.25}$$

Using the above results, we derive an upper bound of a norm of the matrix $A^{-1}J(Q(u))$. A straightforward computation yields

$$\sigma_i \equiv \{A^{-1}J(Q(u))u\}_i = = \alpha^{-1}_{i,N-1} R_{N,N-1} -$$
$$\sum_{j=n+1}^{N-1} \{ \alpha^{-1}_{i,j} - \alpha^{-1}_{i,j-1} \} r_{j-1/2} - \alpha^{-1}_{i,n} r_{n-1/2} \tag{3.26}$$

for $n \leq i \leq N-1$, where

$$R_{N,N-1} \equiv r_{N-1/2} - \frac{\partial r_{N-1/2}}{\partial u_N}.$$

Or

$$\sigma_i = \alpha^{-1}_{i,N-1} R_{N,N-1} + \sum_{j=i+1}^{N-1} \{ \alpha^{-1}_{i,j-1} - \alpha^{-1}_{i,j} \} r_{j-1/2} -$$
$$\sum_{j=n+1}^{i} \{ \alpha^{-1}_{i,j} - \alpha^{-1}_{i,j-1} \} r_{j-1/2} - \alpha^{-1}_{i,n} r_{n-1/2}.$$

Suppose $u_j > 0$, $u_N \equiv 1$, using (3.23) and (3.19) leads to

$$\alpha^{-1}_{i,N-1} R_{N,N-1} - r_M \alpha^{-1}_{i,i} \leq$$

$$\sigma_i < \alpha^{-1}_{i,N-1} R_{N,N-1} + r_M(\alpha^{-1}_{i,i} - \alpha^{-1}_{i,N-1}). \qquad (3.27)$$

where $r_M \equiv \text{Max}\ r_{j-1/2}$,

$$R_{N,N-1} = \frac{u_{N-1}}{2} - \frac{u_{N-1}}{\text{Log}u_{N-1}} - \frac{1-u_{N-1}}{(\text{Log}u_{N-1})^2},$$

and

$$-\frac{1}{4} \leq -\frac{1}{4}(1-u_{N-1}) \leq R_{N,N-1} \leq -\frac{u_{N-1}^{3/4}}{2}(1 - u_{N-1}^{1/4}) < 0.$$

Hence, $(a + \frac{1}{2})^{-1} \geq \alpha^{-1}_{i,i} \geq \alpha^{-1}_{i,N-1}$ because of (3.18), and the inequality (3.27) implies

$$-(r_M + \frac{1}{4}) \leq (a + \frac{1}{2})\ \sigma_i < r_M.$$

From (3.23), $0 \leq r_M \leq \frac{1}{2}$ if $0 < u_j \leq 1$. Finally, we obtain an upper bound

$$\|A^{-1}J(Q(u))\| = \sup_{\|u\|=1} \|A^{-1}J(Q(u))u\| \leq \frac{3}{4(a+1/2)}. \qquad (3.28)$$

Therefore, the following two Theorems are resulted:

<u>Theorem 3.4</u>: The mapping

$$P(u) = A^{-1}\ Q(u) \qquad (3.29)$$

is contractive if

$$a = \frac{\varepsilon}{h} \geq \frac{1}{2} \qquad (3.30)$$

where the matrix A and the vector Q(u) are defined in (3.15). In another word, P(u) has unique fixed point in this case.

<u>Theorem 3.5</u>: When $h \leq 2\varepsilon$, the semilinear scheme (3.10) has unique

solution, and it can be solved by the following 'simple' iteration

$$A\,U^{(0)} = d \tag{3.31}$$

$$A\,U^{(k)} = d + Q(\,U^{(k-1)}\,), \qquad k > 1.$$

Furthermore, from (3.4), U^0_j is a monotocally increasing sequence of j and $Q(U^0)$ is nonnegative. In fact

$$Q(p^{j+1}-1,p^j-1,p^{j-1}-1) = Q(p^{j+1},p^j,p^{j-1}) + Q^*$$

$$\text{where } Q^* = (p^{j+1}-p^j)\{\frac{1}{\text{Logp}} - \frac{1}{\text{Log}[(p^{j+1}-1)/(p^j-1)]}\} -$$

$$(p^j-p^{j-1})\{\frac{1}{\text{Logp}} - \frac{1}{\text{Log}[(p^j-1)/(p^{j-1}-1)]}\}.$$

From Lemma 3.3 (ii), $Q(p^{j+1},p^j,p^{j-1}) > 0$, and since $p >> 1$

$$\text{Log}[(p^{j+1}-1)/(p^j-1)] \sim \text{Logp} - p^{-(2j+1)},$$

$$Q^* \sim \frac{p^j}{\text{Logp}}\;\{(1-p^{-1})p^{-(j-1)} - (p-1)p^{-(j+1)}\} = \frac{(p^{1/2}-p^{-1/2})^2}{\text{Logp}} > 0.$$

By induction, it can be proved that, for this Model problem (3.1), U^k_j in the iteration (3.31) preserve the monotonically increasing with j for each k. Hence, when k tends ∞, the monotonically increasing property still remain true, i.e.,

$$U_j \le U_{j+1}, \quad j = 0,1,2,...$$

Now we consider the convergence in another meaning: the convegence of the solution U^h of the nonlinear difference equations (3.10) to the exact solution u of the differential equation (3.1). In matrix form (3.1) with the local truncation error formula (3.12) or (3.13) can be expressed as

$$A\,u = d + Q(u) + Tr(u). \tag{3.32}$$

Subtracting (3.15) from (3.32) yields the error equation:

$$A(u - U^h) = Q(u) - Q(U^h) + Tr(u). \tag{3.33}$$

When $h < 2\varepsilon$, A^{-1} exists, the above formula is equivalent to

$$u - U^h = A^{-1}(Q(u) - Q(U^h)) + A^{-1}Tr(u). \tag{3.34}$$

According to the Lemma 3.4, for $h < 2\varepsilon$, $A^{-1}Q(u)$ is a contraction mapping in the maximum norm, hence, from (3.28)

$$\|A^{-1}(Q(u) - Q(U^h))\|_\infty \leq \frac{3}{4(a+1/2)}\|u - U^h\|_\infty.$$

Applying Lemma 3.1 and (3.13), we obtained

$$\|u - U^h\|_\infty < \{ 1 - \frac{3}{4(a+1/2)} \}^{-1} \|A^{-1}\|_\infty\|Tr(u)\|_\infty < C\, h^2$$

where the constant

$$C = \frac{1}{3}e^{-1/(2\varepsilon)}\ \varepsilon^{-3} \leq \frac{1}{3}\{\frac{6}{e}\}^3 < 3.6\ . \tag{3.35}$$

Therefore, we have proved the following main result of this section :

<u>Theorem 3.6</u>: The solution of the semi-linear scheme (3.10) converges to the exact solution of the singularly perturbation boundary value problem (3.1). The rate of convergence in the maximum norm is of second order. Moreover,

$$\| U^h - u \|_\infty < Ch^2, \quad \text{if } h \leq 2\varepsilon. \tag{3.36}$$

where the coefficient C defined in (3.35) is uniformly bounded for all ε and C tends to zero as ε does.

Note that the restricted mesh condition $h < 2\varepsilon$, caused by introducing the semi-linear scheme, is only used in the steeper gradient interval $x > 1/2$. Hence, it is possible to restrict the mesh condition only in the interval . As a matter of fact, in block matrix form, the scheme (3.10) can be written as

$$\begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} \begin{vmatrix} V \\ U \end{vmatrix} = \begin{vmatrix} 0 \\ d \end{vmatrix} + \begin{vmatrix} 0 \\ Q(U,V) \end{vmatrix}$$

Let h_1 and h be two different uniform mesh sizes used in the intervals $x < 1/2$ and $x \geq 1/2$, respectively, and

$$a = \frac{\varepsilon}{h}, \quad a_1 = \frac{\varepsilon}{h_1}.$$

Applying Gaussian elimination of matrix form, the above system can be reduced to

$$\tilde{A}_{22}U = d + Q(U, -A_{11}^{-1}A_{12}U) \tag{3.37}$$

where

$$\tilde{A}_{22} = (\alpha^*_{ij}), \quad \alpha^*_{11} = a - \frac{1}{2} + \frac{a+1/2}{a_1+1/2},$$

$$\alpha^*_{ij} = \alpha_{ij} \quad \text{if } (i,j) \neq (1,1).$$

Hence, when $a \geq a_1$, i.e., $h \leq h_1$, the Lemma 3.1 still holds, the above conclusion can be improved further by the next assertion.

<u>Theorem 3.7</u>: If

$$h \leq \min(h_1, 2\varepsilon) \tag{3.38}$$

then the iterative procedure (3.31) converges for the system (3.37) and the error estimate (3.36) is still valid.

4. Linear second order two–point boundary layer problems

Consider a general linear second order singular perturbation problem

$$Lu \equiv -\varepsilon\, u'' + p(x)\, u' + q(x)\, u = f(x), \quad \text{in } [0,1]$$

$$u(0) = u(1) = 0 \tag{4.1}$$

where ε is a small positive parameter and $p(x), q(x)$ and $f(x)$ are sufficiently smooth that their second derivatives are uniformly bounded for all x in [0,1] and for all $\varepsilon > 0$, besides, $p(x) \geq p^* > 0$, $q(x) \geq 0$ and $q(x) - p'(x)/2 \geq \delta > 0$ on [0,1]. Let

$$L_h U^h_j = -(\alpha + p_j/2) U^h_{j-1} + (2\alpha + q_j h^2) U^h_j + (p_j/2 - \alpha) U^h_{j-1}. \tag{4.2}$$

The interval [0,1] is divided into two subintervals : $[0,1] = I_r + I_s$, where I_r is called a regular subinterval over which the first derivative of u(x) is bounded by a control number μ^h, and I_s defines a singular subinterval over which u'(x) may be very large.

