

Time-Splitting for
Advection-Dominated Problems
in One-Space Variable

by

M.F. Wheeler*
C.N. Dawson*
Wendy A. Kinton*

Technical Report 87-24, November 1987

*Mathematical Sciences Department, Rice University, Houston, Texas 77251.

Time-splitting for Advection-Dominated
Parabolic Problems in One-Space Variable

Mary Fanett Wheeler, Wendy A. Kinton, and Clint N. Dawson*

1. Introduction.

In this paper we consider transient problems of the form

$$\frac{\partial s}{\partial t} + \frac{\partial}{\partial x}(g(s)) - \frac{\partial}{\partial x}(a(s) \frac{\partial s}{\partial x}) = 0, \quad x \in I = (0, 1), \quad t > 0. \quad (1.1)$$

$$s(x, 0) = s_0(x), \quad x \in I, \quad (1.2)$$

with the inlet boundary condition

$$s(0, t) = \alpha(t), \quad t > 0, \quad (1.3)$$

and 'absorbing' boundary condition

$$\left. \frac{\partial s}{\partial t} + \frac{\partial}{\partial x}(g(s)) \right|_{x=1} = 0, \quad t > 0. \quad (1.4)$$

We assume $g'(s) \geq 0$,

$$0 \leq a(s) \leq a_1, \quad s \in \mathbf{R},$$

and $g'(s) \gg a(s)$.

The major thrust of this work is to consider a sequential time splitting of (1.1) into two operators; namely,

$$\begin{aligned} \frac{\partial \tilde{s}}{\partial t} + \frac{\partial}{\partial x}(g(\tilde{s})) &= 0, \\ \tilde{s}(0, t) &= \alpha(t) \end{aligned} \quad (1.5)$$

and

$$\begin{aligned} \frac{\partial s^*}{\partial t} - \frac{\partial}{\partial x}(a(s^*) \frac{\partial s^*}{\partial x}) &= 0, \\ s^*(0, t) &= \alpha(t), \\ \frac{\partial s^*}{\partial t} + \frac{\partial g(s^*)}{\partial x} &= 0 \quad \text{at } x = 1. \end{aligned} \quad (1.6)$$

*Mathematical Sciences Department, Rice University, Houston, Texas.

Here we approximate (1.5) using an explicit time-stepping higher order Godunov scheme, taking many (say M) small time steps Δt_s , which satisfy a CFL condition. Subsequently, a diffusive step (approximation solution for (1.6)) using an implicit mixed finite element or cell-centered finite differences is carried out with $\Delta t_B = M \Delta t_s$.

This paper is divided into three additional sections. In Section 2, we establish notation and formulate the above algorithm. Truncation errors for the scheme are also formally derived. Numerical results for the Buckley-Leverett problem with diffusion are presented in Section 3. In the last section extensions and conclusions are discussed.

2. Notation and Algorithm Formulation.

We first discuss the validity of splitting (1.1) into (1.4) and (1.5). Let $\Delta t_s > 0$ and $\Delta t = M \Delta t_s$, M a fixed positive integer. Denote by $S(\Delta t_B)$, $S_A(\Delta t_s)$, $S_D(\Delta t_B)$ the solution operators corresponding to (1.1), (1.4) and (1.5) respectively. Formally, by Taylor series we deduce that

$$(S(\Delta t_B) - S_D(\Delta t_B) \Pi S_A(\Delta t_s))s(x, t) = O((\Delta t_B)^2)$$

We have

$$\begin{aligned} S_A(\Delta t_s)s(x, t) &= S(x, t + \Delta t_s) \\ &= s(x, t) + \Delta t_s s_t + O((\Delta t_s)^2) \\ &= s(x, t) + \Delta t_s (-g(s)_x) + O((\Delta t_s)^2), \end{aligned}$$

and

$$\begin{aligned} S_A(\Delta t_s)S_A(\Delta t_s)s(x, t) &= s(x, t) + 2\Delta t_s(-g(s)_x) + O(\Delta t_s^2) \dots, \\ \left(\prod_{M \text{ times}} S_A(\Delta t_s) \right) s(x, t) &= s(x, t) - \Delta t_B g(s)_x + O(\Delta t_B)^2. \end{aligned}$$

Similarly,

$$S_D(\Delta t_B)s(x, t) = s(x, t) + \Delta t_B((as_x)_x) + O((\Delta t_B)^2).$$

Thus,

$$\begin{aligned}
S_D(\Delta t_B) & \stackrel{M \text{ times}}{\Pi} S_A(\Delta t_s) s(x, t) \\
& = s(x, t) + \Delta t_B (-g(s))_x + (a(s_x))_x + O(\Delta t_B^2) \\
& = S(\Delta t_B) s(x, t) + O(\Delta t_B^2).
\end{aligned}$$

We remark that this is only a “formal” derivation; we have assumed the solutions to (1.1), (1.4) and (1.5) are C^∞ . In addition, we note that the local error is second order; hence, the global error is first order, i.e. $O(\Delta t_B)$. For details regarding time-splitting for linear partial differential equations the reader is referred to LeVeque [5]; for nonlinear equations see Dawson [2].

We adopt the following notation. Let

$$\delta_x: 0 = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \dots < x_{N+\frac{1}{2}} = 1$$

be a partition of $[0, 1]$ with uniform spacing Δx . Set $x_j = \frac{(x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})}{2}$ and $B_j = (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$.

Define

$$M_0^r(\delta_x) = \{v \in C^0(I) \mid v \in P^r(B_j), j = 1, 2, \dots, N\}, \quad r \geq 1$$

where $P^r(E)$, E an interval, is the set of all polynomials of degree less than $r + 1$ defined on E .

In addition denote by

$$M_{-1}^r(\delta_x) = \{v \mid v \in P^r(B_j), j = 1, 2, \dots, N\}, \quad r \geq 0.$$

Let $(\cdot, \cdot)_j$ denote the usual L^2 inner product on B_j , i.e.

$$(\phi, \psi)_j = \int_{B_j} \phi(x) \psi(x) dx, \quad \phi, \psi \in L^2(B_j),$$

$$\text{and } (\phi, \psi) = \int_I \phi(x) \psi(x) dx.$$

Let $\Delta t_s > 0$ and satisfy the CFL condition $\Delta t_s \max(g'(s))/\Delta x \leq 1$. Set $\Delta t_B = M \Delta t_s$, M a positive integer. For a mesh function ϕ defined at the nodes x_j , let

$$\overline{\delta\phi_j} = \min \left\{ \frac{|\delta\phi_{j+1} + \delta\phi_j|}{2}, \delta_{\text{lim}}\phi_j \right\} \text{sgn}(\delta\phi_{j+1} + \delta\phi_j)$$

where

$$\delta \phi_j = \phi_j - \phi_{j-1}$$

and

$$\delta_{\text{lim}} \phi_j = \begin{cases} \alpha_l \min(|\delta \phi_{j+1}|, |\delta \phi_j|), & \delta \phi_{j+1} \delta \phi_j > 0 \\ 0, & \delta \phi_{j+1} \delta \phi_j < 0 \end{cases}$$

We now describe the Godunov mixed finite element algorithm.

- (1) *Initialization*: Let $n = 0$, $l = 0$ and define $Z^{0,0} \in M_{-1}^1(\delta_x)$ by

$$Z_j^{0,0} = \frac{1}{\Delta x} \int_{B_j} s_0(x) dx, \quad j = 1, 2, \dots, N,$$

and

$$Z^{0,0}(x) = Z_j^{0,0} + \frac{(x - x_j)}{\Delta x} \overline{\delta Z_j^{0,0}}, \quad x \in B_j.$$

- (2) *M Advection steps* (higher-order Godunov). For $n \geq 0$ and $l = 0, 1, \dots, M$ define

$Z^{n,l+1} \in M_{-1}^1(\delta_x)$ by

$$Z^{n,l+1}(x) = Z_j^{n,l+1} + \frac{(x - x_j)}{\Delta x} \overline{\delta Z_j^{n,l+1}} \quad (2a)$$

and

$$\begin{aligned} Z^{n,l+1} = & Z_j^{n,l} - \frac{\Delta t_s}{\Delta x} \left[g(Z_j^{n,l}) + \frac{1}{2} \left(1 - \frac{\Delta t_s}{\Delta x} g'(Z_j^{n,l}) \right) \overline{\delta Z_j^{n,l}} \right. \\ & \left. - g(Z_{j-1}^{n,l}) + \frac{1}{2} \left(1 - \frac{\Delta t_s}{\Delta x} g'(Z_{j-1}^{n,l}) \right) \overline{\delta Z_{j-1}^{n,l}} \right] \end{aligned} \quad (2b)$$

(The above approximation (2a)-(2b) is the higher order Godunov procedure with slope-limiting for a flux function $g' \geq 0$. See Bell and Shubin [1].

