APPLIED AND COMPUTATIONAL HARMONIC ANALYSIS **5,** 450–486 (1998)
ARTICLE NO. HA980250

A Multiresolution Method for Numerical Reduction and Homogenization of Nonlinear ODEs

G. Beylkin[1]

Program in Applied Mathematics, University of Colorado at Boulder, Boulder, Colorado 80309-0526

M. E. Brewster[2]

Battelle Pacific Northwest Laboratories, P.O. Box 999, Richland, Washington 99352

and

A. C. Gilbert[3]

Department of Mathematics, Princeton University, Princeton, New Jersey 08544

Communicated by Leslie F. Greengard

Received July 31, 1997

The multiresolution analysis (MRA) strategy for the reduction of a nonlinear differential equation is a procedure for constructing an equation directly for the coarse scale component of the solution. The MRA homogenization process is a method for building a simpler equation whose solution has the same coarse behavior as the solution to a more complex equation. We present two multiresolution reduction methods for nonlinear differential equations: a numerical procedure and an analytic method. We also discuss one possible appproach to the homogenization method.

I. INTRODUCTION

There are many difficult, interesting, and important problems which incorporate multiple scales and which are prohibitively expensive to solve on the finest scales. In many problems of this kind it is sufficient to find the solution on a coarse scale only. However, we cannot disregard the fine scale contributions as the behavior of the solution on the coarse scale is affected by the fine scales. In these problems it is necessary to obtain a procedure for constructing the equations on a coarse scale that account for the contributions from these scales. This amounts to writing an effective equation for the coarse scale

[1] The research was partially supported by ARPA Grant F49620-93-1-0474 and ONR Grant N00014-91-J4037.
[2] The research was partially supported by ARPA Grant F49620-93-1-0474.
[3] The research was partially supported by AT&T Ph.D. Fellowship Program and Associated Western Universities.

1063-5203/98 $25.00

component of the solution which can be solved more economically. Alternatively, we might want to construct simpler fine scale equations whose solutions have the same coarse properties as the solutions of more complicated systems. These simpler equations would also be considerably less expensive to solve. These procedures are generally referred to as homogenization, though the specifics of the approaches vary significantly.

An example of a problem which encompasses many scales and which is difficult to solve on the finest scale is molecular dynamics. The highest frequency motion of a polymer chain under the fully coupled set of Newton's equations determines the largest stable integration time step for the system. In the context of long time dynamics the high frequency motions of the system are not of interest but current numerical methods (see [1, 17]) which directly access the low frequency motions of the polymer are *ad hoc* methods, not methods which take into account the effects of the high frequency behavior. The work of Bornemann and Schütte (see [16, 6]) is a notable exception and appears quite promising.

Let us briefly mention several classical approaches to homogenization. The classical theory of homogenization, developed in part by Bensoussan *et al.* [3], Jikov *et al.* [12], Murat [15], and Tartar [18], poses the problem as follows: Given a family of differential operators L_ϵ, indexed by a parameter ϵ, assume that the boundary value problem

$$L_\epsilon u_\epsilon = f \qquad \text{in } \Omega$$

(with u_ϵ subject to the appropriate boundary conditions) is well-posed in a Sobolev space H for all ϵ and that the solutions u_ϵ form a bounded subset of H so that there is a weak limit u_0 in H of the solutions u_ϵ. The small parameter ϵ might represent the relative magnitude of the fine and coarse scales. The problem of homogenization is to find the differential equation that u_0 satisfies and to construct the corresponding differential operator. We call the homogenized operator L_0 and the equation $L_0 u_0 = f$ in Ω the homogenized equation.

There are several methods for solving this problem. A standard technique is to expand the solution in powers of ϵ, to substitute the asymptotic series into the differential equations and associated boundary conditions, and then to recursively solve for the coefficients of the series given the first order approximation to the solution (see [14, 2, 13] for more details). If we consider a probabilistic interpretation of the solutions to elliptic or parabolic PDEs as averages of functionals of the trajectory of a diffusion process, then homogenization involves the weak limits of probability measures defined by a stochastic process [3]. In [12, 3], the methods of asymptotic expansions and of G-convergence are used to examine families of operators L_ϵ. Murat and Tartar (see [15, 18]) developed the method of compensated compactness. Coifman *et al.* (see [8]) have recently shown that there are intrinsic links between compensated compactness theory and the tools of classical harmonic analysis (such as Hardy spaces and operator estimates).

Using a multiresolution approach, Beylkin and Brewster [7] give a procedure for constructing an equation directly for the coarse scale component of the solution. This process is called reduction. From this effective equation one can determine a simpler equation for the original function with the same coarse scale behavior. Unlike the asymptotic approach for traditional homogenization, the reduction procedure in [7] consists of a reduction operator which takes an equation at one scale and constructs the effective equation at an adjacent scale (the next coarsest scale). This reduction operator

can be used recursively provided that the form of the equation is preserved under the transition. For systems of linear ordinary differential equations a step of the multiresolution reduction procedure consists of changing the coordinate system to split variables into averages and differences (in fact, quite literally in the case of the Haar basis), expressing the differences in terms of the averages, and eliminating the differences from the equations. For systems of linear ODEs there are relatively simple explicit expressions for the coefficients of the resulting reduced system. Because the system is organized so that the form of the equations is preserved, we may apply the reduction step recursively to obtain the reduced system over several scales.

M. Dorobantu [9] and A. Gilbert [10] apply the technique of MRA homogenization to the one-dimensional elliptic problem and derive results which relate the MRA approach to classical homogenization theory. A multiresolution approach to the reduction of elliptic PDEs and eigenvalue problems has been developed in [5]. It is shown in [5] that by choosing an appropriate MRA for a given problem, the small eigenvalues of the reduced operator differ only slightly from those of the original operator.

In this paper we consider a multiresolution strategy for the numerical reduction and homogenization of nonlinear equations. This strategy differs from the classical methods in that we do not require a distinguished parameter ϵ nor do we form an asymptotic expansion (or weak limit) in powers of ϵ. We demonstrate that the numerical reduction procedure can be applied to a small system of nonlinear ordinary differential equations. The main difficulty in performing a reduction step in the nonlinear case as compared to the linear case is that there are no explicit expressions for the differences in terms of the averages. We offer two basic approaches to address this problem. First, it appears possible not to require an analytic substitution for the differences and, instead, to rely on a numerical procedure. Second, we use a series expansion of the nonlinear functions in terms of a small parameter related to the discretization at a given scale (e.g., the step size of the discretization) and obtain analytic recurrence relations for the terms of the expansion. These recurrence relations allow us to reduce repeatedly. A third method is a hybrid of the two basic approaches.

In the first section we present a derivation of the reduction procedure for nonlinear ODEs and the series expansion of the recurrence relations. In the second section we discuss the implementation of the approaches to reduction. We discuss the homogenization procedure for nonlinear equations in the final section. We leave detailed discussions of the results to the appendices.

II. MRA REDUCTION METHODS

II.1. Linear Reduction Method

Let us briefly review the reduction method for linear systems of differential equations presented in [7]. Consider the differential equation

$$\frac{d}{dt}\left(G(t)x(t) + q(t)\right) = F(t)x(t) + p(t), \qquad t \in [0, 1],$$

where F and G are bounded matrix-valued functions and p and q are vector-valued functions (with elements in $L^2([0, 1])$). We will rewrite this differential equation as an integral equation

$$G(t)x(t) + q(t) - \beta = \int_0^t (F(s)x(s) + p(s))ds, \qquad t \in [0, 1] \tag{2.1}$$

(where β is a complex or real vector) since we can preserve the form of this equation under reduction, while we cannot preserve the form of the corresponding differential equation. To express this integral equation in terms of an operator equation on functions in $L^2([0, 1])$, let **F** and **G** be the operators whose actions on functions are pointwise multiplication by F and G and let **K** be the integral operator whose kernel K is

$$K(s, t) = \begin{cases} 1, & 0 \le s \le t \\ 0, & \text{otherwise.} \end{cases}$$

Then Eq. (2.1) can be rewritten as

$$\mathbf{G}x + q - \beta = \mathbf{K}(\mathbf{F}x + p).$$

We will use a general MRA of $L^2([0, 1])$ in this discussion. See Appendix B for definitions. We begin with an initial discretization of our integral equation by applying the projection operator P_n and looking for a solution x_n in V_n. This is equivalent to discretizing our problem at a very fine scale. We have

$$G_n x_n + q_n - \beta = K_n(F_n x_n + p_n), \tag{2.2}$$

where

$$G_n = P_n \mathbf{G} P_n^*, \quad F_n = P_n \mathbf{F} P_n^*, \quad K_n = P_n \mathbf{K} P_n^*, \quad p_n = P_n p, \quad \text{and} \quad q_n = P_n q.$$

We rewrite x_n in terms of its averages ($v_{n-1} \in V_{n-1}$) and differences ($w_{n-1} \in W_{n-1}$),

$$x_n = P_{n-1}x_n + Q_{n-1}x_n = v_{n-1} + w_{n-1},$$

and plug this into our Eq. (2.2):

$$G_n(v_{n-1} + w_{n-1}) + q_n - \beta = K_n(F_n(v_{n-1} + w_{n-1}) + p_n). \tag{2.3}$$

Next, we apply the operators P_{n-1} and Q_{n-1} to Eq. (2.3) to split it into two equations, one with values in V_{n-1} and the other with values in W_{n-1}, and we drop the subscripts:

$$\begin{aligned}(PGP^*)v + (PGQ^*)w + Pq = {} & PKP^*((PFP^*)v + (PFQ^*)w + Pp) \\ & + PKQ^*((QFP^*)v + (QFQ^*)w + Qp)\end{aligned}$$

$$\begin{aligned}(QGP^*)v + (QGQ^*)w + Qq = {} & QKP^*((PFP^*)v + (PFQ^*)w + Pp) \\ & + QKQ^*((QFP^*)v + (QFQ^*)w + Qp).\end{aligned}$$

Let us denote

$$T_{O,j} = P_j O_{j+1} P_j^*, \qquad C_{O,j} = P_j O_{j+1} Q_j^*$$
$$B_{O,j} = Q_j O_{j+1} P_j^*, \qquad A_{O,j} = Q_j O_{j+1} Q_j^*$$

(see [4] for a discussion of the non-standard form or representation of an operator O), so that we may simplify the linear system of equations in v and w. Then we obtain (again dropping the subscript $n-1$)

$$T_G v + C_G w + Pq - \beta = T_K(T_F v + C_F w + Pp) + C_K(B_F v + A_F w + Qp) \tag{2.4}$$

$$B_G v + A_G w + Qq = B_K(T_F v + C_F w + Pp) + A_K(B_F v + A_F w + Qp). \tag{2.5}$$

Let us assume that

$$R = A_G - B_K C_F - A_K A_F$$

is invertible so that we may solve Eq. (2.5) for w and plug the result into Eq. (2.4), giving us a reduced equation in V_{n-1} for v:

$$\begin{aligned}
&(T_G - C_K B_F - (C_G - C_K A_F) R^{-1}(B_G - B_K T_F - A_K B_F))v \\
&\quad + (Pq - C_K Qp - (C_G - C_K A_F) R^{-1}(Qq - B_K Pp - A_K Qp)) - \beta \\
&= T_K[(T_F - C_F R^{-1}(B_G - B_K T_F - A_K B_F))v + Pp - C_F R^{-1}(Qq - B_K Pp - A_K Qp)].
\end{aligned} \tag{2.6}$$

This equation for $v_{n-1} = P_{n-1} x_n$ exactly determines the averages of x_n. That is, we have an exact "effective" equation for the averages of x_n which contains the contribution from the fine scale behavior of x_n. Since we have a linear system and since we assumed that R is invertible, then we can solve Eq. (2.5) exactly for w and substitute the solution into Eq. (2.4). Note that this reduced equation has half as many unknowns as the original system. We call this procedure the reduction step.

Remark. There are differential equations for which $R = A_G - B_K C_F - A_K A_F$ is not invertible. An example of such an equation can be found in [7].

We should point out that under the reduction step the form of the original equations is preserved. Our Eq. (2.6) for v_{n-1} has the form

$$G_{n-1} v_{n-1} + q_{n-1} - \beta = K_{n-1}(F_{n-1} v_{n-1} + p_{n-1}),$$

where

$$\begin{aligned}
G_{n-1} &= T_G - C_K B_F - (C_G - C_K A_F) R^{-1}(B_G - B_K T_F - A_K B_F) \\
F_{n-1} &= T_F - C_F R^{-1}(B_G - B_K T_F - A_K B_F) \\
q_{n-1} &= Pq - C_K Qp - (C_G - C_K A_F) R^{-1}(Qq - B_K Pp - A_K Qp) \\
p_{n-1} &= Pp - C_F R^{-1}(Qq - B_K Pp - A_K Qp).
\end{aligned}$$

This procedure can be repeated up to n times use the recursion formulas:

$$F_j^{(n)} = T_{F,j} - C_{F,j}R_j^{-1}(B_{G,j} - B_{K,j}T_{F,j} - A_{K,j}B_{F,j}), \tag{2.7}$$

$$G_j^{(n)} = T_{G,j} - C_{K,j}B_{F,j} - (C_{G,j} - C_{K,j}A_{F,j})R_j^{-1}(B_{G,j} - B_{K,j}T_{F,j} - A_{K,j}B_{F,j}), \tag{2.8}$$

$$q_j^{(n)} = P_j q - C_{K,j}Q_j p - (C_{G,j} - C_{K,j}A_{F,j})R_j^{-1}(Q_j q - B_{K,j}P_j p - A_{K,j}Q_j p), \tag{2.9}$$

$$p_j^{(n)} = P_j p - C_{F,j}R_j^{-1}(Q_j q - B_{K,j}P_j p - A_{K,j}Q_j p). \tag{2.10}$$

The superscript (n) denotes the resolution level at which we started the reduction procedure and the subscript j denotes the current resolution level.