Define $p_j \dfrac{U_{j+1} - U_{j-1}}{2h} \equiv \tau$. Similar to (ref[3.10]), in this case the semi-linear scheme becomes

$$L_h U^h = \begin{cases} hf_j^h & \text{if } |\tau| \leq \mu^h, \\ hf_j^h - g_j(U^h_{j-1}, U^h_j, U^h_{j+1}) & \text{if } |\tau| > \mu^h \end{cases} \tag{4.3}$$

where

$$g_j(u_{j+1}, u_j, u_{j-1}) = p_j(r(u_{j+1}, u_j; c) - r(u_j, u_{j-1}; c)),$$

$$r(a,b;c) \equiv r(a+c, b+c) = \frac{a+b}{2} - \frac{b - a}{\mathrm{Log}((c+b)/(c+a))}, \tag{4.4}$$

and c is a parameter to be chosen. The purpose of introducing c is two-fold. First, to make the scheme to be well defined; second, to lead to a better approximation.

Now the corresponding matrix A in (3.15) becomes

$$A_n = \begin{vmatrix} 2a+hq_1 & -(a-p_1/2) & & & \\ -(a+p_2/2) & 2a+hq_2 & -(a-p_1/2) & & \\ & \cdots & \cdots & \cdots & \\ & & \cdots & \cdots & \cdots \\ & & -(a+p_{N-2}/2) & 2a+hq_{N-2} & -(a-p_{N-2}/2) \\ & & & -(a+p_{N-1}/2) & 2a+hq_{N-1} \end{vmatrix} \quad (4.5)$$

Denote the determinant of the first j and the last N−i principal determinant by D_j and $D_{i,N-1}$, respectively,

$$\beta_n \equiv \frac{D_{n-1}}{D_n}, \quad \beta_{j,N-1} \equiv \frac{D_{j+1,N-1}}{D_{j,N-1}},$$

then

$$D_n = (2a + hq_n)D_{n-1} - (a - \frac{p_{n-1}}{2})(a + \frac{p_n}{2})D_{n-2},$$

$$D_0 = 1, \; D_1 = 2a + hq_1.$$

$$\beta_n = \frac{1}{2a+hq_n - (a-p_{n-1}/2)(a+p_n/2)\beta_{n-1}}.$$

$$D_{j,N-1} = (2a + hq_j)D_{j+1,N-1} - (a - \frac{p_j}{2})(a + \frac{p_{j+1}}{2})D_{j+2,N-1},$$

$$D_{N,N-1} = 1, \; D_{N-1,N-1} = 2a + hq_{N-1}. \quad (4.6)$$

$$\beta_{j,N-1} = \frac{1}{2a+hq_j - (a-p_j/2)(a+p_{j+1}/2)\beta_{j+1,N-1}}.$$

Lemma 4.1: Assume that

(i) $a = \frac{\varepsilon}{h} \geq \frac{1}{2}\|p\|_\infty$ (4.7)

(ii) $q_j \geq \frac{1}{2h}(p_{j+1} - p_{j-1})$, for all j, (4.8) then

$$\beta_n \leq \frac{1}{a+p_{n+1}/2}, \quad \text{for all } n < N-1.$$

$$\beta_{n,N-1} \leq \frac{1}{a+p_n/2}, \quad \text{for all } n < N-1. \qquad (4.9)$$

Remark: As a discrete form of an inequality $q(x) \geq p'(x)$, (4.8) is a sufficient condition of an elliptic form for the equation (4.1).

Proof:

The first inequality (4.9) is trivial for n=1. By induction, suppose (4.9) holds for n-1, then from (4.6) with the condition (4.8)

$$\beta_n \leq \{2a + \frac{1}{2}(p_{n+1} - p_{n-1}) - (a-p_{n-1}/2)\}^{-1} = \{a + p_{n+1}/2\}^{-1}.$$

The second part of (4.9) can be proved in the same manner.

Definition: f real n x n matrix $A = (\alpha_{i,j})$ with $\alpha_{i,j} \leq 0$ for all $i \neq j$ is called a G-matrix if $A^{-1} \geq 0$ and

$$\alpha^{-1}_{i,j} \geq \alpha^{-1}_{i,j-1} \quad \text{if } i \geq j$$

$$\leq \alpha^{-1}_{i,j-1} \quad \text{if } i < j \qquad (4.10)$$

Remark: A is a G-matrix means that A^{-1} is a good discrete Green's function in the machanical sense.

Theorem 4.2: The matrix A defined in (4.5) is a G-matrix if (4.7) and (4.8) hold.

Proof:

In fact, we have

$$\prod_{k=j}^{i-1} (a+\frac{p_{k+1}}{2})\, D_{i+1,N-1} \frac{D_{j-1}}{D_{N-1}} \quad \text{if } i > j$$

$$\alpha^{-1}_{i,j} = \begin{cases} D_{i+1,N-1} \dfrac{D_{j-1}}{D_{N-1}} & \text{if } i = j \\ \prod_{k=i}^{j-1} (a - \dfrac{p_k}{2}) D_{j+1,N-1} \dfrac{D_{i-1}}{D_{N-1}} & \text{if } i < j. \end{cases} \quad (4.11)$$

Hence, in the case of $i \geq j$

$$\frac{\alpha^{-1}_{i,j}}{\alpha^{-1}_{i,j-1}} = \frac{D_{j-1}}{(a+p_j/2)D_{j-2}} = \frac{1}{(a+p_j/2)\beta_{j-1}} \geq 1,$$

and in the case of $i < j$

$$\frac{\alpha^{-1}_{i,j-1}}{\alpha^{-1}_{i,j}} = \frac{D_{j,N-1}}{(a-p_j/2)D_{j+1,N-1}} > \frac{1}{(a+p_j/2)\beta_{j,N-1}} \geq 1. \quad \text{Q.E.D.}$$

<u>Definition</u>: In a real n-dimensional space R^n a mapping $Q(x)$ is called semi-linear, denoted by $Q \varepsilon SL^n$, if $Q \varepsilon C^1$ and there exists a constant K such that

$$\{J(Q(x))(x+K)\}_j = Q_j(x), \text{ for } 1 < j < n. \quad (4.12)$$

where $J(Q)$ is the Jacobi matrix of Q.

In particular, any linear mapping Ax is semi-linear for every square matrix A when K=0. Hence, the semi-linear mapping defined by the above definition is an extension of the linear transformation.

It is obvious that SL^n forms a linear space, i.e, if both $Q(x)$ and $P(x)$ are semi-linear, and a,b are constants, then $aP(x)+bQ(x)$ is also semi-linear. Furthermore, if a and b are constant vectors with $a=(a_j)$, $b=(b_j)$, then the mapping $(a_jP_j(x) + b_jQ_j(x))$ is also semi-linear.

As an example,

$$R(x) = (R_j(x)), \quad R_j(x) \equiv \frac{x_{j+1} - x_j}{\mathrm{Log}((c+x_{j+1})/(c+x_j))},$$

is semi-linear, because

$$\frac{\partial R_j}{\partial x_j} x_j + \frac{\partial R_j}{\partial x_{j+1}} x_{j+1} = R_j.$$

Moreover, setting a shift transformation $V_j = U^h_j + c$ $(1 \leq j \leq N-1)$, in matrix form the system (4.2) becomes

$$A\ V = d^* + Q(V)$$

where A is defined in (4.5), $Q_j = 0$ or $p_j(r(V_{j+1},V_j) - r(V_j - V_{j-1}))$, depending upon whether $j \varepsilon I_r$ or $j \varepsilon I_s$, respectively. Since R(x) is semi-linear, we have

$$\{J(Q(V))V\}_j = \{Q(V)\}_j, \quad \text{if} \quad j < N-1. \tag{4.13}$$

<u>Theorem</u> 4.<u>3</u>: For given $g_j(x)$ defined in (4.4) and $u_0 \equiv u_N \equiv 0$, the mapping $Q(x) = (g_j(x))$ is semi-linear.

Since

$$\sigma_i \equiv \{A^{-1}J(Q(V))V\}_i = = p_{N-1}\alpha^{-1}{}_{i,N-1}\ R_{N,N-1} - \sum_{j=n}^{N-1} \{\ p_j\alpha^{-1}{}_{i,j} - p_{j-1}\alpha^{-1}{}_{i,j-1}\ \}r_{j-1/2} - p_n\alpha^{-1}{}_{i,n}r_{n-1/2}$$

where

$$r_{j-1/2} \equiv r(V_j,V_{j-1}),\ R_{N,N-1} \equiv r_{N-1/2} + \frac{\partial r_{N-1/2}}{\partial u_N}.$$

it is easily seen that if the condition (3.30) is changed to (4.7) , the Theorem 3.4 will remain true when $p' \geq 0$. Therefore, we obtained an extension of Theorems 3.4 and 3.5:

<u>Theorem</u> 4.<u>4</u>: Suppose

$$p'(x) \geq 0, \text{ and } q_j \geq \frac{1}{2h}(p_{j+1} - p_{j-1}), \text{ for all } j, \tag{4.14}$$

then the mapping $P(V) \equiv A^{-1}Q(V)$ is contractive if (4.7) holds.

