December 2007

Normal Modes Using Maple (Water)

Carl W. David
University of Connecticut, Carl.David@uconn.edu

Normal Modes Using Maple (Water)

C. W. David
Department of Chemistry
University of Connecticut
Storrs, Connecticut 06269-3060
(Dated: December 17, 2007)

I. SYNOPSIS

The brute force Maple code for obtaining normal coördinates is presented here [**?**]

II. NORMAL MODES

```
> #molecular dynamics and normal modes
> restart;
> with(LinearAlgebra):
> with(plottools);
```

We start with the primitive definition of the distances which will be employed in the potential energy function:

```
r1:=sqrt((x2-x1)^2+(y2-y1)^2+(z2-z1)^2);
r2:=sqrt((x3-x2)^2+(y3-y2)^2+(z3-z2)^2);
r1dotr2:=(x2-x1)*(x3-x2)+(y2-y1)*(y3-y2)+
   (z2-z1)*(z3-z2);
theta:=Pi-arccos(r1dotr2/r1/r2);
V:=f1*(r1-r1e)^2+f2*(r2-r2e)^2+f3*(theta-thetae)^2;
```

There are three force constants here. Next, we obtain the matrix of force elements:

$$\frac{\partial^2 V}{\partial x_i \partial x_j}$$

where x_i runs through x, y and z for each atom.

```
f11:=diff(diff(V,x1),x1)/sqrt(m1*m1):
f12:=diff(diff(V,x1),x2)/sqrt(m1*m2):
f13:=diff(diff(V,x1),x3)/sqrt(m1*m3):
f14:=diff(diff(V,x1),y1)/sqrt(m1*m1):
f15:=diff(diff(V,x1),y2)/sqrt(m1*m2):
f16:=diff(diff(V,x1),y3)/sqrt(m1*m3):
f17:=diff(diff(V,x1),z1)/sqrt(m1*m1):
f18:=diff(diff(V,x1),z2)/sqrt(m1*m2):
f19:=diff(diff(V,x1),z3)/sqrt(m1*m3):
f22:=diff(diff(V,x2),x2)/sqrt(m2*m2):
f23:=diff(diff(V,x2),x3)/sqrt(m2*m3):
f24:=diff(diff(V,x2),y1)/sqrt(m2*m1):
f25:=diff(diff(V,x2),y2)/sqrt(m2*m2):
f26:=diff(diff(V,x2),y3)/sqrt(m2*m3):
f27:=diff(diff(V,x2),z1)/sqrt(m2*m1):
f28:=diff(diff(V,x2),z2)/sqrt(m2*m2):
f29:=diff(diff(V,x2),z3)/sqrt(m2*m3):
f33:=diff(diff(V,x3),x3)/sqrt(m3*m3):
f34:=diff(diff(V,x3),y1)/sqrt(m3*m1):
f35:=diff(diff(V,x3),y2)/sqrt(m3*m2):
f36:=diff(diff(V,x3),y3)/sqrt(m3*m3):
f37:=diff(diff(V,x3),z1)/sqrt(m3*m1):
f38:=diff(diff(V,x3),z2)/sqrt(m3*m2):
f39:=diff(diff(V,x3),z3)/sqrt(m3*m3):
f44:=diff(diff(V,y1),y1)/sqrt(m1*m1):
f45:=diff(diff(V,y1),y2)/sqrt(m1*m2):
f46:=diff(diff(V,y1),y3)/sqrt(m1*m3):
f47:=diff(diff(V,y1),z1)/sqrt(m1*m1):
f48:=diff(diff(V,y1),z2)/sqrt(m1*m2):
f49:=diff(diff(V,y1),z3)/sqrt(m1*m3):
f55:=diff(diff(V,y2),y2)/sqrt(m2*m2):
f56:=diff(diff(V,y2),y3)/sqrt(m2*m3):
f57:=diff(diff(V,y2),z1)/sqrt(m2*m1):
f58:=diff(diff(V,y2),z2)/sqrt(m2*m2):
f59:=diff(diff(V,y2),z3)/sqrt(m2*m3):
f66:=diff(diff(V,y3),y3)/sqrt(m3*m3):
f67:=diff(diff(V,y3),z1)/sqrt(m3*m1):
f68:=diff(diff(V,y3),z2)/sqrt(m3*m2):
f69:=diff(diff(V,y3),z3)/sqrt(m3*m3):
f77:=diff(diff(V,z1),z1)/sqrt(m1*m1):
f78:=diff(diff(V,z1),z2)/sqrt(m1*m2):
f79:=diff(diff(V,z1),z3)/sqrt(m1*m3):
f88:=diff(diff(V,z2),z2)/sqrt(m2*m2):
f89:=diff(diff(V,z2),z3)/sqrt(m2*m3):
f99:=diff(diff(V,z3),z3)/sqrt(m3*m3):
```

