INTERNATIONAL CENTRE FOR MECHANICAL SCIENCES

COURSES AND LECTURES - No. 343

COMPUTERIZED SYMBOLIC MANIPULATION IN MECHANICS

EDITED BY

E. KREUZER
TECHNICAL UNIVERSITY OF HAMBURG-HARBURG

SPRINGER - VERLAG WIEN - NEW YORK

PERTURBATION METHODS AND COMPUTER ALGEBRA IN MECHANICS

R.H. Rand
Cornell University, Ithaca, NY, USA

OUTLINE

1. INTRODUCTION

Perturbation methods are a collection of diverse schemes for obtaining approximate analytic solutions to problems which involve a small parameter ϵ, usually in the form of a power series in ϵ. In these notes we will discuss computer algebra treatments of regular perturbations (section 2), and of three singular perturbation methods (composite expansions, section 4, averaging, section 5 and two variable expansion method, section 8).

Computer algebra has had a profound effect on the computations associated with perturbation methods [8,11]. Compared with hand calculations, computer algebra offers improvements in accuracy and speed, as well as in ease of computation. In particular, computer algebra has made major contributions to perturbation methods in the following areas:

a. Obtaining a large number of terms. This raises the question of the convergence of such series, and of the use of techniques for accelerating convergence [14], such as Pade approximants (section 3).

b. Obtaining solutions involving many unevaluated parameters. This permits the search for special relationships between the parameters, i.e. bifurcation theory. E.g. a curve in a parameter plane may separate systems into sets which exhibit qualitatively distinct behavior. A specific example is the Hopf bifurcation (section 6) which gives conditions under which a periodic motion called a limit cycle can be born in a system of nonlinear ODE's.

c. Manipulating special functions, such as Jacobian elliptic functions (section 7), which involve identities so complicated that hand calculation is difficult and discouraging.

2. REGULAR PERTURBATIONS: MATHIEU'S EQUATION

We begin with a mechanics problem: The linear stability of the upside-down pendulum under vertical forcing. Let θ be the angle between the pendulum and the upwards vertical direction. Then the governing linearized differential equation is

$$\frac{d^2\theta}{dt^2} - \left[\frac{g}{L} - \frac{A}{L} \cos t \right] \theta = 0 \tag{2.1}$$

where g is the acceleration of gravity, L is the pendulum's length, and A cos t is the displacement of the pendulum pivot above a fixed datum. The equilibrium at $\theta = 0$ will be said to be stable if all solutions to (2.1) are bounded, and unstable if an unbounded solution exists.

Eq.(2.1) is a special case of Mathieu's equation,

$$\frac{d^2x}{dt^2} + \left[\delta + \epsilon \cos t \right] x = 0 \tag{2.2}$$

in which $\delta = -\frac{g}{L}$ and $\epsilon = \frac{A}{L}$. As is well known [13], for small ϵ the stability chart in the δ–ϵ plane contains smooth "transition" curves separating regions of stability from regions of instability. Pairs of these curves emanate from the positive δ axis at $\delta = \frac{n^2}{4}$, n=1,2,3,..., in Arnold tongues of instability. In addition, there is a single transition curve which passes through $\delta = 0$, $\epsilon = 0$, tangent to the ϵ axis. Since for the forced pendulum we are interested in values of δ which are negative, it is the latter transition curve which we shall be interested in.

From Floquet theory [13] we learn that on these transition curves there exist periodic solutions of the period of the forcer, or of twice the period of the forcer, i.e., of period 2π or 4π. In order to obtain an approximate expression for the transition curve through $\delta = \epsilon = 0$, we shall expand δ and x in a power series in ϵ, a procedure called regular perturbations. (If ϵ appears in the approximate solution in any manner besides that of a power series, the procedure is called singular perturbations.) We set

(2.3) $$\delta = \delta_1 \epsilon + \delta_2 \epsilon^2 + O(\epsilon^3)$$

(2.4) $$x = x_0 + x_1 \epsilon + x_2 \epsilon^2 + O(\epsilon^3)$$

where δ_i are constant and x_i are functions of t. We substitute (2.3),(2.4) into (2.2), collect terms and equate to zero coefficients of ϵ^n for n = 0,1,2,... . This gives:

(2.5) $$\epsilon^0: \quad \ddot{x}_0 = 0$$

(2.6) $$\epsilon^1: \quad \ddot{x}_1 = - \delta_1 x_0 - x_0 \cos t$$

(2.7) $$\epsilon^2: \quad \ddot{x}_2 = - \delta_2 x_0 - \delta_1 x_1 - x_1 \cos t$$

From (2.5) we take $x_0(t) = 1$ for a periodic solution (which may be normalized to unity because eq.(2.2) is linear.) Substituting this into (2.6), we see that $x_1(t)$ will not be periodic unless $\delta_1 = 0$, whereupon $x_1(t) = \cos t$. Substituting all these results into (2.7) gives

(2.8) $$\ddot{x}_2 = - \delta_2 - \cos^2 t$$

Trigonometric reduction of (2.8) gives

$$\ddot{x}_2 = -\delta_2 - \frac{1}{2} - \frac{1}{2}\cos 2t \qquad (2.9)$$

From (2.9) we see that $x_2(t)$ will not be periodic unless $\delta_2 = -\frac{1}{2}$. Thus we have derived the following approximate expression for the desired transition curve:

$$\delta = -\frac{1}{2}\epsilon^2 + O(\epsilon^3) \qquad (2.10)$$

Returning to the upside-down pendulum, eq.(2.1), we see that for small values of g/L, the pendulum can be made stable by choosing A/L appropriately.

By using computer algebra we can formalize this process and thereby obtain many terms of the foregoing series:

$$(2.11) \qquad \delta = -\frac{\epsilon^2}{2} + \frac{7\epsilon^4}{32} - \frac{29\epsilon^6}{144} + \frac{68687\epsilon^8}{294912} - \frac{123707\epsilon^{10}}{409600}$$

$$+ \frac{8022167579\epsilon^{12}}{19110297600} - \frac{286241141477\epsilon^{14}}{468202291200} + \frac{7534554811777337\epsilon^{16}}{8182428094955520}$$

$$- \frac{6364218991597629 6887\epsilon^{18}}{4473742560916930 5600} + \frac{401163280882921989217530 1\epsilon^{20}}{178949702436677222400000 0}$$

$$- \frac{1295373849349047382465956 77\epsilon^{22}}{3608818999139657318400000 00}$$

$$+ \frac{418327797107704762506288342556363\epsilon^{24}}{718391719355135367881687040000 00}$$

$$- \frac{231905723577803781627393585040227 1\,\epsilon^{26}}{24281640114203575434401021952000 0} + \frac{7803188090607711909380744120488641420 7\,\epsilon^{28}}{49354681832129341475552892026880000 00} - \frac{98878725199690876254384088949526344752247 63\,\epsilon^{30}}{3747871151627321868299797738291200000000 0} + \ldots$$

Next we present the record of a **MACSYMA** run in which the steps which we accomplished by hand to $O(\epsilon^2)$ above are repeated using computer algebra. This run is accomplished in an interactive mode, i.e., the instructions are typed from the keyboard. After this run we will present a program which automates the process.

```
(C4) X0:1;
(D4)                                1
(C5) DELTA:DELTA1*E+DELTA2*E^2;

                                   2
(D5)                     DELTA2 E   + DELTA1 E

(C6) DEPENDS([X1,X2],T);
(D6)                        [X1(T), X2(T)]
(C7) X:X0+X1*E+X2*E^2;
                              2
(D7)                         E   X2 + E X1 + 1

(C8) DIFF(X,T,2)+(DELTA+E*COS(T))*X;

            2
         2 d X2                         2
(D8) E   ----- + (E COS(T) + DELTA2 E   + DELTA1 E)
             2
           dT
                                                                    2
                                               2                  d X1
                                             (E   X2 + E X1 + 1) + E ----
                                                                       2
                                                                     dT
```

```
(C9) DE:TAYLOR(%,E,0,2);

                2
               d X1
(D9)/T/      (----- + COS(T) + DELTA1) E
                  2
               dT

                     2
                    d X2                                           2
             + (----- + (COS(T) + DELTA1) X1 + DELTA2) E  + . . .
                       2
                    dT

(C10) EQ1:COEFF(DE,E);

                              2
                             d X1
(D10)/R/                     ----- + COS(T) + DELTA1
                                2
                             dT

(C11) EQ2:COEFF(DE,E,2);

                   2
                  d X2
(D11)/R/          ----- + (COS(T) + DELTA1) X1 + DELTA2
                     2
                  dT

(C12) DELTA1:0;
(D12)                                 0
(C13) EV(EQ1);
                                   2
                                  d X1
(D13)/R/                          ----- + COS(T)
                                     2
                                  dT

(C14) ODE2(%,X1,T);
(D14)                    X1 = COS(T) + %K2 T + %K1

(C15) X1:RHS(%),%K1:%K2:0;
(D15)                                COS(T)
(C16) EV(EQ2);
                            2
                           d X2        2
(D16)/R/                   ----- + COS  (T) + DELTA2
                              2
                           dT
```

```
(C17) EXPAND(TRIGREDUCE(EXPAND(%)));
```

$$\text{(D17)}\qquad \frac{d^2 X2}{dT^2} + \frac{COS(2\ T)}{2} + DELTA2 + \frac{1}{2}$$

```
(C18) DELTA2:-1/2;
```

$$\text{(D18)}\qquad -\frac{1}{2}$$

```
(C19) EV(DELTA);
```

$$\text{(D19)}\qquad -\frac{E^2}{2}$$

One advantage of converting the above run to a program is that the order of truncation may be controlled by assigning a variable (called "trunc" below) which adjusts the size of sums and DO loops. Here is an outline of how the program works, together with the corresponding lines in the above run:

1. Set up expansions (C4–C7)
2. Plug into D.E. and collect terms (C8–C11)

DO LOOP:

3. Apply trig identities (C17)
4. Pick off "secular" terms (which produce non-periodic solutions) and find δ_i (C18)
5. Use O.D.E. package to solve for x_i (C12–C16)