Let us summarize this discussion in the following proposition.

PROPOSITION II.1. *Suppose we have an equation for* $x_{j+1}^{(n)} = P_{j+1}x_n^{(n)}$ *in* V_{j+1},

$$G_{j+1}^{(n)}x_{j+1}^{(n)} + q_{j+1}^{(n)} - \beta = K_{j+1}(F_{j+1}^{(n)}x_{j+1}^{(n)} + p_{j+1}^{(n)}),$$

where the operator $R_j = A_{G,j} - B_{K,j}C_{F,j} - A_{K,j}A_{F,j}$ *is invertible. Then we can write an exact effective equation for* $x_j^{(n)} = P_j x_n^{(n)}$ *in* V_j,

$$G_j^{(n)}x_j^{(n)} + q_j^{(n)} - \beta = K_j(F_j^{(n)}x_j^{(n)} + p_j^{(n)}),$$

using the recursion relations (2.7)–(2.10).

Remark. We initialize the recursion relations with the values

$$G_n = P_n\mathbf{G}P_n^*, \quad F_n = P_n\mathbf{F}P_n^*, \quad K_n = P_n\mathbf{K}P_n^*, \quad p_n = P_n p, \quad \text{and} \quad q_n = P_n q,$$

where $\mathbf{G}$ and $\mathbf{F}$ are the operators whose actions on functions are pointwise multiplication by G and F, bounded matrix-valued functions with elements in $L^2([0, 1])$; $\mathbf{K}$ is the integration operator; and p and q are vector-valued functions with elements in $\mathrm{L}^2([0, 1])$.

Remark. This recursion process involves only the matrices $F_j^{(n)}$, $G_j^{(n)}$, and K_j and the vectors $p_j^{(n)}$ and $q_j^{(n)}$. In other words, we do not have to solve for x at any step in the reduction procedure.

If we apply the reduction procedure n times, we get an equation in V_0,

$$G_0^{(n)}x_0^{(n)} + q_0^{(n)} - \beta = \frac{1}{2}(F_0^{(n)}x_0^{(n)} + p_0^{(n)}),$$

for the coarse scale behavior of $x_0^{(n)}$, which is an easily solved scalar equation. If we are interested in only this average behavior of x, then the reduction process gives us a way of determining the average of x exactly without having to solve the original equation for x and computing its average. This technique is very useful for complicated systems which are computationally expensive to resolve on the finest scale and which solutions we are interested in on only the coarsest scale.

II.2. Nonlinear Reduction Method

We turn now to nonlinear differential equations. Let us begin by highlighting the difficulty in the reduction procedure for nonlinear equations. The reduction procedure begins with a discretization of the nonlinear equation. We choose the Haar basis for illustrative purposes. Just as the initial discretization of a linear ODE is a linear algebraic system, the initial discretization of a nonlinear ODE is a nonlinear system

$$\mathcal{F}_n(x_n) = 0. \tag{2.11}$$

The nonlinear function $\mathcal{F}_n$ maps $\mathbf{R}^N$ to $\mathbf{R}^N$ (for $N = 2^n$) and we denote the kth coordinate of $\mathcal{F}_n(x_n)$ by $\mathcal{F}_n(x_n)(k)$. Similarly, we denote the kth coordinate of x_n by $x_n(k)$. We rewrite x_n in terms of its averages $P_{n-1}x_n = s_{n-1}$ and its differences $Q_{n-1}x_n = d_{n-1}$. We recall that for the Haar basis the action of the operators P_{n-1} and Q_{n-1} amounts to forming averages and differences of the odd and even elements of a vector (normalized by a factor of $\sqrt{2}$). We will modify the Haar basis slightly and normalize the differences by $1/\delta_n$, where $\delta_n = 2^{-n}$. The averages will not be adjusted by any factor. The averages s_{n-1} and the differences d_{n-1} are given in coordinate form by

$$s_{n-1}(k) = \frac{1}{2}\,(x_n(2k+1) + x_n(2k)) \qquad \text{and} \qquad d_{n-1}(k) = \frac{1}{\delta_n}\,(x_n(2k+1) - x_n(2k)).$$

We split our Eq. (2.11) into two equations in the two unknowns s_{n-1} and d_{n-1} by applying P_{n-1} and Q_{n-1} to Eq. (2.11). Our two equations are

$$P_{n-1}(\mathcal{F}_n(s_{n-1}, d_{n-1})) = 0 \tag{2.12}$$

$$Q_{n-1}(\mathcal{F}_n(s_{n-1}, d_{n-1})) = 0. \tag{2.13}$$

Notice that the function $P_{n-1}\mathcal{F}_n$ maps $\mathbf{R}^{N/2} \times \mathbf{R}^{N/2}$ to $\mathbf{R}^{N/2}$ and similarly for $Q_{n-1}\mathcal{F}_n$ but that we cannot split these functions into their actions on $P_{n-1}x_n = s_{n-1}$ and $P_{n-1}x_n = d_{n-1}$ (as we did in the linear case). Instead, we can give the coordinate values for $P_{n-1}\mathcal{F}_n$ and $Q_{n-1}\mathcal{F}_n$ (dropping subscripts),

$$(P\mathcal{F}(s, d))(k) = \frac{1}{2}\,(\mathcal{F}(s, d)(2k+1) + \mathcal{F}(s, d)(2k))$$

$$(Q\mathcal{F}(s, d))(k) = \frac{1}{\delta}\,(\mathcal{F}(s, d)(2k+1) - \mathcal{F}(s, d)(2k))$$

for $k = 0, \ldots, 2^{n-1} - 1$.

As with the linear algebraic system, we must eliminate the differences d from the nonlinear system (2.12)–(2.13). In other words, we must solve Eq. (2.13) for d as a function of s. This equation, however, is a nonlinear equation and may not be easily

solved (if at all). Let us assume that we can solve Eq. (2.13) for d as a function of s and let $\tilde{d}(s)$ denote the solution. We then plug $\tilde{d}(s)$ into Eq. (2.12) to get

$$P\mathcal{F}(s, \tilde{d}(s)) = 0$$

which is the reduced equation for the coarse behavior of x. The form of the original system is preserved under this procedure and we may write the recurrence relation for $\mathcal{F}_j$ as

$$\mathcal{F}_{j-1}(s) = P_{j-1}\mathcal{F}_j(s_{j-1}, \tilde{d}_{j-1}(s_{j-1})),$$

where $\tilde{d}_{j-1}(s_{j-1})$ satisfies $Q_{j-1}\mathcal{F}_j(s_{j-1}, \tilde{d}_{j-1}(s_{j-1})) = 0$ and $0 \leq j \leq n$.

In this subsection we will give the precise form of the nonlinear system (2.13)–(2.12) in d and s, state conditions for (2.13)–(2.12) under which we can solve for d as a function of s, develop two approaches for solving (2.13)–(2.12) for d (a numerical and an analytic approach), and derive formal recurrence relations for the nonlinear function $\mathcal{F}_j$.

We now extend the MRA reduction method to nonlinear ODEs of the form

$$x'(t) = F(t, x(t)), \qquad t \in [0, 1]. \tag{2.14}$$

We will address the difficulties raised in the previous discussion with two approaches, a formal method to be implemented numerically and an asymptotic method. We will assume that F is differentiable as a function of x and as a function of t. The assumption that F is Lipschitz as a function of x guarantees the existence of uniqueness of the solution $x(t)$. For the reduction procedure F must be Lipschitz in t and differentiable in x. We will rewrite this differential equation as an integral equation in a slightly unusual form,

$$G(t, x(t)) - G(0, x(0)) = \int_0^t F(s, x(s))ds, \tag{2.15}$$

where $\partial G/\partial x \neq 0$. The more usual differential equation (2.14) is obtained by setting $G(t, x(t)) = x(t)$ and by differentiating. We choose this integral formulation because we can maintain this form under the reduction procedure.

In our derivations we find it helpful to use an operator notation in addition to the coordinate notation so we write Eq. (2.15) in an operator form,

$$\mathbf{G}(x) = \mathbf{K}(\mathbf{F}(x)), \tag{2.16}$$

where

$$\mathbf{K}(y)(t) = \int_0^t y(s)ds, \qquad \mathbf{G}(y)(t) = G(t, y(t)), \qquad \text{and} \qquad \mathbf{F}(y)(t) = F(t, y(t)).$$

We will use the MRA of $L^2([0, 1])$ associated with the Haar basis to begin our discretization. We discretize Eq. (2.16) in t by applying the projection operator P_n to Eq. (2.16) and seeking a solution $x_n \in V_n$ to the equation

$$G_n(x_n) = K_n F_n(x_n), \tag{2.17}$$

where

$$G_n(x_n) = P_n \mathbf{G}(x_n), \qquad K_n = P_n \mathbf{K} P_n^*, \qquad \text{and} \qquad F_n(x_n) = P_n \mathbf{F}(x_n).$$

Because we are using the Haar basis, x_n is a piecewise constant function with step width $\delta_n = 2^{-n}$. The functions $G_n(x_n)$ and $F_n(x_n)$ are also piecewise constant functions. Note that G_n, F_n, and K_n map V_n to V_n, although G_n and F_n are nonlinear functions. Let $x_n(k)$ denote the value of the function x_n on the interval $k\delta_n < t < (k + 1)\delta_n$, for $k = 0, \ldots, 2^n - 1$. Let $g_n(x_n)(k)$ and $f_n(x_n)(k)$ denote the values of the functions $G_n(x_n)$ and $F_n(x_n)$ on the same interval. That is,

$$g_n(x_n)(k) = \frac{1}{\delta_n} \int_{k\delta_n}^{(k+1)\delta_n} g(s, x_n(k))ds = (P_n \mathbf{G}(x_n))(t),$$

where $k\delta_n < t < (k + 1)\delta_n$, and similarly for $f_n(x)(k)$. We can say that $g_n(x_n)(k)$ is the average value of the function $G(t, \cdot)$ over the time interval $(k\delta_n, (k + 1)\delta_n)$ and evaluated at $x_n(k)$. Notice that $g_n(x_n)(k)$ is shorthand for $g_n(x_n(k))(k)$.

As in [7] we use the integration operator K_n defined by

$$K_n = \delta_n \begin{pmatrix} \frac{1}{2} & 0 & \cdots & 0 \\ 1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 1 & \cdots & 1 & \frac{1}{2} \end{pmatrix}. \tag{2.18}$$

With this notation, the coordinate form of Eq. (2.17) is

$$g_n(x_n)(k) = \delta_n \sum_{k'=0}^{k-1} f_n(x_n)(k') + \frac{\delta_n}{2} f_n(x_n)(k). \tag{2.19}$$

This equation gives the precise form of the nonlinear system $\mathcal{F}(x) = 0$ discussed previously. We are now ready to begin the reduction procedure.

We first split the Eq. (2.17) into two equations, one with values in V_{n-1} and the other with values in W_{n-1}, by applying the projection operators P_{n-1} and Q_{n-1}. We now have the two equations

$$P_{n-1} G_n(x_n) = P_{n-1} K_n(F_n(x_n)) \tag{2.20}$$

$$Q_{n-1} G_n(x_n) = Q_{n-1} K_n(F_n(x_n)). \tag{2.21}$$

At this point let us work with two consecutive levels and drop the index n indicating the multiresolution level (assume that $\delta = \delta_n$). We again modify the Haar basis slightly and normalize the differences by $1/\delta$. The averages will not be adjusted by any factor. By forming successive averages of Eq. (2.19), we can rewrite Eq. (2.20) in coordinate form as

$$\frac{1}{2}\,(g(x)(2k+1) + g(x)(2k)) = \frac{\delta}{2}\sum_{k'=0}^{2k} f(x)(k') + \frac{\delta}{4} f(x)(2k+1) + \frac{\delta}{2}\sum_{k'=0}^{2k-1} f(x)(k') + \frac{\delta}{4} f(x)(2k). \quad (2.22)$$

In the same manner we rewrite Eq. (2.21) by taking successive differences normalized by the step size δ:

$$\frac{1}{\delta}\,(g(x)(2k+1) - g(x)(2k)) = \frac{1}{2}\,(f(x)(2k+1) + f(x)(2k)). \quad (2.23)$$

Let us rearrange the right-hand side of Eq. (2.22) as

$$\begin{aligned}
&\frac{\delta}{2}\sum_{k'=0}^{2k} f(x)(k') + \frac{\delta}{4} f(x)(2k+1) + \frac{\delta}{2}\sum_{k'=0}^{2k-1} f(x)(k') + \frac{\delta}{4} f(x)(2k) \\
&\quad = \delta \sum_{k'=0}^{2k-1} f(x)(k') + \frac{\delta}{4} f(x)(2k+1) + \frac{3\delta}{4} f(x)(2k) \\
&\quad = \delta \sum_{k'=0}^{k-1} (f(x)(2k'+1) + f(x)(2k')) + \frac{\delta}{2}\,(f(x)(2k+1) + f(x)(2k)) \\
&\qquad - \frac{\delta}{4}\,(f(x)(2k+1) - f(x)(2k)).
\end{aligned}$$