Now, we specialize to the geometric situation we want for this molecule (H_2O). We adjust the coördinates slightly, to place the origin at the center of mass, which helps to identify the non-translational and non-rotational modes (which are the vibrational normal modes).

```
x1:=evalf(0.9584*cos((104.5/2)*Pi/180))-
   0.0442:#center of mass adjust
y1:=-evalf(0.9584*sin((104.5/2)*Pi/180)):
z1:=0:# H1 atom
x2:=0.0:# in angstrom
y2:=0:
z2:=0:#location of O atom
x3:=x1:
y3:=-y1:
z3:=0:
r1e:=0.9584:
r2e:=0.9584:
thetae:=evalf(104.5*Pi/180):
m1:=1:
m2:=16:
m3:=1:
f1:=3.76:
```

Typeset by REVTEX

```
f2:=f1:
f3:=0.4*(r1e^2):
print (‘atom1(x,y,z)‘,x1,y1,z1);
print (‘atom2(x,y,z)‘,x2,y2,z2);
print (‘atom3(x,y,z)‘,x3,y3,z3);
```

The values of $f1$ and $f3$ are adjusted until the observed frequencies are consonant with experimental values.

```
F:=Matrix(evalf(evalm(matrix(9,9,[f11, f12, f13, f14, f15, f16, f17, f18, f19,
                            f12, f22, f23, f24, f25, f26, f27, f28, f29,
                            f13, f23, f33, f34, f35, f36, f37, f38, f39,
                            f14, f24, f34, f44, f45, f46, f47, f48, f49,
                            f15, f25, f35, f45, f55, f56, f57, f58, f59,
                            f16, f26, f36, f46, f56, f66, f67, f68, f69,
                            f17, f27, f37, f47, f57, f67, f77, f78, f79,
                            f18, f28, f38, f48, f58, f68, f78, f88, f89,
                            f19, f29, f39, f49, f59, f69, f79, f89, f99])))):
```

This last line of code is print-suppressed on purpose.

Now begins the analysis. We obtain the eigenvectors of this matrix(F) without printing them, and then obtain the eigenvalues separately (they are contained in the evecs output, but the form is so complicated that it appears better to do it the way indicated).

```
evecs:=Eigenvectors(F):
print(‘Eigenvalues:‘);
e := Eigenvalues(F);
lambda1:=Re(e[2]);
lambda2:=Re(e[3]);
lambda3:=Re(e[1]);
```

Using the LinerAlgebra package, the first three eigenvectors and associated eigenvalues are the ones we want, but they are not in the standard order reported in the literature, hence my “re-naming” them.

Next, we compute the frequencies associated with the eigenvalues:

```
#expect 3742, 1597 and 5309 cm^(-1)
nu1:=evalf(sqrt(lambda1*6.022e28)/2/Pi/3e10);#3657
nu2:=evalf(sqrt(lambda2*6.022e28)/2/Pi/3e10);#1595
nu3:=evalf(sqrt(lambda3*6.022e28)/2/Pi/3e10);#3755
```

Now, for each of the three eigenvalues (above) we obtain the eigenvectors, and “normalize” them using the first element of the vector.

```
e1 := [evecs]:
evec1 := Vector(
9,[Re(Column(evecs[2],1)[1]/sqrt(m1)),
Re(Column(evecs[2],1)[2]/sqrt(m2)),
Re(Column(evecs[2],1)[3]/sqrt(m3)),
Re(Column(evecs[2],1)[4]/sqrt(m1)),
Re(Column(evecs[2],1)[5]/sqrt(m2)),
Re(Column(evecs[2],1)[6]/sqrt(m3)),
0.,0.,0.]):
evec1 := evec1/Re(Column(evecs[2],1)[1]);# "normalize"
```

What is the meaning of diagonalizing this matrix? Clearly, instead of having terms of the form $x_1 y_3$ the diagonal terms will be associated with terms of the form x_1^2 or z_3^2 etc.. This means that the eigenvector corresponds to a form of motion in which the coördinates are being combined in quadratic form, which would mean that their motion would be governed by a harmonic oscillator type oscillation with all coördinates “breathing” together in a “normal mode”.