- (3) *Diffusion Step*. Define $(U^{n+1}, S^{n+1}) \in M_0^2(\delta_x) \times M_{-1}^1(\delta_x)$ by

$$\left(\frac{1}{\bar{a}(S^{n+1})} U^{n+1}, v \right) - (s^{n+1}, v) = \alpha_0(t^{n+1}) v(0) - \lambda^{n+1} v(1), \quad v \in M_0^2(\delta_x), \quad (3a)$$

and

$$\left(\frac{S^{n+1} - Z^{n,M}}{\Delta t_B}, w \right) + (U_x^{n+1}, w) = 0, \quad w \in M_{-1}^1(\delta_x) \quad (3b)$$

where $t^{n+1} = (n+1)\Delta t_B$ and

$$\begin{aligned} \lambda^{n+1} = 1.5 & \left[Z_N^{n,M} - \frac{1}{2}(1-g(Z^{n,M}))\frac{\Delta t_s}{\Delta x} \overline{\delta Z_N^{n,M}} \right] \\ & + .5 \left[Z_N^{n,M-1} - \frac{1}{2}(1-g(Z^{n,M-1}))\frac{\Delta t_s}{\Delta x} \overline{\delta Z_N^{n,M-1}} \right] \end{aligned} \quad (3c)$$

(Note the pair (U^{n+1}, S^{n+1}) are approximations to the diffusive flux and $s(\cdot, t^{n+1})$ respectively.)

- (4) set $Z^{n+1,0} = S^{n+1}$ and repeat (2) and (3) with $n = n+1$ until $n = NT$, final time.

In the mixed finite element formulation equation (3a) is derived by multiplying $u = -\bar{a}s_x$ by $\frac{1}{\bar{a}}v$, v a test function in $M_0^2(\delta_x)$, and integrating the result by parts. Equation (3b) is defined by multiplying (1.6) by $w \in M_{-1}^1(\delta_x)$ and integrating.

In step (3) we chose $S^{n+1} \in M_{-1}^1(\delta_x)$ in order to match the advection and diffusion spatial approximating spaces. One could however require $(U^{n+1}, S^{n+1}) \in M_0^1(\delta_x) \times M_{-1}^0(\delta_x)$; i.e. S^{n+1} lies in the space of piecewise discontinuous constants. In this case, the mixed finite element method would reduce to cell-centered finite difference provided the trapezoidal rule of integration is applied to $((1/\bar{a})U^{n+1}, v)_j$ (see Weiser and Wheeler [7]). However, in step (4) one would be required to modify the definition of $Z^{n+1,0}$ to include a step of slope limiting S^{n+1} . Error estimates for this scheme have been derived and will appear in [2].

We also wish to note that λ^{n+1} defined by (3c) involves an approximation to $s(1, t^{n+1})$ using the 'absorbing' boundary condition. We extrapolate the characteristic traces at $(1, t^n + (M + \frac{1}{2})\Delta t_s)$ and $(1, t^n + (M - 1)\Delta t_s)$.

3. Numerical Results.

In this section, we consider (1.1)-(1.4) with

$$g(s) = \frac{s^2}{[4(1-s)^2 + s^2]}, \quad (3.1)$$

$$a(s) = 5 \text{diff}[(1-s)^3 g(s) e^{5(1-s)} + 0.01] \quad (3.2)$$

$$\alpha(t) = 1, \quad t > 0 \quad (3.3)$$

and

$$s(x, 0) = 0, \quad x \in I. \quad (3.4)$$

Here, *diff* is a constant which is varied as 10^{-4} , 10^{-3} , and 10^{-2} .

The algorithm described in Section 2 was implemented using $M_0^2(\delta_x) \times M_{-1}^1(\delta_x)$ for the approximation of $(u = -as_x, s)$. In the Godunov procedure α_l was fixed at 1.5. The nonlinear system was solved by combining the method of substitution with a conjugate gradient procedure.

In Figure 1 the case $a \equiv 0$, no diffusion, is plotted at the time levels $t = t_1 = 0.085775$, $t = t_2 = 0.17155$, and $t = t_3 = 0.25732$. In Figures 2a-2c, 3a-3c, and 4a-4c, plots are exhibited in which $\Delta x^{-1} = 20, 40, 80$ and $\text{diff} = 10^{-4}, 10^{-3}$, and 10^{-2} , respectively. In all of the above cases, we solved for dispersion at every advection step.

In Figures 5 and 6a-6b, we give comparisons of the solution at two different time levels for different values of M , where $M =$ number of advection steps per dispersion step. In Figure 5, $\text{diff} = 10^{-4}$, and $M = 1, 4$, and 16. In Figure 6a, $\text{diff} = 10^{-3}$, and $M = 1, 4$ and 16. In Figure 6b, $\text{diff} = 10^{-3}$ and we show cases where $M = 1, 2$ and where M varies with time. In the latter case we started the simulation with $M = 1$. After four time steps we set $M = 2$, and later set $M = 4$, taking 17 dispersion steps in all, compared to the 48 diffusion steps when $M = 1$ and 24 when $M = 2$.

These figures demonstrate that taking multiple advection steps per dispersion step is a viable procedure. However, as one would expect, there is some correlation between the magnitude of the dispersion term and the choice of M , at least for this nonlinear $a(s)$. In particular, the larger the parameter *diff* is, the more often one needs to solve for dispersion.

In Figure 7 are plots for the Buckley-Leverett problem with the modification $a(s) = \text{diff}$ $= 10^{-3}$, with $m = 1, 4$, and 16. This figure would suggest that the discrepancies demonstrated between the different plots in Figure 6a are caused in part by the nonlinearity of $a(s)$, as given by Equation (3.2).

4 Conclusions and Extensions.

In this paper, we have treated advection-dominated diffusion problems in one space variable using time-splitting. Numerical experimentation indicates that this approach is viable in obtaining accurate numerical approximations to highly nonlinear and even degenerate parabolic problems. Extensions to two and three space variable problems and to problems with nonlinear fast varying reaction terms is straightforward. We are of course, assuming that the higher-order Godunov or some equivalent explicit advection scheme can be generalized to three space variables. In addition, for problems with a self-adjoint diffusion term, the implicit time step would involve solution of a symmetric system of nonlinear algebraic equations.

Also, of interest is that this time-splitting algorithm shows much promise for implementation on parallel computers. Advection and reaction can be treated locally and hence, in parallel. The diffusive step can be solved by coupling a preconditioned conjugate gradient method with domain decomposition. The latter involves breaking a domain into subdomains, solving the subdomain problems, and piecing these subproblems together. Efficient and robust examples of this approach for elliptic operations can be found in [3, 4].

References

1. J.B. Bell and G. R. Shubin, "High-Order Godunov Methods for Reservoir Simulation." EPR Special Report.
2. C.N. Dawson, "Error estimates for Godunov -- mixed methods for nonlinear parabolic equations," in preparation.
3. R. Gonzalez, "Domain decomposition for two-dimensional elliptic operators on vector and parallel machines," Ph.D. thesis, Rice University (1986).
4. J.E. Killough, "A multi-level domain decomposition algorithm suitable for the solution of three dimensional elliptic partial differential equations," Ph.D. thesis, Rice University (1986).
5. R.J. LeVeque, "Time split methods for partial differential equations," Ph.D. thesis, Stanford University (1982).
6. M. Nakata, A. Weiser, and M.F. Wheeler, "Some superconvergence results for mixed finite element methods for elliptic problems on rectangular domains," *MAFELAP Proceedings*, 1984.
7. A. Weiser and M.F. Wheeler, "On convergence of block-centered finite differences for elliptic problems," EPR Report TR-SR-84-14 (1984).