When $p'(x)$ is negative somewhere, the derivation of conditions for the mapping $P(V) \equiv A^{-1}Q(V)$ to be contractive is little more complicated. In this case, set

$$\bar{\sum} \equiv \bar{\sum}^{+} + \bar{\sum}^{-},$$

where the two terms on the righthand side denotes a positive parts and a non-positive part, respectively. Suppose $0<V_j\leq 1$, since $0\leq r_j\leq r_M$, from the Theorem 4.2

$$\sum_{j=n}^{i} \{ p_j\alpha^{-1}_{i,j} + p_{j-1}\alpha^{-1}_{i,j-1} \} r_{j-1/2} + p_n\alpha^{-1}_{i,n} r_{n-1/2}$$

$$\leq r_M(p_i\alpha^{-1}_{i,i} - p_n\alpha^{-1}_{i,n}) + p_n\alpha^{-1}_{i,n} r_{n-1/2} +$$

$$\sum_{j=n}^{i}{}^{+} \{p_{j-1}\alpha^{-1}_{i,j-1} - p_j\alpha^{-1}_{i,j}\}\{r_M - r_{j-1/2}\}$$

$$< r_M\alpha^{-1}_{i,i}\{ p_i + \sum_{j=n+1}^{i}{}^{+} (p_{j-1}-p_j) \}.$$

Hence

$$-\sigma_i < \frac{3}{4} + \frac{1}{2(a+p_{i+1}/2)} \sum_{j=n+1}^{i}{}^{+} (p_{j-1}-p_j).$$

In the same manner of the derivation in the last section, we obtain the following result.

<u>Theorem</u> <u>4.5</u>: The conclution in Theorem 4.4 still holds even if the condition $p'\geq 0$ is removed provided that the iequality (4.7) is changed to

$$a \geq \mathrm{Max}\{ \frac{\|p\|_\infty}{2},\ 2\sum_{j=n}^{N-1}{}^{+} (p_{j-1}-p_j) - \frac{1}{2}p_N\}. \tag{4.15}$$

In order to get an error estimatation, using (3.34), it is only need to find a bound for $\|A^{-1}Tr(u)\|_\infty$ when the mapping is contractive.

This reduces the error estimates to estimate the bound of the truncation error. Substituting the exact solution u in the scheme (4.3) yields

$$L_h u_j = hf_j + \frac{h^3}{12}\{-\varepsilon u^{(4)} + 2pu^{(3)}\}_\xi,$$

if $j \in I_r$, and

$$L_h u_j = hf_j - g_j(u_{j-1}, u_j, u_{j+1}) + Tr_j(u),$$

$$Tr_j = \frac{h^3}{12}\{-\varepsilon u^{(4)} + 2pu^{(3)} - p(\frac{u'^2}{c+u})'\}|_\eta, \qquad (4.16)$$

if $j \in I_s$, where $0<\xi,\eta<1$.

It has been noted that the exact solution has a factorization which consists of two parts : one is regular, another is singular (see Hon Honde[5], Lemma 1), hence

$$u(x) = \gamma\{W(x) + Z(x)\}$$

$$W(x) = e^{-p(1)(1-x)/\varepsilon} - x - (1-x)e^{-p(1)/\varepsilon} \qquad (4.17)$$

where $\gamma = \lim_{\varepsilon \to 0} \lim_{x \to 1} \varepsilon u'(x)/p(1)$ is a constant bounded uniformly for all $0<\varepsilon<1$, and

$$|Z(x)| \le C, \quad |Z'(x)| \le C, \quad |Z''(x)| \le C\{1+\frac{1}{\varepsilon} e^{-\beta(1-x)/\varepsilon}\} \qquad (4.18)$$

C is a constant independent of ε, and $0<\beta\le p^*$.

Set c in (4.4) equal to γ which can be also obtained in computing process. Because the singularity of the exact solution u(x) is only near x=1, the width of the boundary layer in which u(x) has large derivatives is less than k times ε, where k is a constant no matter how small ε is, this implies that we can take $I_s = (1-k\varepsilon,1)$, and on $[0,1-k\varepsilon]$ u(x) and its first fourth derivatives are uniformly bounded. Hence we only need to consider the maximum error within the boundary layer. From (4.17), when $h \leq \varepsilon\|p\|_\infty/2$, in the interval I_s

$$-\varepsilon u^{(4)} + pu^{(3)} = c\{ \frac{p(1)^3}{\varepsilon^3}[p(x)-p(1)]e^{-p(1)(1-x)/\varepsilon} - \varepsilon Z^{(4)} + pZ^{(3)} \} = O(h^{-2}),$$

$$u'' - \frac{u'^2}{c+u} = \frac{1}{u+c}\{u''(u+c) - u'^2\}$$

$$= \frac{c^2}{u+c}\{[\frac{p(1)^2}{\varepsilon^2}e^{-p(1)(1-x)/\varepsilon} + Z''][e^{-p(1)(1-x)/\varepsilon} + Z + c] - [\frac{p(1)}{\varepsilon}e^{-p(1)(1-x)/\varepsilon} + Z]^2\}$$

$$= \frac{c^2}{u+c}\, e^{-p(1)(1-x)/\varepsilon}\frac{p(1)}{\varepsilon}[\frac{p(1)}{\varepsilon}(Z+1) - 2] + \dots$$

$$\{u'' - \frac{u'^2}{c+u}\}' = \frac{c^2}{u+c}e^{-p(1)(1-x)/\varepsilon}(\frac{p(1)}{\varepsilon})^2[\frac{p(1)}{\varepsilon}(Z+1) - 2] + \dots$$

Since $u(1)=0$, $Z(1)\doteq -1$, so

$$u'' - \frac{u'^2}{c+u} = O(h^{-1}), \quad \{u'' - \frac{u'^2}{c+u}\}' = O(h^{-2}). \tag{4.19}$$

Substituting into (4.16) yields

$$\max_{I_s}|Tr(x)| = O(h).$$

However, for the usual linear central difference scheme the corresponding truncation error is $O(h^0)$. Hence, introducing the semi-linear scheme gives one more order of precision for the maximum local truncation error. Considering that the width of the boundary layer I_s is only $k\varepsilon$, the number of knots using the semi-linear scheme is always less than a constant if the ratio $\frac{h}{\varepsilon}$ remains a constant. Hence

$$\|A^{-1}Tr(x)\|_\infty = O(h) \tag{4.20}$$

Since the error system

$$A\ (u - U^h) = Q(u) - Q(U) + Tr(u),$$

so

$$\|u - U^h\| \le \|A^{-1}(Q(u)-Q(U^h))\| + \|A^{-1}Tr(x)\|,$$

$$\|u - U^h\| < \{ 1 - \|A^{-1}J(Q(u))\| \}^{-1}\|A^{-1}Tr(x)\|.$$

if $\|A^{-1}J(Q(u))\|\ \|u - U^h\| < 1$.

Therefore, we have extended the error estimate Theorem 3.6 for the general problem (4.1).

<u>Theorem</u> <u>4.6</u>: The solution of the semi-linear scheme (4.3) converges to the exact solution of the singular perturbation problem (4.1) as h tends zero if the (4.8) holds. Moreover, if the ratio of h to ε is less than a constant, as given in (4.7) or (4.15), according to the condition of Theorems 4.4 and 4.5 respectively, then there exist an error estimation such that

$$\| U^h - u \|_\infty < Ch, \tag{4.21}$$

where the coefficient C is uniformly bounded for all $\varepsilon>0$.

Using the same block matrix technique described in the last section, the mesh constrain (4.7) or (4.15) can be limited only in the interval I_s. In the regular interval, a larger mesh step can be allowed.

5. Computational results

In this section, three numerical examples are presented to show the effect of the semi-linear scheme. Only uniform mesh sizes are used throughout this section. The emphasis is on the comparison among the semi-linear scheme, the corresponding linear centered difference scheme, and an 'Upwind method' described by Christie and el. in [3].

EXAMPLE 1. -- Model Problem

$$Lu = -\varepsilon u'' + u' = 0 \qquad \text{in } (0,1)$$

$$u(0) = 0, \; u(1) = 1 \qquad (3.1)$$

Numerical results for the model problem (3.1) are tablated in Table 1-3, and a comparison of three different methods is given for $\varepsilon = 1/60$ and $h = 1/40$ in the sense of the pointwise error. For the linear centered difference scheme, there is a relative error of 36% at the last interior mesh point $x = 0.975$. The 'Upwind Symmetric Quadratics Method' has a relative error 1.2% at the same point. The advantage of the semi-linear scheme is clear. Three or seven digits can be improved for an iteration error of 0.1D-3 or 0.1D-8, respectively. For a given accuracy, the required CPU time using the semi-linear scheme is twice less than using the usual linear scheme.

For the same ε, Table 2 describes the convergence rate. If the given admissible error is 0.5D-4, the semi-linear scheme requires a mesh size h approximately 1/30, while the linear scheme requires a mesh size 1/2000. The total CPU time between these two schemes differed by a factor of 2 to 3. If we consider the error of the first derivative, the difference between the two schemes is even greater because we use a semi-linear differentiation formula to match the semi-linear scheme (3.10) in order to get the discrete first steep derivative.

The influence of the the convergence of the semi-linear discrization is presented in Table 3, depending upon the ratio of ε to h. We consider three cases: $h/\varepsilon = 2$, $3/2$, and 1. According to Theorem 3.2, the iteration converges if $h/\varepsilon \leq 2$. These numerical results satisfy the theory. The iteration is also convergent for $h = 2\varepsilon$, however, this is not recommended because the rate of convergence is too slow. For a practical choice of $h/\varepsilon = 1.5$, a higher precision seems with obtainable less CPU time.

Hence, to get the same accuracy, one can solve a small semi-linear systems instead of the original large linear system. However, the error in the two schemes behaves differently. For the linear scheme, when the ratio ε/h is constant, the error is also constant independent of the size of ε. For the semi-linear scheme the error converges to zero as ε does. The examples to follow will show the same phenomenon.