END OF DO LOOP

6. Output results (C19)

Here is the program:

```
/* set up expansions */
trunc:read("truncation order")$
depends(x,t);
xx:sum(x[i]*e^i,i,0,trunc);
delta:sum(del[i]*e^i,i,0,trunc);
results:[x[0]=1,del[0]=0];
/* prepare null function for later */
null(t):=0;

/* plug into d.e. and collect terms */
de1:diff(xx,t,2)+(delta+e*cos(t))*xx$
de2:taylor(de1,e,0,trunc)$
for i:0 thru trunc do eq[i]:coeff(de2,e,i)$

/* main loop */
for i:1 thru trunc do (
temp1:ev(eq[i],results),

/* apply trig identities */
temp2:expand(trigreduce(expand(temp1))),

/* pick off secular terms */
temp3:subst(null,cos,temp2),
temp4:ev(temp3,x[i]=0),

/* solve for del[i] */
temp5:solve(temp4,del[i]),
results:append(results,temp5),
temp6:ev(temp2,results),
temp7:ratsimp(temp6),

/* use o.d.e. package to find x[i] */
temp8:ode2(temp7,x[i],t),
/* zap arbitrary constants */
temp9:ev(temp8,%k1=0,%k2=0),
results:append(results,[temp9]))$
/* end of main loop */

/* output results */
deltafinal:ev(delta,results);
```

And here is a sample run:

```
(C3) /* set up expansions */
trunc:read("truncation order")$
truncation order
8;

(C4) depends(x,t);

(D4)                    [X(T)]

(C5) xx:sum(x[i]*e^i,i,0,trunc);

           8         7         6         5
(D5) X    E  + X    E  + X    E  + X    E
      8         7         6         5

           4         3         2
     + X    E  + X    E  + X    E  + X    E + X
        4         3         2         1        0

(C6) delta:sum(del[i]*e^i,i,0,trunc);

      8          7          6
(D6) E  DEL   + E  DEL   + E  DEL
           8          7          6

  5          4          3          2
+ E  DEL   + E  DEL   + E  DEL   + E  DEL
        5          4          3          2

+ E DEL   + DEL
       1       0

(C7) results:[x[0]=1,del[0]=0];

(D7)          [X  = 1, DEL  = 0]
                0          0

JUMP TO END OF RUN...

/* output results */
deltafinal:ev(delta,results);

                  8        6       4    2
         68687 E     29 E     7 E     E
(D13)    -------- - ------ + ----- - --
          294912      144      32     2
```

3. PADE APPROXIMANTS

Pade approximants are a tool for improving the convergence of perturbation series [1]. Given a truncated power series of the form

$$f = a_0 + a_1 \epsilon + a_2 \epsilon^2 + \cdots + a_n \epsilon^n \tag{3.1}$$

a Pade approximant is a <u>rational</u> <u>fraction</u> which has the same truncated Taylor series as f up to $O(\epsilon^n)$. For example, the truncated series

$$1 + \frac{\epsilon}{2} - \frac{\epsilon^2}{8} \text{ has the Pade approximant } \frac{3\epsilon + 4}{\epsilon + 4} \tag{3.2}$$

since a Taylor expansion gives

$$\frac{3\epsilon + 4}{\epsilon + 4} = 1 + \frac{\epsilon}{2} - \frac{\epsilon^2}{8} + \frac{\epsilon^3}{32} - \frac{\epsilon^4}{128} + \cdots \tag{3.3}$$

The process of finding a Pade approximant involves solving a system of linear equations [1], but **MACSYMA** can do this for you automatically.

As a model example, suppose that the function $\sqrt{1+\epsilon}$ plays the role of an exact solution to a perturbation problem. We normally do not have access to the exact solution, but rather we solve for its perturbation expansion, the role of which in this case is played by the following Taylor series:

$$\sqrt{1+\epsilon} = 1 + \frac{\epsilon}{2} - \frac{\epsilon^2}{8} + \frac{\epsilon^3}{16} - \frac{5}{128}\epsilon^4 + \frac{7}{256}\epsilon^5 - \frac{21}{1024}\epsilon^6 + \cdots \tag{3.4}$$

Now this series diverges for $|\epsilon|>1$ (since $\sqrt{1+\epsilon}$ has a branch point at $\epsilon = -1$). For example, at $\epsilon = 2$, the "exact solution" gives $\sqrt{1+\epsilon} = \sqrt{3} = 1.7320$, while the "perturbation solution" gives

$1+\frac{\epsilon}{2}-\cdots-\frac{21}{1024}\epsilon^6=0.9375$. Now we attempt to improve the convergence of the series by using a Pade approximant. In **MACSYMA**, the command is

(3.5) PADE (taylor series, m, n);

where m is the degree of the polynomial in the numerator of the Pade approximant, and n is the corresponding value for the denominator. These should be chosen so that m+n equals the degree of the truncated Taylor series. In this example we take m = n = 3 and obtain:

```
(C4) sqrt(1+e);
(D4)          SQRT(E + 1)
(C5) taylor(%,e,0,6);
                   2     3      4      5      6
              E   E     E    5 E    7 E   21 E
(D5)/T/ 1 + - - -- + -- - ---- + ---- - ---- + . . .
              2   8    16   128    256   1024
(C6) pade(%,3,3);
          3       2
       7 E  + 56 E  + 112 E + 64
(D6)  [-------------------------]
        3       2
       E  + 24 E  + 80 E + 64
(C7) [D4,D5,D6],e = 2,numer;
(D7)  [1.73205, 0.9375, [1.73171]]
```

That is, at $\epsilon = 2$ the Pade approximant gives 1.73171, and thus very nearly recovers the exact solution from the divergent perturbation series!

Besides offering an improved approximation, there is another important use for Pade approximants. The <u>zeros</u> of the <u>denominator</u> of the Pade approximant often give useful information about the <u>location</u> and nature of the nearest <u>singularity</u> in the perturbation solution. To continue with the above model example, the zeros of the denominator of the Pade approximant in (D6), $\epsilon^3+24\epsilon^2+80\epsilon+64=0$, all lie on the negative ϵ axis:

```
(C8) DENOM(PART(D6,1));

              3        2
(D8)        E  + 24 E  + 80 E + 64

(C9) ALLROOTS(%);

(D9) [E = - 1.23191, E = - 2.57242, E = - 20.19566]
```

It turns out that the closest zero to the origin ($\epsilon=0$) approaches the branch point $\epsilon=-1$ as the number of terms increases.

My experience is that Pade approximants often give excellent results in real perturbation problems. As an example, we take the transition curve in the Mathieu equation that we found at the end of the previous section. Numerical integration of the differential equation (2.2) shows that for $\delta = -0.4$, for example, the transition curve passes close to $\epsilon = 1.035$. Substituting this value of ϵ into the $O(\epsilon^8)$ perturbation expansion for the transition curve gives

$$\delta = -\frac{\epsilon^2}{2} + \frac{7\epsilon^4}{32} - \frac{29\epsilon^6}{144} + \frac{68687\epsilon^8}{294912} = -0.225 \text{ for } \epsilon = 1.035$$

This value is far from the numerically-obtained value of $\delta = -0.4$. Now we use Pade approximants:

```
(C7) TC;
              2      4       6          8
             E    7 E    29 E    68687 E
(D7)/T/   - -- + ---- - ----- + -------- + . . .
             2    32     144     294912
```

```
(C8) PADE(TC,4,4);
(D8)
                      4              2
            2092824 E   + 2244096 E
     [- ———————————————————————————————————]
                4              2
        882545 E   + 6149232 E   + 4488192
```

which gives the value $\delta = -0.398$ for $\epsilon = 1.035$, which is very close to the numerical value of $\delta = -0.4$.

We may also use the Pade approximant to estimate the radius of convergence of the perturbation power series:

```
(C9) DENOM(PART(D8,1));

                4              2
(D9) 882545 E   + 6149232 E   + 4488192
(C10) ALLROOTS(%);
(D10) [E = 0.91014 %I, E = - 0.91014 %I,
       E = 2.47775 %I, E = - 2.47775 %I]
```

The closest root of the denominator of the Pade approximant to the expansion point $\epsilon=0$ is a distance 0.91 away. Thus we expect a radius of convergence in ϵ of about 0.9. Taking more terms gives a value of about 0.73. Thus the value we used above, namely $\epsilon = 1.035$, was outside the radius of convergence, which explains why the perturbation series did not deliver a meaningful value. However, note how well the Pade approximant improved the convergence of this divergent series.

4. METHOD OF COMPOSITE EXPANSIONS (MATCHED ASYMPTOTICS) [9]

The method of composite expansions is a singular perturbation method which is an alternative to matched aymptotic expansions for differential equations of the form:

$$\epsilon y'' + a(x) y' + b(x) y = 0 \tag{4.1}$$

with two boundary conditions:

$$y(0) = \alpha, \quad y(1) = \beta, \tag{4.2}$$

where primes represent differentiation with respect to x. In (4.1), $\epsilon << 1$ and a(x) and b(x) are analytic functions of x on [0,1]. We assume that $a(x) > 0$ throughout [0,1], in which case a boundary layer (b.l.) will occur at $x = 0$. (In the opposite case that $a(x) < 0$, the b.l. will occur at $x = 1$, but changing independent variable to $\xi = 1-x$ will bring the b.l. to $\xi = 0$. We assume that a(x) does not change sign in [0,1], since this case leads to more complicated behavior (internal b.l.'s), see [1,7].)