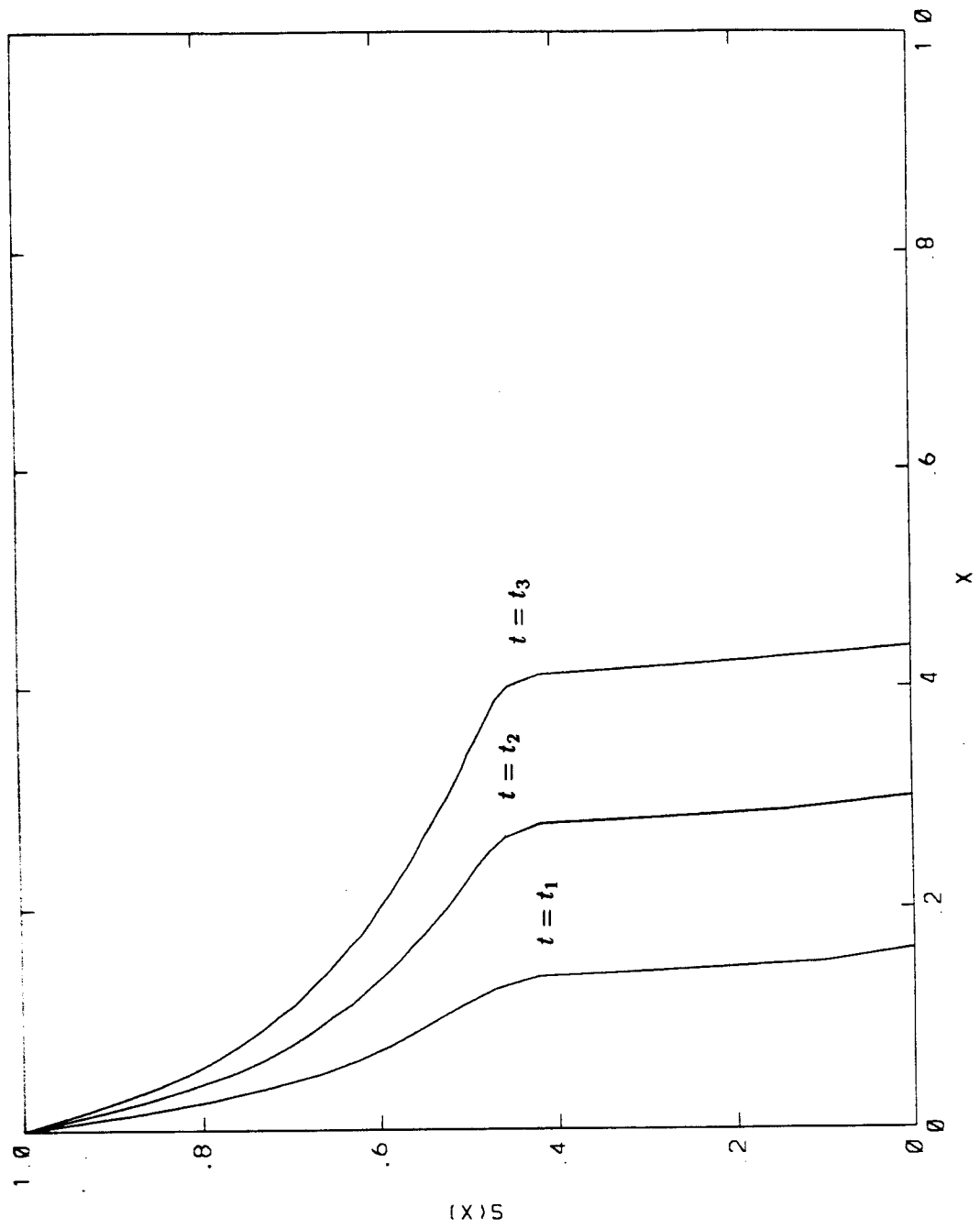


Figure 1. No diffusion, $\Delta x = 1/80$.

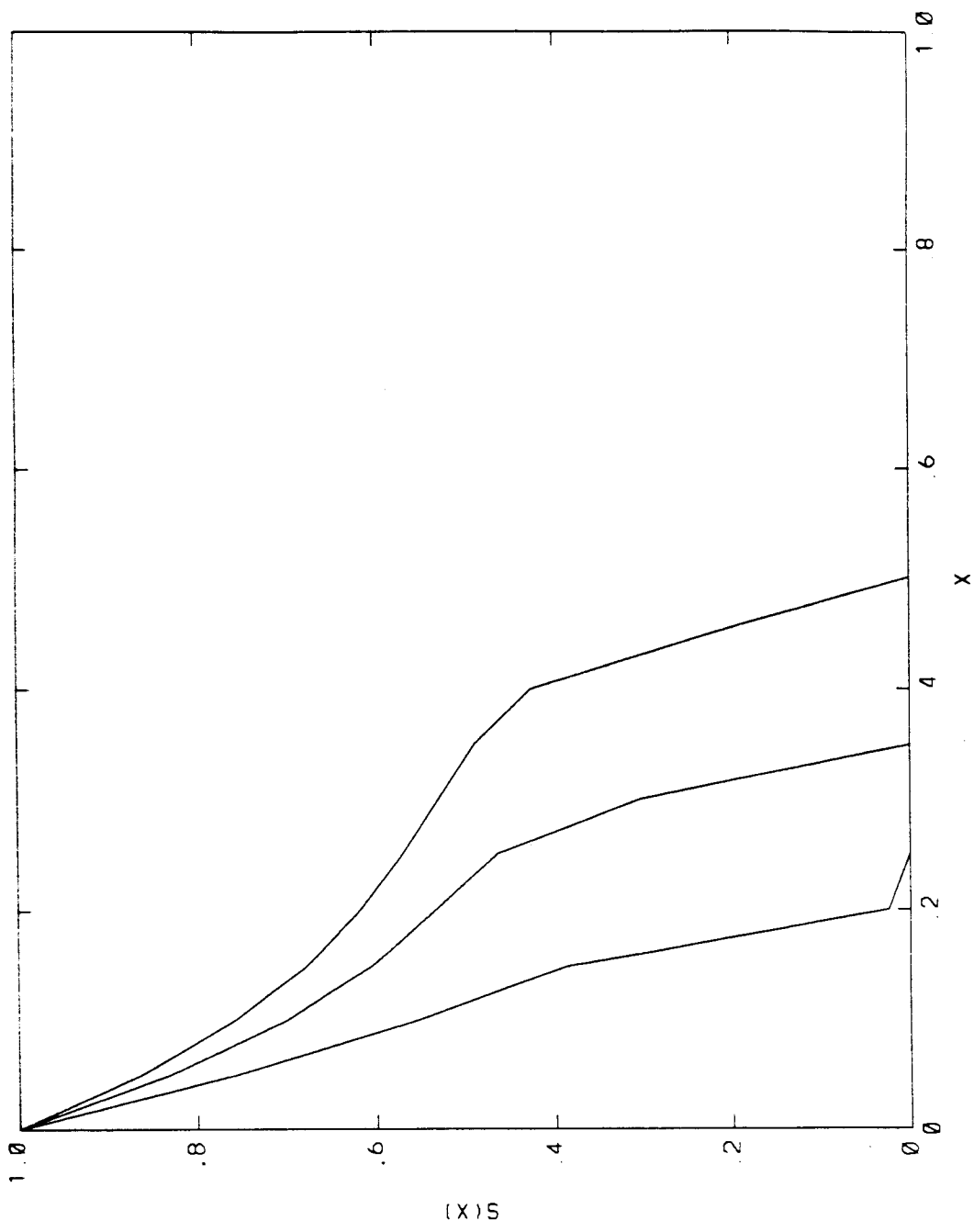


Figure 2a. $\Delta x^{-1} = 20$, $diff = 10^{-4}$, $M = 1$.