Table 1.
(ε = 1/60, h = 1/40)

x	Theoretical solution	Linear scheme	Semi-linear ei = 0.1-3	Semi-linear ei = 0.1-8	Upwind method
0.800	0.000006144	0.000000173	0.000006141	0.000006144	-
0.850	0.000123410	0.000008500	0.000123447	0.000123410	-
0.875	0.000553084	0.000059499	0.000553305	0.000553085	-
0.900	0.002478752	0.000416493	0.002477749	0.002478752	0.0026
0.925	0.011108997	0.002915452	0.011110675	0.011108997	0.0115
0.950	0.049787068	0.020408163	0.049785741	0.049787068	0.0510
0.975	0.223130160	0.142857143	0.223130576	0.223130160	0.2258
1.000	1.0	1.0	1.0	1.0	1.0
Max error of u		-0.8027-01	0.1679-5	0.2044-9	0.27-2
Location of max error		0.975	0.925	0.875	0.975
Max error of u'		0.56+1	0.95-4	0.11-7	-
Number of iterations		1	12	22	1
CPU TIME (seconds)		0.07	0.15	0.26	-

Remark: ei -- the admissible error for iterations of the semi-linear scheme.

Table 2
(ε = 1/60, ei(SL1) = 0.1-4, ei(SL2) = 0.1-8)

	1/h / M	20	30	40	60	640	1920
Max error of u	L	-0.25+0	-0.14+0	-0.80-1	-0.35-1	-0.27-3	-0.30-4
	SL1	-0.46+1	0.54-5	0.17-5	-0.35-6	-0.37-9	-0.51-12
	SL2	-0.46+1	0.32-9	0.20-9	-0.51-10	0.46-12	0.46-16
Max error of u'	L	0.70+1	0.66+1	0.56+1	0.39+1	0.80-1	-0.25-1
		-0.10+4	0.31-3	0.95-4	-0.21-4	-0.13-6	0.50-9
	SL2	-0.10+4	0.19-7	0.11-7	-0.28-8	0.16-9	0.51-13
CPU TIME	L	0.01	0.02	0.07	0.08	0.46	1.30
	SL1	0.93 (100)	0.51 (35)	0.15 (12)	0.16 (7)	0.90 (3)	2.78 (3)
	SL2	0.93 (100)	0.74 (71)	0.28 (22)	0.24 (12)	1.10 (4)	3.21 (4)

Remark. 1. The number in the bracket is the number of iterations required to reduce the error to be less than the admissible range ei.
2. Max u' = 60.

Table 3-1 h/ε = 2

	1/ε ei	10 0.1-4	20 0.1-4	60 0.1-8	100 0.1-12
Error of u	L	<------------ -0.14+0 ------------>			
	SL	-0.45+0	-0.96-4	0.32-9	0.20-12
Error of u'	L	0.11+1	0.22+1	0.66+1	0.11+2
	SL	0.45+1	-0.20-2	0.19-7	0.18-10
CPU TIME	L	0.01	0.01	0.03	0.11
	SL	0.21	0.35	0.70	1.76
		(100)	(100)	(71)	(100)

Remark. 1. The number in the bracket is the number of iterations required to reduce the error to be less than the admissible range ei.

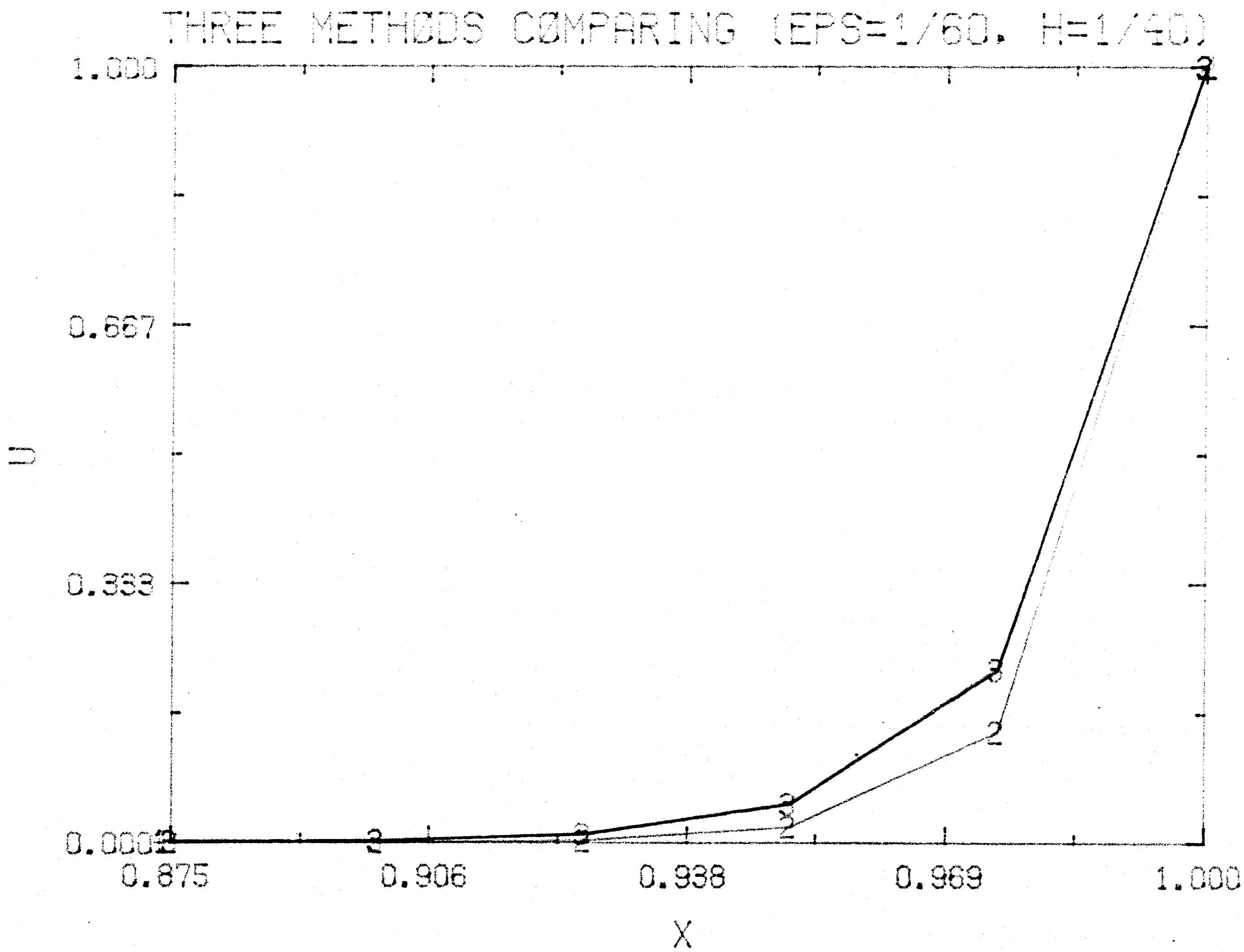
Table 3-2 h/ε = 3/2

	1/ε ei	15 0.1-4	30 0.1-4	60 0.1-8	90 0.1-12
Error of u	L	<------------ -0.80-1 ------------>			
	SL	-0.23-3	0.15-5	0.20-9	0.28-13
Error of u'	L	0.14+1	0.28+1	0.56+1	0.84+1
	SL	-0.39-2	0.43-4	0.11-7	0.24-11
CPU TIME	L	0.01	0.01	0.03	0.11
	SL	0.03	0.07	0.25	0.70
		(12)	(12)	(22)	(30)

Table 3-3 h/ε = 1

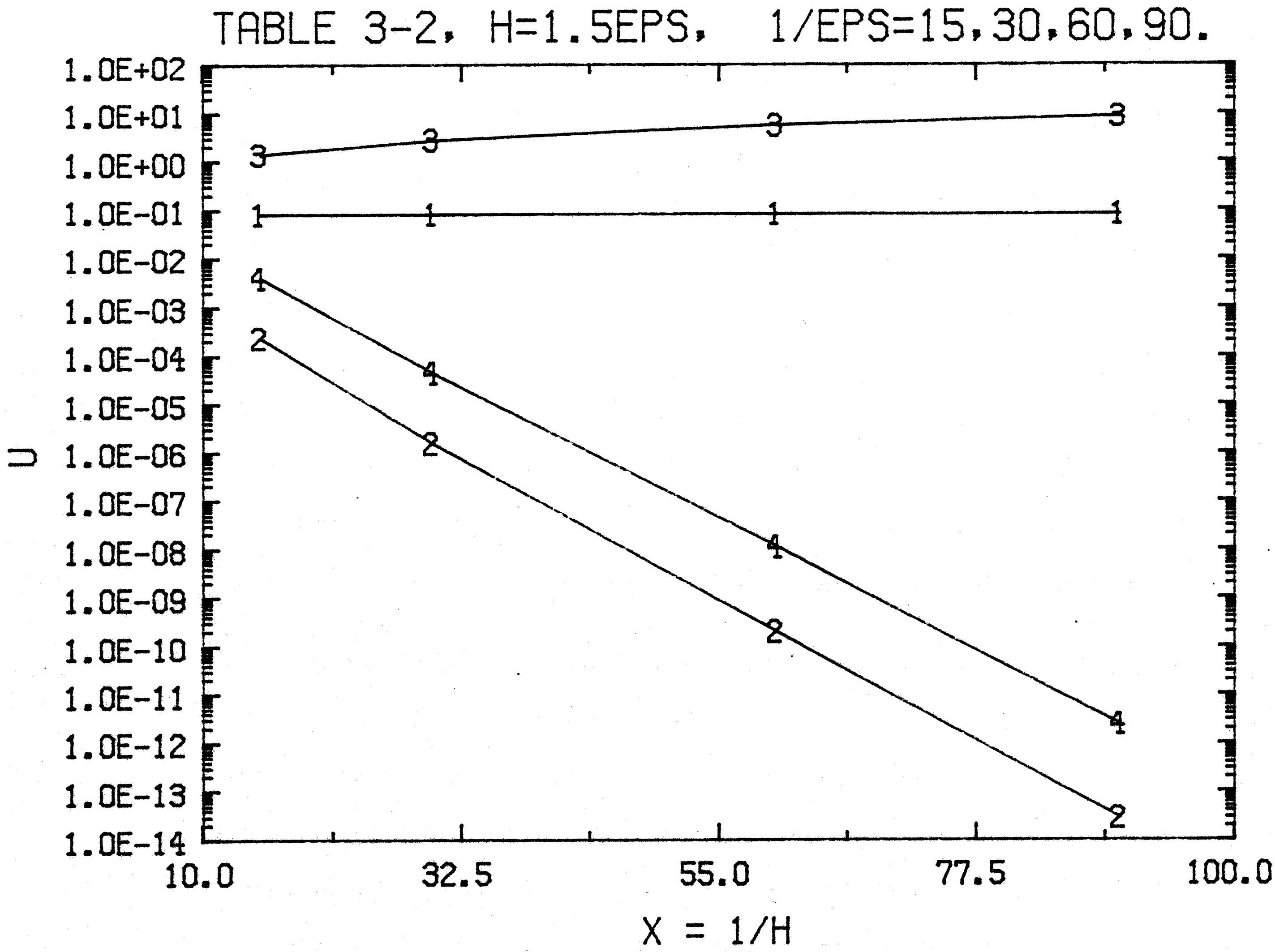
	1/ε ei	5 0.1-4	10 0.1-4	20 0.1-4	60 0.1-8	100 0.1-12
Error of u	L	<------------ -0.35-1 ------------>				
	SL	-0.49-2	-0.92-3	-0.64-5	-0.51-10	-0.69-14
Error of u'	L	0.32+0	0.64+0	0.13+1	0.38+1	0.64+1
	SL	0.33-1	-0.11-1	-0.15-3	-0.28-8	-0.59-12
CPU TIME	L	0.01	0.01	0.02	0.06	0.21
	SL	0.01	0.02	0.05	0.23	0.71
		(7)	(7)	(7)	(12)	(17)