What follows is the output from the above code, followed by coding to allow visualization of the resultant three (important) normal modes.

[*arc*, *arrow*, *circle*, *cone*, *cuboid*, *curve*, *cutin*, *cutout*, *cylinder*, *disk*, *dodecahedron*, *ellipse*, *ellipticArc*, *hemisphere*, *hexahedron*, *homothety*, *hyperbola*, *icosahedron*, *line*, *octahedron*, *pieslice*, *point*, *polygon*, *project*, *rectangle*, *reflect*, *rotate*, *scale*, *semitorus*, *sphere*, *stellate*, *tetrahedron*, *torus*, *transform*, *translate*, *vrml*]

$$r1 := \sqrt{x2^2 - 2\,x2\,x1 + x1^2 + y2^2 - 2\,y2\,y1 + y1^2 + z2^2 - 2\,z2\,z1 + z1^2}$$

$$r2 := \sqrt{x3^2 - 2\,x3\,x2 + x2^2 + y3^2 - 2\,y3\,y2 + y2^2 + z3^2 - 2\,z3\,z2 + z2^2}$$

$$r1dotr2 := (x2 - x1)\,(x3 - x2) + (y2 - y1)\,(y3 - y2) + (z2 - z1)\,(z3 - z2)$$

$$\theta := \pi - \arccos(((x2 - x1)\,(x3 - x2) + (y2 - y1)\,(y3 - y2) + (z2 - z1)\,(z3 - z2))/(\sqrt{x2^2 - 2\,x2\,x1 + x1^2 + y2^2 - 2\,y2\,y1 + y1^2 + z2^2 - 2\,z2\,z1 + z1^2}\sqrt{x3^2 - 2\,x3\,x2 + x2^2 + y3^2 - 2\,y3\,y2 + y2^2 + z3^2 - 2\,z3\,z2 + z2^2}))$$

$$V := f1\,(\sqrt{x2^2 - 2\,x2\,x1 + x1^2 + y2^2 - 2\,y2\,y1 + y1^2 + z2^2 - 2\,z2\,z1 + z1^2} - r1e)^2 + f2\,(\sqrt{x3^2 - 2\,x3\,x2 + x2^2 + y3^2 - 2\,y3\,y2 + y2^2 + z3^2 - 2\,z3\,z2 + z2^2} - r2e)^2 + f3\,(\pi - \arccos(((x2 - x1)\,(x3 - x2) + (y2 - y1)\,(y3 - y2) + (z2 - z1)\,(z3 - z2))/(\sqrt{x2^2 - 2\,x2\,x1 + x1^2 + y2^2 - 2\,y2\,y1 + y1^2 + z2^2 - 2\,z2\,z1 + z1^2}\sqrt{x3^2 - 2\,x3\,x2 + x2^2 + y3^2 - 2\,y3\,y2 + y2^2 + z3^2 - 2\,z3\,z2 + z2^2})) - thetae)^2$$

$$atom1(x, y, z),\ 0.5425490411,\ -0.7577968876,\ 0$$

$$atom2(x, y, z),\ 0.,\ 0,\ 0$$

$$atom3(x, y, z),\ 0.5425490411,\ 0.7577968876,\ 0$$

$$Eigenvalues:$$

$$e := \begin{bmatrix} 8.13401069112464192 + 0.\,I \\ 7.85493216263778394 + 0.\,I \\ 1.59223647549554824 + 0.\,I \\ -0.222145814780197015 + 0.\,I \\ 0.468246115420671980\,10^{-9} + 0.\,I \\ 0.153976430546731994\,10^{-9} + 0.\,I \\ -0.339398490515555594 + 0.\,I \\ -0.167615550899999994 + 0.\,I \\ 0.175555546092411312\,10^{-9} + 0.\,I \end{bmatrix}$$