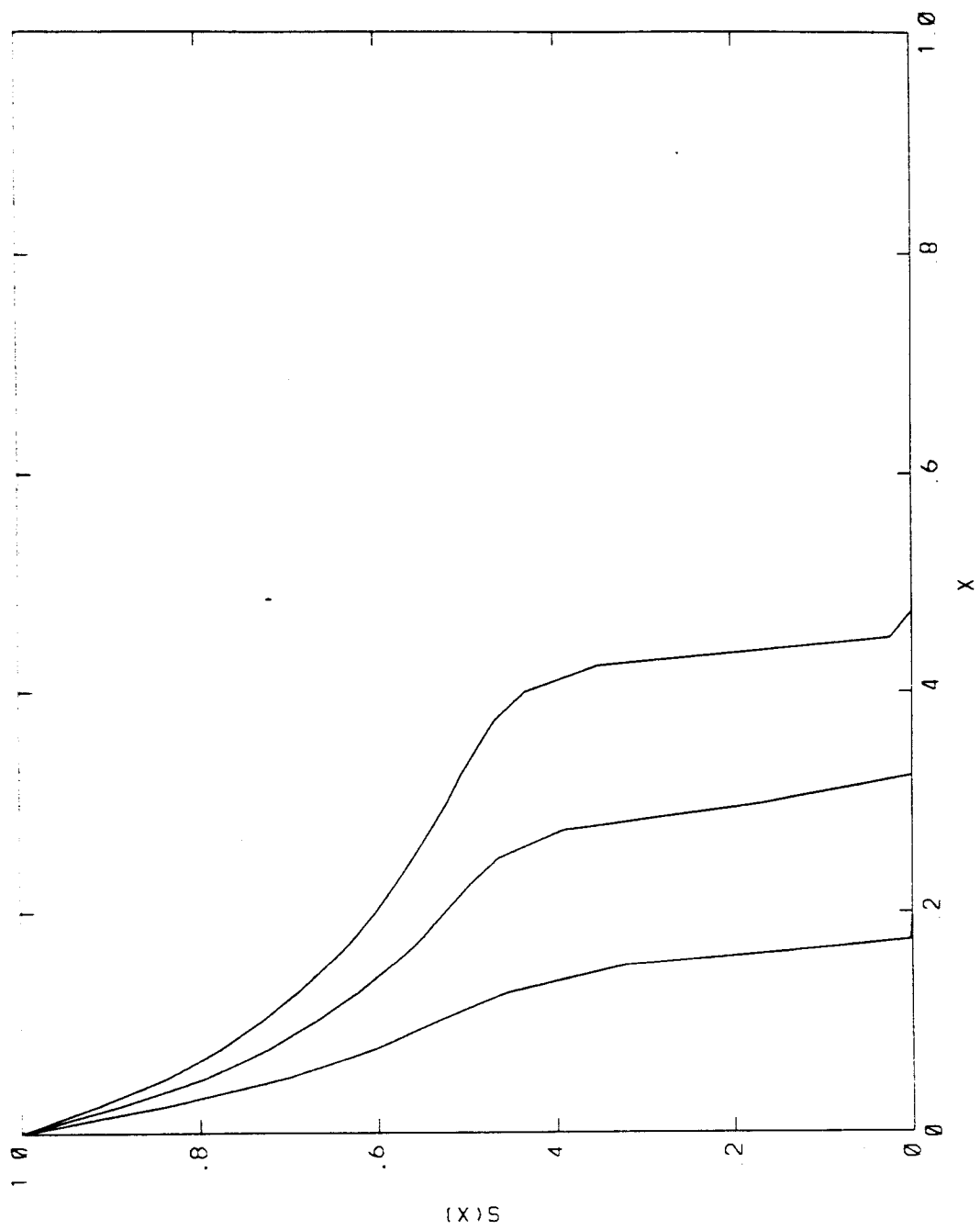


Figure 2b. $\Delta x^{-1} = 40$, $diff = 10^{-4}$, $M = 1$.

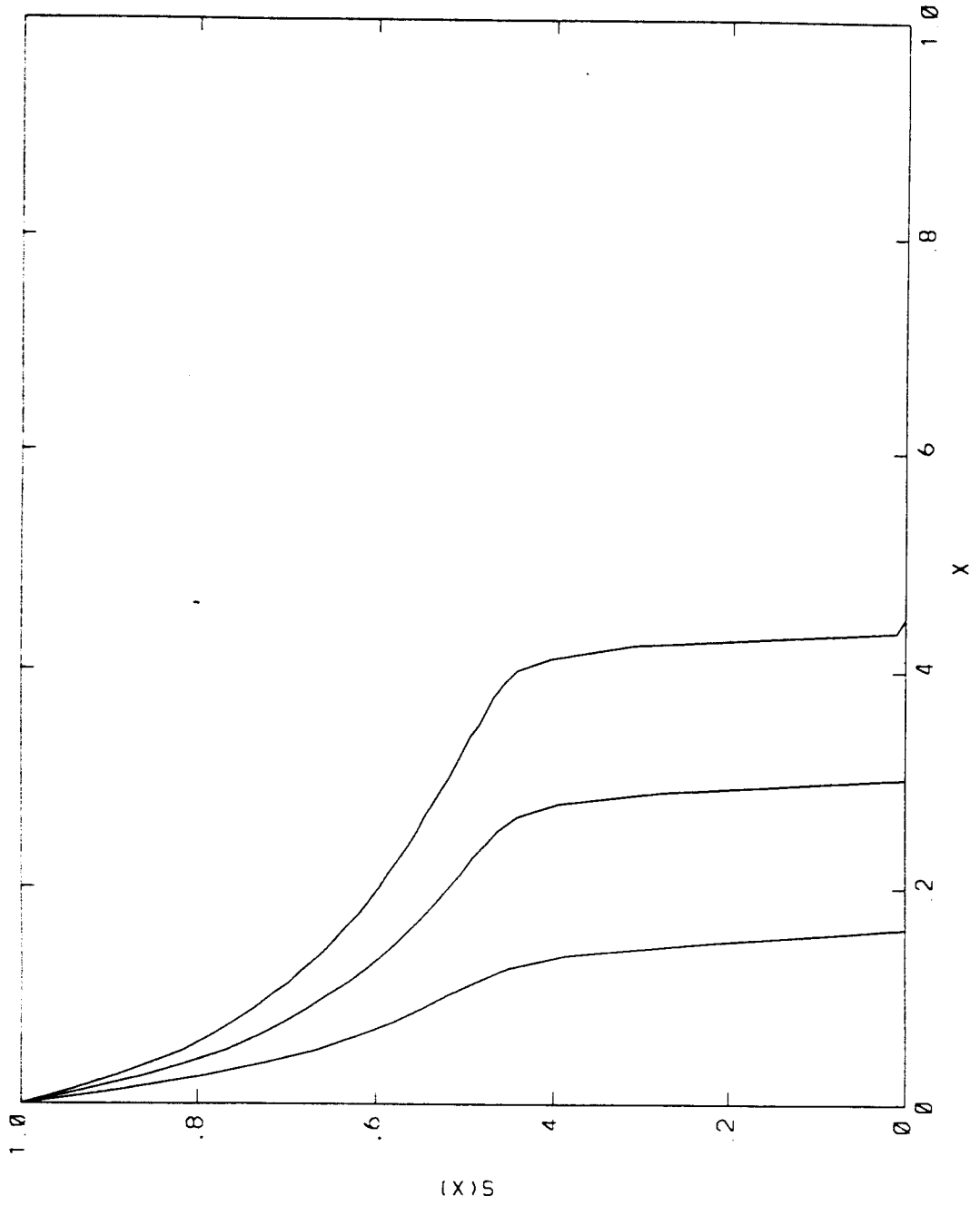


Figure 2c. $\Delta x^{-1} = 80$, $diff = 10^{-4}$, $M = 1$.

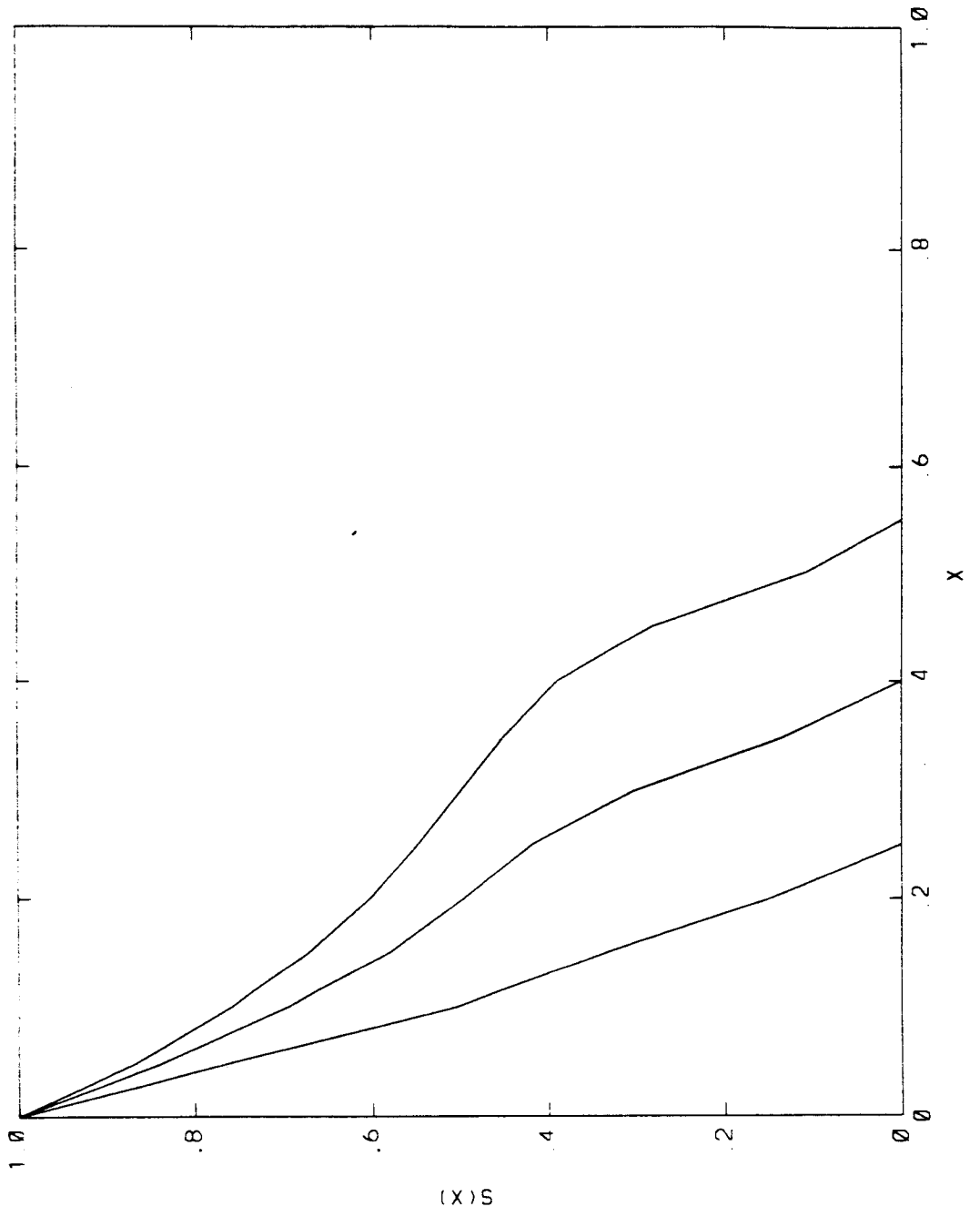


Figure 3a. $\Delta x^{-1} = 20$, $diff = 10^{-3}$, $M = 1$.