The following description of the method is based on Nayfeh [7], pp.148–149: We look for a solution to (4.1) and (4.2) in the form:

$$y = \sum_{n=0}^{\infty} \epsilon^n f_n(x) + e^{-g(x)/\epsilon} \sum_{n=0}^{\infty} \epsilon^n h_n(x) \tag{4.3}$$

in which the functions f_n, g and h_n are to be found. We require $g(x) \sim x$ in the limit $x \to 0$, since the b.l. is known to have thickness x. The procedure is to substitute (4.3) into (4.1) and (4.2), collect terms, and equate the coefficients of ϵ^n and $\epsilon^n e^{-g/\epsilon}$ to zero. We proceed directly to an example [3]:

(4.4) $$\epsilon\, y'' + (2x+1)\, y' + 2\, y = 0$$

for which $a(x) = 2x+1$ and $b(x) = 2$. Substituting (4.3) into (4.4) and multiplying by ϵ gives the following lowest order terms:

(4.5) $e^{-g/\epsilon}$: $$g'\, h_0\, (g' - 2x - 1) = 0$$

(4.6) $\epsilon\, e^{-g/\epsilon}$: $$h_0\, (2 - g'') + h_0'\, (2x + 1 - 2g') = 0$$

(4.7) ϵ: $$2\, f_0 + (2x + 1)\, f_0' = 0$$

Substituting (4.3) into the b.c. (4.2) gives:

(4.8) $$f_0(0) + h_0(0) = \alpha \quad \text{and} \quad f_0(1) = \beta$$

Eq.(4.5) gives either $h_0 = 0$, g = constant (both of which must be rejected), or:

(4.9) $$g(x) = x^2 + x$$

in which the arbitrary constant of integration has been taken as zero in order that $g(x) \sim x$. Substituting (4.9) into (4.6) gives

(4.10) $$h_0' = 0 \quad \text{or} \quad h_0(x) = \text{constant}.$$

Next, (4.7) may be integrated to give:

(4.11) $$f_0(x) = \frac{\text{constant}}{2x + 1}.$$

The arbitrary constants of integration in (4.10) and (4.11) are found by using the b.c.(4.8):

(4.12) $$f_0(x) = \frac{3\beta}{2x+1}, \quad h_0(x) = \alpha - 3\beta$$

Substituting (4.9) and (4.12) into (4.3) gives the approximate result:

(4.13) $$y = \frac{3\beta}{2x+1} + (\alpha - 3\beta)\, e^{-\frac{x^2+x}{\epsilon}} + O(\epsilon)$$

This type of computation may be automated using computer algebra. The **MACSYMA** program "composite" which we present here accomplishes this task for general functions a(x) and b(x), to arbitrary order of truncation. Here is a sample run on the previous example:

```
composite();
The d.e. is: ey''+a(x)y'+b(x)y=0
with b.c. y(0)=y0 and y(1)=y1
enter a(x) > 0 on [0,1]
2*x+1;
enter b(x)
2;
enter y0
alpha;
enter y1
beta;

The d.e. is: ey''+( 2 x + 1 )y'+( 2 )y=0
with b.c. y(0)= alpha and y(1)= beta

enter truncation order
3;

                     3                 2
     85312 beta e       928 beta e        16 beta e
(- ------------- - -------------- - ----------
          243                27                3

                          2
                         x  + x
                      - --------
                           e
 - 3 beta + alpha) %e

   3             6                5                 4
 - e  (5120 beta x  + 15360 beta x  + 21504 beta x
```

$$+ 17408\ \mathrm{beta}\ x^3 + 16032\ \mathrm{beta}\ x^2 + 9888\ \mathrm{beta}\ x$$

$$- 85312\ \mathrm{beta})/(31104\ x^7 + 108864\ x^6 + 163296\ x^5$$

$$+ 136080\ x^4 + 68040\ x^3 + 20412\ x^2 + 3402\ x + 243)$$

$$- e^2\ (128\ \mathrm{beta}\ x^4 + 256\ \mathrm{beta}\ x^3 + 336\ \mathrm{beta}\ x^2$$

$$+ 208\ \mathrm{beta}\ x - 928\ \mathrm{beta})/(864\ x^5 + 2160\ x^4 + 2160\ x^3$$

$$+ 1080\ x^2 + 270\ x + 27)$$

$$- \frac{e\ (8\ \mathrm{beta}\ x^2 + 8\ \mathrm{beta}\ x - 16\ \mathrm{beta})}{24\ x^3 + 36\ x^2 + 18\ x + 3} + \frac{3\ \mathrm{beta}}{2\ x + 1}$$

In order to check the accuracy of the results, we used finite differences to obtain a numerical solution to (4.4),(4.2) for $\alpha = 1$, $\beta = 2$. The following table lists the values of y'(0) obtained from various truncations of the composite expansion, along with the finite difference value, for $\epsilon = 0.01, 0.05$ and 0.09:

<u>Table of approximate values of y'(0)</u>

truncation order	$\epsilon = 0.01$	$\epsilon = 0.05$	$\epsilon = 0.09$
0	488	88.0	43.55
1	497.97	95.2	47.98
2	498.59	96.88	48.48
3	498.65	97.40	46.94
finite diff.	498.66	98.64	54.16

Note that while additional accuracy is obtained by taking more terms in the expansion for $\epsilon = 0.01$ and $\epsilon = 0.05$, this is no longer true for $\epsilon = 0.09$. This behavior is typical of asymptotic series (as opposed to convergent series.) The two limiting processes of letting $\epsilon \to 0$ and truncating the series after $N \to \infty$ terms are in competition. In an asymptotic series, we fix N and let $\epsilon \to 0$, while in a convergent series, we fix ϵ and let $N \to \infty$. In terms of the Table, if we fix the truncation order at N = 3 and take ϵ smaller, the difference between the asymptotic series approximation and the exact solution (represented by the finite differences result) becomes smaller. If we fix ϵ, however, we cannot expect that increasing the order of truncation N will necessarily increase the accuracy of the approximation, cf. $\epsilon = 0.09$.

Here is the program listing:

```
/* method of composite expansions */

composite():=(

/* input problem from keyboard */
print("The d.e. is: ey''+a(x)y'+b(x)y=0"),
print("with b.c. y(0)=y0 and y(1)=y1"),
a:read("enter a(x) > 0 on [0,1]"),
b:read("enter b(x)"),
y0:read("enter y0"),
y1:read("enter y1"),
print("The d.e. is: ey''+(",a,")y'+(",b,")y=0"),
print("with b.c. y(0)=",y0,"and y(1)=",y1),
trunc:read("enter truncation order"),

/* set up basic form of solution */
y:sum(f[n](x)*e^n,n,0,trunc)+
  %e^(-g(x)/e)*sum(h[n](x)*e^n,n,0,trunc),

/* substitute into d.e. */
de1:e*diff(y,x,2)+a*diff(y,x)+b*y,

/* expand and collect terms */
de2:expand(e*de1),
gstuff:coeff(de2,%e^(-g(x)/e)),
other:expand(de2-gstuff*%e^(-g(x)/e)),
gstuff2:taylor(gstuff,e,0,trunc+1),
other2:taylor(other,e,0,trunc+1),
for i:0 thru trunc+1 do
   geq[i]:coeff(gstuff2,e,i),
for i:1 thru trunc+1 do
   otheq[i]:coeff(other2,e,i),
```

```
/* find g(x) */
geq:geq[0]/h[0](x)/diff(g(x),x),
gsol:ode2(geq,g(x),x),
resultlist:[g(x)=ev(rhs(gsol),%c=0)],

/* find h[i](x)'s */
/* hh[i] = constant of integration */
for i:0 thru trunc do(
   heq[i]:ev(geq[i+1],resultlist,diff),
   hsol[i]:ode2(heq[i],h[i](x),x),
   hsol[i]:ev(hsol[i],%c=hh[i]),
   resultlist:append(resultlist,[hsol[i]])),

/* find f[i](x)'s */
/* ff[i] = constant of integration */
for i:0 thru trunc do(
   feq[i]:ev(otheq[i+1],resultlist,diff),
   fsol[i]:ode2(feq[i],f[i](x),x),
   fsol[i]:ev(fsol[i],%c=ff[i]),
   resultlist:append(resultlist,[fsol[i]])),

/* boundary conditions */
/* y(0)=y0 and y(1)=y1 */
bc1:ev(y=y0,resultlist,x=0),
bc2:ev(y=y1,resultlist,x=1),
/* kill exponentially small %e terms */
bc2:subst(0,%e,bc2),
for i:0 thru trunc do
   (bc1[i]:ratcoef(bc1,e,i),
    bc2[i]:ratcoef(bc2,e,i)),

/* solve for unknown consts */
bceqs:makelist(bc1[i],i,0,trunc),
bceqs:append(bceqs,makelist(bc2[i],i,0,trunc)),
bcunk:makelist(hh[i],i,0,trunc),
bcunk:append(bcunk,makelist(ff[i],i,0,trunc)),
const:solve(bceqs,bcunk),

/* substitute back */
resultlist:ratsimp(ev(resultlist,const)),
ysol:ev(y,resultlist))$
```

5. METHOD OF AVERAGING: VAN DER POL'S EQUATION

The method of averaging [12] is a powerful perturbation method based on choosing a near-identity transformation to simplify the approximate equations of motion as much as possible. In contrast to regular perturbations, where the solution is expanded in a power series in ϵ, in averaging it is the near-identity transformation which is expanded in a power series in ϵ. The result of the method is a simplified differential equation rather than an approximate expression for the solution of the differential equation.

We shall use the method of averaging to treat systems of the form:

$$\ddot{z} + z = \epsilon \, F(z,\dot{z}) \tag{5.1}$$

The first step is to assume a solution in the form

$$z = a \cos \varphi \,, \quad \dot{z} = - a \sin \varphi \tag{5.2}$$

in which $a=a(t)$ and $\varphi=\varphi(t)$, i.e., the method of variation of parameters. Substituting (5.2) into (5.1) gives

$$\dot{a} = - \epsilon \; s \; F(a\; c,-a\; s) \;, \quad \dot{\varphi} = 1 - \frac{\epsilon}{a} \, c \; F(a\; c,-a\; s) \tag{5.3}$$

where $s = \sin \varphi$, $c = \cos \varphi$.

Next we posit a near-identity transformation of the form:

$$a = \bar{a} + \epsilon \, w_1 + \epsilon^2 \, v_1 + \epsilon^3 \, u_1 + O(\epsilon^4), \tag{5.4}$$

$$\varphi = \bar{\varphi} + \epsilon \, w_2 + \epsilon^2 \, v_2 + \epsilon^3 \, u_2 + O(\epsilon^4) \tag{5.5}$$

where the <u>generating functions</u> v_i, v_i and u_i depend on $\bar{a}$, $\bar{\varphi}$. We must substitute eqs.(5.4),(5.5) into eq.(5.3), expand in ϵ, collect terms, and equate to zero the coefficient of ϵ^n, for n=0,1,2,... . Then our goal is to choose the generating functions so as to simplify the resulting differential eqs. on $\bar{a}$ and $\bar{\varphi}$.