Figure 2



1 SØLUTIØN X = 0.975, U = 0.2231
2 LINEAR SCHEME U = 0.1429
3 SEMILINEAR SCHEME U = 0.2231
4 UPWIND U = 0.2258

Figure 3



1 LINEAR SCHEME FUNCTIØN ERRØR = 0.081
2 SEMILINEAR SCHEME FUNCTIØN ERRØR 0.23E-3
3 LINEAR SCHEME FIRST DERIVATIVE ERRØR
4 SEMILINEAR SCHEME FIRST DERIVATIVE ERRØR

EXAMPLE 2. A linear singular perturbation problem with constant coefficients

$$Lu = -\varepsilon u'' + u' + (1+\varepsilon)u = f(x), \quad \text{in } (0,1)$$

$$u(0) = u(1) = 0 \qquad (4.1)$$

where $f(x) = (1+\varepsilon)(a-b)x - \varepsilon a - b$, $a = 1 + e^{-(1+\varepsilon)/\varepsilon}$, $b = 1 + e^{-1}$, with exact solution

$$u(x) = e^{-(1+\varepsilon)(1-x)/\varepsilon} + e^{-x} - a + (a-b)x$$

It is not difficulty to find that the coefficient γ in (4.17) here is equal to 1 without knowing the exact solution in advance. Hence, we assign the constant $c = 1$ using the semi-linear scheme (4.2).

The result listed in Table 4 shows that iterations converge monotonically if the ratio $h/\varepsilon \leq 2$ and that the calculated results agree well with the theoritical analysis. For the same mesh size, required CPU time using semi-linear scheme is little more than using the usual linear scheme (about 10% to 20% for small ε), but more than one significant digit is obtained. More small ε is contained, more advantage the semi-linear scheme has. Therefore, with the same accuracy, using semi-linear scheme, an large linear system arisen from the linear scheme is replaced by a smaller semi-linear system. For instance, when $\varepsilon=0.01$, the maximum error using the semilinear scheme with N=60 (after 13 iterations) is less 50% than of the usual linear scheme with N=200, meanwhile the ratio of CPU time is 0.27 : 1.12(sec.). When $\varepsilon=0.001$, the maximum error using the semilinear scheme with N=600 is less ten times than of the usual linear scheme with N=2000, and the ratio of CPU time is 3.22 : 7.87(sec.).

Table 4-1 $\varepsilon = 0.1$ ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
5	L	0.800	0.1935D+00	0.1247D+01	0.01	1
	SL	0.800	0.1935D+00	0.1247D+01	0.01	2
6	L	0.833	0.1474D+00	0.1163D+01	0.01	1
	SL	0.833	0.4765D-01	0.8202D-01	0.03	10
10	L	0.900	0.6548D-01	0.7837D+00	0.01	1
	SL	0.800	0.2255D-01	-0.2642D+00	0.08	7
20	L	0.950	0.1974D-01	0.3247D+00	0.02	1
	SL	0.850	0.6793D-02	-0.1112D+00	0.04	4
40	L	0.975	0.5493D-02	0.1057D+00	0.05	1
	SL	0.850	0.1499D-02	-0.3305D-01	0.10	4

Remark. 1. NE is the number of iterations required to reduce the error to be less than the admissible range ei.

Table 4-2 $\varepsilon = 0.01$, ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
50	L	0.980	0.1595D+00	0.1116D+02	0.15	1
	SL	0.960	0.7958D-02	-0.1829D+01	0.40	24
60	L	0.983	0.1230D+00	0.1020D+02	0.18	1
	SL	0.967	0.6016D-02	-0.1543D+01	0.27	13
100	L	0.990	0.5561D-01	0.6581D+01	0.29	1
	SL	0.970	0.2872D-02	-0.6184D+00	0.43	7
200	L	0.995	0.1688D-01	0.2624D+01	0.57	1
	SL	0.970	0.6339D-03	-0.1742D+00	0.69	4
400	L	0.998	0.4705D-02	0.8364D+00	1.12	1
	SL	0.970	0.1681D-03	-0.5743D-01	1.36	4

Table 4-3 $\varepsilon = 0.001$, ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
500	L	0.998	0.1568D+00	0.1102D+03	2.05	1
	SL	0.992	-0.1981D-02	-0.2046D+01	3.42	24
600	L	0.998	0.1210D+00	0.1006D+03	2.47	1
	SL	0.995	0.9313D-03	-0.2414D+01	3.22	14
1000	L	0.999	0.5475D-01	0.6459D+02	3.93	1
	SL	0.995	0.2977D-03	-0.8986D+00	5.16	7
1200	L	0.999	0.4045D-01	0.5219D+02	4.77	1
	SL	0.995	0.2150D-03	-0.6389D+00	5.58	6
2000	L	0.999	0.1663D-01	0.2565D+02	7.87	1
	SL	0.996	0.9066D-04	-0.2710D+00	9.00	4