To simplify our notation, let us define **S** and **D** as "average" and "difference" operators which act on $g(x)$ and $f(x)$ by taking successive averages and differences of elements $g(x)(k)$ and $f(x)(k)$. We define **S** and **D** as

$$\mathbf{S}g(x)(k) = \frac{1}{2}\,(g(x)(2k+1) + g(x)(2k))$$

$$\mathbf{D}g(x)(k) = \frac{1}{\delta}\,(g(x)(2k+1) - g(x)(2k)).$$

Then we may write the coordinate form of Eqs. (2.20)–(2.21) in a compact form

$$\mathbf{S}g(x)(k) + \frac{\delta^2}{4}\mathbf{D}f(x)(k) = 2\delta \sum_{k'=0}^{k-1} \mathbf{S}f(x)(k') + \delta \mathbf{S}f(x)(k) \tag{2.24}$$

$$\mathbf{D}g(x)(k) = \mathbf{S}f(x)(k). \tag{2.25}$$

We have split Eq. (2.19) into two sets and now we split the variables accordingly. We define the averages s_{n-1} and the scaled differences d_{n-1} as

$$s_{n-1}(k) = \frac{1}{2}\left(x_n(2k+1) + x_n(2k)\right) \qquad \text{and} \qquad d_{n-1}(k) = \frac{1}{\delta}\left(x_n(2k+1) - x_n(2k)\right).$$

Notice that since x_n is a piecewise constant function with step width δ_n, then s_{n-1} and d_{n-1} are piecewise constant functions with step width $2\delta_n = \delta_{n-1}$. We will now change variables in Eqs. (2.24) and (2.25) and replace x with

$$x(2k+1) = s(k) + \frac{\delta}{2}d(k) \qquad \text{and} \qquad x(2k) = s(k) - \frac{\delta}{2}d(k).$$

We will abuse our own notation slightly for clarity and denote the change of variables by

$$\mathbf{S}g(s, d)(k) = \frac{1}{2}\left(g\left(s + \frac{\delta}{2}d\right)(2k+1) + g\left(s - \frac{\delta}{2}d\right)(2k)\right)$$

$$\mathbf{D}g(s, d)(k) = \frac{1}{\delta}\left(g\left(s + \frac{\delta}{2}d\right)(2k+1) - g\left(s - \frac{\delta}{2}d\right)(2k)\right).$$

Note that when we write $g(x)(k)$, this is shorthand for $g(x(k))(k)$; so $g(x)(2k+1)$ stands for $g(x(2k+1))(2k+1)$. When we replace $x(2k+1)$ with $s(k) + \frac{\delta}{2}d(k)$ and write $g(x)(2k+1) = g(s + \frac{\delta}{2}d)(2k+1)$, this is shorthand for the expression

$$g(x(2k+1))(2k+1) = g\left(s(k) + \frac{\delta}{2}d(k)\right)(2k+1).$$

The shorthand notation $g(s - \frac{\delta}{2}d)(2k)$ is similar. Then our system of two equations in the two variables s and d is given by

$$\mathbf{S}g(s, d)(k) + \frac{\delta^2}{4}\mathbf{D}f(s, d)(k) = 2\delta \sum_{k'=0}^{k-1} \mathbf{S}f(s, d)(k') + \delta \mathbf{S}f(s, d)(k) \tag{2.26}$$

$$\mathbf{D}g(s, d)(k) = \mathbf{S}f(s, d)(k). \tag{2.27}$$

Our goal, as in the linear case, is to eliminate the variables d from Eqs. (2.26)–(2.27) to obtain a single equation for s. We consider (2.27) as an equation for d which we have

to solve in order to find d in terms of s. Let us assume that we can solve (2.27) for d and let $\tilde{d}$ represent this solution. Notice that Eq. (2.27) is a nonlinear equation for d so that $\tilde{d}$ is a nonlinear function of s. We will discuss how this is implemented numerically in Section III and how this is implemented analytically in Subsection II.3. In the linear case, $\tilde{d}$ is a linear function of s and it can be easily computed explicitly. Provided that we have $\tilde{d}$, we substitute this into Eq. (2.26) and obtain

$$\mathbf{S}g(s, \tilde{d})(k) + \frac{\delta^2}{4}\mathbf{D}f(s, \tilde{d})(k) = 2\delta \sum_{k'=0}^{k-1} \mathbf{S}f(s, \tilde{d})(k') + \delta \mathbf{S}f(s, \tilde{d})(k). \tag{2.28}$$

Observe that we may arrange Eq. (2.28) as

$$g_{n-1}(k)(s_{n-1}) = \delta_{n-1} \sum_{k'=0}^{k-1} f_{n-1}(k')(s_{n-1}) + \frac{\delta_{n-1}}{2} f_{n-1}(k)(s_{n-1}), \tag{2.29}$$

where

$$g_{n-1}(k)(s_{n-1}) = \mathbf{S}g_n(k)(s_{n-1}, \tilde{d}_{n-1}) + \frac{\delta_n^2}{4}\mathbf{D}f_n(k)(s_{n-1}, \tilde{d}_{n-1}) \tag{2.30}$$

and

$$f_{n-1}(k)(s_{n-1}) = \mathbf{S}f_n(k)(s_{n-1}, \tilde{d}_{n-1}). \tag{2.31}$$

In other words, the reduced equation (2.29) is the effective equation for the averages s_{n-1} of x_n. It is important to note that this equation has the same form as the original discretization.

Let us switch now to operator notation to present the recurrence relations for the reduction procedure. We use the solution $\tilde{d}$ of Eq. (2.27) to write Eq. (2.29) in operator form as

$$G_{n-1}^{(n)}(s_{n-1}) = K_{n-1}F_{n-1}^{(n)}(s_{n-1}),$$

where $s_{n-1} = P_{n-1}x$ and the nonlinear operators $G_{n-1}^{(n)}$ and $F_{n-1}^{(n)}$ map V_{n-1} to V_{n-1}. The superscript (n) on the operators denotes the level at which we start the reduction procedure and the subscript $n-1$ denotes the current level of resolution. The operators $G_{n-1}^{(n)}$ and $F_{n-1}^{(n)}$ are defined as the operators which act elementwise according to Eqs. (2.30) and (2.31), respectively. Notice that they have the same form as the operators $G_n^{(n)}$ and $F_n^{(n)}$; both functions $G_{n-1}^{(n)}(s_{n-1})$ and $F_{n-1}^{(n)}(s_{n-1})$ are piecewise constant functions with step width δ_{n-1}. In particular, the kth element of $G_{n-1}^{(n)}(s_{n-1})$ depends only on the arguments through the kth element of $s_{n-1}(k)$. Because the form of the discretization is preserved under reduction, we can consider Eqs. (2.31) and (2.30) as recurrence relations for the operators $G_{n-1}^{(n)}$ and

$F^{(n)}_{n-1}$ and, as such, may be applied recursively to obtain a sequence of operators $G^{(n)}_j$ and $F^{(n)}_j$, $j \le n$. The recurrence relations for $G^{(n)}_j$ and $F^{(n)}_j$ (for $j \le n$) in operator form are given by

$$G^{(n)}_j = P_j G^{(n)}_{j+1} + \frac{\delta^2_{j+1}}{4} Q_j F^{(n)}_{j+1} \tag{2.32}$$

$$F^{(n)}_j = P_j F^{(n)}_{j+1}, \tag{2.33}$$

provided the solution $\tilde{d}_j$ of the equation $Q_j G^{(n)}_{j+1} = P_j F^{(n)}_{j+1}$ exists. Observe that the operator forms of the "average" and "difference" operators **S** and **D**, which we introduced in working with the coordinate forms of our expressions, are the projections P_j and Q_j. We emphasize that this is a formal derivation of the recurrence relations. We show in Section III how to implement numerically this formal procedure. In Subsection II.3 we derive analytic expressions for these recurrence relations.

Let us now address the existence of the solution $\tilde{d}_j$ to the equation $Q_j G^{(n)}_{j+1} = P_j F^{(n)}_{j+1}$. We will write this equation in coordinate form as follows (dropping subscripts),

$$\mathcal{F}(s, d)(k) = \mathbf{D}g(s, d)(k) - \mathbf{S}f(s, d)(k) = 0,$$

where $\mathcal{F}: E \to \mathbf{R}^{2^j}$, $(s, d) \in E$ an open set in $\mathbf{R}^{2^j} \times \mathbf{R}^{2^j}$, and $k = 0, \ldots, 2^j - 1$. Assume that g and f are both differentiable functions so that $\mathcal{F} \in C^1(E)$. Suppose that there is a pair $(s^0, d^0) \in E$ such that

$$\mathcal{F}(s^0, d^0)(k) = \mathbf{D}g(s^0, d^0)(k) - \mathbf{S}f(s^0, d^0)(k) = 0$$

and that the Jacobian of $\mathcal{F}$ with respect to d at (s^0, d^0) does not vanish. (We know that such a pair $(s^0, d^0) \in E$ must exist since a unique solution to our ODE exists.) The Implicit Function Theorem tells us that there is a neighborhood S of s^0 in $\mathbf{R}^{2^j}$ and a unique function $\tilde{d}: S \to \mathbf{R}^{2^j}$ ($\tilde{d} \in C^1(S)$) such that $\tilde{d}(s^0) = d^0$ and $\mathcal{F}(s, \tilde{d}(s)) = 0$ for $s \in S$.

Let us investigate what it means for the Jacobian of $\mathcal{F}$ with respect to d at (s^0, d^0) to be nonzero. Notice that the kth coordinate of $\mathcal{F}$, $\mathcal{F}(s, d)(k)$, depends only on the kth coordinates of s and d

$$\mathcal{F}(s, d)(k) = \mathbf{D}g(s, d)(k) - \mathbf{S}f(s, d)(k).$$

In turn, $s(k)$ and $d(k)$ depend on $x(2k + 1)$ and $x(2k)$ and we may write $\mathcal{F}(s, d)(k)$ in terms of $x(2k + 1)$ and $x(2k)$. In particular, we can write

$$\mathbf{D}g(s, d)(k) = \frac{1}{\delta}\,(g(x)(2k + 1) - g(x)(2k))$$

$$\mathbf{S}f(s, d)(k) = \frac{1}{2}\,(f(x)(2k + 1) + f(x)(2k)),$$

where

$$x(2k+1) = s(k) + \frac{\delta}{2}\, d(k) \qquad \text{and} \qquad x(2k) = s(k) - \frac{\delta}{2}\, d(k).$$

When we differentiate $\mathcal{F}(s, d)(k)$ with respect to $d(k)$, we can apply the chain rule and differentiate with respect to $x(2k+1)$ and $x(2k)$ instead. Therefore, the derivative of the term $\mathbf{D}g(s, d)(k)$ with respect to $d(k)$ is

$$\frac{\partial}{\partial d(k)}\, \mathbf{D}g(s, d)(k) = \frac{1}{2}\frac{dg(x)(2k+1)}{dx(2k+1)} + \frac{1}{2}\frac{dg(x)(2k)}{dx(2k)} = \mathbf{S}g'(s, d)(k).$$

We calculate a similar expression for the derivative of $\mathbf{S}f(s, d)(k)$. Hence, the Jacobian of $\mathcal{F}$ with respect to d is given by the matrix $J_{\mathcal{F}}$ with entries (k, l):

$$\begin{aligned} J_{\mathcal{F}}(s, d)(k, l) &= \frac{\partial \mathcal{F}(k)}{\partial d(l)} = \frac{\partial}{d(l)}\left(\mathbf{D}g(s, d)(k) - \mathbf{S}f(s, d)(k)\right) \\ &= \begin{cases} \mathbf{S}g'(s, d)(k) - \dfrac{\delta^2}{4}\mathbf{D}f'(s, d)(k), & k = l, \\ 0, & k \neq l. \end{cases} \end{aligned}$$

Requiring the Jacobian of $\mathcal{F}$ to be nonsingular at (s^0, d^0) is equivalent to stipulating that the product below be nonzero; i.e.,

$$\prod_{k=0}^{2^j-1} \left(\mathbf{S}g'(s^0, \tilde{d}^0)(k) - \frac{\delta^2}{4}\, \mathbf{D}f'(s^0, \tilde{d}^0)(k) \right) \neq 0.$$

In other words, the quantity $\mathbf{S}g'(s^0, d^0)(k) - \frac{\delta^2}{4}\mathbf{D}f'(s^0, \tilde{d}^0)(k)$ must be nonzero for every $k = 0, \ldots, 2^j - 1$ to find a solution $\tilde{d}(s)$ for each k. If δ^2 is sufficiently small, the product $\Pi_{k=0}^{2^j-1}\, \mathbf{S}g'(s^0, d^0)(k) \neq 0$ dominates the condition. We will see this condition reappear in the analytic reduction procedure.

We summarize the above derivation as

PROPOSITION II.2. *Given an equation of the form (2.19) on some scale $j+1$ (with dyadic intervals of size $2^{-(j+1)}$), we arrange the reduction of this equation to an equation at scale j as*

$$g_j(k)(s_j) = \delta_j \sum_{k'=0}^{k-1} f_j(k')(s_j) + \frac{\delta_j}{2}\, f_j(k)(s_j), \tag{2.34}$$

where

$$g_j(k)(s_j) = \mathbf{S}g_{j+1}(k)(s_j, \tilde{d}_j) + \frac{\delta_{j+1}^2}{4}\, \mathbf{D}f_{j+1}(k)(s_j, \tilde{d}_j) \tag{2.35}$$

and

$$f_j(k)(s_j) = \mathbf{S}f_{j+1}(k)(s_j, \tilde{d}_j). \tag{2.36}$$

The solution $\tilde{d}_j$ *to the equation* $\mathbf{D}g_{j+1}(k)(s_j, d_j) - \mathbf{S}f_{j+1}(k)(s_j, d_j)$ *exists provided there is a pair* (s_j^0, d_j^0) *which satisfies the equation and the product below does not vanish:*

$$\prod_{k=0}^{2^j-1} \left(\mathbf{S}g_{j+1}(k)(s_j^0, \tilde{d}_j^0) - \frac{\delta_{j+1}^2}{4} \mathbf{D}f_{j+1}(k)(s_j^0, \tilde{d}_j^0) \right) \neq 0. \tag{2.37}$$

Remark. We have stated the proposition for a scalar differential equation but it also holds for a system of differential equations, assuming that the product (2.37) is non-singular.