We shall illustrate the method by applying it to a particular example, van der Pol's equation:

$$\ddot{z} + z = \epsilon\, \dot{z}\, (1-z^2) \tag{5.6}$$

which is in the form of (5.1) with $F(z,\dot{z},t) = \dot{z}\,(1-z^2)$. This d.e. exhibits a periodic solution called a <u>limit cycle</u>. Making the substitution (2) we obtain the following differential eqs.(5.3), which are exact so far:

$$\dot{a} = \epsilon\, a\, s^2(1-a^2c^2) \ , \quad \dot{\varphi} = 1 + \epsilon\, c\, s(1-a^2c^2) \tag{5.7}$$

Substituting (5.4),(5.5) into (5.7), and temporarily neglecting terms of $O(\epsilon^2)$, gives

$$\dot{\bar{a}} + \epsilon\frac{\partial w_1}{\partial \bar{a}}\,\dot{\bar{a}} + \epsilon\frac{\partial w_1}{\partial \bar{\varphi}}\,\dot{\bar{\varphi}} = \epsilon\, \bar{a}\, \sin^2\bar{\varphi}\, [1-\bar{a}^2\cos^2\bar{\varphi}] + O(\epsilon^2) \tag{5.8}$$

and a similar equation on $\dot{\bar{\varphi}}$. Solving for $\dot{\bar{a}}$ and $\dot{\bar{\varphi}}$ gives

$$\dot{\bar{a}} = -\,\epsilon\frac{\partial w_1}{\partial \bar{\varphi}} + \epsilon\, \bar{a}\, \sin^2\bar{\varphi}\, [1-\bar{a}^2\cos^2\bar{\varphi}] + O(\epsilon^2) \tag{5.9}$$

$$\dot{\bar{\varphi}} = 1 - \epsilon\frac{\partial w_2}{\partial \bar{\varphi}} + \epsilon\, \sin\bar{\varphi}\, \cos\bar{\varphi}\, [1-\bar{a}^2\cos^2\bar{\varphi}] + O(\epsilon^2) \tag{5.10}$$

Trigonometric reduction of eqs.(5.9),(5.10) gives

$$\text{(5.11)}\quad \dot{\bar{a}} = -\epsilon\frac{\partial w_1}{\partial\bar{\varphi}} + \epsilon\,\bar{a}\left[\frac{1}{2} - \frac{\bar{a}^2}{8} - \frac{1}{2}\cos 2\bar{\varphi} + \frac{\bar{a}^2}{8}\cos 4\bar{\varphi}\right] + O(\epsilon^2)$$

and a similar eq. on $\dot{\bar{\varphi}}$. Now we may choose w_1 and w_2 to eliminate the trigonometric terms. From (5.11) this amounts to integrating the trig terms in $\bar{\varphi}$, holding $\bar{a}$ constant (which gives the same result as averaging these terms over one period, hence the name method of averaging):

$$\text{(5.12)}\qquad w_1 = \int \bar{a}\left[-\frac{1}{2}\cos 2\bar{\varphi} + \frac{\bar{a}^2}{8}\cos 4\bar{\varphi}\right] d\bar{\varphi}$$

$$\text{(5.13)}\qquad w_1 = -\frac{\bar{a}}{4}\sin 2\bar{\varphi} + \frac{\bar{a}^3}{32}\sin 4\bar{\varphi} + K_1$$

where K_1 is an arbitrary function of $\bar{a}$. Similarly we find

$$\text{(5.14)}\qquad w_2 = \left[-\frac{1}{4} + \frac{\bar{a}^2}{8}\right]\cos 2\bar{\varphi} + \frac{\bar{a}^2}{32}\cos 4\bar{\varphi} + K_2$$

These choices for w_1 and w_2 give the following simplified eqs. on $\dot{\bar{a}}$ and $\dot{\bar{\varphi}}$:

$$\text{(5.15)}\qquad \dot{\bar{a}} = \epsilon\,\bar{a}\left[\frac{1}{2} - \frac{\bar{a}^2}{8}\right] + O(\epsilon^2)\ ,\qquad \dot{\bar{\varphi}} = 1 + O(\epsilon^2)$$

The limit cycle in van der Pol's eq.(5.6) corresponds to the non-zero equilibrium of the $\dot{\bar{a}}$ eq.(5.15), namely $\bar{a} = 2 + O(\epsilon)$.

The process thus far is called first-order averaging. To extend the method to second-order averaging, we again substitute (5.4),(5.5) into (5.7), but this time retain terms of $O(\epsilon^2)$, and use the expressions (5.13),(5.14) for $w_1 . w_2$. This will yield eqs. on $\dot{\bar{a}}$ and $\dot{\bar{\varphi}}$ which may be simplified by choosing v_1 and v_2 to kill any trig terms, as was done for w_1 and w_2 above. The process is then continued to third-order averaging in which u_1 and u_2 are similarly determined, and so on. As a time-saving short-cut, note that for the final round of averaging, we may avoid solving for the associated generating functions. E.g., for third-order averaging, the functions u_1 and u_2 need not actually be computed, but rather may be taken into account by simply removing the $O(\epsilon^3)$ periodic terms.

Here is a **MACSYMA** program which automates the foregoing computation:

```
/* third order averaging on vdp's eq */
depends([a,phi],t)$
z:a*cos(phi)$
zd:-a*sin(phi)$
f:(1-z^2)*zd$
de1:[diff(a,t)=-e*sin(phi)*f,diff(phi,t)=1-e/a*cos(phi)*f]$
depends([w1,w2,v1,v2],[abar,phibar])$
transf:[a=abar+e*w1+e^2*v1,phi=phibar+e*w2+e^2*v2]$
depends([abar,phibar],t)$
de2:de1,transf,diff$
de3:solve(%,[diff(abar,t),diff(phibar,t)])$
de4:taylor(de3,e,0,3)$

/* first order averaging: */
ws1:ratcoef(%,e)$
expand(trigreduce(expand(%)))$
integrate(%,phibar)$
ws2:%,%integconst1:k1,%integconst2:k2$
ws3:ws2-ratcoef(ws2,phibar)*phibar$
ws4:solve(part(%,1),[w1,w2])$
de5:de4,ws4,diff$
```

```
/* second order averaging: */
vs1:ratcoef(%,e,2)$
de6:expand(trigreduce(expand(%)))$
integrate(%,phibar)$
vs2:%,%integconst3:k3,%integconst4:k4$
vs3:vs2-ratcoef(vs2,phibar)*phibar$
vs4:solve(part(%,1),[v1,v2])$
de7:de5,vs4,diff$

/* third order averaging: */
de8:expand(trigreduce(expand(%)))$
subst(s,sin,%)$
de9:subst(c,cos,%)$
c(any):=0$
s(any):=0$
de10:ev(de9);
```

The result of this program is the following simplified ("averaged") d.e.'s on $\dot{\bar{a}}$ and $\dot{\bar{\varphi}}$:

$$(5.16)\quad \frac{dABAR}{dT} = -\frac{3\ ABAR^2\ E^3\ K3}{8} + \frac{E^3\ K3}{2} - \frac{3\ ABAR\ E^3\ K1^2}{8} - \frac{3\ ABAR^2\ E^2\ K1}{8} + \frac{E^2\ K1}{2} - \frac{43\ ABAR^7\ E^3}{16384} + \frac{39\ ABAR^5\ E^3}{2048} - \frac{9\ ABAR^3\ E^3}{256} - \frac{ABAR^3\ E}{8} + \frac{ABAR\ E}{2},$$

$$\frac{dPHIBAR}{dT} = -\frac{11\ ABAR^3\ E^3\ K1}{64} + \frac{3\ ABAR\ E^3\ K1}{8} - \frac{11\ ABAR^4\ E^2}{256} + \frac{3\ ABAR^2\ E^2}{16} - \frac{E^2}{8} + 1$$

Note that the form of the averaged d.e.'s depends on the choice of the arbitrary functions of $\bar{a}$, K_1 and K_3. This interesting point has been discussed by Kahn and his associates (see e.g. [6]), who show that there are various ways to select the constants K_i. For example, if we take all $K_i=0$, then we achieve the simplest choice for the generating functions (which relate the transformed coordinates to the original coordinates). However, a judicious choice of the K_i can make the averaged eqs.(5.16) easier to handle. For example, if we take $K_1=0$ and

(5.17) $$K_3 = \frac{43\,\bar{a}^7 - 312\,\bar{a}^5 + 576\,\bar{a}^3}{-6144\,\bar{a}^2 + 8192}$$

then the averaged eqs.(5.16) become:

(5.18) $$\dot{\bar{a}} = \epsilon\,\frac{\bar{a}}{8}\,(4 - \bar{a}^2) + O(\epsilon^4), \quad \dot{\bar{\varphi}} = 1 - \epsilon^2\,\frac{11\,\bar{a}^4 - 48\,\bar{a}^2 + 32}{256} + O(\epsilon^4)$$

Note that in (5.18) the ϵ^2 term in the $\dot{\bar{a}}$ eq. has been eliminated by choosing $K_1=0$ and the ϵ^3 term has been eliminated by choosing K_3 as in eq.(5.17).

6. HOPF BIFURCATIONS

As an example of the application of computer algebra to bifurcation theory, we consider the following example [11]:

(6.1) $$\ddot{z} + z = \epsilon\left[\alpha_1\, z^2 + \alpha_2\, z\,\dot{z} + \alpha_3\,\dot{z}^2\right] + \epsilon^2\left[\mu\,\dot{z} + \beta_1\, z^3 + \beta_2\, z^2\dot{z} + \beta_3\, z\,\dot{z}^2 + \beta_4\,\dot{z}^3\right]$$

where μ, α_i, β_i are parameters, and where $\epsilon \ll 1$. As in the case of van der Pol's equation considered in the previous section, eq.(6.1) may exhibit a periodic limit cycle solution for small ϵ. We shall be interested in understanding how such a periodic solution comes to be born as the parameters are varied.

We use second-order averaging on eq.(6.1). As in the previous section, we set

(6.2) $$z = a\cos\varphi\,, \quad \dot{z} = -\,a\sin\varphi$$

and apply a near-identity transformation as in the previous section, resulting in the averaged eqs.:

$$\dot{\bar{a}} = \epsilon^2 \left[\frac{\mu \bar{a}}{2} + \frac{1}{8} \left[\beta_2 + 3\,\beta_4 + \alpha_2(\alpha_1+\alpha_3)\right] \bar{a}^3 \right] + 0(\epsilon^3) \tag{6.3}$$

and a similar eq. on $\dot{\bar{\varphi}}$ which we shall not be concerned with. A limit cycle solution to eq.(6.1) will correspond to a non-zero equilibrium solution to (6.3), i.e. to a real root $\bar{a}$ of the equation:

$$\frac{\mu}{2} + \frac{1}{8} \left[\beta_2 + 3\,\beta_4 + \alpha_2(\alpha_1+\alpha_3) \right] \bar{a}^2 = 0 \tag{6.4}$$

Thus a limit cycle is predicted to occur if $\beta_2+3\,\beta_4+\alpha_2(\alpha_1+\alpha_3)$ has the opposite sign of μ. If $\beta_2+3\,\beta_4+\alpha_2(\alpha_1+\alpha_3) \neq 0$, then a limit cycle will be born out of the origin $\bar{a} = 0$ as μ is swept through the value $\mu = 0$, a scenario which is known as a Hopf bifurcation [5].