Figure 4

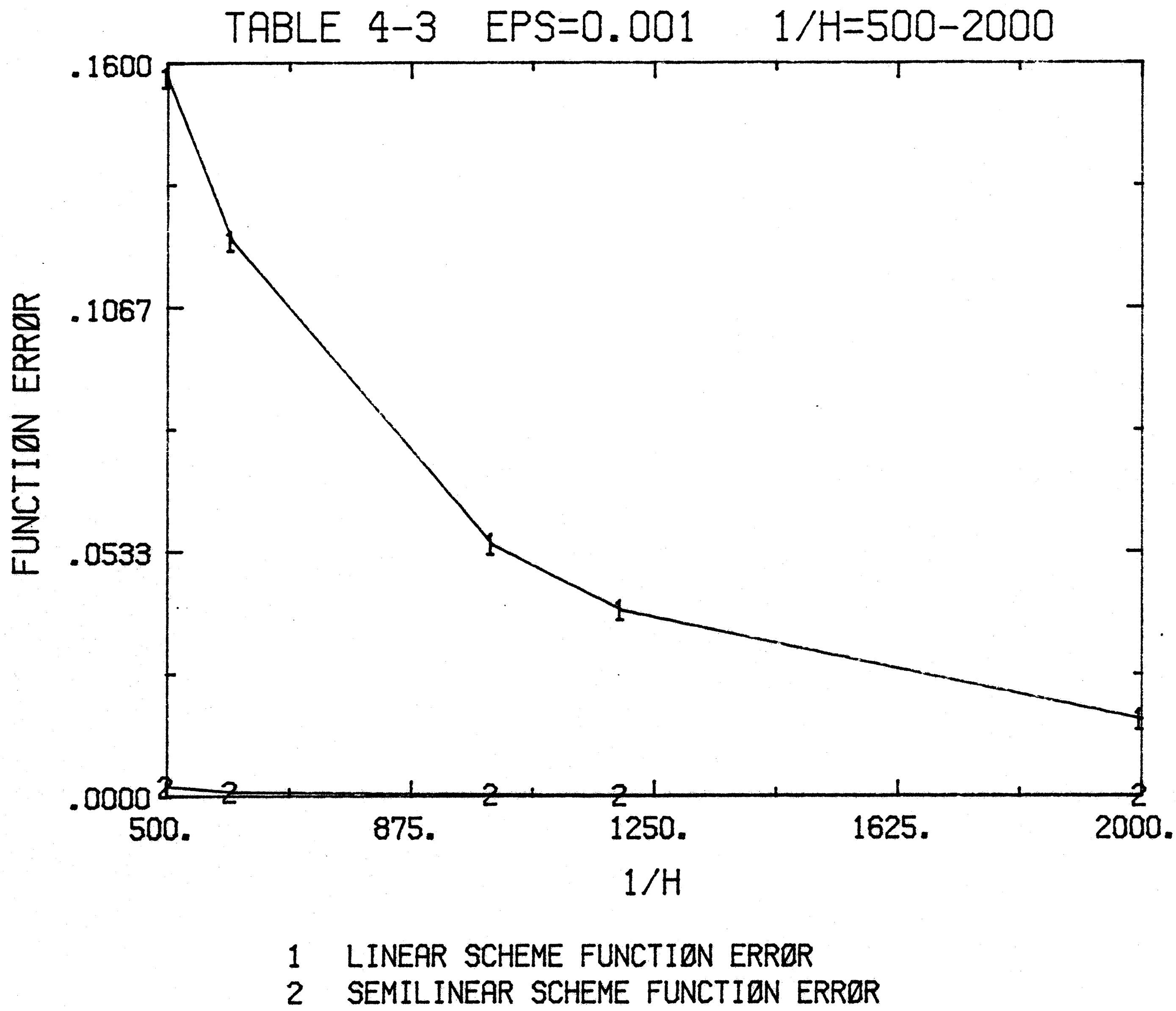
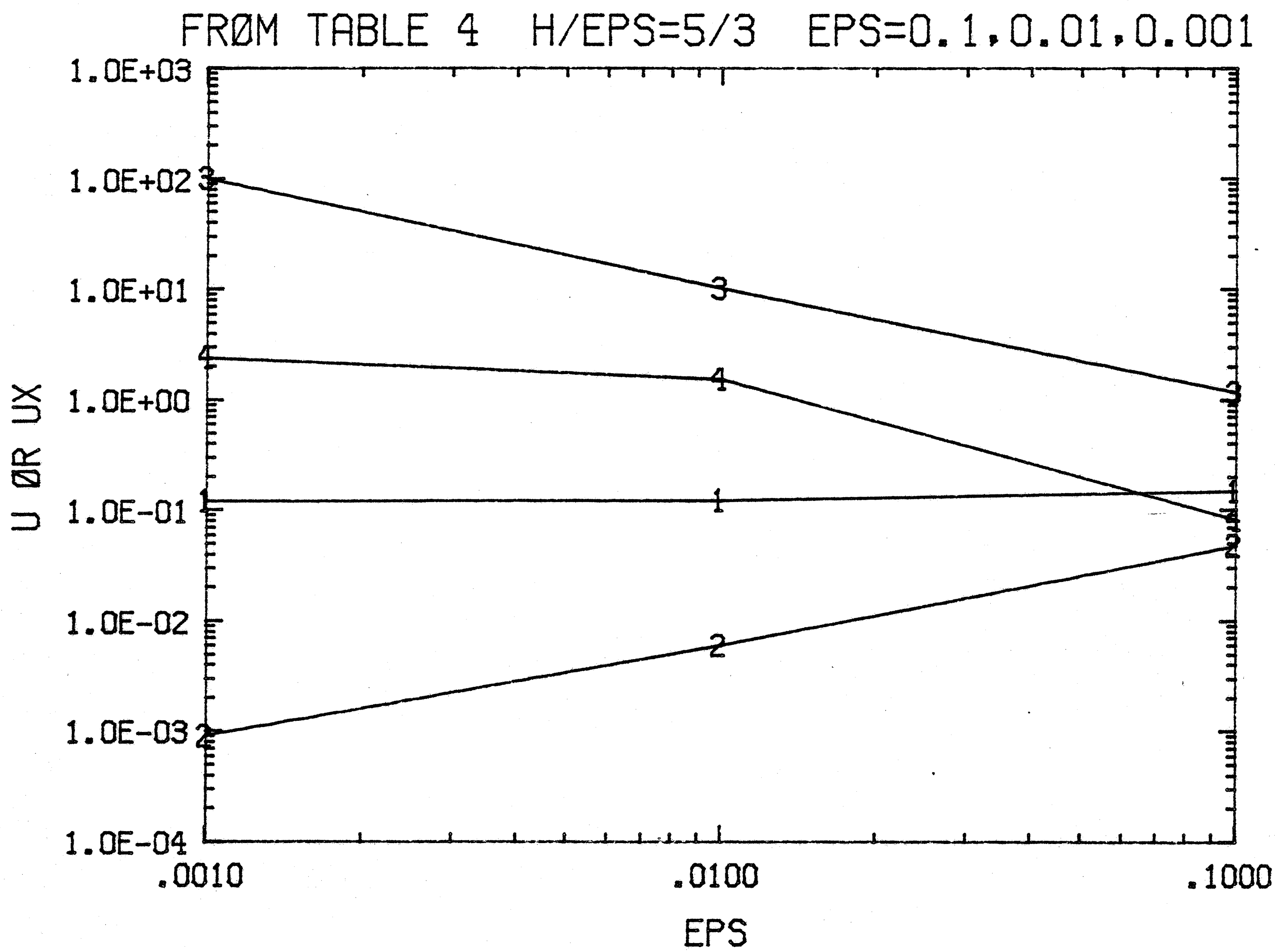


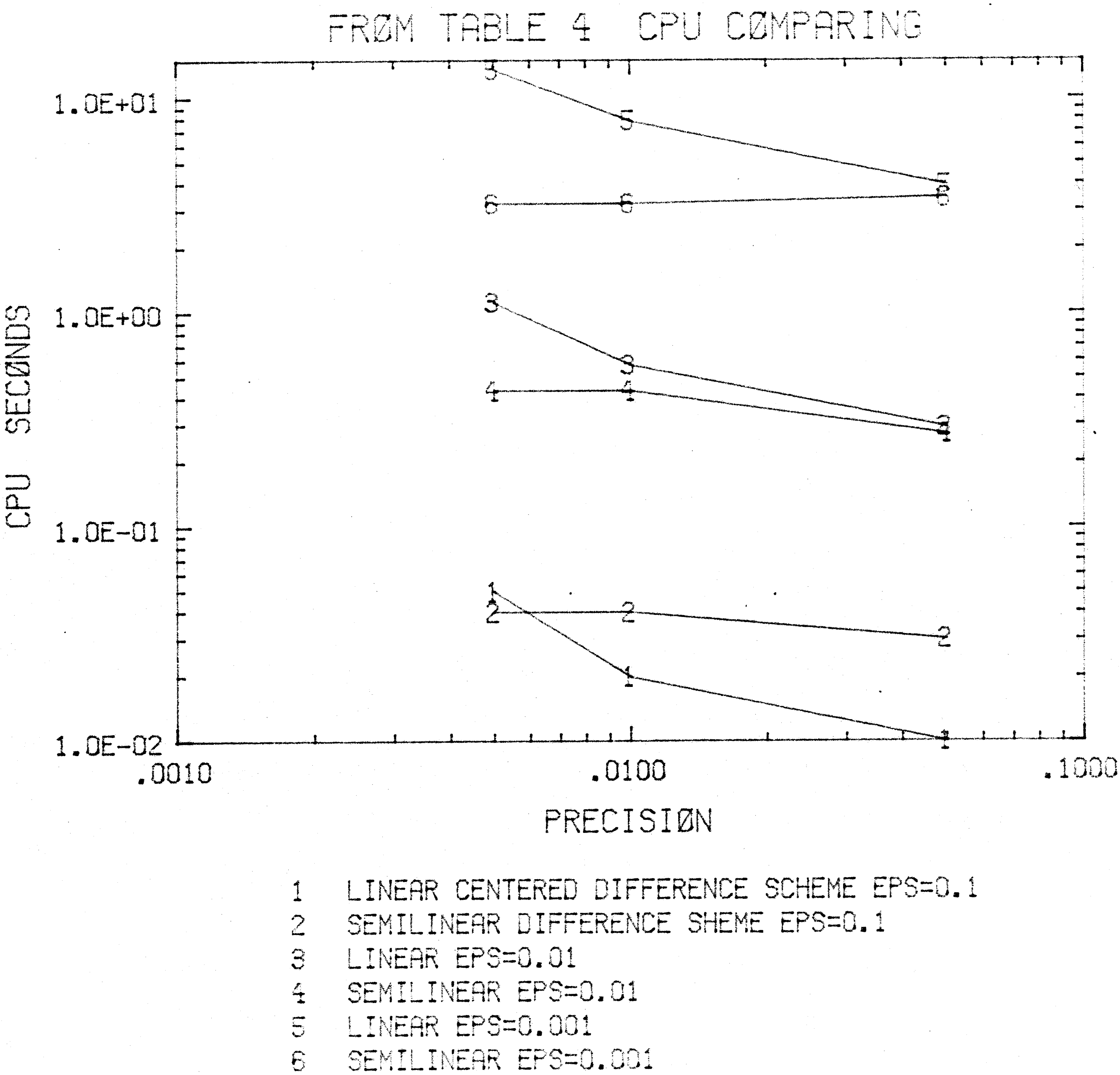
TABLE 4-3 EPS=0.001 1/H=500-2000
.1600
.1067
.0533
.0000
FUNCTIØN ERRØR
500.
875.
1250.
1625.
2000.
1/H
1 LINEAR SCHEME FUNCTIØN ERRØR
2 SEMILINEAR SCHEME FUNCTIØN ERRØR

Figure 5



1 LINEAR SCHEME FUNCTIØN ERRØR
2 SEMILINEAR SCHEME FUNCTIØN ERRØR
3 LINEAR SCHEME FIRST DERIVATIVE ERRØR
4 SEMILINEAR SCHEME FIRST DERIVATIVE ERRØR

Figure 6


FRØM TABLE 4 CPU CØMPARING
1.0E+01
1.0E+00
1.0E-01
1.0E-02
CPU SECØNDS
.0010
.0100
.1000
PRECISIØN
1 LINEAR CENTERED DIFFERENCE SCHEME EPS=0.1
2 SEMILINEAR DIFFERENCE SHEME EPS=0.1
3 LINEAR EPS=0.01
4 SEMILINEAR EPS=0.01
5 LINEAR EPS=0.001
6 SEMILINEAR EPS=0.001

EXAMPLE 3. A semi-linear singular perturbation problem

$$Lu = -\varepsilon u'' + u' + (1+\varepsilon)u = f(x,u), \quad \text{in } (0,1)$$

$$u(0) = u(1) = 0 \tag{5.1}$$

with the same solution as Example 2, where

$$f(x,u) = a - b - (1+\varepsilon)\{e^{-x} - u + \frac{c}{u+a-(a-b)x-e^{-x}}\},$$

$$a=1+e^{-(1+\varepsilon)/\varepsilon},\ b=1+e^{-1},\ c = e^{2(1+\varepsilon)(1-x)/\varepsilon}.$$

This example shows the advantage of using SL- scheme over L-scheme more than example 2, because both schemes have to be solved by iteration for this semi-linear problem, and in this case SL-scheme requires less CPU time than in L-scheme for the same mesh size.