II.3. Series Expansion of the Recurrence Relations

In the previous subsection we derived recurrence relations for the functions $g_j(k)(s_j)$ and $f_j(k)(s_j)$ (2.35)–(2.36) which depended on the existence of $\tilde{d}_j$. In this subsection we derive analytic expressions for these recurrence relations (2.35)–(2.36) and an explicit expression for $\tilde{d}_j$.

Let us begin at the initial discretization scale $\delta_n = 2^{-n}$ and examine the reduction from scale n to scale $n - 1$. We will not include the subscripts n and $n - 1$ unless they are necessary for clarity. Assume that $\delta = \delta_n$. The equation which determines $\tilde{d}_{n-1}$ is given by

$$\mathbf{D}g_n(s_{n-1}, d_{n-1})(k) = \mathbf{S}f_n(s_{n-1}, d_{n-1})(k). \tag{2.38}$$

Below it will be convenient to expand $g(x)(2k + 1)$ as

$$\begin{aligned} g(x(2k+1))(2k+1) &= g\left(s(k) + \frac{\delta}{2} d(k)\right)(2k+1) \\ &= g(s(k))(2k+1) + g'(s(k))(2k+1)\frac{\delta}{2} d(k) + O(\delta^2). \end{aligned}$$

We will then use a slight abuse of notation and write $g(s(k))(2k + 1)$ as $g(s)(2k + 1)$ (and $g'(s(k))(2k + 1)$ as $g'(s)(2k + 1)$). The reader should beware that the notation convention for $g(x)$ and $g(s)$ is thus slightly different. To solve this equation for $\tilde{d}$, we will first expand $g(s, d)$ and $f(s, d)$ in Taylor series about $s(k)$ (for each $k = 0, \ldots, 2^{n-1} - 1$) and keep only the terms which are of order $O(1)$ in δ. Observe that we may expand the left side of Eq. (2.38) as

$$\begin{aligned} &\frac{1}{\delta}\left(g\left(s + \frac{\delta}{2} d\right)(2k+1) - g\left(s - \frac{\delta}{2} d\right)(2k) \right) \\ &\quad = \frac{1}{\delta}\left(g(s)(2k+1) - g(s)(2k)\right) + \frac{d(k)}{2}\left(g'(s)(2k+1) + g'(s)(2k)\right) + O(\delta^2), \end{aligned}$$

and similarly for the right side. After expanding both sides of Eq. (2.38) and retaining only terms of order $O(1)$ in δ, we have the equation

$$\mathbf{D}g(s)(k) + \mathbf{S}g'(s)(k)d(k) = \mathbf{S}f(s)(k),$$

which we may solve for $\tilde{d}(s)(k)$:

$$\tilde{d}(s)(k) = \frac{\mathbf{S}f(s)(k) - \mathbf{D}g(s)(k)}{\mathbf{S}g'(s)(k)} + O(\delta^2).$$

Next we expand the recursion relations for $g_{n-1}(s_{n-1})$ and $f_{n-1}(s_{n-1})$ in Taylor series about s_{n-1} and keep only the terms which are of order $O(1)$ in δ_{n-1}. This gives us the following expressions for g_{n-1} and f_{n-1}:

$$g_{n-1}(s_{n-1})(k) = \mathbf{S}g_n(s_{n-1})(k) \qquad \text{and} \qquad f_{n-1}(s_{n-1})(k) = \mathbf{S}f_n(s_{n-1})(k).$$

Notice that if we retain terms which are only of order $O(1)$ in δ_{n-1}, the recursion relations do not depend on $\tilde{d}_{n-1}$! These equations simply reproduce the discretization procedure without incorporating any information from the fine scale. In operator form, we have done nothing other than project onto the next coarsest scale, reducing $P_n\mathbf{G}(x_n) = \mathbf{K}_nP_n\mathbf{F}(x_n)$ to $P_{n-1}\mathbf{G}(x_{n-1}) = \mathbf{K}_{n-1}P_{n-1}\mathbf{F}(x_{n-1})$. Therefore, we have to include higher order terms in the recurrence relations to determine any contribution from the fine scales.

Let us expand the recurrence relations for $g_{n-1}(s_{n-1})$ and $f_{n-1}(s_{n-1})$ in Taylor series again, but this time we will retain terms of order $O(1)$ and $O(\delta_{n-1}^2)$. This gives us recurrence relations of the form

$$g_{n-1}(s)(k) = \mathbf{S}g_n(s)(k) + \left(\frac{\tilde{d}(s)(k)}{16}\left(\mathbf{D}g_n'(s)(k) + \mathbf{S}f_n'(s)(k)\right)\right.$$
$$\left. + \frac{1}{16}\mathbf{D}f_n(s)(k) + \frac{\tilde{d}^2(s)(k)}{32}\mathbf{S}g_n''(s)(k)\right)\delta_{n-1}^2$$

$$f_{n-1}(s)(k) = \mathbf{S}f_n(s)(k) + \left(\frac{\tilde{d}(s)(k)}{16}\mathbf{D}f'(s)(k) + \frac{\tilde{d}^2(s)(k)}{32}\mathbf{S}f''(s)(k)\right)\delta_{n-1}^2.$$

Notice that these equations do include information from the fine scale. If we solve Eq. (2.38) for $\tilde{d}_{n-1}(s)(k)$ to order $O(1)$ and substitute $\tilde{d}_{n-1}(s)(k)$ into the recursion relations for $g_{n-1}(s)$ and $f_{n-1}(s)$, we may split the functions $g_{n-1}(s)$ and $f_{n-1}(s)$ into two terms, one of order $O(1)$ in δ_{n-1} and one of order $O(\delta_{n-1}^2)$,

$$g_{n-1}(s)(k) = \gamma_0(s)(k) + \gamma_1(s)(k)\delta_{n-1}^2 \qquad \text{and} \qquad f_{n-1}(s)(k) = \theta_0(s)(k) + \theta_1(s)(k)\delta_{n-1}^2,$$

where

$$\gamma_0 = \mathbf{S} g_n$$

$$\theta_0 = \mathbf{S} f_n$$

$$\gamma_1 = \frac{\tilde{d}}{16} (\mathbf{D} g'_n + \mathbf{S} f'_n) + \frac{1}{16} \mathbf{D} f_n + \frac{\tilde{d}^2}{32} \mathbf{S} g''_n$$

$$\theta_1 = \frac{\tilde{d}}{16} \mathbf{D} f'_n + \frac{\tilde{d}^2}{32} \mathbf{S} f''_n$$

$$\tilde{d} = \frac{\mathbf{S} f_n - \mathbf{D} g_n}{\mathbf{S} g'_n}.$$

We summarize the previous discussion in the following proposition.

PROPOSITION II.3. *If F and G are twice continuously differentiable as functions of x and if F is a Lipschitz function in both t and x, then we can obtain analytic expressions, at least up to order δ_j^2, for the recurrence relations and for $\tilde{d}$. Let us again introduce a superscript (n) on the functions to denote the level at which we started the reduction procedure, the subscript j, as before, signifies the current level of resolution. If the functions $g_{j+1}^{(n)}(s)$ and $f_{j+1}^{(n)}(s)$ at some scale $j+1$ consist of two terms, one of order $O(1)$ and the other of order $O(\delta_{j+1}^2)$,*

$$g_{j+1}^{(n)}(s)(k) = \gamma_{0,j+1}^{(n)}(s)(k) + \gamma_{1,j+1}^{(n)}(s)(k)\delta_{j+1}^2 \tag{2.39}$$

and

$$f_{j+1}^{(n)}(s)(k) = \theta_{0,j+1}^{(n)}(s)(k) + \theta_{1,j+1}^{(n)}(s)(k)\delta_{j+1}^2, \tag{2.40}$$

then we may arrange the reduction of these function to functions $g_j^{(n)}(s)$ and $f_j^{(n)}(s)$ at scale j as

$$g_j^{(n)}(s)(k) = \gamma_{0,j}^{(n)}(s)(k) + \gamma_{1,j}^{(n)}(s)(k)\delta_j^2 \qquad \text{and} \qquad f_j^{(n)}(s)(k) = \theta_{0,j}^{(n)}(s)(k) + \theta_{1,j}^{(n)}(s)(k)\delta_j^2, \tag{2.41}$$

where (dropping superscripts)

$$\gamma_{0,j} = \mathbf{S}\gamma_{0,j-1} \tag{2.42}$$

$$\theta_{0,j} = \mathbf{S}\theta_{0,j-1} \tag{2.43}$$

$$\gamma_{1,j} = \frac{1}{4}\mathbf{S}\gamma_{1,j-1} + \frac{\tilde{d}_j}{16}(\mathbf{D}\gamma'_{0,j-1} + \mathbf{S}\theta'_{0,j-1}) + \frac{1}{16}\mathbf{D}\theta_{0,j-1} + \frac{(\tilde{d}_j)^2}{32}\mathbf{S}\gamma''_{0,j-1} \tag{2.44}$$

$$\theta_{1,j} = \frac{1}{4}\mathbf{S}\theta_{1,j-1} + \frac{\tilde{d}_j}{16}\mathbf{D}\theta'_{0,j-1} + \frac{(\tilde{d}_j)^2}{32}\mathbf{S}\theta''_{0,j-1} \tag{2.45}$$

$$\tilde{d}_j = \frac{\mathbf{S}\theta_{0,j-1} - \mathbf{D}\gamma_{0,j-1}}{\mathbf{S}\gamma''_{0,j-1}}. \tag{2.46}$$

In other words, at level j, we arrange the functions $g_j^{(n)}$ *and* $f_j^{(n)}$ *so that they consist of two terms of the appropriate orders and we write recurrence relations for each of these two terms.*

Remark. We usually initialize the reduction procedure with the $O(1)$ terms,

$$\gamma_{0,n}^{(n)}(s)(k) = g_n^{(n)}(s)(k), \qquad \theta_{0,n}^{(n)}(s)(k) = f_n^{(n)}(s)(k),$$

and the $O(\delta_n^2)$ terms,

$$\gamma_{1,n}^{(n)}(s)(k) = 0, \qquad \theta_{1,n}^{(n)}(s)(k) = 0.$$

This can be modified, however.

Remark. Higher order expansions may be obtained in the same manner. We supply an algorithm implemented in Maple in Subsection VI.2 to compute the recurrence relations for sufficiently high order terms.

III. IMPLEMENTATION AND EXAMPLES

In this section we present the numerical implementation of our formal reduction procedure, which we derived in Subsection II.2, and three examples to evaluate the accuracy of our reduction methods and to explore "patching" together the series expansion of the recursion relations and the numerical reduction procedure. We also determine numerically the long-term effect of a small perturbation in a nonlinear forced equation.

III.1. Implementation of the Reduction Procedure

We initialize our numerical reduction procedure with two tables of values, one table for each of the discretizations of the functions F and G at the starting resolution level n. The first coordinate k in our table enumerates the averages in time of the functions F and G, the functions $g_n(s_n)(k)$ and $f_n(s_n)(k)$, for $k = 0, \ldots, 2^n - 1$. Notice that these are still functions of s_n which is unknown, so we also discretize in s_n. In other words, from the start, we look at a range of possible values $s_n(k, i)$ $(i = 0, \ldots, N - 1)$ for each k, and work with all of them together. This discretization gives us the second coordinate i for our tables. We then have values $g_n(s_n(k, i))(k)$ and $f_n(s_n(k, i))(k)$ for $k = 0, \ldots, 2^n - 1$ and $i = 0, \ldots, N - 1$. To look at a range of possible values in $s_n(k)$, we must have some *a priori* knowledge of the bounds on the solution of the differential equation.

Next we form the equation (dropping the subscript n) which determines $\tilde{d}$ on the interval $k\delta_{n-1} < t < (k + 1)\delta_{n-1}$ (see Eq. (2.27)):

$$\mathbf{D}g(s(k, i), d(k, i))(k) = \mathbf{S}f(s(k, i), d(k, i))(k). \tag{3.47}$$

Notice that this is a sampled version of Eq. (2.27) and for each sample value $s(k, i)$ and for each $k = 0, \ldots, 2^{n-1} - 1$ we must solve (3.47) for $\tilde{d}(k, i)$. That is, our unknowns

$\tilde{d}(k, i)$ form a two-dimensional array. To solve for each $\tilde{d}(k, i)$ we must interpolate among the known values $g(s(k, i))(k)$ since we need to know the value $g(s(k, i) + \frac{\delta}{2}\tilde{d}(k, i))(2k + 1)$ (and similarly for $g(s(k, i) - \frac{\delta}{2}\tilde{d}(k, i))(2k)$) and we only have the values at the sample points $s(k, i)$ for $i = 0, \ldots, N - 1$. For higher order interpolation schemes, we need fewer grid points in s to achieve a desired accuracy which reduces the size of the system with which we have to work.