7. AVERAGING WITH ELLIPTIC FUNCTIONS [10]

In the previous sections we applied regular perturbations and the method of averaging to d.e.'s which reduced to the simple harmonic oscillator when $\epsilon=0$, i.e. we perturbed off of sine and cosine solutions. In this section we shall apply the method of averaging to the equation

$$\ddot{x} + x + x^2 = \epsilon\, g(x,\dot{x}), \quad \epsilon<<1 \tag{7.1}$$

This involves perturbing off of the Jacobian elliptic function solutions of the equation

$$\ddot{x} + x + x^2 = 0 \tag{7.2}$$

This work illustrates the utility of computer algebra in facilitating the necessary computations.

We begin by briefly discussing Jacobian elliptic functions [2]. The functions cn and sn are the elliptic counterparts of the trigonometric functions cos and sin. Like cos, cn is even, while like sin, sn is odd. The identity

$$\text{sn}^2 + \text{cn}^2 = 1 \tag{7.3}$$

reminds us of the corresponding relation between sin and cos. The functions cn and sn are actually a family of functions which are parameterized by a "square-modulus" m:

$$\text{cn} = \text{cn}(u,m), \quad \text{sn} = \text{sn}(u,m) \tag{7.4}$$

In fact, cn and sn reduce to cos and sin for $m = 0$. The derivatives of cn and sn, however, introduce a new function, dn, which has no trigonometric counterpart:

$$\frac{d}{du}\,\text{cn} = -\,\text{sn}\,\text{dn}, \quad \frac{d}{du}\,\text{sn} = \text{cn}\,\text{dn} \tag{7.5}$$

In the trigonometric limit $m = 0$, dn reduces to unity. The function dn satisfies the following equations:

$$m\,\text{sn}^2 + \text{dn}^2 = 1, \quad \frac{d}{du}\,\text{dn} = -\,m\,\text{sn}\,\text{cn} \tag{7.6}$$

The functions cn and sn are periodic with period $4K(m)$, where $K(m)$ is a tabulated function called the complete elliptic integral of the first kind. For m going from 0 to 1, $K(m)$ goes from $\pi/2$ to infinity. The function dn has period $2K(m)$.

A related quantity is E(m), the complete elliptic integral of the second kind. For m going from 0 to 1, E(m) goes from $\pi/2$ to 1. E(m) enters this work through the integral of cn^2 over one period 4K(m):

$$\int_0^{4K} cn^2\, du = \frac{4}{m}\,[E(m) - (1-m)\,K(m)] \tag{7.7}$$

Next we obtain the solutions of eq.(7.2) which we are perturbing off of. Although the general solution to (7.2) may be found by utilizing the conservation of energy, evaluating the resulting elliptic integral and inverting, we find it more instructive to proceed as follows: Assume a solution in the form:

$$x = a_1 + a_2\, sn^2, \qquad u = \omega\, t + b \tag{7.8}$$

where sn = sn(u,m) and where a_1, a_2, ω, b and m are constants. Differentiating (7.8) and using eq.(7.5) gives

$$\dot{x} = 2\, a_2\, \dot{u}\, sn\, sn' = 2\, a_2\, \omega\, sn\, cn\, dn \tag{7.9}$$

where ' represents differentiation with respect to u. Differentiating (7.9) and using eqs.(7.2)-(7.6) gives

$$\ddot{x} = 2\, a_2\, \omega^2\, [(m-1) + 2(1-2m)\, cn^2 + 3m\, cn^4] \tag{7.10}$$

Substituting (7.10) into (7.2), collecting terms, and equating to zero the coefficients of cn^m, m=0,2,4, gives 3 eqs. for a_1, a_2 and ω, which may be solved as [10]:

(7.11) $$a_1 = \frac{1 + m - \sqrt{\lambda}}{2 \sqrt{\lambda}}$$

(7.12) $$a_2 = - \frac{3 m}{2 \sqrt{\lambda}}$$

(7.13) $$\omega = \frac{1}{2 \lambda^{1/4}}$$

where

(7.14) $$\lambda = m^2 - m + 1$$

Thus, the general solution, eq.(7.8), to eq.(7.2) involves the two arbitrary constants m and b. Here b determines the phase of the solution and m (where $0 \leq m \leq 1$) determines its amplitude.

In order to set up the method of averaging for eq.(7.1), we begin by using variation of parameters to express the effect of the order ϵ terms on the slow evolution of the square–modulus m, now considered a function of t.

We look for a solution to eq.(7.1) in the form of eq.(7.8), in which the two arbitrary constants m and b are allowed to vary in time. This results in first–order differential equations on m(t) and b(t). For brevity, we shall consider only the equation on m(t), the equilibria of which will correspond to periodic motions in eq.(7.1) (limit cycles).

Let us write the general solution, eq.(7.8), in the abstract form:

(7.15) $$x = x(t,m,b)$$

Differentiating eq.(7.15) gives

(7.16) $$\frac{dx}{dt} = \frac{\partial x}{\partial t} + \frac{\partial x}{\partial m} \frac{dm}{dt} + \frac{\partial x}{\partial b} \frac{db}{dt}$$

As usual in variation of parameters, we require that $\dot{x}$ be given by eq.(7.9), i.e.,

(7.17) $$\frac{dx}{dt} = \frac{\partial x}{\partial t}$$

giving

(7.18) $$\frac{\partial x}{\partial m}\frac{dm}{dt} + \frac{\partial x}{\partial b}\frac{db}{dt} = 0$$

Differentiating eq.(7.17) gives

(7.19) $$\frac{d^2x}{dt^2} = \frac{\partial^2 x}{\partial t^2} + \frac{\partial^2 x}{\partial m \partial t}\frac{dm}{dt} + \frac{\partial^2 x}{\partial b \partial t}\frac{db}{dt}$$

in which the unperturbed solution satisfies eq.(7.2), now written as

(7.20) $$\frac{\partial^2 x}{\partial t^2} + x + x^2 = 0$$

since m and b are held constant in eq.(7.20). In eq.(7.1), however, m and b are now permitted to depend on t, so that

(7.21) $$\frac{d^2x}{dt^2} + x + x^2 = \epsilon\ g$$

Substituting Eqs.(7.20) and (7.21) into Eq.(7.19) gives

(7.22) $$\frac{\partial^2 x}{\partial m \partial t}\frac{dm}{dt} + \frac{\partial^2 x}{\partial b \partial t}\frac{db}{dt} = \epsilon\ g$$

Eqs.(7.18) and (7.22) may be solved simultaneously for $\dot{m}$ as follows:

(7.23) $$\dot{m} = \frac{\epsilon\ g\ x_b}{x_{mt} x_b - x_{bt} x_m}$$

where subscripts represent partial differentiation.

The following **MACSYMA** program computes the partial derivatives in eq.(7.23) and simplifies the result:

```
depends([cn,sn,dn],[u,m]);
depends(u,[t,w,b]);
depends([a1,a2,w],m);
derivs:['diff(cn,u)=-sn*dn,'diff(sn,u)=cn*dn,'diff(dn,u)=-m*sn*cn,
        'diff(cn,m)=sn*dn*f,'diff(sn,m)=-cn*dn*f,
        'diff(dn,m)=m*sn*cn*f-sn^2/(2*dn)];
idents:[sn=sqrt(1-cn^2), dn= sqrt(1-m+m*cn^2)];
x:a1+a2*sn^2;
xt:diff(x,t);
xt:xt,derivs;
xbt:diff(xt,b);
xbt:xbt,derivs;
xb:diff(x,b);
xb:xb,derivs;
xm:diff(x,m);
xm:xm,derivs;
xmt:diff(xt,m);
xmt:xmt,derivs;
dmdt:-xb*e*g/(xm*xbt-xmt*xb);
dmdt:ratsimp(dmdt);
dmdt:dmdt,idents,diff;
dmdt:dmdt,derivs;
dmdt:dmdt,u=w*t+b,diff;
dmdt:dmdt,
a1 = -(sqrt(m^2-m+1)-m-1)/(2*sqrt(m^2-m+1)),
a2 = -3*m/(2*sqrt(m^2-m+1)),w = 1/(2*(m^2-m+1)^(1/4)),diff;
dmdt:factor(ratsimp(dmdt));
```

In writing the above program, the following identities were used ([2], p.283, formulas 710):

$$\frac{\partial sn}{\partial m} = -\ cn\ dn\ f,\ \frac{\partial cn}{\partial m} = cn\ dn\ f,\ \frac{\partial dn}{\partial m} = m\ sn\ cn\ f - \frac{1}{2}\frac{sn^2}{dn}$$

where

$$f = f(u,m) = \frac{E(u)-(1-m)u}{2m(1-m)} - \frac{sn\ cn}{2(1-m)dn}$$

Here is a record of the final expression produced by the program:

```
                   2                 2               2          7/4
 8 CN SQRT(1 - CN ) E G SQRT((CN  - 1) M + 1) (M  - M + 1)
 ------------------------------------------------------------------
                          3 (M - 1)
```

That is,

$$\dot{m} = -\frac{8}{3}\frac{\lambda^{7/4}}{1-m}\ sn\ cn\ dn\ \epsilon\ g \tag{7.24}$$

where $g = g(x,\dot{x}) = g(a_1+a_2\ sn^2, 2a_2\omega\ sn\ cn\ dn)$.

The right hand side (rhs) of eq.(7.24) is periodic in t and thus in the appropriate form for averaging. For first-order averaging, we may replace the rhs of eq.(7.24) by its average value taken over one period of length 4K(m):

$$\dot{m} = -\frac{8}{3}\frac{\lambda^{7/4}}{1-m}\ \epsilon\ \frac{1}{4K}\int_0^{4K} sn\ cn\ dn\ g\ du \tag{7.25}$$

where $g = g(a_1+a_2\ sn^2, 2a_2\omega\ sn\ cn\ dn)$ and $sn = sn(u,m)$, $cn = cn(u,m)$ and $dn = dn(u,m)$.