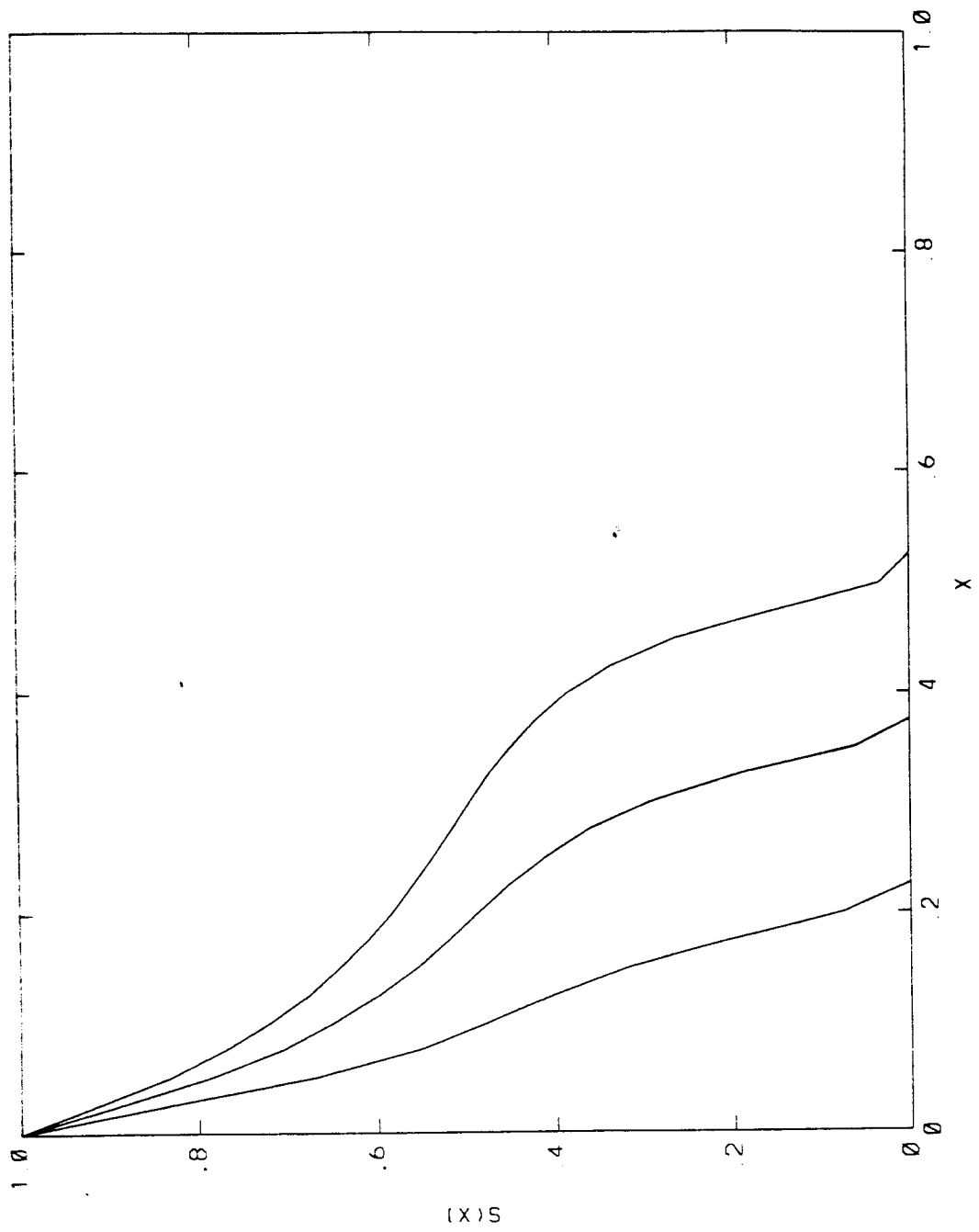


Figure 3b. $\Delta x^{-1} = 40$, $diff = 10^{-3}$, $M = 1$.

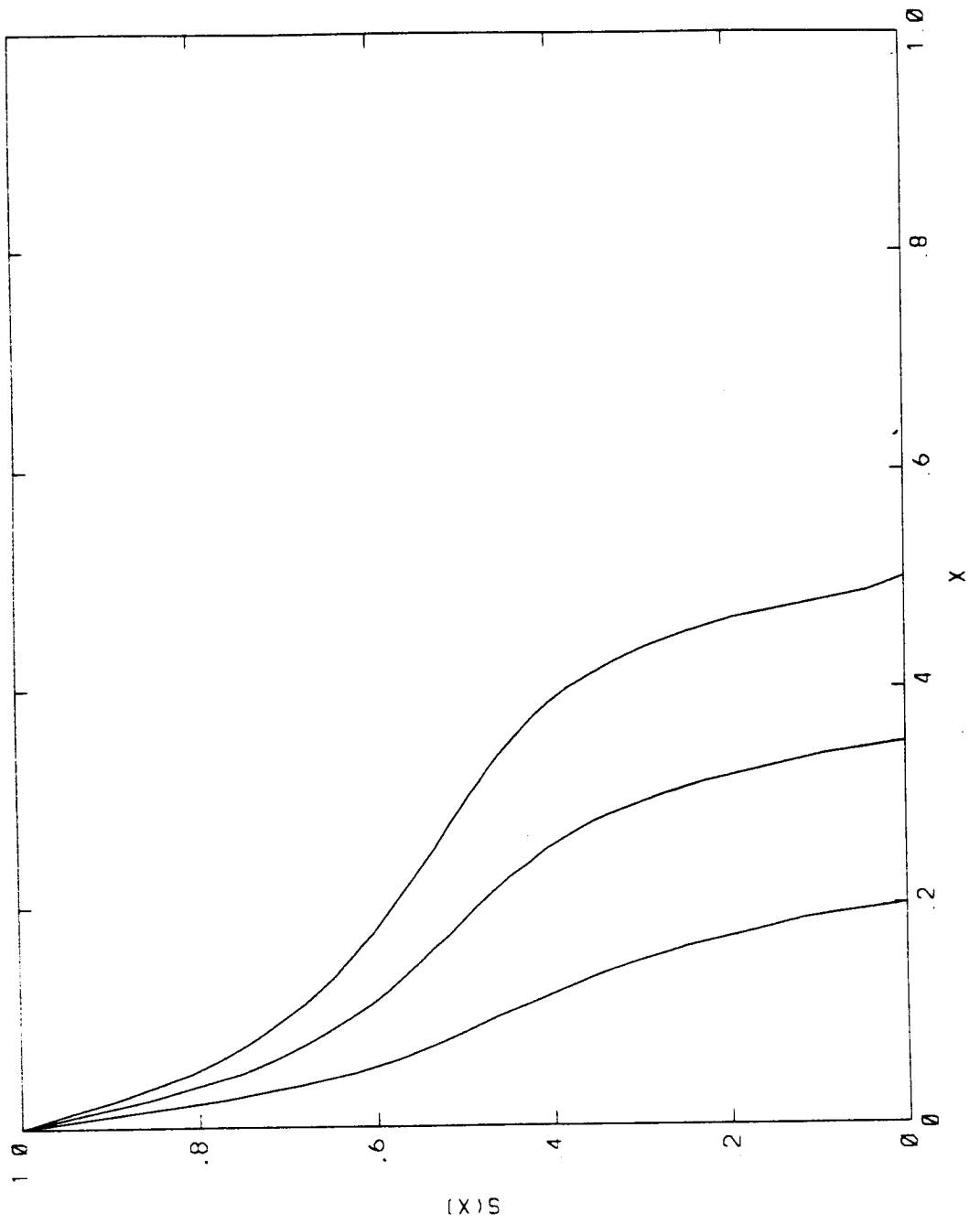


Figure 3c. $\Delta x^{-1} = 80$, $diff = 10^{-3}$, $M = 1$.