$$\lambda 1 := 7.85493216263778394$$
$$\lambda 2 := 1.59223647549554824$$
$$\lambda 3 := 8.13401069112464192$$
$$\nu 1 := 3648.719452$$
$$\nu 2 := 1642.755904$$
$$\nu 3 := 3712.971554$$

$$evec1 := \begin{bmatrix} 1.00000000012054268 \\ -0.117630578423092292\,10^{-11} \\ -1.00000000012054268 \\ -1.36841811400762214 \\ 0.171052264157478567 \\ -1.36841811400762214 \\ 0. \\ 0. \\ 0. \end{bmatrix}$$

```
l1:=
arrow([x1,y1,z1],[evec1[1]+x1,evec1[4]+y1,evec1[7]+z1],.04,.1,.1,color
=green):
l2:=
arrow([x3,y3,z3],[evec1[3]+x3,evec1[6]+y3,evec1[9]+z3],.04,.1,.1,color
=red);
c1 := sphere([.5425490411,-.7577968876,0], 0.1):
c2 := sphere([.5425490411, .7577968876,0], 0.1):
c3 := sphere([0,0,0], 0.3,color=red):
b1 := line([0,0,0], [.5425490411,-.7577968876,0], color=red,
linestyle=2):
b2 := line([0,0,0], [.5425490411,.7577968876,0], color=red,
linestyle=2):

l[3]:=
-arrow([evec1[7],evec1[8],evec1[9]],[0,0,0],.04,.06,.1,color=blue):
plots[display](\{l1,c1,l2,c2,c3,b1,b2
\},axes=boxed,scaling=CONSTRAINED);
```

l2 := POLYGONS([[0.5586968515, 0.7459965393, 0.],
[0.5264012307, 0.7695972359, 0.], [−0.3735987690, −0.4619790664, 0.],
[−0.3413031482, −0.4855797624, 0.]], [[−0.3978204845, −0.4442785434, 0.],
[−0.4574509588, −0.6106212264, 0.], [−0.3170814327, −0.5032802854, 0.]],
STYLE(*PATCHNOGRID*), COLOUR(*RGB*, 1.00000000, 0., 0.))

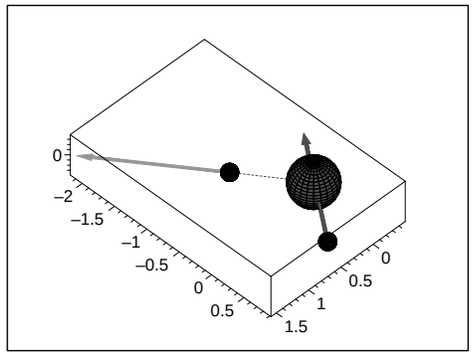

FIG. 1: The highest energy normal mode.

```
> evec2 := Vector(
> 9,[Re(Column(evecs[2],2)[1]/sqrt(m1)),Re(Column(evecs[2],2)[2]/sqrt(m2
> )),
> Re(Column(evecs[2],2)[3]/sqrt(m3)),Re(Column(evecs[2],2)[4]/sqrt(m1)),
> Re(Column(evecs[2],2)[5]/sqrt(m2)),Re(Column(evecs[2],2)[6]/sqrt(m3)),
> 0.,0.,0.]):
> evec2 := evec2/Re(Column(evecs[2],2)[1]);
> l1:=
> arrow([x1,y1,z1],[evec2[1]+x1,evec2[4]+y1,evec2[7]+z1],.04,.1,.1,color
> =green):
> l2:=
> arrow([x3,y3,z3],[evec2[3]+x3,evec2[6]+y3,evec2[9]+z3],.04,.1,.1,color
> =red);
> c1 := sphere([.5425490411,-.7577968876,0], 0.1):
> c2 := sphere([.5425490411, .7577968876,0], 0.1):
> c3 := sphere([0,0,0], 0.3,color=red):
> b1 := line([0,0,0], [.5425490411,-.7577968876,0], color=red,
> linestyle=2):
> b2 := line([0,0,0], [.5425490411,.7577968876,0], color=red,
> linestyle=2):
> l[3]:=
> -arrow([evec2[7],evec2[8],evec2[9]],[0,0,0],.04,.06,.1,color=blue):
> plots[display]({l1,c1,l2,c2,c3,b1,b2
> },axes=boxed,scaling=CONSTRAINED);
```

$$evec2 := \begin{bmatrix} 1.00000000013801804 \\ -0.124999999987226912 \\ 1.00000000013801804 \\ -1.32659964351433502 \\ -0.476805414484634736\,10^{-12} \\ 1.32659964351433502 \\ 0. \\ 0. \\ 0. \end{bmatrix}$$