If $g(x,\dot{x})$ is a polynomial in x and $\dot{x}$, then the evaluation of the integral in eq.(7.25) may be readily accomplished. Terms in g of the form $x^N\ \dot{x}^M$ for M even lead to integrals of the form

(7.26) $$\int_0^{4K} \mathrm{cn}^P \,\mathrm{sn}\,\mathrm{dn}\,du$$

where P is an integer, which vanish due to the oddness of the integrand. On the other hand, terms in g of the form $x^N \dot{x}^M$ for M odd lead to integrals of the form

(7.27) $$\int_0^{4K} \mathrm{cn}^{2P}\,du$$

which may be evaluated by using the following results from [2], pp.192–3, formulas 312: Define

(7.28) $$C_{2P} = \int_0^{4K} \mathrm{cn}^{2P}\,du$$

Then

(7.29) $$C_0 = 4K, \quad C_2 = \frac{4}{m}\,[E-(1-m)K]$$

and

(7.30) $$C_{2P+2} = \frac{2P}{2P+1}\,\frac{2m-1}{m}\,C_{2P} + \frac{2P-1}{2P+1}\,\frac{1-m}{m}\,C_{2P-2}$$

where E = E(m) and K = K(m) are complete elliptic integrals.

An equilibrium point of the averaged $\dot{m}$ equation, eq.(7.25), corresponds to a limit cycle in the original equation (1). Thus if $m = m_0$ is a root of the equation

(7.31) $$\int_0^{4K} \mathrm{sn}\,\mathrm{cn}\,\mathrm{dn}\,g\,du = 0$$

then the averaged equation predicts that for small ϵ, a limit cycle coincides with the solution (7.8) of eq.(7.2) associated with a value of $m = m_0$.

For an arbitrary polynomial $g(x,\dot{x})$, the limit cycle integral condition, eq.(7.31), may be efficiently evaluated by using the following **MACSYMA** program:

```
g:read("Enter g(x,y), where y=dx/dt");
/* eliminate even powers of y */
g1:ev(g,y=-y);
g2:(g-g1)/2;
/* form integrand of limit cycle integral, eq.(7.31) */
integrand:g2*sn*cn*dn,x=a1+a2*sn^2,y=2*a2*w*sn*cn*dn$
/* use identities to express sn,dn in terms of cn */
idents:[sn=sqrt(1-cn^2), dn= sqrt(1-m+m*cn^2)];
integrand1:expand(ev(integrand,idents))$
/* set up rules for evaluating the integral of cn^r */
c[0]:4*k(m);
c[2]:4*(e(m)-(1-m)*k(m))/m;
c[r]:=(r-2)*(2*m-1)/((r-1)*m)*c[r-2]+(r-3)*(1-m)/((r-1)*m)*c[r-4];
/* perform the integration by replacing cn^r by c[r] */
max:hipow(integrand1,cn);
for i:0 thru max step 2 do b[i]:coeff(integrand1,cn,i);
result:sum(b[2*i]*c[2*i],i,0,max/2)$
/* substitute the parameters a1,a2,w in terms of m */
result1:result,
        a1 = -(sqrt(m^2-m+1)-m-1)/(2*sqrt(m^2-m+1)),
        a2 = -3*m/(2*sqrt(m^2-m+1)), w = 1/(2*(m^2-m+1)^(1/4))$
/* simplify the result */
result2:ratsimp(result1);
```

As an example let us take

(7.32) $$\ddot{x} + x + x^2 = \epsilon\ (k_{01}\ \dot{x} + k_{11}\ x\ \dot{x})$$

in which k_{01} and k_{11} are given parameters.

Here is a partial record of a sample run in which only the first and last lines of the program are displayed:

```
(C3) g:read("Enter g(x,y), where y=dx/dt");
Enter g(x,y), where y=dx/dt
K01*Y+K11*X*Y;
(D3)                    K11 X Y + K01 Y
                                      . . .
(C16) /* simplify the result */
result2:ratsimp(result1);
                 2                   2
(D16) - (SQRT(M   - M + 1) ((7 K11 M   - 21 K11 M + 14 K11) K(M)
                2
 + (- 14 K11 M   + 14 K11 M - 14 K11) E(M))
          2                     2
 + SQRT(M   - M + 1) ((- 14 K01 M   + 42 K01 M - 28 K01) K(M)
              2
 + (28 K01 M   - 28 K01 M + 28 K01) E(M))
             3            2
 + (5 K11 M   + 5 K11 M   - 20 K11 M + 10 K11) K(M)
                3             2
 + (- 10 K11 M   + 15 K11 M   + 15 K11 M - 10 K11) E(M))
    2          1/4      3       2
/((M   - M + 1)    (35 M   - 35 M   + 35 M))
```

This condition may be expressed in a convenient form by solving for k_{01}:

```
(C18) SOLVE(RESULT2,K01);
```

$$\text{(D18)}\quad [\mathrm{K01} = ((5\ \mathrm{K11}\ \mathrm{M}^3 + \mathrm{SQRT}(\mathrm{M}^2 - \mathrm{M} + 1)\ (7\ \mathrm{K11}\ \mathrm{M}^2 - 21\ \mathrm{K11}\ \mathrm{M} + 14\ \mathrm{K11}) + 5\ \mathrm{K11}\ \mathrm{M}^2 - 20\ \mathrm{K11}\ \mathrm{M} + 10\ \mathrm{K11})\ \mathrm{K}(\mathrm{M}) + (-\ 10\ \mathrm{K11}\ \mathrm{M}^3 + \mathrm{SQRT}(\mathrm{M}^2 - \mathrm{M} + 1)\ (-\ 14\ \mathrm{K11}\ \mathrm{M}^2 + 14\ \mathrm{K11}\ \mathrm{M} - 14\ \mathrm{K11}) + 15\ \mathrm{K11}\ \mathrm{M}^2 + 15\ \mathrm{K11}\ \mathrm{M} - 10\ \mathrm{K11})\ \mathrm{E}(\mathrm{M}))/(\mathrm{SQRT}(\mathrm{M}^2 - \mathrm{M} + 1)\ (14\ \mathrm{M}^2 - 42\ \mathrm{M} + 28)\ \mathrm{K}(\mathrm{M}) + (-\ 28\ \mathrm{M}^2 + 28\ \mathrm{M} - 28)\ \mathrm{SQRT}(\mathrm{M}^2 - \mathrm{M} + 1)\ \mathrm{E}(\mathrm{M}))]$$

This result can be rewritten in the form:

$$\text{(7.33)}\qquad \frac{k_{01}}{k_{11}} = \frac{1}{2} + \frac{5}{14}\,\frac{(K-2E)m^3+(K+3E)m^2+(-4K+3E)m+2(K-E)}{\sqrt{\lambda}\ \ [(K-2E)m^2+(-3K+2E)m+2(K-E)]}$$

where λ is given by eq.(7.14) and where K and E are complete elliptic integrals.

Numerical evaluation of eq.(7.33) shows [10] that averaging predicts no limit cycle if $k_{01}/k_{11}>1/7$ or if $k_{01}/k_{11}<0$, and a single limit cycle for $0<k_{01}/k_{11}<1/7$. In order to check this result, we numerically integrated eq.(7.32) using a Runge-Kutta scheme and found excellent agreement with the averaging predictions.

8. TWO VARIABLE EXPANSION METHOD

The two variable expansion method [7,11] is a popular singular perturbation method which offers a convenient alternative to averaging. It is essentially equivalent to the method of multiple scales [7]. The method may be motivated by noting that the method of averaging produces approximate solutions of the form (cf.eqs.(5.2),(5.3)):

$$z = a \cos \varphi \tag{8.1}$$

where

$$\frac{da}{dt} = O(\epsilon) , \qquad \frac{d\varphi}{dt} = 1 + O(\epsilon) \tag{8.2}$$

That is, $a = a(\epsilon t)$ and $\varphi = t + \psi(\epsilon t) = \varphi(t,\epsilon t)$. Thus we are led to the key ansatz that $z = z(t,\epsilon t)$. In order to simplify notation, we will use ξ and η to represent ordinary time and slow time, respectively:

$$z = z(\xi,\eta), \qquad \xi = t, \quad \eta = \epsilon t \tag{8.3}$$

In order to offer a comparison with the method of averaging presented in section 5, we will use the two variable expansion method on the same class of problems, cf.eq.(5.1):

$$\frac{d^2z}{dt^2} + z = \epsilon\, F(z,\frac{dz}{dt}) \tag{8.4}$$

Substitution of eq.(8.3) into (8.4) turns the o.d.e.(8.4) into a p.d.e., since now both and ξ and η are treated as independent variables:

$$\frac{dz}{dt} = \frac{\partial z}{\partial \xi}\frac{d\xi}{dt} + \frac{\partial z}{\partial \eta}\frac{d\eta}{dt} = \frac{\partial z}{\partial \xi} + \epsilon\, \frac{\partial z}{\partial \eta} \tag{8.5}$$

(8.6) $$\frac{d^2z}{dt^2} = \frac{\partial^2 z}{\partial \xi^2} + 2\epsilon \frac{\partial^2 z}{\partial \xi \partial \eta} + \epsilon^2 \frac{\partial^2 z}{\partial \eta^2}$$

which gives

(8.7) $$\frac{\partial^2 z}{\partial \xi^2} + 2\epsilon \frac{\partial^2 z}{\partial \xi \partial \eta} + \epsilon^2 \frac{\partial^2 z}{\partial \eta^2} + z = \epsilon\ F(z, \frac{\partial z}{\partial \xi} + \epsilon \frac{\partial z}{\partial \eta})$$

Now we expand z in a power series in ϵ:

(8.8) $$z(\xi,\eta) = z_0(\xi,\eta) + \epsilon\ z_1(\xi,\eta) + O(\epsilon^2)$$

Substituting eq.(8.8) into (8.7) and collecting terms gives:

(8.9) $$\frac{\partial^2 z_0}{\partial \xi^2} + z_0 = 0$$

(8.10) $$\frac{\partial^2 z_1}{\partial \xi^2} + z_1 = -2 \frac{\partial^2 z_0}{\partial \xi \partial \eta} + F(z_0, \frac{\partial z_0}{\partial \xi})$$

The general solution to the p.d.e.(8.9) is:

(8.11) $$z_0 = a(\eta)\ \cos(\xi + \psi(\eta))$$

Note that since (8.9) is a p.d.e., the "constants of integration" a and ψ become arbitrary functions of the variable η.