Once we have computed the values $\tilde{d}(k, i)$, we calculate the reduced tables of values $g_{n-1}(s(k, i))(k)$ and $f_{n-1}(s(k, i))(k)$, where $k = 0, \ldots, 2^{n-1} - 1$ and $i = 0, \ldots, N - 1$, according to the sampled versions of the recurrence relations (2.35)–(2.36):

$$g_{n-1}(s(k, i))(k) = \mathbf{S}g_n(k)(s(k, i), \tilde{d}(k, i)) + \frac{\delta_n^2}{4}\mathbf{D}f_n(k)(s(k, i), \tilde{d}(k, i))$$

$$f_{n-1}(s(k, i))(k) = \mathbf{S}f_n(k)(s(k, i), \tilde{d}(k, i)).$$

Notice that the tables are reduced in width in k by a factor of two and that this procedure can be applied repeatedly.

Remark. Observe that when $i = 0$ (respectively, $i = N - 1$), we cannot interpolate to calculate the values $g(s(k, i) - \frac{\delta}{2}\tilde{d}(k, i))$ (respectively, $g(s(k, i) + \frac{\delta}{2}\tilde{d}(k, i))$). We must either extrapolate (and then ignore the resulting "boundary effects" which propagate through the reduction procedure) or adjust the grid in the s variable at each resolution level. An alternate approach could be to use asymptotic formulas valid for large s.

We implemented this algorithm in Matlab as a prototype to test the following examples.

III.2. Examples

III.2.1. Accuracy. With the first example we verify our numerical reduction procedure and determine how the accuracy of the method depends on the step-size $\delta_n = 2^{-n}$ of the initial discretization. We also evaluate the accuracy of the linear versus cubic interpolation in the context of our approach. We use a simple separable equation

$$x'(t) = (1/\epsilon)x^2(t)\cos(t/\epsilon) \qquad \text{and} \qquad x(0) = x_0 \tag{3.48}$$

with the solution available analytically. We observe that the solution $x(t)$ to Eq. (3.48) oscillates about its initial value x_0. We choose $\epsilon = 1/(4\pi)$ and the initial value $x_0 = 1/2$. The exact solution is given by

$$x(t) = \frac{x_0}{1 - x_0\sin(t/\epsilon)},$$

which we use to verify our reduction procedure. In particular we check if the averages of $x(t)$ satisfy the difference equation derived via reduction.

Let us assume that we reduce to resolution level $\delta_j = 2^{-j}$ so that we have two tables

TABLE 1
Errors as a Function of the Initial Resolution

Initial resolution = δ_n	Average error
2^{-2}	0.0774
2^{-3}	0.0290
2^{-4}	0.0069
2^{-5}	0.0019

of values for $f_j(s(k, i))(k)$ and $g_j(s(k, i))(k)$. If $x_j(k)$ is the average of x over the interval $k2^{-j} < t < (k + 1)2^{-j}$, then the following equation should hold

$$g_j(x_j)(k) = \delta_j \sum_{k'=0}^{k-1} f_j(x_j)(k') + \frac{\delta_j}{2} f_j(x_j)(k).$$

We denote by $e_j(k)$ the error over each interval $k\delta_j < t < (k + 1)\delta_j$ and define $e_j(k)$ by

$$e_j(k) = \left| g_j(x_j)(k) - \delta_j \sum_{k'=0}^{k-1} f_j(x_j)(k') - \frac{\delta_j}{2} f_j(x_j)(k) \right|.$$

Note that we have only sampled values for $g_j(s(k, i))(k)$ and $f_j(s(k, i))(k)$ and so we must interpolate among these values to calculate $g_j(x_j)(k)$ for a specific value $x_j(k)$. We want to know how the errors $e_j(k)$ depend on the level of resolution at which we begin the reduction procedure. We reduce to resolution level with $\delta_j = 2^{-1}$ and calculate the errors $e_j(0)$ and $e_j(1)$ using the averages $x_j(0) = x_j(1) = 0.5774$. We fix the number of sample points in s to be 50 and use linear interpolation. Table 1 lists the errors as a function of the initial resolution. If we exclude the errors associated with the initial resolution $\delta_n = 2^{-2}$ and plot the logarithm of the remaining errors as a function of $\log(\delta_n)$, the slope of the fitted line is 1.9660. We can conclude that the accuracy of our numerical reduction scheme increases with the square of the initial resolution.

As we described above, we have to interpolate between known function values in the tables. We used both linear and cubic interpolation methods. We would like to know how the interpolation affects the error of the method and the minimum number of sample points in s we need for both interpolation methods. We use Eq. (3.48) again with the same values for x_0 and ϵ. We fix the initial resolution at $\delta_n = 2^{-5}$. For technical reasons, with cubic interpolation we can reduce only to resolution level $\delta_j = 2^{-2}$. Table 2 lists the errors as a function of the number of sample points in s for both linear and cubic interpolation. In Fig. 1 we have plotted the average error as a function of the number of sample points in s for the two methods of interpolation. We can see that with cubic interpolation the minimum number of grid points in s is 15 and that with linear interpolation we can achieve the same accuracy with 50 grid points. We can also see from the

TABLE 2
Error as a Function of the Number of Sample Points in *s*, with Linear Interpolation and with Cubic Interpolation

	Average error	
No. of sample points in *s*	Linear	Cubic
6	0.0238	0.0045
10	0.0098	0.0020
15	0.0052	0.0020
25	0.0029	0.0020
30	0.0024	—
50	0.0019	—

graph that increasing the number of grid points (past 15) will yield no gain in the accuracy of the cubic interpolation method.

III.2.2. Hybrid reduction method. In the second example we will combine the analytic reduction procedure with the numerical procedure. We begin at a very fine resolution $\delta_{n_0} = 2^{n_0}$ and reduce analytically to a coarser resolution level $\delta_{n_1} = 2^{n_1}$. From this level we reduce numerically to the final coarse level δ_j. The analytic reduction procedure is computationally inexpensive compared to the numerical procedure and we want to take advantage of this efficiency as much as possible. However, we must balance computational expense with accuracy. With this example we will determine the resolution level δ_{n_1} at which this balance is achieved. Again we use a separable equation given by

$$x'(t) = x^2(t)\cos(t/\epsilon), \qquad x_0 = 0.1, \qquad \epsilon = \frac{1}{4\pi}. \tag{3.49}$$

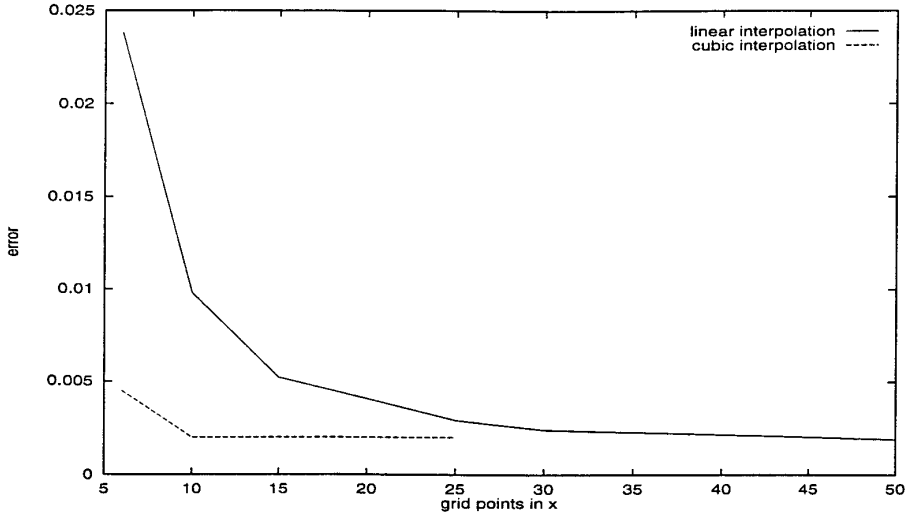

FIG. 1. The error as a function of the number of sample points in *s* for linear and cubic interpolation methods.

TABLE 3
Errors as a Function of the Intermediate Resolution

Intermediate Resolution = n_1	Average error
2^{-2}	0.00106
2^{-3}	0.00093
2^{-4}	0.00092
2^{-5}	0.00092

The solution to Eq. (3.49) is

$$x(t) = \frac{x_0}{1 - \epsilon x_0 \sin(t/\epsilon)}.$$

We begin with analytic reduction at resolution $\delta_{n_0} = 2^{-10}$. We choose the final resolution level to be $\delta_j = 2^{-2}$ and we let n_1, the resolution as which we switch to the numerical procedure, range from 2 to 5. Table 3 lists the errors as a function of n_1. Note that we have used cubic interpolation and ten grid points in x. Figure 2 is a graph of the average error as a function of the intermediate resolution. We can see from this graph that the biggest gain in accuracy occurs at the intermediate resolution $\delta_{n_1} = 2^{-3}$. In other words, at the finer intermediate levels ($n_1 = 4, 5$) we give a small gain in accuracy compared to the computational expense of the additional resolution levels in the numerical reduction. To balance accuracy with computational time for this particular example, we should reduce analytically to resolution $\delta_{n_1} = 2^{-3}$ and then switch to the numerical reduction to reach the final level $\delta_j = 2^{-2}$. The analytic procedure allows us to reduce our problem with very little computational expense (compared to the numerical procedure) and then for the

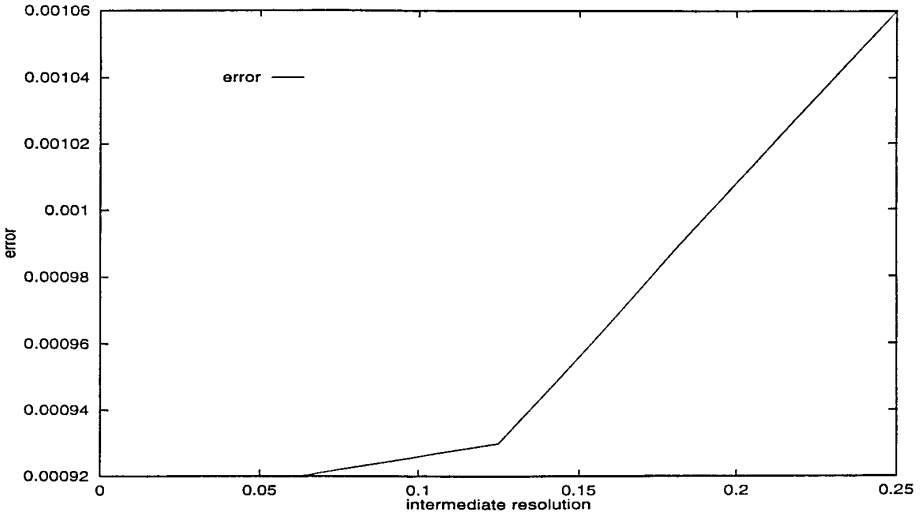

FIG. 2. The error as a function of the intermediate resolution level at which we switch from the analytic reduction method to the numerical reduction method.

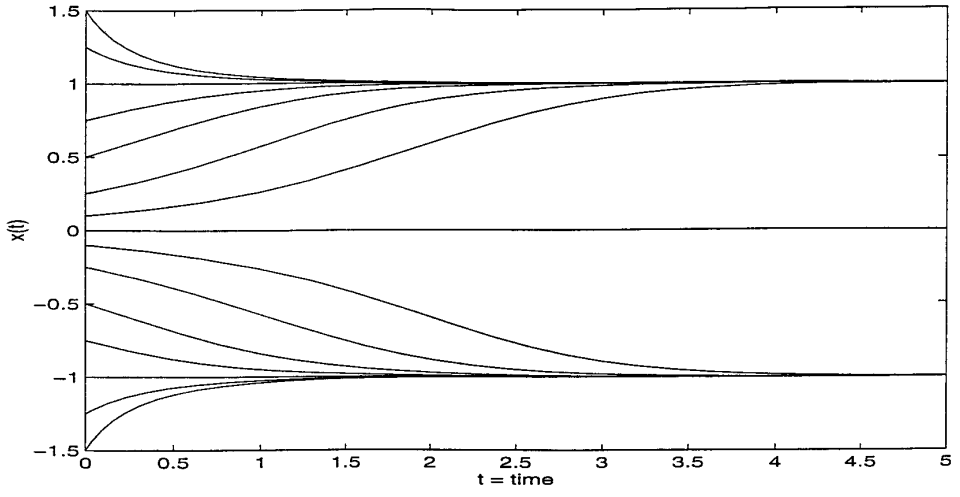

FIG. 3. The flows for Eq. (3.50) with zero forcing.

additional accuracy needed we can use only one relatively more expensive numerical reduction step.

III.2.3. Bifurcation and stability analysis. The third example we will consider is the equation

$$x'(t) = (1 - x^2(t))\,x(t) + A\,\sin(t/\epsilon), \qquad x(0) = x_0, \tag{3.50}$$

where ϵ is a small parameter associated to the scale of the oscillation in the forcing term. If the amplitude $A = 0$, then the solution $x(t)$ has one unstable equilibrium point at $x_0 = 0$ and two stable equilibria at $x_0 = -1, 1$ (see Fig. 3).