$l2 :=$ POLYGONS([[0.5585198100, 0.7457580114, 0.], [0.5265782722, 0.7698357638, 0.], [1.426578272, 1.963775443, 0.], [1.458519810, 1.939697691, 0.]], [[1.402622119, 1.981833758, 0.], [1.542549041, 2.084396532, 0.], [1.482475963, 1.921639376, 0.]], STYLE($PATCHNOGRID$), COLOUR(RGB, 1.00000000, 0., 0.))

```
> evec3 := Vector(
> 9,[Re(Column(evecs[2],3)[1]/sqrt(m1)),Re(Column(evecs[2],3)[2]/sqrt(m2
> )),
> Re(Column(evecs[2],3)[3]/sqrt(m3)),Re(Column(evecs[2],3)[4]/sqrt(m1)),
> Re(Column(evecs[2],3)[5]/sqrt(m2)),Re(Column(evecs[2],3)[6]/sqrt(m3)),
> 0.,0.,0.]):
> evec3 := evec3/Re(Column(evecs[2],3)[1]);
> l1:=
> arrow([x1,y1,z1],[evec3[1]+x1,evec3[4]+y1,evec3[7]+z1],.04,.1,.1,color
> =green):
> l2:=
> arrow([x3,y3,z3],[evec3[3]+x3,evec3[6]+y3,evec3[9]+z3],.04,.1,.1,color
> =red);
> c1 := sphere([.5425490411,-.7577968876,0], 0.1):
> c2 := sphere([.5425490411, .7577968876,0], 0.1):
> c3 := sphere([0,0,0], 0.3,color=red):
> b1 := line([0,0,0], [.5425490411,-.7577968876,0], color=red,
> linestyle=2):
> b2 := line([0,0,0], [.5425490411,.7577968876,0], color=red,
> linestyle=2):
> l[3]:=
> -arrow([evec3[7],evec3[8],evec3[9]],[0,0,0],.04,.06,.1,color=blue):
> plots[display]({l1,c1,l2,c2,c3,b1,b2
> },axes=boxed,scaling=CONSTRAINED);
```

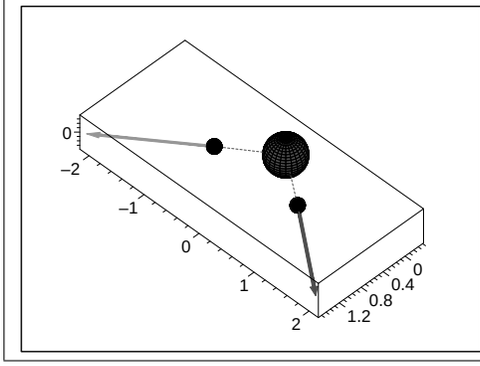

FIG. 2: The second highest energy normal mode.

$$evec3 := \begin{bmatrix} 0.99999999978243114 \\ -0.124999999804760176 \\ 0.99999999978243114 \\ 0.848032792191763974 \\ 0.177020195925372626\,10^{-11} \\ -0.848032792191763974 \\ 0. \\ 0. \\ 0. \end{bmatrix}$$

l2 := POLYGONS([[0.5296135018, 0.7425433037, 0.],
[0.5554845804, 0.7730504715, 0.], [1.455484580, 0.0098209584, 0.],
[1.429613502, −0.0206862094, 0.]], [[1.474887890, 0.0327013343, 0.],
[1.542549041, −0.0902359046, 0.], [1.410210193, −0.0435665853, 0.]],
STYLE(*PATCHNOGRID*), COLOUR(*RGB*, 1.00000000, 0., 0.))