Substitution of eq.(8.11) into (8.10) gives a p.d.e. on $z_1(\xi,\eta)$:

(8.12) $$\frac{\partial^2 z_1}{\partial \xi^2} + z_1 = 2 \frac{\partial a}{\partial \eta} \sin(\xi+\psi) - 2a \frac{\partial \psi}{\partial \eta} \cos(\xi+\psi)$$

$$+ F(a \cos(\xi+\psi), - a \sin(\xi+\psi))$$

For a particular choice of the function $F(z,\dot{z})$, the right-hand side of eq.(8.12) may be trigonometrically reduced, and terms may be collected as follows:

$$\frac{\partial^2 z_1}{\partial \xi^2} + z_1 = (\ldots)\ \sin(\xi+\psi) + (\ldots)\ \cos(\xi+\psi) + \text{higher harmonics} \tag{8.13}$$

In order to avoid "secular" terms, i.e., terms which grow linearly in ξ and which prevent the resulting approximate solution from being uniformly valid in the limit as t approaches infinity, we require the coefficients $(\ldots)$ of $\sin(\xi+\psi)$ and $\cos(\xi+\psi)$ to vanish. This gives two o.d.e.'s on $a(\eta)$ and $\psi(\eta)$, which, when solved and substituted into eq.(8.11), provide the general solution to the original problem (8.4), accurate to $O(\epsilon)$.

As an example, we again take van der Pol's eq.(5.6):

$$\frac{d^2 z}{dt^2} + z = \epsilon\ \frac{dz}{dt}\ (1 - z^2) \tag{8.14}$$

Here the function $F(z,\dot{z})$ of (8.4) becomes

$$F(z,\dot{z}) = \frac{dz}{dt}\ (1 - z^2) \tag{8.15}$$

F appears on the right-hand side of eq.(8.12) in the form:

$$F(a\ \cos(\xi+\psi), -\ a\ \sin(\xi+\psi)) = -\ a\ \sin(\xi+\psi)\ (1 - a^2\cos^2(\xi+\psi)) \tag{8.16}$$

Trigonometric reduction of (8.16) gives:

$$F(a\ \cos(\xi+\psi), -\ a\ \sin(\xi+\psi)) = (-\ a + \frac{a^3}{4})\ \sin(\xi+\psi) + \frac{a^3}{4}\ \sin 3(\xi+\psi) \tag{8.17}$$

Elimination of secular terms in eq.(8.12) gives the two o.d.e.'s:

$$\text{(8.18)} \quad \sin(\xi+\psi):\ 2\,\frac{\partial a}{\partial \eta} = a - \frac{a^3}{4}\,, \qquad \cos(\xi+\psi):\ 2a\,\frac{\partial \psi}{\partial \eta} = 0$$

Note that these eqs. are equivalent to eqs.(5.15) obtained by the method of averaging. Since η represents slow time, eqs.(8.18) are called a slow-flow. Computer algebra may be used to solve the slow-flow on $a(\eta)$:

```
(C2) depends(a,eta);
(D2)                      [A(ETA)]
(C3) 2*diff(a,eta) = a-a^3/4;
                                3
                      dA       A
(D3)                2 ──── = A - ──
                     dETA       4
(C4) ode2(%,a,eta);
(D4)     - LOG(A + 2) + 2 LOG(A) - LOG(A - 2) = ETA + %C
(C5) soln1:logcontract(%);
                          2
                         A
(D5)               LOG(──────) = ETA + %C
                        2
                       A  - 4

(C6) %e^lhs(%) = %e^rhs(%);
                        2
                       A         ETA + %C
(D6)                 ────── = %E
                      2
                     A  - 4

(C7) soln2:solve(%,a^2);
                                ETA + %C
                      2     4 %E
(D7)                [A  = ─────────────────]
                               ETA + %C
                            %E          - 1
```

```
     /* set initial condition */
(C8) soln1,a = a0,eta = 0;
                                    2
                                  A0
(D8)                        LOG(-------) = %C
                                  2
                                A0  - 4
(C9) solve(%,%C);
                                         2
                                       A0
(D9)                        [%C = LOG(-------)]
                                       2
                                     A0  - 4
(C10) ratsimp(ev(soln2,%));

                                   2  ETA
                         2     4 A0  %E
(D10)                  [A  = - ------------------]
                              2  ETA      2
                            A0  %E    - A0  + 4
```

That is,

$$a^2(\eta) = \frac{4\ a^2(0)\ e^{\eta}}{a^2(0)\ e^{\eta} - a^2(0) + 4} \tag{8.19}$$

This result, when substituted into eq.(8.11), gives the approximate solution:

$$z(t) = \frac{2}{\sqrt{1 + \left[\frac{4}{a^2(0)} - 1\right] e^{-\epsilon t}}} \cos(t + \psi(0)) + O(\epsilon) \tag{8.20}$$

Note that this solution contains two arbitrary constants $a(0)$, $\psi(0)$, and thus represents an approximate general solution to van der Pol's eq.(8.14). Note also that as $t \to +\infty$, $z \to 2\cos(t+\psi(0))$. For large t, binomial expansion of eq.(8.20) shows that the approach to the limit cycle goes like $e^{-\epsilon t}$. In the limit as $t \to -\infty$, there are two possibilities. If $a(0) < 2$, then $z \to 0$ as $t \to -\infty$. However, if $a(0) > 2$, then z blows up in finite time! This behavior should not surprise us, and is due to the polynomial nature of the nonlinearity in eq.(8.14). For example, the simple model equation

$$\frac{dz}{dt} = z^2 \tag{8.21}$$

has the general solution

$$z(t) = \frac{1}{\frac{1}{z(0)} - t} \tag{8.22}$$

which blows up in finite time t = 1/z(0). Thus the approximate solution to van der Pol's eq., as generated by the two variable expansion method, captures numerous features of the behavior of the real solution. Numerical integration of eq.(8.14) shows good agreement with (8.20) for small ϵ.

The foregoing procedure may be formalized into a computer algebra program:

```
/* two variable expansion on z''+z=e f(z,z') */
depends([a,psi],eta);
z0:a*cos(xi+psi);
/* define f=f(z,zdot) here: */
f(z,zdot):=zdot*(1-z^2);
/* rhs= right hand side of z1 eq. */
rhs:-2*diff(z0,xi,1,eta,1)+f(z0,diff(z0,xi));
rhs2:expand(trigreduce(expand(%)));
eq1:coeff(rhs2,sin(xi+psi));
eq2:coeff(rhs2,cos(xi+psi));
solve([eq1,eq2],[diff(a,eta),diff(psi,eta)]);
```

Here is a sample run:

```
(C3) /* two variable expansion on z''+z=e f(z,z') */

depends([a,psi],eta);
(D3)                      [A(ETA), PSI(ETA)]
```

```
(C4) z0:a*cos(xi+psi);
(D4)                      A COS(XI + PSI)

(C5) /* define f=f(z,zdot) here: */

f(z,zdot):=zdot*(1-z^2);
                                                2
(D5)                 F(Z, ZDOT) := ZDOT (1 - Z )

(C6) /* rhs= right hand side of z1 eq. */

rhs:-2*diff(z0,xi,1,eta,1)+f(z0,diff(z0,xi));
              dA                         dPSI
(D6) - 2 (- ———— SIN(XI + PSI) - A ———— COS(XI + PSI))
             dETA                        dETA

                                  2     2
                      - A (1 - A  COS  (XI + PSI)) SIN(XI + PSI)

(C7) rhs2:expand(trigreduce(expand(%)));
       3
      A  SIN(3 XI + 3 PSI)        dA
(D7) ——————————————————————— + 2 ———— SIN(XI + PSI)
               4                  dETA

       3
      A  SIN(XI + PSI)                                    dPSI
   + ———————————————— - A SIN(XI + PSI) + 2 A ———— COS(XI + PSI)
              4                                          dETA

(C8) eq1:coeff(rhs2,sin(xi+psi));
                                      3
                             dA      A
(D8)                       2 ———— + ——— - A
                            dETA     4

(C9) eq2:coeff(rhs2,cos(xi+psi));
                                  dPSI
(D9)                          2 A ————
                                  dETA

(C10) solve([eq1,eq2],[diff(a,eta),diff(psi,eta)]);

                                3
                    dA         A   - 4 A   dPSI
(D10)             [[———— = - ——————————, ———— = 0]]
                   dETA            8       dETA
```

These results agree with eqs.(8.18).

Next we apply the foregoing program to the following example:

$$\frac{d^2z}{dt^2} + z = \epsilon \left[b_1 \frac{dz}{dt} + b_2 \left[\frac{dz}{dt}\right]^3 + b_3 \left[\frac{dz}{dt}\right]^5\right] \tag{8.23}$$

where the b_i are given parameters. This entails replacing the line in the program which defines f, with the line:

(8.24)
```
f(z,zdot):=b1*zdot+b2*zdot^3+b3*zdot^5;
```

The resulting program gives the following slow-flow:

$$\text{(D10)} \quad [[\frac{dA}{dETA} = \frac{5\ A^5\ B3 + 6\ A^3\ B2 + 8\ A\ B1}{16}, \frac{dPSI}{dETA} = 0]]$$

Limit cycles correspond to (non-zero) equilibria in the $dA/d\eta$ equation:

$$8\ b_1 + 6\ b_2\ \rho + 5\ b_3\ \rho^2 = 0, \qquad \rho = a^2 \tag{8.25}$$

A positive root ρ of (8.25) corresponds to a limit cycle. There may be 0, 1 or 2 limit cycles. For example, for $b_1 = 1$, $b_2 = -3$ and $b_3 = 1$, eq.(8.25) predicts the existence of two limit cycles with amplitudes a = 0.72 and 1.75, in agreement with numerical integration of eq.(8.23) for small ϵ.