A small perturbation in the forcing term will effect large changes in the asymptotic behavior as t tends to infinity. Therefore, the behavior of the solution on a fine scale will affect the large scale behavior. In particular, if the amplitude A is nonzero but small, then the solution $x(t)$ has three periodic orbits. Two of the periodic orbits are stable while one is unstable (see Fig. 4). As we increase the amplitude A, there is a pitchfork bifurcation—the three periodic orbits merge into one stable periodic orbit (see Fig. 5). We would like to know if we can determine numerically the initial values of these periodic orbits from the reduction procedure and if those periodic solutions are stable or unstable. We will compare these results derived from the reduction procedure with those from the asymptotic expansion of x for initial values near $x_0 = 0$ and for small ϵ. Let us begin with the asymptotic expansion of x for small values of ϵ. Assume we have an expansion of the form

$$x(t;\,\epsilon) \sim 0 + \epsilon x_1(t,\,\tau) + \epsilon^2 x_2(t,\,\tau) + \cdots, \tag{3.51}$$

where the fast time scale τ is given by $\tau = t/\epsilon$. If we substitute the expansion (3.51) into the Eq. (3.50), we have the equation

$$\frac{\partial x_1}{\partial \tau} + \epsilon\left(\frac{\partial x_1}{\partial t} + \frac{\partial x_2}{\partial \tau}\right) = A\,\sin\tau + \epsilon x_1 + O(\epsilon^2).$$

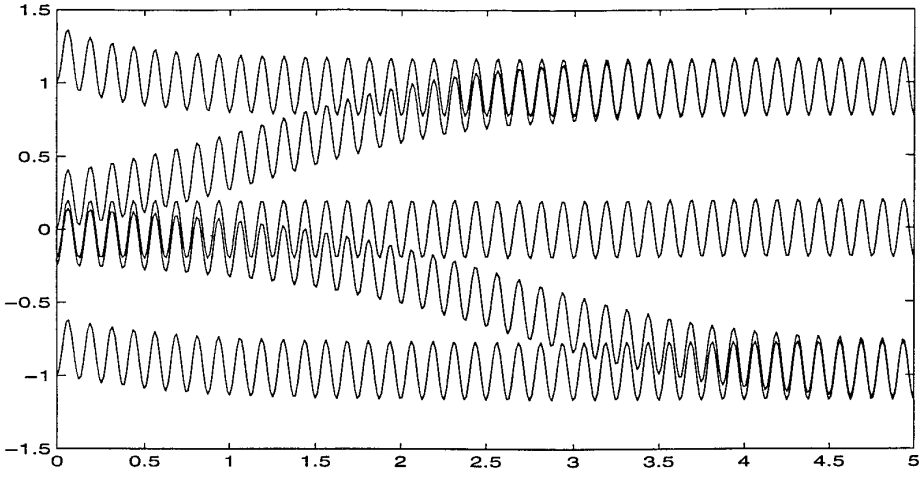

FIG. 4. The flows for Eq. (3.50) with small but nonzero forcing. Notice that there are three periodic orbits: two stable and one unstable.

Equating terms of order one in ϵ, we have $\frac{\partial x_1}{\partial \tau} = A \sin \tau$, which has the solution $x_1(t, \tau) = -A \cos \tau + \omega(t)$. The function ω is determined by a secularity condition which we impose on the terms of order ϵ. Equating the terms of order ϵ gives us the equation

$$\frac{\partial x_2}{\partial \tau} = -A \cos \tau + \omega(t) - \omega'(t).$$

The non-oscillatory term $\omega - \omega'$ in the above equation is "secular" because if it were nonzero, we would have a linear term in τ which is incompatible with the assumed form

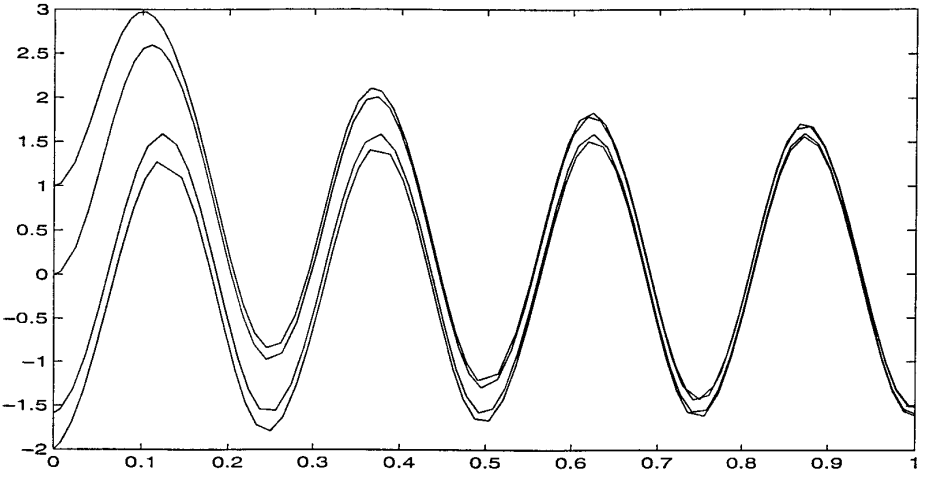

FIG. 5. The flows for Eq. (3.50) with large amplitude A. Notice that there is only one (stable) periodic orbit in this diagram as the system has undergone a pitchfork bifurcation.

of the expansion (3.51). Therefore, we set this term equal to zero, $\omega - \omega' = 0$, and determine that $\omega(t) = C_1 \, e^t$. So we have obtained an asymptotic expansion for x

$$x(\tau; \epsilon) \sim 0 + \epsilon(-A \cos \tau + C_1 \, e^t). \tag{3.52}$$

Note that this asymptotic expansion is valid only for $t \leq |\log \epsilon|$. We can, however, determine the behavior of x for large time by examining the direction of the growth in x since the direction signifies which stable periodic orbit (1 or -1) captures the solution. Observe that the sign of the coefficient C_1 depends on the initial value x_0. In particular, if $x_0 > -A\epsilon$ then $C_1 > 0$ and if $x_0 < -A\epsilon$ then $C_1 < 0$. In other words, if ϵ is sufficiently small, there is a separation point x_0^*, defined as the largest value such that if $x_0 < x_0^*$, then $x(t) < 0$ as t tends to infinity. According to the asymptotic expansion (3.52), the separation point x_0^* as ϵ goes to zero is given by

$$x_0^* \sim -A\epsilon.$$

This is an approximation of the initial value of the unstable periodic solution.

Let us derive another approximation for the separation point by linearizing Eq. (3.50) about $x_0 = 0$. The linearized differential equation is the equation

$$x'(t) = x(t) + A \sin(t/\epsilon), \qquad x(0) = x_0,$$

which has the solution $x(t)$ given by

$$x(t) = e^t \left(x_0 + \int_0^t A e^{-s} \sin(s/\epsilon) ds \right).$$

The sign of the factor $x_0 + \int_0^t Ae^{-s} \sin(s/\epsilon) ds$ as t tends to infinity determines the direction of growth in $x(t)$. In other words, the separation point x_0^* is the value for which the following is true

$$\lim_{t \to \infty} \left(x_0^* + \int_0^t A e^{-s} \sin(s/\epsilon) ds \right) = 0.$$

If we evaluate the integral in the above expression, we determine that x_0^* satisfies

$$\lim_{t \to \infty} \left(x_0^* + \frac{A\epsilon^2}{1 + \epsilon^2} e^{-t} \sin(t/\epsilon) - \frac{A\epsilon}{1 + \epsilon^2} (e^{-t} \cos(t/\epsilon) - 1) \right) = 0.$$

Thus the separation point is given by

$$x_0^* \sim -A\epsilon \qquad \text{for } \epsilon \text{ sufficiently small.}$$

We have derived two approximations for the initial value near $x_0 = 0$ of the unstable periodic orbit. We will compare these two approximations with the values we determine numerically from the reduction procedure.

We now turn to the numerical reduction procedure. Assume that we can reduce the problem to a resolution level $\delta_j = 2^{-j}$ where it no longer depends on time (i.e., the problem is now autonomous). This means that the tables $g_j(s(k, i))(k)$ and $f_j(s(k, i))(k)$ depend only on i and not on k. Let $x_j(k)$ denote the average of the solution x over the interval $k\delta_j < t < (k + 1)\delta_j$. Observe that for Eq. (3.50) the functions G and F are given by

$$G(t, x(t)) = x(t) - x_0 \qquad \text{and} \qquad F(t, x(t)) = (1 - x^2(t))x(t) + A \sin(t/\epsilon)$$

so that the initial value x_0 is simply a parameter in the numerical reduction scheme and we may take $G(t, x(t)) = x(t)$.

If the solution $x(t)$ is periodic and if δ_j is an integer multiple of that period, the averages $x_j(k)$ will be all equal to the value x_e (call that the average); that is, $x_j(1) = x_j(2) = \cdots = x_e$. Therefore the value of $g_j(\cdot)(k)$ at each average $x_j(k)$ is the same:

$$g_j(x_j)(1) = g_j(x_j)(2) = \cdots = g_j(x_e).$$

Since this holds for all k, we will drop the parameter. We will also drop the subscript j for clarity. If we take the expressions for g evaluated at two successive averages $x(l)$ and $x(l + 1)$ and subtract them, we find that $f(x_e)$ must satisfy

$$0 = g(x_e) - g(x_e) = g(x(l + 1)) - g(x(l)) = \frac{\delta}{2}(f(x(l + 1)) + f(x(l))) = f(x_e).$$

This gives us a criterion for finding the average value x_e. We know that the average value of the periodic solution x is a zero of f. Finally, the separation point x_0^* is the initial value such that $g(x_e) - x_0^* = 0$.

To determine if the separation point x_0^* is stable or unstable, we will perturb it by a small value λ. Set the new initial value x_0 equal to $x_0^* + \lambda$. Let $(\Delta x_e)_l$ denote the deviation from the average value x_e in the average of x over the interval $l\delta_j < t < (l + 1)\delta_j$. Then, the discretization scheme relates the difference between $(\Delta x_e)_l$ and $(\Delta x_e)_{l+1}$:

$$g(x_e + (\Delta x_e)_{l+1}) - g(x_e + (\Delta x_e)_l) = \frac{\delta}{2}(f(x_e + (\Delta x_e)_{l+1}) + f(x_e + (\Delta x_e)_l)).$$

If we linearize the above equation, the following holds:

$$g'(x_e)((\Delta x_e)_{l+1} - (\Delta x_e)_l) = \frac{\delta}{2} f'(x_e)((\Delta x_e)_{l+1} + (\Delta x_e)_l),$$

TABLE 4

ϵ	A	x_e	Sep. pts.	Ratios
$\frac{1}{16\pi}$	1	4.0×10^{-7}	-0.0199	1.1354
		1.0006	0.9807	0.7796
		-1.0006	-1.0204	0.7796
$\frac{1}{16\pi}$	10	4.0×10^{-6}	-0.1989	1.1276
		0.9746	0.7759	0.7868
		-0.9746	-1.1732	0.7868
$\frac{1}{16\pi}$	20	4.0×10^{-6}	-0.3978	1.1056
		0.8927	0.4951	0.8185
		-0.8927	-1.2901	0.8186
$\frac{1}{8\pi}$	1	3.0×10^{-6}	-0.0397	1.1345
		0.9991	0.9595	0.7815
		-0.9991	1.0387	0.7815
$\frac{1}{8\pi}$	40	-1.4×10^{-5}	-1.5891	0.7594

Note. The entry x_e is the value of $x(t)$ for the corresponding initial value x_0, which we call a separation point. If the ratio is greater than one, the separation point is unstable and if the ratio is less than one, the separation point is stable. These three columns are calculated using the effective equation.

or equivalently, we may use the ratio

$$\frac{(\Delta x_e)_{l+1}}{(\Delta x_e)_l} = \frac{g'(x_e) + (\delta/2) f'(x_e)}{g'(x_e) - (\delta/2) f'(x_e)}.$$

to test the stability of the separation point x_0^*.

Table 4 below lists several values for ϵ, the amplitude A, and the corresponding average values x_e for the periodic orbits, separation points, and ratios. The separation point which has a corresponding ratio greater than one is the unstable periodic orbit with initial value x_0^*. We reduce to a level where the problem is autonomous and use cubic interpolation. We compare the calculated separation points x_0^* with those determined by the two analytic methods in Table 5. Notice that for the values $A = 40$ and $\epsilon = 1/(8\pi)$ we have only one stable periodic solution. In other words, the two stable periodic orbits have merged with the unstable one to create one stable periodic solution. Clearly, this merging of solutions shows that the fine scale behavior of the solution has a large effect on the coarse scale (or long time) behavior. Furthermore, we have detected numerically this large effect. In order to determine the value of x_0^* asymptotically for $A = 40$ and $\epsilon = 1/8\pi$, we had to resort to a different asymptotic expansion from the one used previously.

IV. HOMOGENIZATION

In the previous sections we discussed only the MRA reduction procedure for nonlinear ODEs. In this section we construct the MRA homogenization scheme for nonlinear ODEs. In the multiresolution approach to homogenization, the homogenization step is a proce-

TABLE 5

ϵ	A	x_0^* (MRA)	x_0^* (asymp.)	x_0^* (linear)
$\frac{1}{16\pi}$	1	−0.0199	−0.01989	$-\frac{1}{16\pi}$
$\frac{1}{16\pi}$	10	−0.1989	−0.1989	$-\frac{10}{16\pi}$
$\frac{1}{16\pi}$	20	−0.3978	−0.3978	$-\frac{20}{16\pi}$
$\frac{1}{8\pi}$	1	−0.0397	−0.0395	$-\frac{1}{8\pi}$
$\frac{1}{8\pi}$	40	−1.5891	−1.592	—

Note. We determine the unstable separation point x_0^* with three methods: one numerical method using the reduced equation and two analytic methods. For $\epsilon = 1/8\pi$ and $A = 40$ we could not apply the linearization method and we had to use a different asymptotic expansion.

dure by which the original system is replaced by some other system with desired properties (perhaps a "simpler" system). By making sure that both systems produce the same reduced equations at some coarse scale, we observe that as far as *the solution* at the coarse scale is concerned, the two systems are indistinguishable. We should emphasize that this is a preliminary investigation of the homogenization method for nonlinear ODEs. There are many different approaches to homogenizing a nonlinear ODE and many different possibilities for a "simpler" system depending on the problem. We explore one of these possibilities.