Two different results of two semi-linear schemes are listed in Table 5. SL1 scheme is described in section 4. SL2 scheme comes from a Semi-linear Galerkin method [Jiachang Sun, 11]. Using the same simple iterative procedure for the same size h, SL2 seems need less number of iterations required to reduce the error to be less than the same admissible range than SL1, but SL2 costs a little more CPU time to compute its coefficients. At any rate, the advantage of both semi-linear schemes over the linear scheme is clear. The stronger the singularity of the solution of the differential equation, the more the advantage.

Table 5-1 $\varepsilon = 0.1$, ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
6	L	0.833	0.1413D+00	0.1163D+01	0.25	100
	SL2	0.667	0.1913D-01	0.1468D+00	0.16	11
10	L	0.900	0.1111D+00	0.7837D+00	0.43	100
	SL	0.900	0.2463D-01	-0.2786D+00	0.60	100
	SL2	0.800	0.1715D-01	0.2356D+00	0.17	13
20	L	0.950	0.2517D-01	0.3247D+00	0.86	100
	SL	0.900	0.1110D-01	-0.1286D+00	1.12	100
	SL2	0.850	0.5012D-02	-0.9715D-01	0.32	13
40	L	0.975	0.6984D-02	0.1057D+00	0.30	14
	SL	0.900	0.2445D-02	-0.4427D-01	0.35	14
	SL2	0.825	0.1004D-02	-0.2500D-01	0.70	13
80	L	0.987	0.1812D-02	0.3023D-01	0.63	14
	SL	0.900	0.6265D-03	-0.1872D-01	0.73	14
	SL2	0.837	0.2561D-03	-0.7389D-02	1.33	13

Table 5-2 $\varepsilon = 0.01$, ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
50	L	0.980	0.1391D+00	0.1116D+02	4.33	100
	SL1	0.960	0.8372D-02	-0.1886D+01	4.72	100
	SL2	0.960	0.7927D-02	-0.1662D+01	11.67	100
60	L	0.983	0.1255D+00	0.1020D+02	7.37	100
	SL2	0.967	0.5378D-02	-0.1375D+01	1.47	10
100	L	0.990	0.5766D-01	0.6581D+01	1.08	10
	SL1	0.970	0.3158D-02	-0.6582D+00	1.11	9
	SL2	0.960	0.2617D-02	-0.5572D+00	2.20	9
200	L	0.995	0.1733D-01	0.2624D+01	2.05	9
	SL1	0.975	0.6970D-03	-0.1894D+00	2.14	9
	SL2	0.965	0.5763D-03	-0.1500D+00	4.44	9
400	L	0.998	0.4812D-02	0.8364D+00	4.13	9
	SL1	0.973	0.1845D-03	-0.6287D-01	4.51	9
	SL2	0.968	0.1536D-03	-0.4842D-01	8.86	9

Table 5-3 $\varepsilon = 0.001$, ei = 1.0-5

h=1/N		x	Max(Er(u))	Max(Er(u'))	CPU	NE
500	L	0.998	0.2943D-01	0.1102D+03	58.29	100
	SL1	0.994	0.1020D-02	-0.3495D+01	59.29	100
	SL2	0.994	0.1129D-02	-0.3240D+01	153.63	100
600	L	0.998	0.1211D+00	0.1006D+03	71.83	100
	SL2	0.995	0.9154D-03	-0.2334D+01	23.22	11
1000	L	0.999	0.5494D-01	0.6459D+02	14.16	9
	SL1	0.995	0.3019D-03	-0.9171D+00	14.05	9
	SL2	0.995	0.2907D-03	-0.8533D+00	28.66	9
2000	L	0.999	0.1667D-01	0.2565D+02	27.56	9
	SL1	0.996	0.9201D-04	-0.2771D+00	27.70	9
	SL2	0.995	0.8845D-04	-0.2594D+00	57.68	9

Figure 7

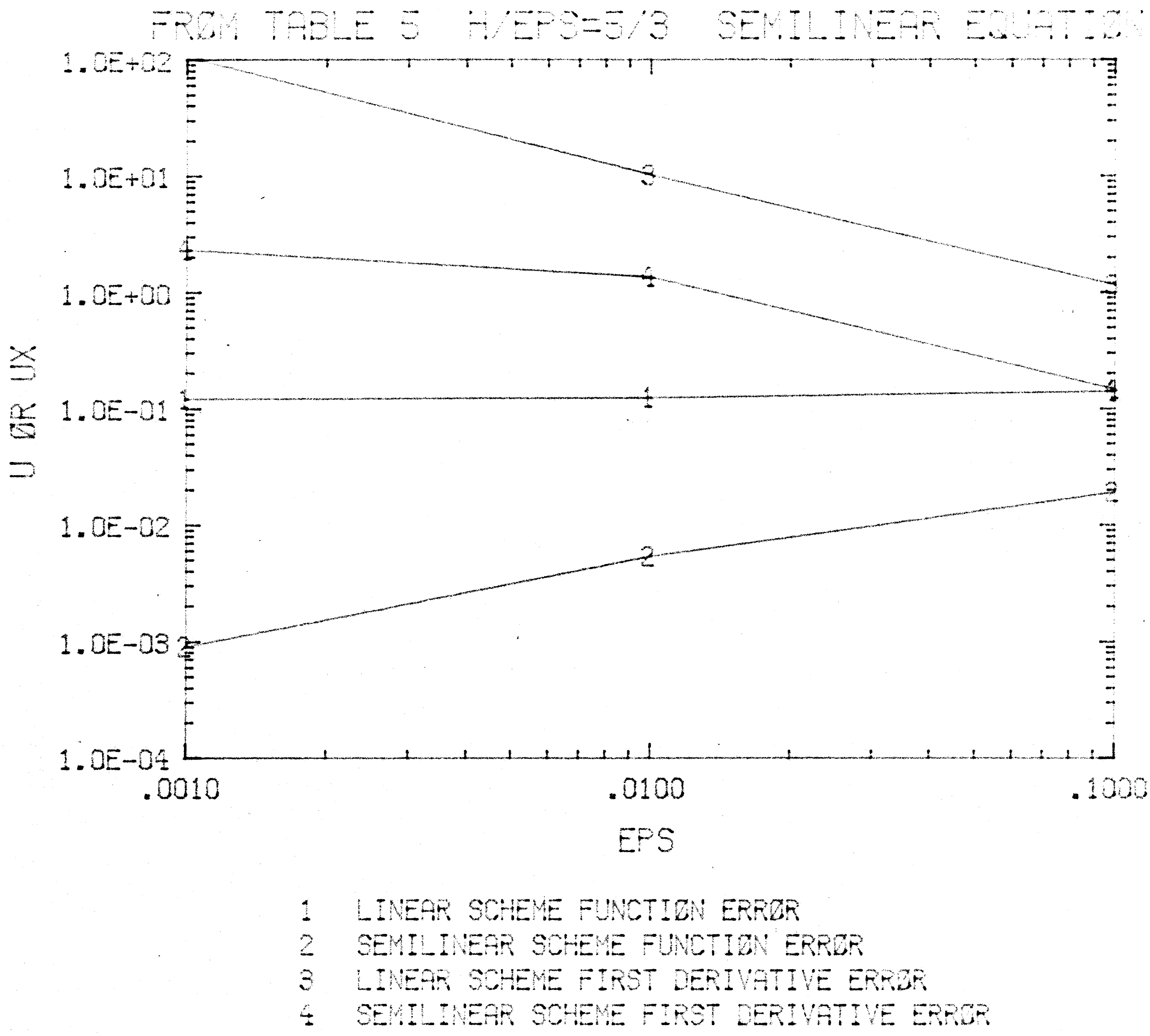

FRØM TABLE 5 H/EPS=5/3 SEMILINEAR EQUATIØN
1.0E+02
1.0E+01
1.0E+00
1.0E-01
1.0E-02
1.0E-03
1.0E-04
U ØR UX
.0010
.0100
.1000
EPS
1 LINEAR SCHEME FUNCTIØN ERRØR
2 SEMILINEAR SCHEME FUNCTIØN ERRØR
3 LINEAR SCHEME FIRST DERIVATIVE ERRØR
4 SEMILINEAR SCHEME FIRST DERIVATIVE ERRØR