9. APPLICATION TO COUPLED OSCILLATORS

In this section we apply the two variable expansion method to a system of n coupled oscillators of the form:

$$\frac{d^2z_i}{dt^2} + z_i = \epsilon\ F_i(z_j,\frac{dz_j}{dt})\ , \quad i,j=1,2,\ldots,n \tag{9.1}$$

Note that the n oscillators are identical for $\epsilon = 0$. As in section 8, we begin by asserting that

$$z_i = z_i(\xi,\eta), \qquad \xi = t, \quad \eta = \epsilon t \tag{9.2}$$

Then eq.(9.1) becomes, neglecting terms of $O(\epsilon^2)$:

$$\frac{\partial^2 z_i}{\partial\xi^2} + 2\epsilon\ \frac{\partial^2 z_i}{\partial\xi\partial\eta} + z_i = \epsilon\ F_i(z_j,\frac{\partial z_j}{\partial\xi}) \tag{9.3}$$

We expand z_i in a power series in ϵ:

$$z_i(\xi,\eta) = z_{i0}(\xi,\eta) + \epsilon\ z_{i1}(\xi,\eta) + O(\epsilon^2) \tag{9.4}$$

which gives

$$\frac{\partial^2 z_{i0}}{\partial\xi^2} + z_{i0} = 0 \tag{9.5}$$

$$\frac{\partial^2 z_{i1}}{\partial\xi^2} + z_{i1} = -\ 2\ \frac{\partial^2 z_{i0}}{\partial\xi\partial\eta} + F_i(z_{j0},\frac{\partial z_{j0}}{\partial\xi}) \tag{9.6}$$

The general solution of eq.(9.5) is

$$z_{i0} = a_i(\eta) \cos(\xi + \psi_i(\eta)) \tag{9.7}$$

Substitution of (9.7) into (9.6) for a particular choice of the F_i permits terms to be arranged as follows:

$$\frac{\partial^2 z_{i1}}{\partial \xi^2} + z_{i1} = (\ldots) \sin \xi + (\ldots) \cos \xi + \text{higher harmonics} \tag{9.8}$$

Elimination of the secular terms (...) in each of the n eqs.(9.8) results in 2n first order o.d.e.'s governing the 2n unknowns

$$a_1, \psi_1, a_2, \psi_2, \ldots, a_n, \psi_n.$$

As an example, we consider a <u>ring</u> of n van der Pol oscillators with "diffusion" coupling:

$$\frac{d^2 z_i}{dt^2} + z_i = \epsilon \frac{dz_i}{dt} (1 - z_i^2) + \epsilon\, \alpha\, (z_{i+1} - 2\, z_i + z_{i-1}) \tag{9.9}$$

where i=1,...,n, and where we use the convention $z_0 \equiv z_n$ and $z_{n+1} \equiv z_1$, because the oscillators are arranged in a ring.

The following **MACSYMA** program sets up the right-hand side of eq.(9.6), substitutes eq.(9.7) and collects secular terms as in eq.(9.8):

```
/* two variable expansion on a system of coupled oscillators */
/* n=number of oscillators */
n:read("enter n");
depends([a,psi],eta);
for i:1 thru n do
  z0[i]:a[i]*cos(xi+psi[i]);
/* define f[i] here: */
f[1]:diff(z0[1],xi)*(1-z0[1]^2)+alpha*(z0[2]-2*z0[1]+z0[n]);
for i:2 thru n-1 do
  f[i]:diff(z0[i],xi)*(1-z0[i]^2)
       +alpha*(z0[i+1]-2*z0[i]+z0[i-1]);
f[n]:diff(z0[n],xi)*(1-z0[n]^2)+alpha*(z0[1]-2*z0[n]+z0[n-1]);
/* rhs[i]= right hand side of z1[i] eq. */
for i:1 thru n do(
  rhs[i]:-2*diff(z0[i],xi,1,eta,1)+f[i],
  rhs2:expand(trigexpand(rhs[i])),
  rhs3:expand(trigreduce(rhs2,xi)),
  eq1:coeff(rhs3,sin(xi)),
  eq2:coeff(rhs3,cos(xi)),
  eqs[i]:solve([eq1,eq2],[diff(a[i],eta),diff(psi[i],eta)]));
slowflow:makelist(eqs[i],i,1,n)$
trigident:makelist(sin(psi[i])^2=1-cos(psi[i])^2,i,1,n)$
slowflow2:slowflow,trigident$
slowflow3:ratsimp(slowflow2)$
slowflow4:expand(trigreduce(expand(slowflow3)));
```

Although the computation can only be accomplished for a specific value of n, we can, by comparing the resulting slow-flows for n=3,4,5, obtain the following slow-flow which is valid for a general value of n:

$$\frac{da_i}{d\eta} = \frac{a_i}{2} - \frac{a_i^3}{8} + \frac{\alpha}{2}\left[a_{i+1}\sin(\psi_{i+1}-\psi_i) + a_{i-1}\sin(\psi_{i-1}-\psi_i)\right] \tag{9.10}$$

$$\frac{d\psi_i}{d\eta} = \alpha - \frac{\alpha}{2a_i}\left[a_{i+1}\cos(\psi_{i+1}-\psi_i) + a_{i-1}\cos(\psi_{i-1}-\psi_i)\right] \tag{9.11}$$

Eqs.(9.10) and (9.11) are valid for i=1,2,...,n with the convention that $a_0 \equiv a_n$, $\psi_0 \equiv \psi_n$, $a_{n+1} \equiv a_1$, $\psi_{n+1} \equiv \psi_1$. The complete analysis of these 2n slow-flow equations has not, to my knowledge, been accomplished. Here we discuss two simple solutions corresponding to equilibria of the slow-flow, i.e., to periodic motions of the original system (9.9).

In the case that $\alpha = 0$, i.e., no coupling, each oscillator exhibits, to $O(\epsilon)$, an amplitude of 2. We seek solutions such that when $\alpha \neq 0$ this behavior persists:

$$a_1 = a_2 = \cdots = a_n = 2 \tag{9.12}$$

Substitution of (9.12) into the slow-flow (9.10),(9.11) gives:

$$0 = \alpha \left[\sin(\psi_{i+1}-\psi_i) + \sin(\psi_{i-1}-\psi_i)\right] \tag{9.13}$$

$$\frac{d\psi_i}{d\eta} = \alpha - \frac{\alpha}{2} \left[\cos(\psi_{i+1}-\psi_i) + \cos(\psi_{i-1}-\psi_i)\right] \tag{9.14}$$

One solution to eqs.(9.13),(9.14) is

$$\psi_1 = \psi_2 = \cdots = \psi_n \tag{9.15}$$

whereupon

$$\frac{d\psi_i}{d\eta} = 0 \tag{9.16}$$

This motion corresponds to each of the oscillators vibrating in phase with the others. Their common frequency is, to $O(\epsilon)$, unity, cf.(9.7).

Another solution to eqs.(9.13),(9.14) is

$$\psi_{i+1} = \psi_i + \frac{2\pi}{n}, \qquad i = 1,2,\ldots,n-1 \tag{9.17}$$

or, equivalently,

$$\psi_i = \psi_1 + \frac{2\pi}{n}(i-1), \qquad i = 2,3,\ldots,n \tag{9.18}$$

Note that the slow-flow (9.10),(9.11) is 2π-periodic in ψ_i, so that all angles ψ_i are understood mod 2π. Thus there is no jump in (9.18) for i=n, i.e., $\psi_n = \psi_1 + \frac{2\pi}{n}(n-1)$ means $\psi_1 = \psi_n + \frac{2\pi}{n} \pmod{2\pi}$.

Substituting eq.(9.17) into (9.14) gives

(9.19) $$\frac{d\psi_i}{d\eta} = \alpha \left[1 - \cos \frac{2\pi}{n}\right]$$

This motion corresponds to a wave moving around the ring. The phase of each oscillator differs from that of its neighbors by a constant amount, and each has a common frequency equal to

(9.20) $$1 + \epsilon\ \alpha \left[1 - \cos \frac{2\pi}{n}\right]$$

For large n, this frequency may be approximated by $1 + \frac{2\pi^2 \epsilon \alpha}{n^2}$.

The two solutions (9.15) and (9.17) may be viewed as special cases of a more general class of solutions to (9.13),(9.14). Consider the solution:

(9.21) $$\psi_{i+1} = \psi_i + \frac{2m\pi}{n}, \qquad i = 1,2,\ldots,n-1$$

where $m < n$ is an integer. This solution may also be viewed as a wave moving around the ring, but now with m complete wavelengths exhibited in one ring circumference. From this point of view, the uniform solution (9.15) corresponds to m=0 and the solution (9.17) corresponds to m=1.

A related question concerns the stability and bifurcation of all of these motions. Although we will not consider this here, we note that this computation is more conveniently accomplished via the slow-flow (9.13),(9.14), where the motion in question is an equilibrium point, than via the original eqs.(9.9), where the motion is a function of time.

REFERENCES

1. C.M.Bender and S.A.Orszag, "Advanced Mathematical Methods for Scientists and Engineers", McGraw-Hill (1978)

2. P.Byrd and M.Friedman, "Handbook of Elliptic Integrals for Engineers and Physicists", Springer (1954)

3. V.T.Coppola and R.H.Rand,"Averaging Using Elliptic Functions: Approximation of Limit Cycles", Acta Mechanica, 81:125-142 (1990)

4 V.T.Coppola and R.H.Rand, "MACSYMA Program to Implement Averaging Using Elliptic Functions", in Computer Aided Proofs in Analysis, eds. K.R.Meyer and D.S.Schmidt, pp.71–89, Springer (1991)

5. J.Guckenheimer and P.Holmes, "Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields", Springer (1983)

6. P.B.Kahn and Y.Zarmi, "Minimal Normal Forms in Harmonic Oscillations with Small Nonlinear Perturbations", Physica D, 54:65-74 (1991)

7. A.Nayfeh, "Perturbation Methods", Wiley (1973)

8. R.H.Rand, "Computer Algebra in Applied Mathematics: An Introduction to MACSYMA", Pitman (1984)

9. R.H.Rand, "The Use of Symbolic Computation in Perturbation Analysis", in Symbolic Computation in Fluid Mechanics and Heat Transfer, ed. H.H.Bau et al., ASME publication HTD-105/AMD-97, pp.41-45 (1988)

10. R.H.Rand, "Using Computer Algebra to Handle Elliptic Functions in the Method of Averaging", in Symbolic Computations and Their Impact on Mechanics, eds. A.K.Noor,I.Elishakoff, G.Hulbert, pp.311–326, Amer.Soc.Mech.Eng., PVP-Vol.205, (1990)

11. R.H.Rand and D.Armbruster, "Perturbation Methods, Bifurcation Theory and Computer Algebra", Springer (1987)

12. J.A.Sanders and F.Verhulst, "Averaging Methods in Nonlinear Dynamical Systems", Springer (1985)

13. J.J.Stoker, "Nonlinear Vibrations", Interscience (1950)

14. M.Van Dyke, "Analysis and Improvement of Perturbation Series", Q.J.Mech.Appl.Math., 27:423–450 (1974)