Suppose we reduce our problem to level j, using the series expansion of the recurrence relations, and have a discretization of the form

$$g_j(s_j)(k) = \delta_j \sum_{k'=0}^{k-1} f_j(s_j)(k') + \frac{\delta_j}{2} f_j(s_j)(k), \tag{4.53}$$

where the functions $g_j(s_j)$ and $f_j(s_j)$ are expanded in powers of δ_j,

$$g_j(s_j)(k) = \gamma_{0,j}(s_j)(k) + \gamma_{1,j}(s_j)(k)\delta_j^2 \qquad \text{and} \qquad f_j(s_j)(k) = \theta_{0,j}(s_j)(k) + \theta_{1,j}(s_j)(k)\delta_j^2.$$

We want to find 2^j functions $\tilde{G}(s)(k)$ and $\tilde{F}(s)(k)$ (indexed by $k = 0, \ldots, 2^j - 1$) with expansions

$$\tilde{G}(s)(k) = \tilde{G}_0(s)(k) + \delta_j^2 \tilde{G}_1(s)(k) \qquad \text{and} \qquad \tilde{F}(s)(k) = \tilde{F}_0(s)(k) + \delta_j^2 \tilde{F}_1(s)(k)$$

such that for each k and all $s_j \in V_j$ we have

$$\begin{aligned} g_j(s_j)(k) &= \gamma_0(s_j)(k) + \delta_j^2 \gamma_1(s_j)(k) = \tilde{G}_0(s_j)(k) + \delta_j^2 \tilde{G}_1(s_j)(k) \\ f_j(s_j)(k) &= \theta_0(s_j)(k) + \delta_j^2 \theta_1(s_j)(k) = \tilde{F}_0(s_j)(k) + \delta_j^2 \tilde{F}_1(s_j)(k), \end{aligned} \tag{4.54}$$

where

$$\tilde{G}_1(x)(k) = \frac{1}{24}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)^2 \tilde{G}_0''(x)(k) + \frac{1}{12}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)\tilde{F}_0'(x)(k)$$

and

$$\tilde{F}_1(x)(k) = \frac{1}{24}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)^2 \tilde{F}_0''(x)(k).$$

In other words, on each interval $(k)2^{-j} < t < (k+1)2^{-j}$ we want to find two functions $\tilde{G}(x)(k)$ and $\tilde{F}(x)(k)$ which depend only on x such that the reduction scheme applied to these functions on each interval yields the same discretization (4.53) as the original. We know what the fixed point or limiting value of the reduction process for autonomous equations is (see Appendix A) so we may use this exact form to specify $\tilde{G}_1(x)(k)$ and $\tilde{F}_1(x)(k)$ in terms of $\tilde{G}_0(x)(k)$ and $\tilde{F}_0(x)(k)$. We can eliminate $\tilde{G}_1(x)(k)$ and $\tilde{F}_1(x)(k)$ from Eqs. (4.54) to get the following coupled system of differential equations for each k

$$\frac{g_j(x)(k) - \tilde{G}_0(x)(k)}{\delta_j^2} = \frac{1}{24}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)^2 \tilde{G}_0''(x)(k) + \frac{1}{12}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)\tilde{F}_0''(x)(k)$$

$$\frac{f_j(x)(k) - \tilde{F}_0(x)(k)}{\delta_j^2} = \frac{1}{24}\left(\frac{\tilde{F}_0(x)(k)}{\tilde{G}_0'(x)(k)}\right)^2 \tilde{F}_0''(x)(k).$$

We may pick out the non-oscillatory solution to the system of differential equations and obtain

$$\tilde{G}_0 = \gamma_0 + \delta_j^2\left(\gamma_1 - \frac{1}{24}\left(\frac{\theta_0}{\gamma_0'}\right)^2 \gamma_0'' - \frac{1}{12}\left(\frac{\theta_0}{\gamma_0'}\right)\theta_0'\right)$$

$$\tilde{F}_0 = \theta_0 + \delta_j^2\left(\theta_1 - \frac{1}{24}\left(\frac{\theta_0}{\gamma_0'}\right)^2 \theta_0''\right).$$

This homogenization procedure will yield a simplified equation which is autonomous over intervals of length 2^{-j} and whose solution has the same average over these intervals as the solution to the original, more complicated differential equation. One can replace the original equation by this homogenized equation and be assured that the coarse behavior of the homogenized equation is asymptotically equal to the coarse behavior of the original solution.

V. CONCLUSIONS

We can extend the MRA reduction and homogenization strategies to small systems of nonlinear differential equations. The main difficulty in extending the reduction procedure to nonlinear equations is that there are no explicit expressions for the fine scale behavior of the solution in terms of the coarse scale behavior. We resolve this problem with two

approaches; a numerical reduction procedure and a series expansion of the recurrence relations which gives us an analytic reduction procedure.

The numerical procedure requires some *a priori* knowledge of the bounds on the solution since it entails using a range of possible values for the solution and its average behavior and working with all of them together. The accuracy of this scheme increases with the square of the initial resolution but it is computationally feasible for small systems of equations only. We can use the reduced equation, which we compute numerically, to find the periodic orbits of a periodically forced system and to determine the stability of the orbits.

One reduction step in the analytic method consists of expanding the recurrence relations in Taylor series about the averages of the solution. We gather the terms in the series which are all of the same order in δ_j, the step size, and identify them as one term in the series so that we have a power series in δ_j. Then we write recurrence relations for each term in the series so that the nonlinear functions which determine the solution on the next coarsest scale are themselves power series in the next coarsest step size δ_{j-1}. We determine the recurrence relations for an arbitrary term in this power series, show that the recurrence relations converge if applied repeatedly, and investigate the convergence of the power series for linear ODEs.

The homogenization procedure for nonlinear differential equations is a preliminary one. We replace the original equation with an equation which is autonomous on the coarse scale at which we want the solutions to agree. If we are interested in the behavior of our solution only on a scale 2^{-j}, then our simpler equation which we use in place of the original equation does not depend on t over intervals of size 2^{-j}. Unlike the linear case where a constant coefficient equation (or an equation with piecewise constant coefficients) is clearly simpler than a variable coefficient equation, there are many possible kinds of "simpler" equations which can replace a nonlinear equation. We present one candidate type for a simpler equation and leave others untouched.

VI. APPENDIX A

In this appendix we present several detailed discussions of the series expansion of the recursion relations. The first is a derivation of the fixed point of the recurrence relations for autonomous equations. The second is an algorithm for generating the relations for higher order terms in the power series expansions.

More detailed discussions can be found in [11]. The results include the general forms of the coefficients $\gamma_{0,j}^{(n)}(s)$, $\gamma_{1,j}^{(n)}(s)$, $\theta_{0,j}^{(n)}(s)$, and $\theta_{1,j}^{(n)}(s)$ in the expansions of $g_j^{(n)}(s)$ and $f_j^{(n)}(s)$ for non-autonomous differential equations. Conditions for the convergence as n tends to $-\infty$ of the recurrence relations for the two lowest order coefficients are also discussed. The altered recurrence relations for the case when the left side of the differential equation (2.14), $F(t, x(t))$, is not Lipschitz as a function of t are given. Finally, the recurrence relations for the general coefficients $\gamma_{i,j}^{(n)}$ and $\theta_{i,j}^{(n)}$ are discussed along with the convergence of the series expansions

$$g_j^{(n)}(x)(k) = \sum_{i=0}^{\infty} \gamma_{i,j}^{(n)}(x)(k)\delta_j^{2i} \qquad \text{and} \qquad f_j^{(n)}(x)(k) = \sum_{i=0}^{\infty} \theta_{i,j}^{(n)}(x)(k)\delta_j^{2i}$$

under the reduction process.

VI.1. Recursion Relations for Autonomous Equations

We will now apply the reduction procedure to the autonomous integral equation

$$G(x(t)) = \int_0^t F(x(s))ds \tag{6.55}$$

and examine the series expansions for the recurrence relations when applied to this autonomous integral equation. We will consider only the first two terms in the expansions; higher order discretization schemes can be obtained if we keep higher order terms in the expansions.

THEOREM VI.1. *Let us assume that the functions F and G are both twice continuously differentiable as functions of x and that dG/dx ≠ 0. Then the coefficients* $\gamma_{0,j}^{(n)}$, $\gamma_{1,j}^{(n)}$, $\theta_{0,j}^{(n)}$, *and* $\theta_{1,j}^{(n)}$ *are given by*

$$\gamma_{0,j}^{(n)} = G, \qquad \gamma_{1,j}^{(n)} = \frac{(1/3)(2^{2m}-1)}{2^{2+2m}}\left(\frac{F}{G'}\right)F' + \frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\left(\frac{F}{G'}\right)^2 G''$$

$$\theta_{0,j}^{(n)} = F, \qquad \theta_{1,j}^{(n)} = \frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\left(\frac{F}{G'}\right)^2 F'',$$

where $m = n - j$. *Furthermore, in the limit as m tends to infinity, the coefficients converge to*

$$\gamma_{0,j}^{(-\infty)} = G, \qquad \gamma_{1,j}^{(-\infty)} = \frac{1}{12}\left(\frac{F}{G'}\right)F' + \frac{1}{24}\left(\frac{F}{G'}\right)^2 G''$$

$$\theta_{0,j}^{(-\infty)} = F, \qquad \theta_{1,j}^{(-\infty)} = \frac{1}{24}\left(\frac{F}{G'}\right)^2 F''.$$

Proof. Because the functions G and F do not depend explicitly on time, the terms $g_n(x_n)(k)$ and $f_n(x_n)(k)$ in the initial discretization

$$g_n(x_n)(k) = \delta_n \sum_{k'=0}^{k-1} f_n(x_n)(k') + \frac{\delta_n}{2} f_n(x_n)(k)$$

are simply the values of G and F evaluated at $x_n(k)$. In the non-autonomous case, the terms $g_n(x_n)(k)$ and $f_n(x_n)(k)$ are the averages of the function $G(t, \cdot)$ and $F(t, \cdot)$ over the time interval $k\delta_n < t < (k+1)\delta_n$ and evaluated at $x_n(k)$. Because the values $g_n(s_{n-1}(k))(2k+1)$ and $g_n(s_{n-1}(k))(2k)$ are equal, the difference operator **D** applied to g_n and evaluated at s_{n-1} yields zero,

$$\mathbf{D}g_n(s_{n-1})(k) = \frac{1}{\delta}\left(g_n(s_{n-1}(k))(2k+1) - g_n(s_{n-1}(k))(2k)\right) = 0,$$

and the average operator **S** applied to g_n and evaluated at s_{n-1} yields $g_n(s_{n-1})(k)$,

$$\mathbf{S}g_n(s_{n-1})(k) = \frac{1}{2}\left(g_n(s_{n-1}(k))(2k+1) + g_n(s_{n-1}(k))(2k)\right) = g_n(s_{n-1})(k).$$

We will drop the parameter k in what follows for this reason and simply write $G(x_n)$ and $F(x_n)$ instead of $g_n(x_n)(k)$ and $f_n(x_n)(k)$ and we will simplify the recursion relations.

We begin with an initial discretization of our integral equation at resolution level $n = 1$ and initialize the coefficients as

$$\gamma_{0,1}^{(1)}(x_1) = G(x_1), \qquad \gamma_{1,1}^{(1)}(x_1) = 0$$

$$\theta_{0,1}^{(1)}(x_1) = F(x_1), \qquad \theta_{1,1}^{(1)}(x_1) = 0.$$

We reduce one level to $j = 0$ so that the difference in resolution $(n - j)$ is one. Using the simplified recursion relations, we calculate the reduced coefficients:

$$\gamma_{0,0}^{(1)}(x_0) = G(x_0), \qquad \gamma_{1,0}^{(1)}(x_0) = \frac{1}{16}\left(\frac{F(x_0)}{G'(x_0)}\right)F'(x_0) + \frac{1}{32}\left(\frac{F(x_0)}{G'(x_0)}\right)^2 G''(x_0)$$

$$\theta_{0,0}^{(1)}(x_0) = G(x_0), \qquad \theta_{1,0}^{(1)}(x_0) = \frac{1}{32}\left(\frac{F(x_0)}{G'(x_0)}\right)^2 F''(x_0).$$

We want to find the forms of the coefficients for an arbitrary difference in resolution $(n - j) = m$. We proceed by induction. Assume that for $(n - j) = m$ we have

$$\gamma_{0,j}^{(n)} = G, \qquad \gamma_{1,j}^{(n)} = \frac{(1/3)(2^{2m}-1)}{2^{2+2m}}\left(\frac{F}{G'}\right)F' + \frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\left(\frac{F}{G'}\right)^2 G'' \tag{6.56}$$

$$\theta_{0,j}^{(n)} = F, \qquad \theta_{1,j}^{(n)} = \frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\left(\frac{F}{G'}\right)^2 F''. \tag{6.57}$$