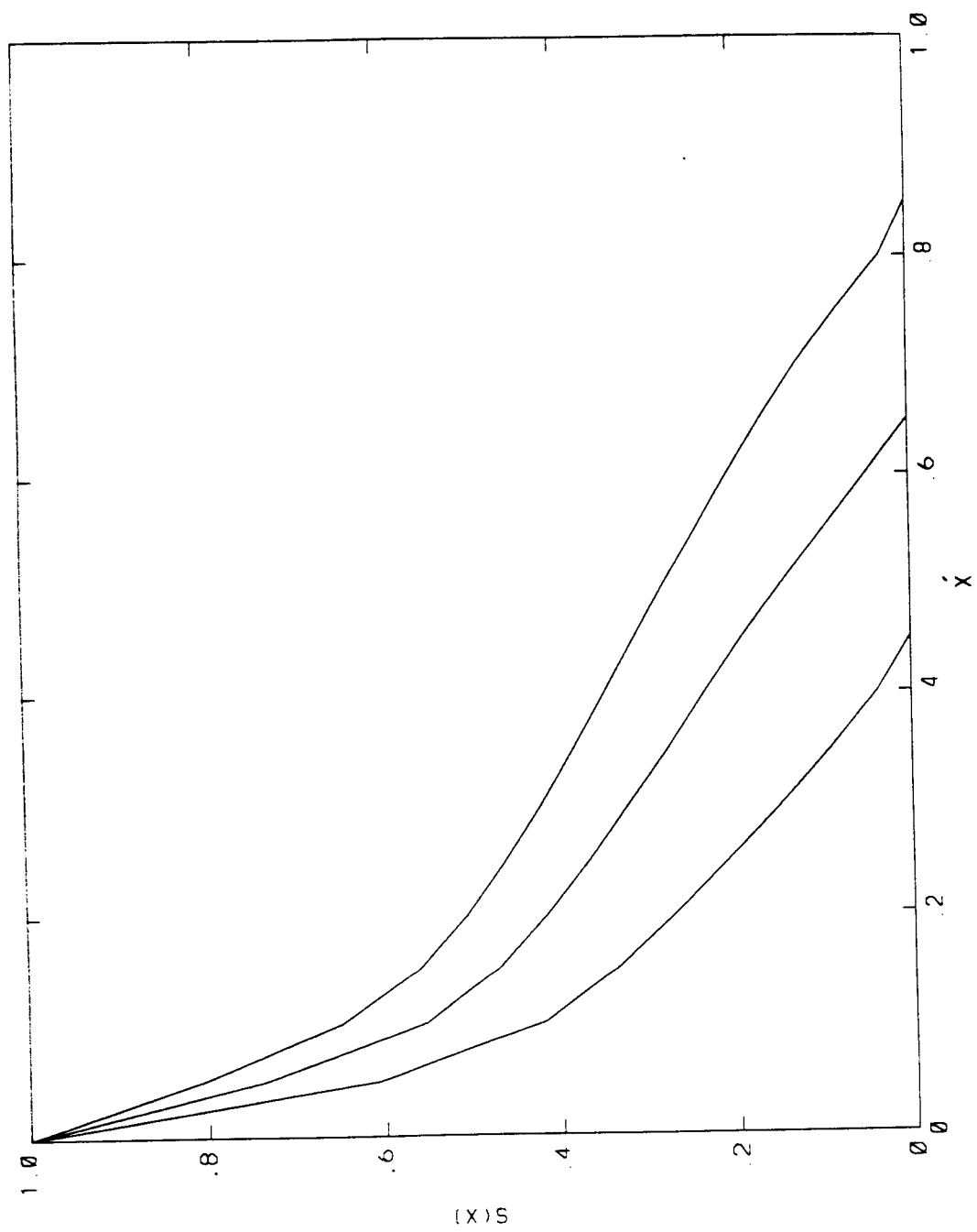


Figure 4a. $\Delta x^{-1} = 20$, $diff = 10^{-2}$, $M = 1$.

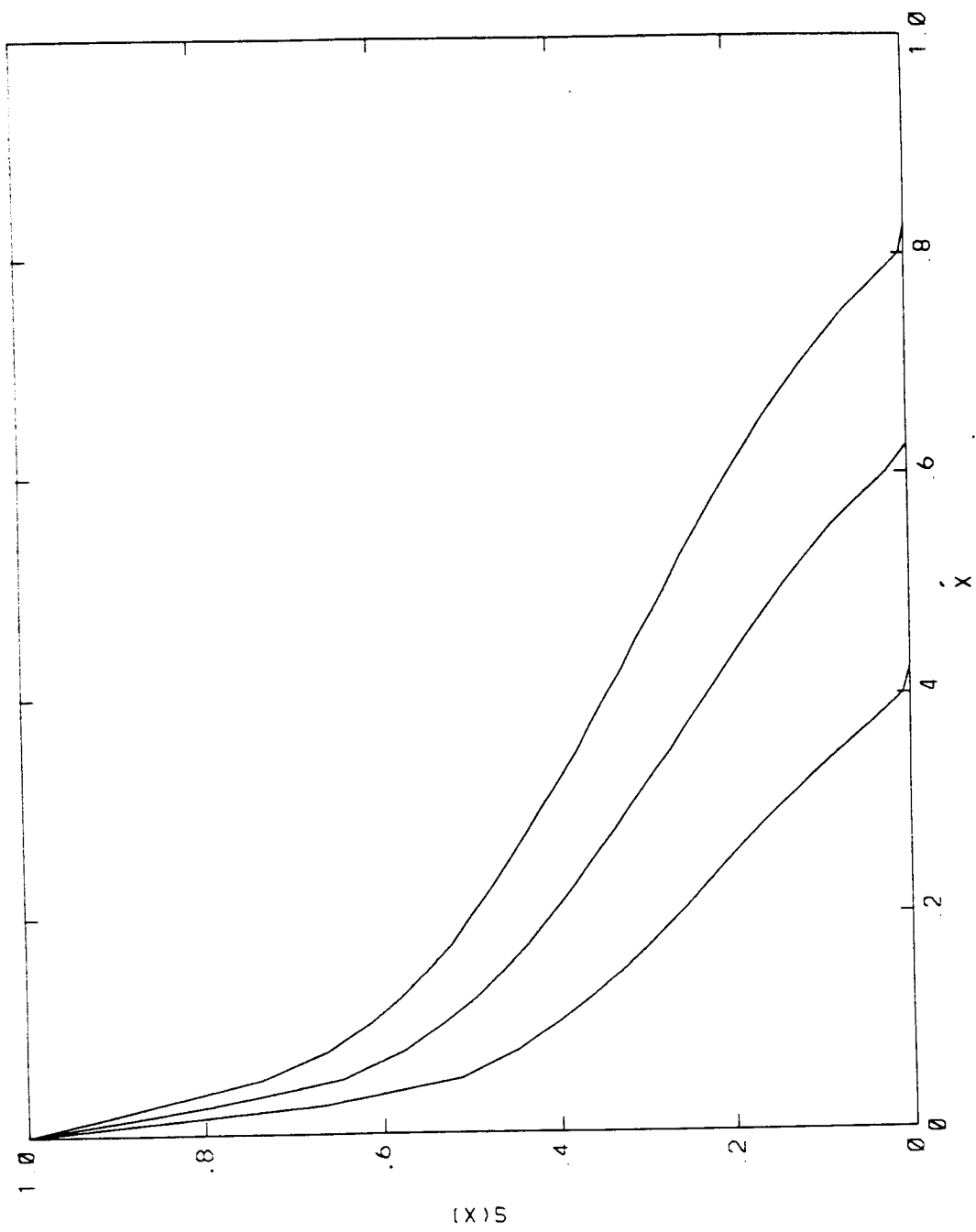


Figure 4b. $\Delta x^{-1} = 40$, $diff = 10^{-2}$, $M = 1$.