Figure 8

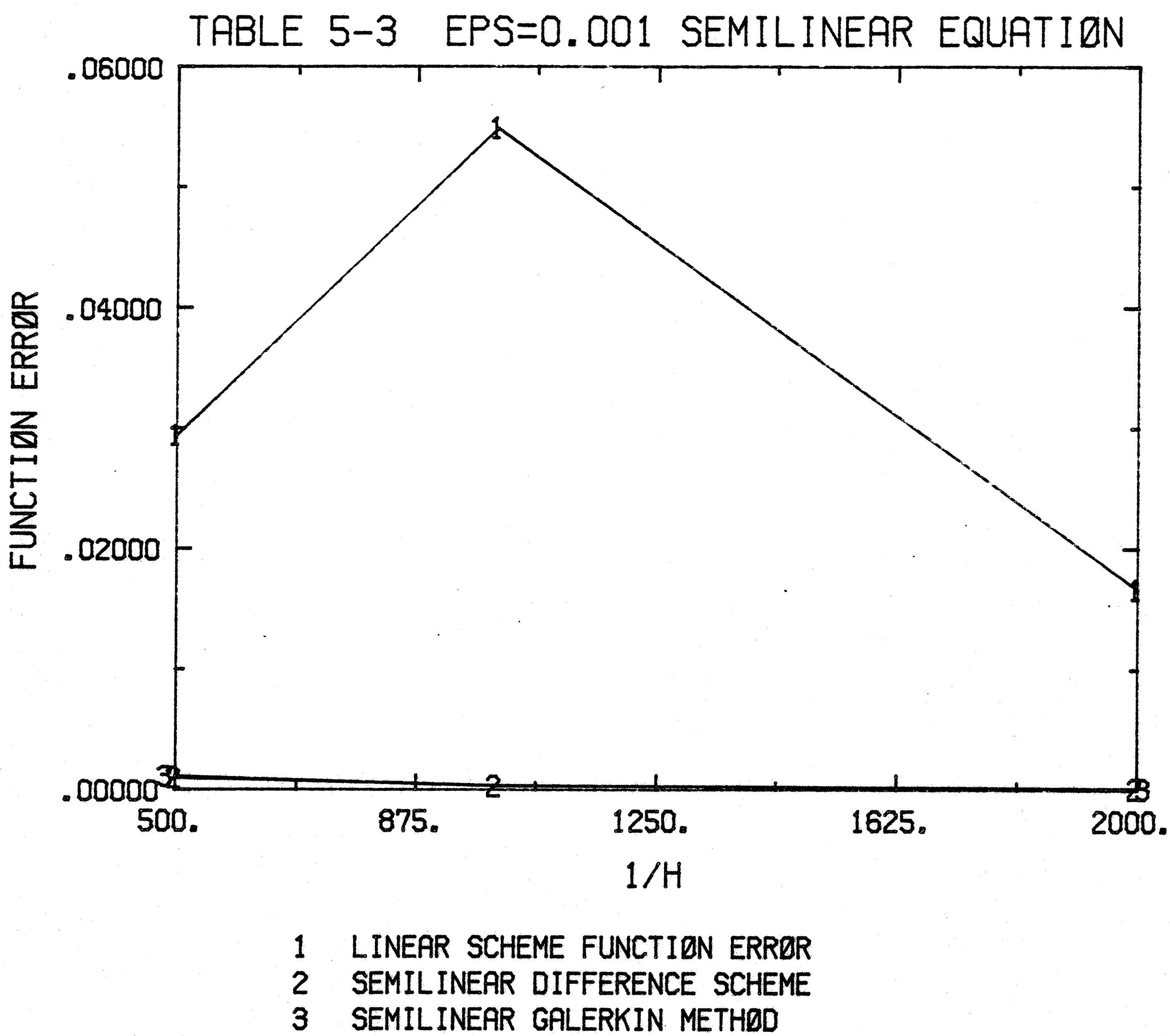
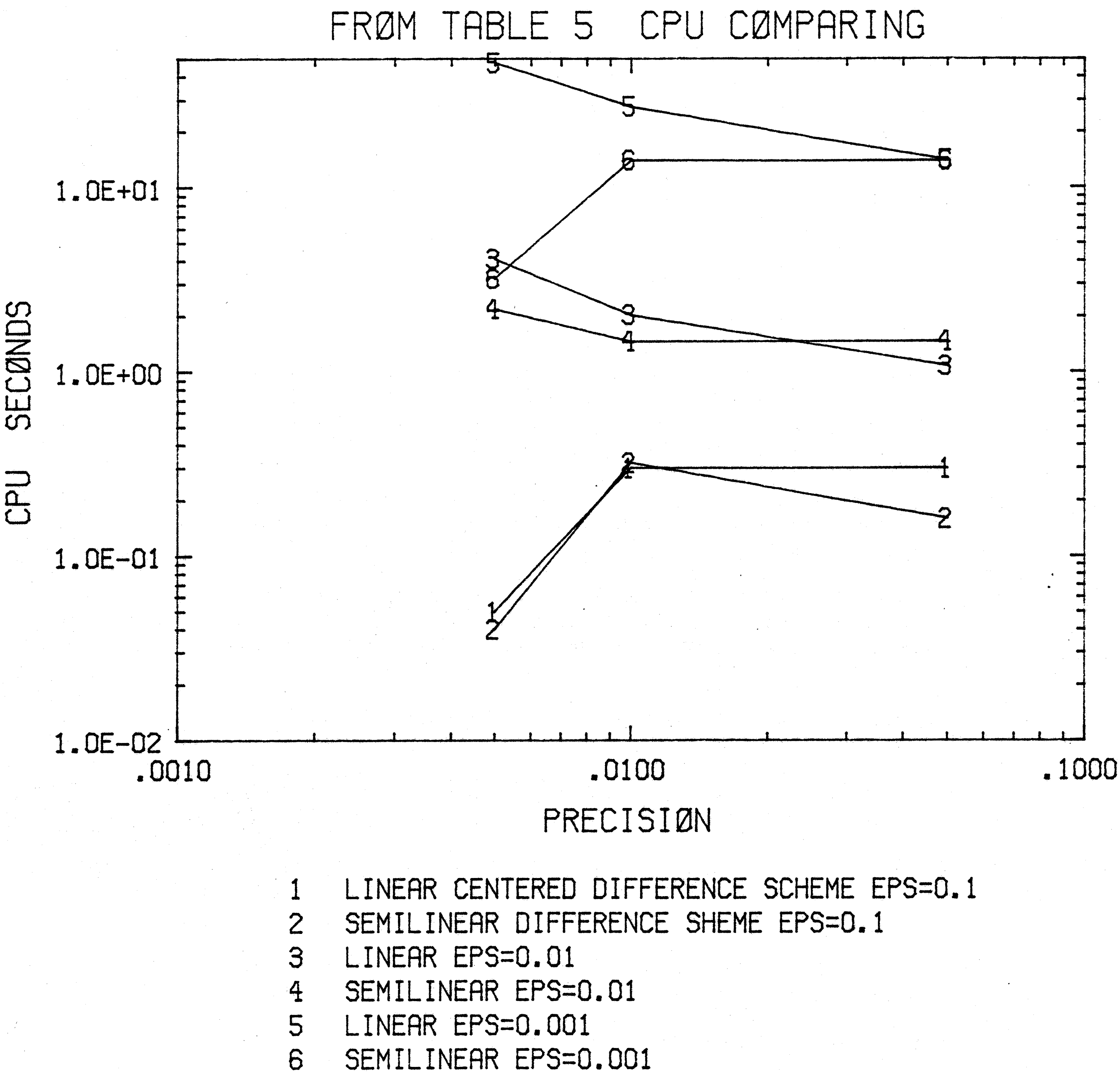


TABLE 5-3 EPS=0.001 SEMILINEAR EQUATIØN
.06000
.04000
.02000
.00000
FUNCTIØN ERRØR
500.
875.
1250.
1625.
2000.
1/H
1 LINEAR SCHEME FUNCTIØN ERRØR
2 SEMILINEAR DIFFERENCE SCHEME
3 SEMILINEAR GALERKIN METHØD

Figure 9


FRØM TABLE 5 CPU CØMPARING
CPU SECØNDS
1.0E+01
1.0E+00
1.0E-01
1.0E-02
.0010
.0100
.1000
PRECISIØN
1 LINEAR CENTERED DIFFERENCE SCHEME EPS=0.1
2 SEMILINEAR DIFFERENCE SHEME EPS=0.1
3 LINEAR EPS=0.01
4 SEMILINEAR EPS=0.01
5 LINEAR EPS=0.001
6 SEMILINEAR EPS=0.001

REFERENCES

1. Barrett, J.W., Morton, K.W., Optimal Finite Element Solutions to Diffusion-Convection Problems in One Dimension, Int. J. Num. Meth. Engng. 15(1980), 1457-1474.

2. Christie, J., Mitchell, A.R., Upwinding of High Order Gelerkin Methods in Conduction-Convection Problems, Int. J. Num. Meths. in Engng. 12(1978), 1764-1771.

3. Christie, J., Griffith, D.F., Mitchell,A.R., Zienkienwicz, O.C., Finite Element Methods for Second Order Differential Equations with Significant First Derivatives, Int. J. Num. Meths. in Engng. 10(1976), 1389-1396.

4. Davis, S.F., Flaherty, J.E., An Adaptive Finite Element Method for Initial-Boundary Value Problems for Partial Differential Equations. Technical Report 81-13, 1981, March, Universities Space Research Association.

5. Han Honde, The Finite Element Method for a Singular Perturbation Problem Using Enriched Subspace. Technical Report, 1981, October, University of Maryland.

6. Heinrich, J.C., Huyakorn, P.S., Zienkienwicz, O.C., Mitchell, A.R., An 'Upwind' Finite Element Scheme for Two-Dimensional Convective Transport Equation, Int. J. Num. Meths. in Engng. 11(1977), 131-143.

7. Schultz, M.H., ''Spline Analysis'', Prentice-Hall, Inc., 1973.

8. Sun Jiachang, The Spline Functions in Local Coordinates and Circular Spline Curve, Mathematicae Sinica, 20(1977), 28-40.

9. Sun Jiachang, Generalizations of the Means and their Inequalities. Department of Mathematics, University of California, Santa Barbara, May, 1981. (to appear).

10. Sun Jiachang, Semi-linear One-step Implicit Schemes for Solving Initial Value Problem of Ordinary Differential Equations with Steep Gradients. Department of Computer Science, Yale University, Technical Report No.215. 1982.

11. Sun Jiachang, A Galerkin Method on Nonlinear Subsets and Its Application to a Singular Perturbation Problem. Department of Computer Science, Yale University, Technical Report No.217. 1982.

12. Sun Jiachang, Semi-linear Difference Schemes and Semi-linear Finite Element Methods for Singular Perturbation Problems in two dimensions. (in preparation)