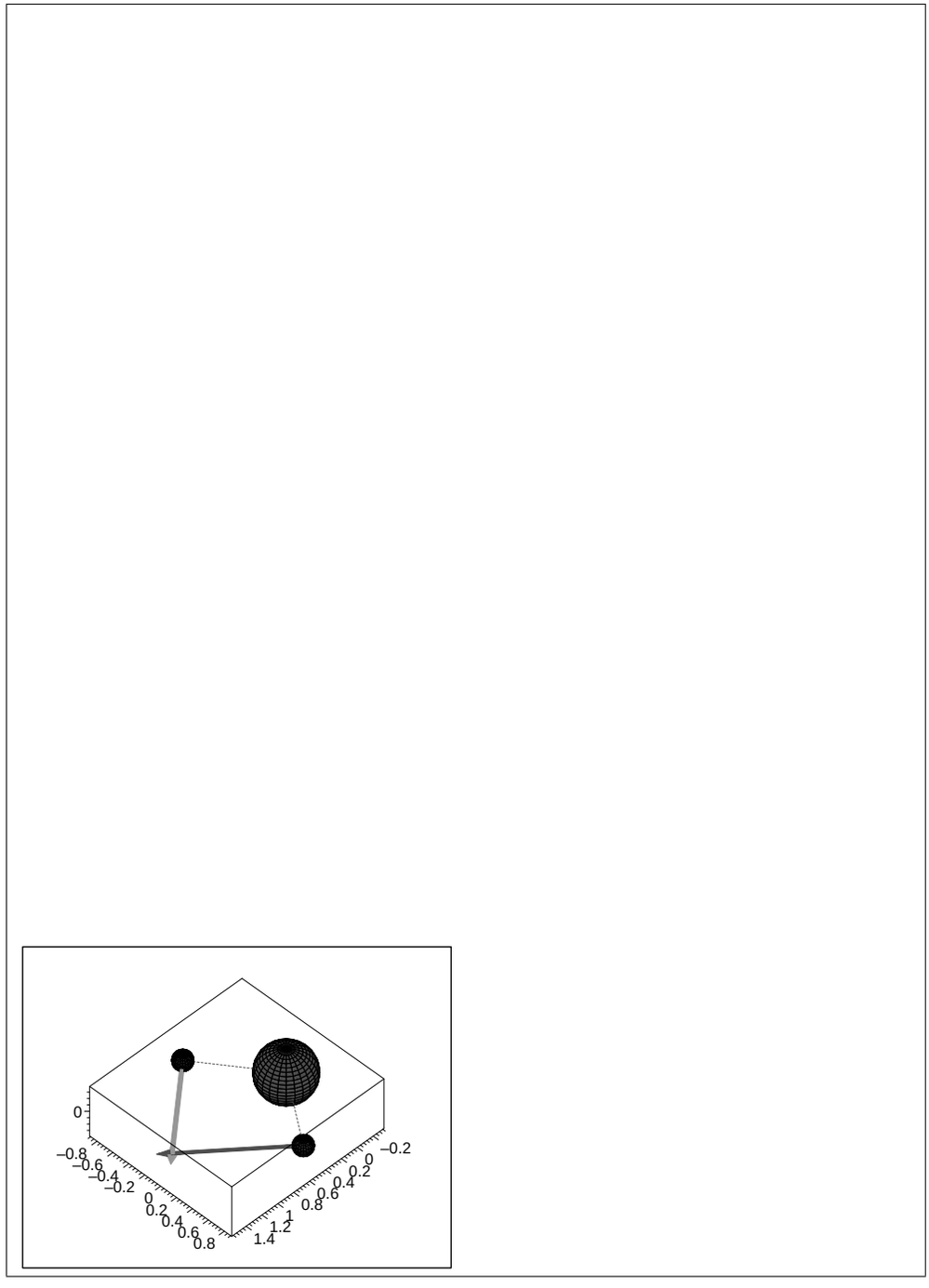

FIG. 3: The third highest energy normal mode.

III. DISCUSSION

The graphical representation of the normal vibrational modes is so interesting and thought provoking, that one wonders why this method has not received more public-

ity.

Unfortunately, I can't publish it in a proper Journal until I can find out the identity of the original author. This seems to be so sad.

[] Unfortunately, I've used someone else's work here, and the original citation is lost in the chaos of my office. I've searched in vain for hours on the WWW, to no avail. If the author of the original Maple spreadsheet will contact me, I will be more than happy to cite his/her work.
And my humble apologies for "plagarizing" someone else's work.
cwd