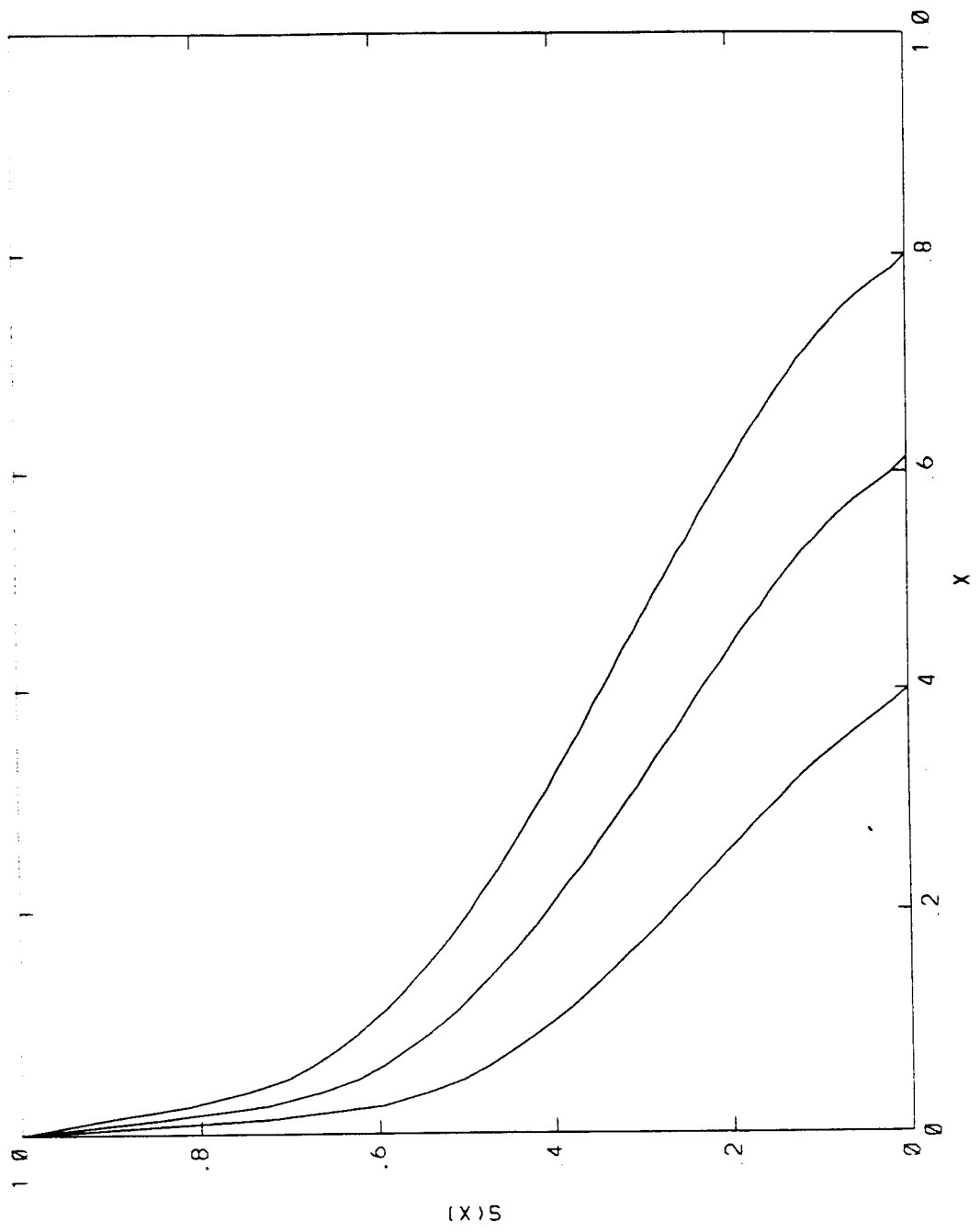


Figure 4c. $\Delta x^{-1} = 80$, $diff = 10^{-2}$, $M = 1$.

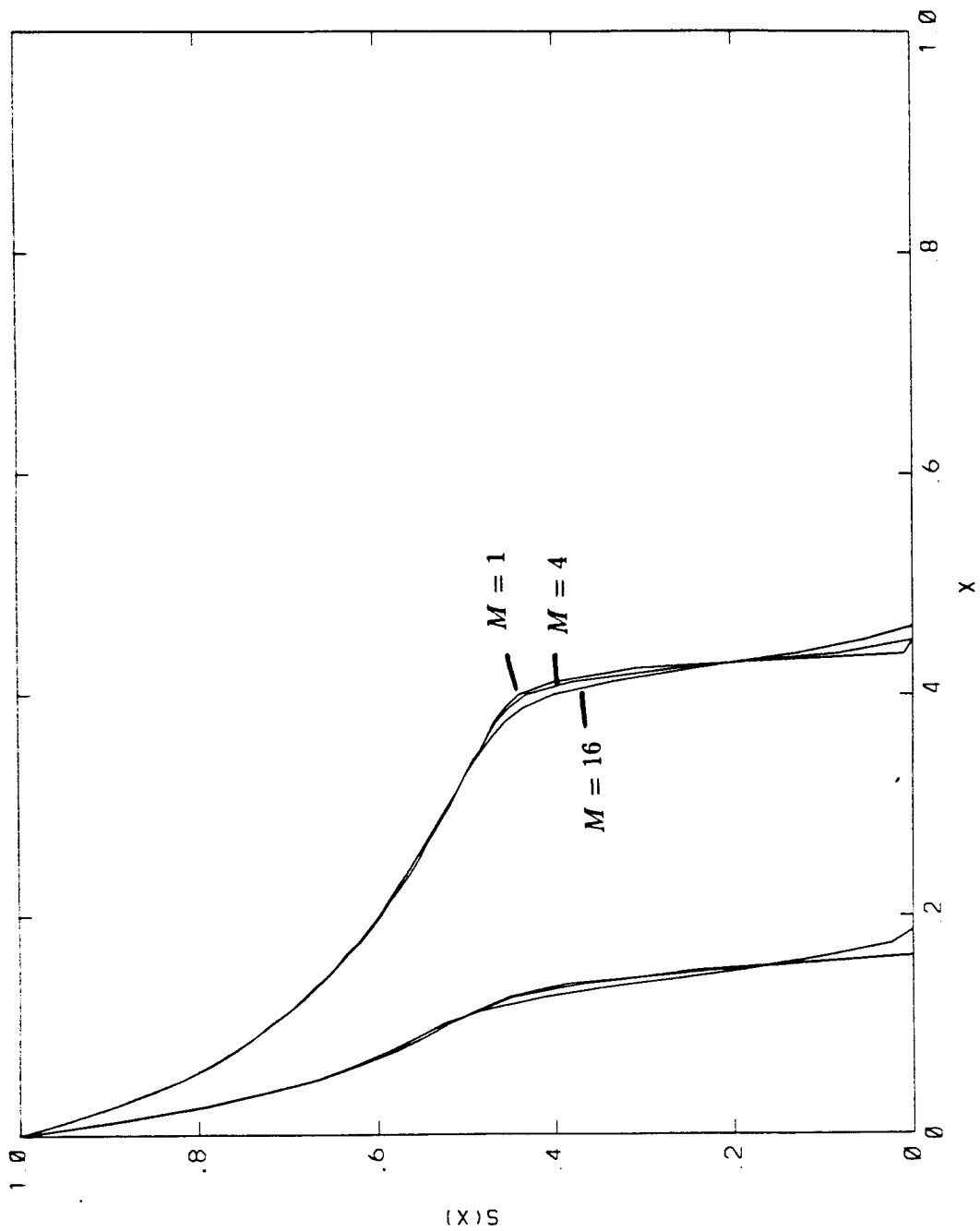


Figure 5. $\Delta x^{-1} = 80$, $diff = 10^{-4}$, $M = 1, 4, 16$.

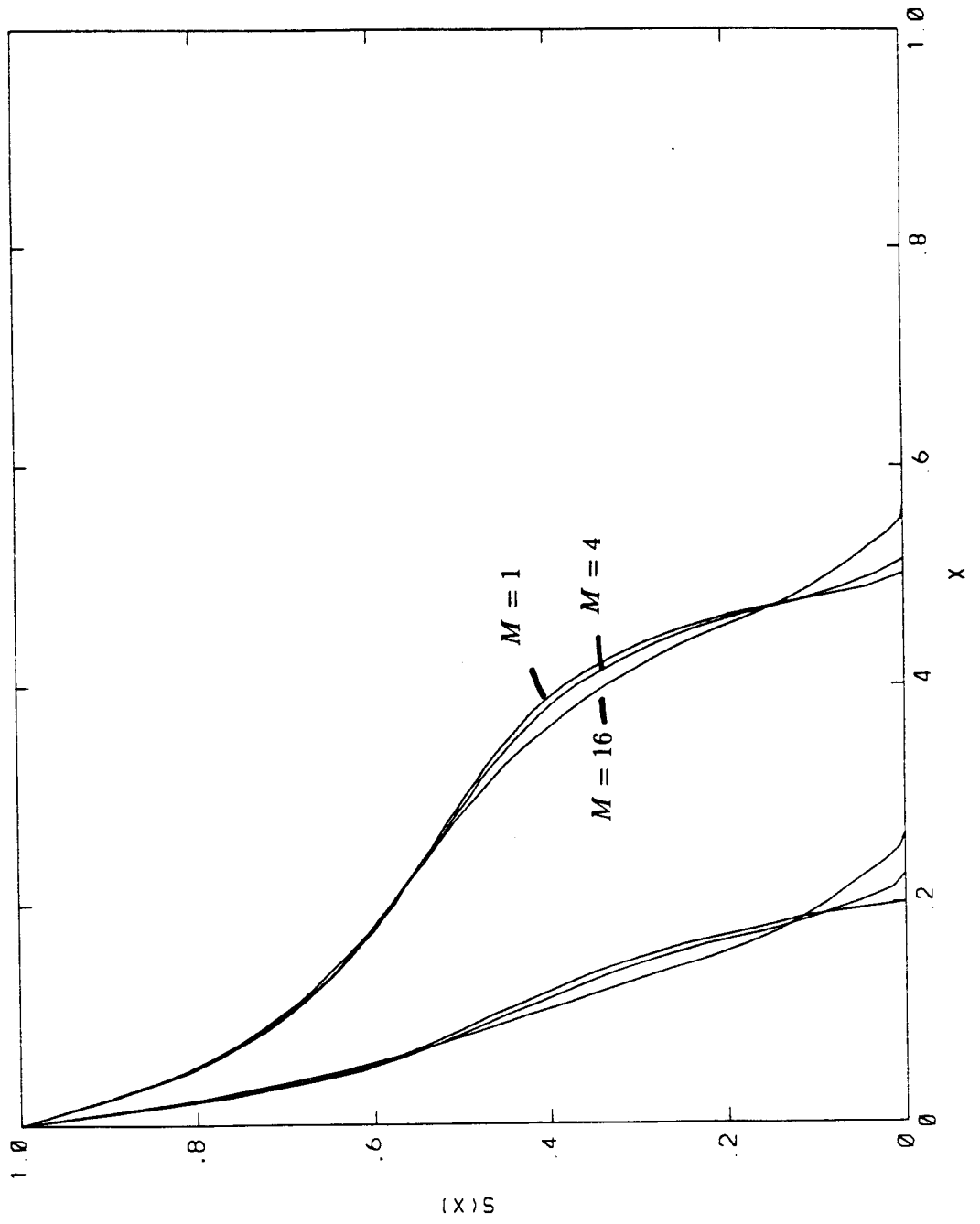


Figure 6a. $\Delta x^{-1} = 80$, $diff = 10^{-3}$, $M = 1, 4, 16$.