We will apply the simplified recursion relations to these coefficients and reduce one more level so that $n - (j-1) = m + 1$. It is clear that $\gamma_{0,j-1}^{(n)} = G$ and $\theta_{0,j-1}^{(n)} = F$. The simplified recursion relations tell us that

$$\begin{aligned}
\gamma_{1,j-1}^{(n)} &= \frac{1}{4}\,\mathbf{S}\gamma_{1,j}^{(n)} + \frac{1}{16}\left(\frac{F}{G'}\right)F' + \frac{1}{32}\left(\frac{F}{G'}\right)^2 G'' \\
&= F'\left(\frac{F}{G'}\right)\left(\frac{1}{16} + \frac{1}{4}\frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\right) + G''\left(\frac{F}{G'}\right)^2\left(\frac{1}{32} + \frac{1}{4}\frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\right) \\
&= \frac{(1/3)(2^{2(m+1)}-1)}{2^{2+2(m+1)}}\left(\frac{F}{G'}\right)F' + \frac{(1/3)(2^{2(m+1)}-1)}{2^{3+2(m+1)}}\left(\frac{F}{G'}\right)^2 G'',
\end{aligned}$$

and

$$\theta_{1,j-1}^{(n)} = \frac{1}{4}\mathbf{S}\theta_{1,j}^{(n)} + \frac{1}{32}\left(\frac{F}{G'}\right)^2 F'' = F''\left(\frac{F}{G'}\right)^2\left(\frac{1}{32} + \frac{1}{4}\frac{(1/3)(2^{2m}-1)}{2^{3+2m}}\right)$$

$$= \frac{(1/3)(2^{2(m+1)}-1)}{2^{3+2(m+1)}}\left(\frac{F}{G'}\right)^2 G''.$$

This proves the formulas (6.56)–(6.57) for all $m = (n - j)$. Note that these forms depend only on the difference in resolution levels $n - j$. In the limit as m tends to infinity, we find that the coefficients converge to

$$\gamma_{0,j}^{(-\infty)} = G, \qquad \gamma_{1,j}^{(-\infty)} = \frac{1}{12}\left(\frac{F}{G'}\right)F' + \frac{1}{24}\left(\frac{F}{G'}\right)^2 G''$$

$$\theta_{0,j}^{(-\infty)} = F, \qquad \theta_{1,j}^{(-\infty)} = \frac{1}{24}\left(\frac{F}{G'}\right)^2 F''.$$

Additionally, the limiting values of these coefficients eliminate the error of the initial discretization, give us expressions independent of resolution level j, and contribute errors only from the truncations of the original Taylor series. The reduced equation at level j is then given by

$$g_j(x_j)(k) = \delta_j \sum_{k'=0}^{k-1} f_j(x_j)(k') + \frac{\delta_j}{2} f_j(x_j)(k), \qquad \text{where (dropping } j\text{)} \tag{6.58}$$

$$g(x)(k) = \gamma_0^{(\infty)}(x(k)) + \gamma_1^{(\infty)}(x(k))\delta^2 \tag{6.59}$$

and

$$f(x)(k) = \theta_0^{(\infty)}(x(k)) + \theta_1^{(\infty)}(x(k))\delta^2. \quad \blacksquare \tag{6.60}$$

VI.2. Algorithm to Generate Recurrence Relations

In Subsection II.3, we limited our expansions to $O(\delta^2)$ terms. In this subsection we present an algorithm (implemented in Maple) to compute the recurrence relations for the terms of the power series expansions including higher powers of δ

$$g_j(s_j)(k) = \sum_{i=0}^{I} \gamma_{i,j}(s_j)(k)\delta_j^{2i}, \tag{6.61}$$

$$f_j(s_j)(k) = \sum_{i=0}^{I} \theta_{i,j}(s_j)(k)\delta_j^{2i}, \tag{6.62}$$

$$\tilde{d}_j(s_j)(k) = \sum_{i=0}^{I} \eta_{i,j}(s_j)(k)\delta_j^{2i}. \tag{6.63}$$

```
ord := 2:
ge := sum((SG(i,x) + h*DG(i,x)/2)*h^(2*i), i = 0..ord):
go := sum((SG(i,x) - h*DG(i,x)/2)*h^(2*i), i = 0..ord):
fe := sum((SF(i,x) + h*DF(i,x)/2)*h^(2*i), i = 0..ord):
fo := sum((SF(i,x) - h*DF(i,x)/2)*h^(2*i), i = 0..ord):
QG := (subs(x = s + h/2*d,ge) - subs(x = s - h/2*d,go))/h:
QF := (subs(x = s + h/2*d,fe) + subs(x = s - h/2*d,fo))/2:
dsub := {d = sum(d(i)*(2*h)^(2*i), i=0..ord)}:
eq1 := taylor(subs(dsub, QG - QF), h, 2*ord + 2):
solve(coeff(eq1, h, 0), d(0));
newf := taylor(subs(h = h/2, subs(dsub, QF)), h, 2*ord + 2):
coeff(newf, h, ord);
PG := (subs(x = s + h/2*d, ge) + subs(x = s - h/2*d, go))/2 +
      h/4*(subs(x = s + h/2*d, fe) - subs(x = s - h/2*d,fo)):
newg := taylor(subs(h = h/2, subs(dsub, PG)), h, 2*ord + 2):
coeff(newg, h, ord);
```

FIG. 6. Maple code to compute recurrence relations for coefficients up to any specified order in series expansions of g and f. The specified order for the example is ord: = 2. The variable h stands for the δ used in the text.

In other words, if we group the terms in g_j, f_j, and $\tilde{d}_j$ by their order in δ_j and if we stipulate that the terms in g_{j-1} and f_{j-1} must be grouped in the same fashion, then we can determine the recurrence relations for the coefficients $\gamma_{i,j-1}(s_{j-1})(k)$ $(i = 0, \ldots, I)$ in the series expansion of g_{j-1} (and similarly for the coefficients $\theta_{i,j-1}$).

In the program shown in Fig. 6, we first specify the order I of the expansions. In the example program the order is four. Next the four quantities `ge`, `go`, `fe`, and `fo` are defined. Notice that we are using the fact that

$$(\mathbf{S}g)(x)(k) = \frac{1}{2}\,(g(x)(2k+1) + g(x)(2k))$$

$$(\mathbf{D}g)(x)(k) = \frac{1}{\delta}\,(g(x)(2k+1) - g(x)(2k))$$

to express `ge` $= g(x)(2k)$, the even-numbered values of $g(x)$, and `go` $= g(x)(2k+1)$, the odd-numbered values. The step-size δ is accorded the variable `h` in the program. Next we form the two sides of the equation `QG - QF = 0` which determines $\hat{d}$; at the same time we substitute `x(2k + 1) = s(k) + h/2d(k)` and `x(2k) = s(k) - h/2d(k)` into `ge` and `fe` (respectively, `go` and `fo`). Into the expression `QG - QF`, we substitute the series expansion for $\tilde{d}$,

```
d=sum(d(i)*(2*h)(2*i),    i=0 . . . ord).
```

We expand the expression `QG - QF` in a Taylor series and we peel off the zeroth-order coefficient in `h` and solve for `d(0)`, which gives us the first term in our expansion for $\tilde{d}$. This is the recurrence relation for $\tilde{\eta}_0$. To determine higher order terms in the expansion of $\tilde{d}$, we use, for example,

```
simplify(solve(coeff(eq1, h, 2), d(1)));
```

Recall that the recurrence relation for f_j is $f_j = \mathbf{S}f_{j+1}$ and notice that $\mathbf{S}f_{j+1}$ is the same as `QF` so we simply substitute the expansion for $\tilde{d}$ into `QF`. Then we let `h = h/2` to adjust the resolution size for the next step and finally expand the expression in a Taylor series. (Recall that g_{j+1} and f_{j+1} are expanded in powers of $\delta_{j+1} = \delta_j/2$ and g_j and f_j are expanded in powers of δ_j.) To determine the recurrence relation for the coefficient $\theta_i(s)(k)$, we peel off the ith coefficient (for $i \leq$ `ord`):

```
coeff(newf, h, i);.
```

The recurrence relation for g_j is given by $g_j = \mathbf{S}g_{j+1} + h^2/4\mathbf{D}f_{j+1}$ which we denote by `PG`. Again we substitute `x(2k + 1) = s(k) + h/2d(k)` and `x(2k) = s(k) − h/2d(k)` into `ge` and `fe` (respectively, `go` and `fo`) and we substitute the expansion for $\tilde{d}$ into `PG`. Finally we rescale `h` and expand `PG` in a Taylor series. We determine recurrence relations for $\gamma_i(s)(k)$ in the same fashion as before:

```
coeff(newg, h, i);.
```

We should point out that this is an algorithm for determining the recurrence relation for the coefficients in the series (6.61)–(6.63); however, it does not give a closed form for the recurrence relations.

VII. APPENDIX B

A multiresolution analysis (MRA) of $L^2([0, 1])$ is a decomposition of the space into a chain of closed subspaces

$$V_0 \subset V_1 \subset \cdots \subset V_n \cdots$$

such that

$$\overline{\bigcup_{j\geq 0} V_j} = L^2([0, 1])$$

and

$$\bigcap_{j\geq 0} V_j = \{V_0\}.$$

If we let P_j denote the orthogonal projection operator onto V_j, then $\lim_{j\to\infty} P_j f = f$ for all $f \in L^2([0, 1])$. We have the additional requirements that each subspace V_j $(j > 0)$ is a rescaled version of the base space V_0:

$$f \in V_j \Leftrightarrow f(2^j \cdot) \in V_0.$$

Finally, we require that there exists $\phi \in V_0$ (called the scaling function) so that ϕ forms an orthonormal basis of V_0. We can conclude that the set $\{\phi_{j,k} | k = 0, \ldots, 2^j -$

1} is an orthonormal basis for each subspace V_j. Here $\phi_{j,k}$ denotes a translation and dilation of ϕ:

$$\phi_{j,k} = 2^{j/2}\phi(2^j x - k).$$

As a consequence of the above properties, there is an orthonormal wavelet basis

$$\{\psi_{j,k} | \, j \geq 0, \, k = 0, \ldots, 2^j - 1\}$$

of $L^2([0, 1])$, $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$, such that for all f in $L^2([0, 1])$

$$P_{j+1} f = P_j f + \sum_{k=0}^{2^j - 1} \langle f, \psi_{j,k} \rangle \psi_{j,k}.$$

If we define W_j to be the orthogonal complement of V_j in V_{j+1}, then

$$V_{j+1} = V_j \oplus W_j.$$

We have, for each fixed j, an orthonormal basis $\{\psi_{j,k} | k = 0, \ldots, 2^j - 1\}$ for W_j. Finally, we may decompose $L^2([0, 1])$ into a direct sum

$$L^2([0, 1]) = V_0 \bigoplus_{j \geq 0} W_j.$$

The operator Q_j is the orthogonal projection operator onto the space W_j.

The Haar wavelet ψ and its associated scaling function ϕ are defined as follows:

$$\phi(x) = \begin{cases} 1, & x \in [0,1) \\ 0, & \text{elsewhere} \end{cases} \quad \text{and} \quad \psi(x) = \begin{cases} 1, & x \in [0, 1/2) \\ -1, & x \in [1/2, 1) \\ 0, & \text{elsewhere.} \end{cases}$$

REFERENCES

1. A. Askar, B. Space, and H. Rabitz, The subspace method for long time scale molecular dynamics, preprint, 1995.
2. C. M. Bender and S. A. Orszag, "Advanced Methods for Scientists and Engineers," McGraw–Hill, New York, 1978.
3. A. Bensoussan, P. L. Lions, and G. Papnicolaou, "Asymptotic Analysis for Periodic Structures," North-Holland, The Netherlands, 1978.
4. G. Beylkin, R. Coifman, and V. Rohklin, Fast wavelet transforms and numerical algorithms, I, *Comm. Pure Appl. Math.* **44** (1991).
5. G. Beylkin and N. Coult, A multiresolution strategy for reduction of elliptic PDE's and eigenvalue problems, preprint, 1996.
6. F. Bornemann and C. Schütte, Homogenization of hamiltonian systems with a strong constraining potential, *Phys. D* **102,** Nos. 1–2 (1997).
7. M. E. Brewster and G. Beylkin, A multiresolution strategy for numerical homogenization, *Appl. Comput. Harmon. Anal.* **2** (1995).

8. R. Coifman, P. L. Lions, Y. Meyer, and S. Semmes, Compensated compactness and hardy spaces, *J. Math. Pures Appl.* **72** (1993).
9. M. Dorobantu, "Wavelet-Based Algorithms for Fast PDE Solvers," Ph.D. thesis, Royal Institute of Technology, Stockholm University, 1995.
10. A. C. Gilbert, A comparison of multiresolution and classical one-dimensional homogenization schemes, *Appl. Comput. Harmon. Anal.* **5** (1998), 1–35.
11. A. C. Gilbert, "Multiresolution Homogenization Schemes for Differential Equations and Applications," Ph.D. thesis, Princeton University, 1997.
12. V. V. Jikov, S. M. Kozlov, and O. A. Oleinik, "Homogenization of Differential Operators and Integral Functionals," Springer-Verlag, New York, 1994.
13. J. Kervorkian and J. D. Cole, "Perturbation Methods in Applied Mathematics," Springer-Verlag, New York, 1985.
14. P. A. Lagerstrom, "Matched Asymptotic Expansions, Ideas and Techniques," Springer-Verlag, New York, 1988.
15. F. Murat, Compacité par compensation, *Ann. Scuola Norm. Sup. Pisa Cl. Sci. (4)* **5** (1978).
16. C. Schütte and F. Bornemann, Homogenization approach to smoothed molecular dynamics, preprint, 1997.
17. B. Space, H. Rabitz, and A. Askar, Long time scale molecular dynamics subspace integration method applied to anharmonic crystals and glasses, *J. Chem. Phys.* **99,** No. 11 (1993).
18. L. Tartar, Compensated compactness and applications to partial differential equations, *in* "Heriot-Watt Sympos., IV, 1979."