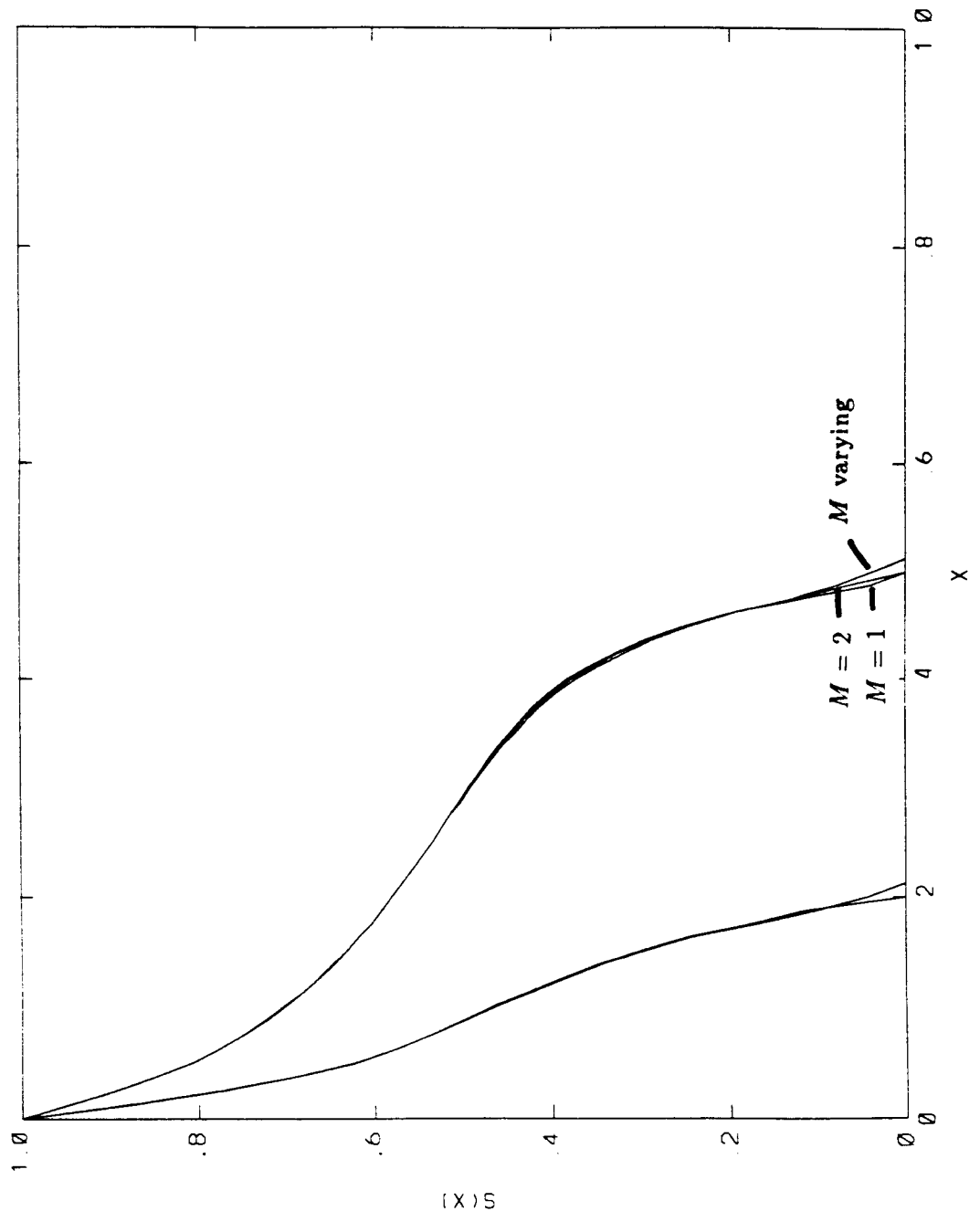


Figure 6b. $\Delta x^{-1} = 80$, $diff = 10^{-3}$, $M = 1, 2$, varying.

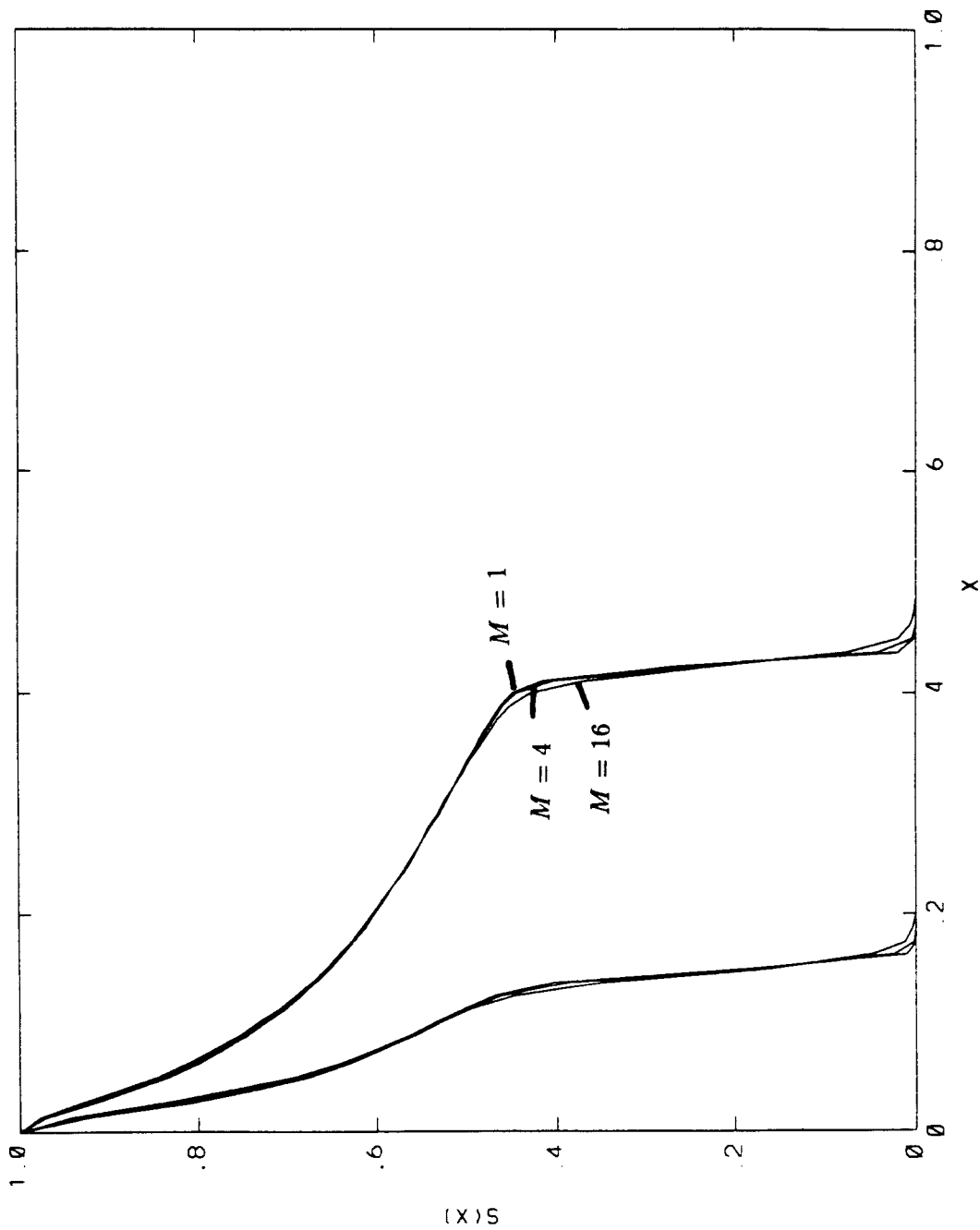


Figure 7. $\Delta x^{-1} = 80$, $a(s) \equiv 10^{-3}$, $M = 1, 4, 16$.